RELAXING THE ADDITIVITY CONSTRAINTS IN DECENTRALIZED NO-REGRET HIGH-DIMENSIONAL BAYESIAN OPTIMIZATION

Anthony Bardou* **& Patrick Thiran**
IC, EPFL
Lausanne, Switzerland
{anthony.bardou,patrick.thiran}@epfl.ch

Thomas Begin
ENS Lyon, UCBL, CNRS, LIP
Lyon, France
thomas.begin@ens-lyon.fr

ABSTRACT

Bayesian Optimization (BO) is typically used to optimize an unknown function f that is noisy and costly to evaluate, by exploiting an acquisition function that must be maximized at each optimization step. Even if provably asymptotically optimal BO algorithms are efficient at optimizing low-dimensional functions, scaling them to high-dimensional spaces remains an open problem, often tackled by assuming an additive structure for f. By doing so, BO algorithms typically introduce additional restrictive assumptions on the additive structure that reduce their applicability domain. This paper contains two main contributions: (i) we relax the restrictive assumptions on the additive structure of f *without* weakening the maximization guarantees of the acquisition function, and (ii) we address the over-exploration problem for decentralized BO algorithms. To these ends, we propose DuMBO, an asymptotically optimal decentralized BO algorithm that achieves very competitive performance against state-of-the-art BO algorithms, especially when the additive structure of f comprises high-dimensional factors.

1 INTRODUCTION

Many real-world applications involve optimizing an unknown, noisy, costly-to-evaluate objective function f. Examples of such tasks include hyper parameters tuning in deep neural networks (Bergstra et al., 2013), robotics (Lizotte et al., 2007), networking (Hornby et al., 2006) and computational biology (González et al., 2014). In such applications, f can be seen as a black box that can only be discovered by successive queries. This prevents the use of traditional first-order approaches to optimize f.

Bayesian Optimization (BO) has become a highly effective framework for black-box optimization. In general, a BO algorithm tackles this problem by modeling f as a Gaussian process (GP) and by leveraging this model to query f at specific inputs. The challenge of querying f is to trade off exploration (*i.e.* to adequately query an input that improves the quality of the GP regression of f) for exploitation (*i.e.* to query an input that is thought to be the maximal argument of f). To achieve this trade-off at time t, a BO algorithm maximizes an acquisition function $\varphi_t(\boldsymbol{x})$, built by leveraging the information provided by the GP model, to select a query $\boldsymbol{x}^t$.

Although BO has shown its efficiency at optimizing black-box functions, so far it has mostly found success with low-dimensional input spaces (Wang et al., 2013). However, real-world applications, such as computer vision, robotics or networking, often involve a high-dimensional objective function f. Scaling BO algorithms to such input spaces remains a great challenge as the cost of finding $\arg\max \varphi_t$ grows exponentially with the input space dimension d. A classical way to circumvent that issue is to cap the complexity of the maximization by assuming an additive decomposition of f (e.g. see Kandasamy et al. (2015)) with a low *Maximum Factor Size* (MFS), denoted by $\bar{d}$, which is the maximum number of dimensions for a factor of the decomposition. Unfortunately, assuming an additive decomposition with low MFS may lead to the optimization of a coarse approximation of f.

*Some of this work was done while AB was at ENS Lyon.

The performance of BO algorithms under the assumption of low MFS has been extensively studied (e.g. see Kandasamy et al. (2015); Hoang et al. (2018); Mutny & Krause (2018)), with Hoang et al. (2018) detailing efforts to relax the assumption on MFS. In this paper, we demonstrate that it is possible to completely relax the low-MFS assumptions that limit the applicability domain of asymptotically optimal BO algorithms while still providing provable global maximization guarantees on the acquisition function. To illustrate this, we propose DuMBO, a decentralized, message-passing, asymptotically optimal BO algorithm able to infer a complex additive decomposition of f without any assumption regarding its MFS. As far as we know, this is the first BO algorithm to display such desirable property. Additionally, we provide an efficient way to approximate the well-known GP-UCB acquisition function (Srinivas et al., 2012) in a decentralized context. Finally, we evaluate DuMBO and establish its superiority against several state-of-the-art solutions on both synthetic and real-world problems wherein the noisy objective function f may or may not be decomposed.

2 BACKGROUND

2.1 STATE OF THE ART

Given a black-box objective function $f : \mathcal{D} \subset \mathbb{R}^d \to \mathbb{R}$, the goal of a BO algorithm is to find $\boldsymbol{x}^* = \arg\max_{\boldsymbol{x}} f(\boldsymbol{x})$ using as few queries as possible. The quality of the optimization is measured with the immediate regret $r_t = f(\boldsymbol{x}^*) - f(\boldsymbol{x}^t)$, and the cumulative regret $R_t = \sum_{i=1}^{t} r_i$. A BO algorithm is said to be asymptotically optimal if $\lim_{t\to+\infty} R_t/t = 0$, which implies that the BO algorithm will asymptotically reach $\boldsymbol{x}^*$ and hence guarantees *no-regret* performance.

A BO algorithm typically uses a GP to infer a posterior distribution for the value of $f(\boldsymbol{x})$ at any point $\boldsymbol{x} \in \mathcal{D}$ and selects, at each time step t, a query $\boldsymbol{x}^t$. The BO algorithm bases its querying policy on the maximization of an acquisition function that quantifies the benefits of observing $f(\boldsymbol{x})$ in terms of exploration and exploitation. Common acquisition functions include probability of improvement (Jones et al., 1998), expected improvement (Mockus, 1994) and upper confidence bound (Auer, 2003). Like many other acquisition functions, the latter leads to an asymptotically optimal application to GPs, called GP-UCB (Srinivas et al., 2012) and defined as

$$\varphi_t(\boldsymbol{x}) = \mu_t(\boldsymbol{x}) + \beta_t^{\frac{1}{2}} \sigma_t(\boldsymbol{x}). \tag{1}$$

It involves an exploitation term $\mu_t(\boldsymbol{x})$, which is the posterior mean of the GP at input $\boldsymbol{x}$, and an exploration term $\sigma_t(\boldsymbol{x})$, which is the posterior standard deviation of the GP at input $\boldsymbol{x}$. The scalar $\beta_t^{1/2}$ handles the exploration-exploitation trade-off in order to guarantee the asymptotic optimality of GP-UCB with high probability.

As stated before, scaling BO algorithms to high-dimensional functions is challenging because of the exponential complexity of the optimization algorithms used to maximize φ_t. To tackle this problem, BO algorithms generally fall into one of the two following categories (with the exception of TuRBO proposed by Eriksson et al. (2019), which uses trust regions to maximize f).

Embedding BO algorithms assume that only a few dimensions significantly impact f and project the high-dimensional space of f into a low-dimensional one where the optimization is actually performed. REMBO (Wang et al., 2016) and ALEBO (Letham et al., 2020) use random matrices to embed the high-dimensional space while SAASBO (Eriksson & Jankowiak, 2021) uses sparse GPs defined on subspaces and LineBO (Kirschner et al., 2019) exploits successive line-searches in random directions. Other approaches such as Gómez-Bombarelli et al. (2018); Moriconi et al. (2020) are based, respectively, on Variational Auto-Encoders and on manifold GPs to learn an embedding. Gupta et al. (2020) propose to perform the optimization in two orthogonal subspaces. Finally, some approaches select a subset of dimensions of the input space to project onto. Such recent methods include Dropout (Li et al., 2017) and MCTS-VS (Song et al., 2022).

Decomposing BO algorithms assume an additive structure for f and optimize the factors of the induced decomposition. Classical approaches such as MES (Wang & Jegelka, 2017), ADD-GPUCB (Kandasamy et al., 2015) or QFF (Mutny & Krause, 2018) assume a decomposition with a MFS equal to 1 and orthogonal domains. More recent approaches like DEC-HBO (Hoang et al., 2018) are able to optimize decompositions with larger MFS and shared input components. Still, the MFS of the decomposition must be low to avoid a prohibitive computational complexity. Note that,

Table 1: Comparison of state-of-the-art decomposing BO algorithms with DuMBO on relevant criteria. Here, n is the number of factors in the decomposition, d the number of dimensions of f, $\bar{d}$ the MFS of the decomposition, t the optimization step, ζ the desired accuracy when maximizing φ_t and N_A a constant defined in Appendix E. N_m is a constant defined in Hoang et al. (2018).

Solution	Complexity	MFS Assumption	Find $\arg\max \varphi_t$
ADD-GPUCB	$\mathcal{O}\left(t^3 + nt^2 + n^2\zeta^{-1}\right)$	$\bar{d} = 1$	Yes
QFF	$\mathcal{O}\left((\zeta^{-1}t^{3/2}(\log t)^{\bar{d}})^{\bar{d}}\right)$	$\bar{d} = 1$	Yes
DEC-HBO	$\mathcal{O}\left(N_m\zeta^{-\bar{d}}n(t^3 + n)\right)$	Low $\bar{d}$	Under assumptions
DuMBO	$\mathcal{O}\left(\bar{d}N_A n t^3 \zeta^{-1}\right)$	None	Yes

under some assumptions on f, these approaches are provably asymptotically optimal and a subset of them, namely ADD-GPUCB (Kandasamy et al., 2015) and DEC-HBO (Hoang et al., 2018), can be used in a decentralized context. Finally, note that in a recent work, (Ziomek & Ammar, 2023) showed that, in an adversarial context, exploiting random decompositions is optimal on average.

2.2 DuMBO (Decentralized Message-passing Bayesian Optimization algorithm)

We propose DuMBO, a decomposing algorithm that relaxes the low MFS constraint on the assumed additive decomposition of f. Table 1 gathers the main differences between DuMBO and state-of-the-art decomposing algorithms. Note that ADD-GPUCB and QFF require the simplest form of additive decompositions (*i.e.*, $\bar{d} = 1$). As a consequence, when optimizing a complex objective function f, they often need to coarsely approximate it. In return, they are able, at each time step t, to query $\arg\max \varphi_t$. In contrast, DEC-HBO tolerates more complex decompositions (*i.e.*, with $\bar{d} > 1$), but is no longer guaranteed to find the global maximum of φ_t, because it uses a max-sum algorithm (Rogers et al., 2011) that requires f to have a sparse additive decomposition to converge. Finally, DuMBO is the only algorithm that is able to completely relax any assumption on the MFS without weakening the maximization guarantees on the acquisition function. This allows DuMBO to be simultaneously asymptotically optimal and able to handle arbitrarily complex decompositions.

The remaining of this article is devoted to formulating the BO problem (Section 3), presenting DuMBO (Section 4), providing theoretical guarantees (Section 5) and comparing its empirical performance with state-of-the-art BO algorithms (Section 6).

3 Problem Formulation and First Results

In this section, we introduce the core assumptions about the black-box objective function $f : \mathcal{D} \to \mathbb{R}$ to obtain an additive decomposition (Section 3.1). Next, we exploit these assumptions to derive inference formulas (Section 3.2) and to adapt GP-UCB to a decentralized context (Section 3.3).

3.1 Core Assumptions

In order to optimize f in a decentralized fashion, we make several assumptions.

Assumption 3.1. *The unknown objective function f can be decomposed into a sum of factor functions $\left(f^{(i)}\right)_{i\in[\![1,n]\!]}$, with compact domains $\left(D^{(i)}\right)_{i\in[\![1,n]\!]}$, such that $\mathcal{D} = \cup_{i=1}^{n} D^{(i)}$ and*

$$f = \sum_{i=1}^{n} f^{(i)}. \tag{2}$$

Any decomposition can be represented by a factor graph where each factor and variable node denote, respectively, one of the n factors of the decomposition and one of the d input components of f. An edge exists between a factor node i and a variable node j if and only if $f^{(i)}$ uses x_j as an input component. We use $\mathcal{V}_i, i \in [\![1, n]\!]$, and $\mathcal{F}_j, j \in [\![1, d]\!]$, to denote respectively the set of variable

nodes connected to factor node i and the set of factor nodes connected to variable node j. Please refer to Appendix A for a detailed example regarding additive decompositions and factor graphs.

To make predictions about the factor functions without any prior knowledge, we need a model that maps the previously collected inputs with their noisy outputs. Denoting $\boldsymbol{x}_{\mathcal{V}_i} = (x_j)_{j \in \mathcal{V}_i}$, let us introduce the following assumption.

Assumption 3.2. *Factor functions $f^{(i)}$ are independent $\mathcal{GP}\left(\mu_0^{(i)}, k^{(i)}\left(\boldsymbol{x}_{\mathcal{V}_i}, \boldsymbol{x}'_{\mathcal{V}_i}\right)\right)$, with prior mean $\mu_0^{(i)} = 0$ and covariance function $k^{(i)}$.*

Since f is a sum of independent GPs, Assumption 3.2 implies that f is also $\mathcal{GP}\left(\mu_0, k(\boldsymbol{x}, \boldsymbol{x'})\right)$ with prior mean $\mu_0 = 0$ and covariance function $k(\boldsymbol{x}, \boldsymbol{x'}) = \sum_{i=1}^{n} k^{(i)}\left(\boldsymbol{x}_{\mathcal{V}_i}, \boldsymbol{x}'_{\mathcal{V}_i}\right)$.

Finally, to ensure the no-regret property of DuMBO (see Section 5), we introduce the following assumption on each $k^{(i)}$.

Assumption 3.3. *For any $i \in [\![1, n]\!]$, $k^{(i)}$ is an L-Lipschitz, twice differentiable function on $\mathcal{D}^{(i)}$. Furthermore, it exists $H > 0$ such that, for any $\boldsymbol{x}, \boldsymbol{x}' \in \mathcal{D}^{(i)}$ we have*

$$||\nabla^2 k^{(i)}(\boldsymbol{x}, \boldsymbol{x}')||_2 \leq H. \tag{3}$$

Note that Assumption 3.3 is mild: a large class of covariance functions satisfy it, such as the Matérn class (with $\nu \geq 5/2$), the squared-exponential function or even the rational quadratic function. Please refer to Williams & Rasmussen (2006) for details on these covariance functions.

3.2 Inference Formulas

For any $\boldsymbol{x} \in \mathcal{D}$ and given the previous input queries $(\boldsymbol{x}^1, \cdots, \boldsymbol{x}^t)$, the vector $(f(\boldsymbol{x}), f(\boldsymbol{x}^1), \cdots, f(\boldsymbol{x}^t))^\top$ is Gaussian. Given the t-dimensional vector of noisy outputs $\boldsymbol{y} = (y_1, \cdots, y_t)^\top$, with $y_i = f(\boldsymbol{x}^i) + \epsilon$ and ϵ a centered Gaussian variable of variance σ^2, the posterior distribution of the factor $f^{(i)}(\boldsymbol{x})$ is also Gaussian. Since f can be decomposed, the posterior mean $\mu_{t+1}^{(i)}(\boldsymbol{x}_{\mathcal{V}_i})$ and variance $(\sigma_{t+1}^{(i)}(\boldsymbol{x}_{\mathcal{V}_i}))^2$ of the factor $f^{(i)}$ at time $t+1$ can be expressed with the posterior means and covariance functions of the factor functions involved in decomposition (2).

Proposition 3.4. *Let $\mu_{t+1}^{(i)}(\boldsymbol{x}_{\mathcal{V}_i})$ and $(\sigma_{t+1}^{(i)}(\boldsymbol{x}_{\mathcal{V}_i}))^2$ be the posterior mean and variance of $f^{(i)}$ at input $\boldsymbol{x}_{\mathcal{V}_i}$, respectively. Then, for the decomposition (2),*

$$\mu_{t+1}^{(i)}(\boldsymbol{x}_{\mathcal{V}_i}) = \boldsymbol{k}_{\boldsymbol{x}_{\mathcal{V}_i}}^{(i)\top}\left(\boldsymbol{K} + \sigma^2 \boldsymbol{I}\right)^{-1} \boldsymbol{y} \tag{4}$$

$$(\sigma_{t+1}^{(i)}(\boldsymbol{x}_{\mathcal{V}_i}))^2 = k^{(i)}(\boldsymbol{x}_{\mathcal{V}_i}, \boldsymbol{x}_{\mathcal{V}_i}) - \boldsymbol{k}_{\boldsymbol{x}_{\mathcal{V}_i}}^{(i)\top}\left(\boldsymbol{K} + \sigma^2 \boldsymbol{I}\right)^{-1} \boldsymbol{k}_{\boldsymbol{x}_{\mathcal{V}_i}}^{(i)} \tag{5}$$

with $t \times 1$ vectors $\boldsymbol{k}_{\boldsymbol{x}_{\mathcal{V}_i}}^{(i)} = (k^{(i)}(\boldsymbol{x}_{\mathcal{V}_i}, \boldsymbol{x}_{\mathcal{V}_i}^j))_{j \in [\![1,t]\!]}$, $t \times t$ matrix $\boldsymbol{K} = (k(\boldsymbol{x}^j, \boldsymbol{x}^k))_{j,k \in [\![1,t]\!]}$ and $\boldsymbol{I}$ the $t \times t$ identity matrix.

For the sake of generality, Proposition 3.4 only requires an additive decomposition of f. Appendix B describes how such a decomposition can be inferred from data, using the method proposed by Gardner et al. (2017). Note that Proposition 3.4 does *not* assume a corresponding additive decomposition of the observed outputs in $\boldsymbol{y}$. However a large class of real-world applications naturally come up with such an additive output decomposition (e.g., network throughput maximization (Bardou & Begin, 2022), energy consumption minimization (Bourdeau et al., 2019) or UAVs-related applications (Xie et al., 2018)). As shown by Wang et al. (2020), having access to a decomposed output can only improve the predictive performance of the GP surrogate model. Therefore, we derive the inference formulas to handle the case where the output decomposition is known in Appendix C. Also, we explore the benefits of having access to the decomposed output of f in Section 6.

3.3 Proposed Acquisition Function

Having defined a surrogate model for f, we now turn to finding an optimal policy for querying the objective function. In this section, we exploit the decomposition of f and its associated factor graph (see Appendix A) to build an acquisition function for our BO algorithm that approximates GP-UCB in a decentralized context. Proofs for all the presented results can be found in Appendix D.

Recall that GP-UCB is defined by (1) as the sum of an exploitation term $\mu_t(\boldsymbol{x})$ and an exploration term $\sigma_t(\boldsymbol{x})$ weighted by some scalar $\beta_t^{1/2}$. Finding an additive decomposition for GP-UCB is hard, because Assumption 3.1 allows $\mu_t(\boldsymbol{x})$ to be expressed as a sum, but not $\sigma_t(\boldsymbol{x})$. To circumvent this caveat, Kandasamy et al. (2015) proposed to apply GP-UCB to each factor of the additive decomposition of f, with $\varphi_t^{(i)} = \mu_t^{(i)} + \beta_t^{1/2}\sigma_t^{(i)}$. Then, they proved that their algorithm ADD-GPUCB offers no-regret performance by taking $\sum_{i=1}^n \varphi_t^{(i)} = \mu_t + \beta_t^{1/2}\sum_{i=1}^n \sigma_t^{(i)}$ as the acquisition function. Although the exploitation term μ_t is preserved, the exploration term is now overweighted since $\sum_{i=1}^n \sigma_t^{(i)} \geq \sqrt{\sum_{i=1}^n \left(\sigma_t^{(i)}\right)^2} = \sigma_t$. To reach better empirical performance, one could look for a tighter additive upper bound of σ_t^2. This is the purpose of this section.

In a given factor graph, a factor node i can access information about another factor node if they share a common variable node j (see Figure 3 in Appendix A). We gather all the indices of the factor nodes that share at least one variable node with the factor node i in $\mathcal{N}_i = \cup_{j\in\mathcal{V}_i}\mathcal{F}_j$. Then, we propose the following approximation for $\sigma_t(\boldsymbol{x})$:

$$\sum_{i=1}^n \sqrt{\sum_{k\in\mathcal{N}_i} \frac{\left(\sigma_t^{(k)}(\boldsymbol{x}_{\mathcal{V}_k})\right)^2}{|\mathcal{N}_k|^2}}. \tag{6}$$

On the one hand, this approximation is exact and equal to $\sigma_t(\boldsymbol{x})$ for a complete factor graph (*i.e.*, $\forall i \in [\![1,n]\!], |\mathcal{N}_i| = n$). On the other hand, given a decomposition made only of one-dimensional factors with orthogonal domains (*i.e.*, $\forall i \in [\![1,n]\!], |\mathcal{N}_i| = 1$), it boils down to the approximation proposed by Kandasamy et al. (2015), that is, $\sum_{i=1}^n \sigma_t^{(i)}(\boldsymbol{x}_{\mathcal{V}_i})$. A benefit of approximation (6) is to better exploit the structure of the factor graph. Indeed, the following result shows that (6) is a tighter upper bound of $\sigma_t(\boldsymbol{x})$ than the one proposed in Kandasamy et al. (2015).

Theorem 3.5. *Let Assumptions 3.1 and 3.2 hold. Then, for any factor graph and any $\boldsymbol{x} \in \mathcal{D}$,*

$$\sigma_t(\boldsymbol{x}) \leq \sum_{i=1}^n \sqrt{\sum_{k\in\mathcal{N}_i} \frac{\left(\sigma_t^{(k)}(\boldsymbol{x}_{\mathcal{V}_k})\right)^2}{|\mathcal{N}_k|^2}} \leq \sum_{i=1}^n \sigma_t^{(i)}(\boldsymbol{x}_{\mathcal{V}_i}). \tag{7}$$

From the bounds of Theorem 3.5, one can expect the acquisition function

$$\varphi_t(\boldsymbol{x}) = \mu_t(\boldsymbol{x}) + \beta_t^{\frac{1}{2}} \sum_{i=1}^n \sqrt{\sum_{k\in\mathcal{N}_i} \frac{\left(\sigma_t^{(k)}(\boldsymbol{x}_{\mathcal{V}_k})\right)^2}{|\mathcal{N}_k|^2}} \tag{8}$$

to be less prone to over-exploration. Thus, a BO algorithm using (8) behaves more like GP-UCB than ADD-GPUCB or DEC-HBO. Note that (8) has a natural decomposition $\sum_{i=1}^n \varphi_t^{(i)}(\boldsymbol{x}_{\mathcal{V}_i})$ with

$$\varphi_t^{(i)}(\boldsymbol{x}_{\mathcal{V}_i}) = \mu_t^{(i)}(\boldsymbol{x}_{\mathcal{V}_i}) + \beta_t^{\frac{1}{2}} \sqrt{\frac{\left(\sigma_t^{(i)}(\boldsymbol{x}_{\mathcal{V}_i})\right)^2}{|\mathcal{N}_i|^2} + c_i} \tag{9}$$

with $c_i = \sum_{\substack{k\in\mathcal{N}_i \\ k\neq i}} \frac{\left(\sigma_t^{(k)}(\boldsymbol{x}_{\mathcal{V}_k})\right)^2}{|\mathcal{N}_k|^2}$ computed by message-passing with the variable nodes in $\mathcal{V}_i$.

4 DuMBO

In this section, we describe DuMBO, a BO algorithm that exploits the results from Section 3 to find $\arg\max_{\boldsymbol{x}\in\mathcal{D}} \sum_{i=1}^n \varphi_t^{(i)}(\boldsymbol{x}_{\mathcal{V}_i})$. Optimizing $\varphi_t(\boldsymbol{x}) = \sum_{i=1}^n \varphi_t^{(i)}(\boldsymbol{x}_{\mathcal{V}_i})$ while ensuring the consistency between shared input components is equivalent to solving the constrained optimization problem

$$\max \sum_{i=1}^n \varphi_t^{(i)}\left(\boldsymbol{x}^{(i)}\right) \text{ such that } \boldsymbol{x}^{(i)}_{\mathcal{V}_i\cap\mathcal{V}_j} = \boldsymbol{x}^{(j)}_{\mathcal{V}_i\cap\mathcal{V}_j}, \forall i,j \in [\![1,n]\!] \tag{10}$$

with $\boldsymbol{x}^{(1)}, \cdots, \boldsymbol{x}^{(n)}$ being the inputs (whose dimension indices are respectively listed in $\mathcal{V}_1, \cdots, \mathcal{V}_n$) of the factor functions $\varphi_t^{(1)}, \cdots, \varphi_t^{(n)}$.

To simplify the equality constraints in (10), we introduce a consensus variable $\bar{\boldsymbol{x}} \in \mathcal{D}$ and we reformulate the optimization problem as

$$\max \sum_{i=1}^{n} \varphi_t^{(i)} \left(\boldsymbol{x}^{(i)} \right) \text{ such that } \boldsymbol{x}^{(i)} = \bar{\boldsymbol{x}}_{\mathcal{V}_i}, \forall i \in [\![1, n]\!] . \tag{11}$$

We now turn the problem (11) into an unconstrained optimization problem by considering its augmented Lagrangian $\mathcal{L}_\eta(\boldsymbol{x}^{(1)}, \cdots, \boldsymbol{x}^{(n)}, \bar{\boldsymbol{x}}, \boldsymbol{\lambda})$:

$$\mathcal{L}_\eta = \sum_{i=1}^{n} \varphi_t^{(i)}(\boldsymbol{x}^{(i)}) - \boldsymbol{\lambda}^{(i)\top}(\boldsymbol{x}^{(i)} - \bar{\boldsymbol{x}}_{\mathcal{V}_i}) - \frac{\eta}{2} ||\boldsymbol{x}^{(i)} - \bar{\boldsymbol{x}}_{\mathcal{V}_i}||_2^2 \tag{12}$$

with $\boldsymbol{\lambda}_k^{(i)}$ a column vector of dual variables with $|\mathcal{V}_i|$ components and a hyperparameter $\eta > 0$, which can be set dynamically following the procedure detailed in Boyd et al. (2011).

To maximize (12), we consider the Alternating Direction Method of Multipliers (ADMM), proposed by Gabay & Mercier (1976). ADMM is an iterative method that proposes, at iteration k, to solve sequentially the problems

$$\begin{aligned}
\boldsymbol{x}_{k+1}^{(1)} &= \arg\max_{\boldsymbol{x}^{(1)}} \mathcal{L}(\boldsymbol{x}^{(1)}, \cdots, \boldsymbol{x}_k^{(n)}, \bar{\boldsymbol{x}}_k, \boldsymbol{\lambda}_k) \\
&\vdots \\
\boldsymbol{x}_{k+1}^{(n)} &= \arg\max_{\boldsymbol{x}^{(n)}} \mathcal{L}(\boldsymbol{x}_{k+1}^{(1)}, \cdots, \boldsymbol{x}_{k+1}^{(n-1)}, \boldsymbol{x}^{(n)}, \bar{\boldsymbol{x}}_k, \boldsymbol{\lambda}_k) \\
\bar{\boldsymbol{x}}_{k+1} &= \arg\max_{\bar{\boldsymbol{x}}} \mathcal{L}(\boldsymbol{x}_{k+1}^{(1)}, \cdots, \boldsymbol{x}_{k+1}^{(n)}, \bar{\boldsymbol{x}}, \boldsymbol{\lambda}_k) && (13) \\
\boldsymbol{\lambda}_{k+1} &= \arg\max_{\boldsymbol{\lambda}} \mathcal{L}(\boldsymbol{x}_{k+1}^{(1)}, \cdots, \boldsymbol{x}_{k+1}^{(n)}, \bar{\boldsymbol{x}}_{k+1}, \boldsymbol{\lambda}). && (14)
\end{aligned}$$

Note that $\boldsymbol{x}_{k+1}^{(1)}, \cdots, \boldsymbol{x}_{k+1}^{(n)}$ can be found concurrently by each factor node of the factor graph of f. We propose to proceed by gradient ascent (e.g. with ADAM (Kingma & Ba, 2015)) of

$$\mathcal{L}_\eta^{(i)} = \varphi_t^{(i)}(\boldsymbol{x}^{(i)}) - \boldsymbol{\lambda}^{(i)\top}(\boldsymbol{x}^{(i)} - \bar{\boldsymbol{x}}_{\mathcal{V}_i}) - \frac{\eta}{2} ||\boldsymbol{x}^{(i)} - \bar{\boldsymbol{x}}_{\mathcal{V}_i}||_2^2. \tag{15}$$

Next, each factor node i sends $\left(\boldsymbol{x}_{k+1}^{(i)}, \frac{\left(\sigma_t^{(i)}(\boldsymbol{x}_{k+1}^{(i)})\right)^2}{|\mathcal{N}_i|^2} \right)$ to its variable nodes in $\mathcal{V}_i$. Each variable node j uses the received data to compute (13) and (14). In fact, if $\forall i \in [\![1, n]\!]$, $\sum_{j \in \mathcal{F}_i} \lambda_{0,i}^{(j)} = 0$, it is known (see Boyd et al. (2011)) that the closed-forms for (13) and (14) are

$$\bar{\boldsymbol{x}}_{k+1} = \left(\frac{1}{|\mathcal{F}_i|} \sum_{j \in \mathcal{F}_i} x_{k+1,i}^{(j)} \right)_{i \in [\![1,d]\!]} \tag{16}$$

$$\boldsymbol{\lambda}_{k+1} = \left(\boldsymbol{\lambda}_k^{(i)} + \eta \left(\boldsymbol{x}_{k+1}^{(i)} - \bar{\boldsymbol{x}}_{k+1,\mathcal{V}_i} \right) \right)_{i \in [\![1,n]\!]} . \tag{17}$$

Finally, each variable node j sends $\left(\boldsymbol{\lambda}_{k+1}^{(i)}, \bar{x}_{k+1,j} \right)$ as well as $\left(\frac{\left(\sigma_t^{(l)}(\boldsymbol{x}_{k+1}^{(l)})\right)^2}{|\mathcal{N}_l|^2} \right)_{l \in \mathcal{F}_j}$ to its factor node i, $i \in \mathcal{F}_j$. This allows each factor node i to update its dual variables $\boldsymbol{\lambda}^{(i)}$ as well as the value of term c_i in (9).

These results describe a fully decentralized message-passing algorithm, called DuMBO, which can run on the factor graph of f. The detailed algorithm (Algorithm 1), as well as a discussion about its time complexity, are provided in Appendix E. Since DuMBO relies on ADMM to maximize φ_t, let us briefly discuss its maximization guarantees. It is well known that ADMM converges towards the global maximum of a convex φ_t. ADMM has also demonstrated very good performance at optimizing non-convex functions (Liavas & Sidiropoulos, 2015; Lai & Osher, 2014; Chartrand & Wohlberg, 2013). This is explained by recent works such as Wang et al. (2019), which extends the global maximization guarantee of ADMM to the class of *restricted prox-regular* functions that satisfy the Kurdyka-Lojasiewicz condition. We demonstrate that the acquisition function φ_t simultaneously satisfies these conditions in the next section.

5 Asymptotic Optimality

In this section, we demonstrate the asymptotic optimality of DuMBO. First, we prove that, at each iteration, ADMM is always able to globally maximize the acquisition function φ_t with Theorem 5.1. Then, we demonstrate that DuMBO has a lower immediate regret than another asymptotically optimal BO algorithm with Theorem 5.2. With these two theorems, we establish the asymptotical optimality of DuMBO, stated in Corollary 5.3.

Let us start with the global maximization guarantee of ADMM, whose proof can be found in Appendix F.

Theorem 5.1. *Under Assumption 3.3, $\forall i \in [\![1, n]\!], \varphi_t^{(i)}$ (see (9)) is a restricted-prox regular function. Furthermore, the augmented Lagrangian $\mathcal{L}_\eta$ (see (12)) is a Kurdyka-Lojasiewicz (KL) function.*

Now that we have the guarantee that φ_t is always maximized, we can properly establish the asymptotic optimality of DuMBO. We start by providing an upper bound on its immediate regret $r_t = f(\boldsymbol{x}^*) - f(\boldsymbol{x}_t)$ for a finite, discrete domain $\mathcal{D}$. Its proof can be found in Appendix G.

Theorem 5.2. *Let $r_t = f(\boldsymbol{x}^*) - f(\boldsymbol{x}^t)$ denote the immediate regret of DuMBO. Let $\delta \in (0, 1)$ and $\beta_t = 2 \log \left(\frac{|\mathcal{D}| \pi^2 t^2}{6\delta} \right)$. Then $\forall t \in \mathbb{N}$, with probability at least $1 - \delta$,*

$$r_t \leq 2\beta_t^{\frac{1}{2}} \sum_{i=1}^{n} \sqrt{\sum_{k \in \mathcal{N}_i} \frac{\left(\sigma_t^{(k)}(\boldsymbol{x}_{\mathcal{V}_k}) \right)^2}{|\mathcal{N}_k|^2}}, \tag{18}$$

We demonstrate the asymptotic optimality of DuMBO by piggybacking on the asymptotic optimality of DEC-HBO (Hoang et al., 2018). The latter is a decomposing BO algorithm with an immediate regret bound of $2\beta_t^{1/2} \sum_{i=1}^{n} \sigma_t^{(i)}(\boldsymbol{x}^t)$ over a finite, discrete domain (see Theorem 1 in Hoang et al. (2018)). Interestingly, Theorem 3.5 directly implies that the immediate regret bound (18) is lower than the immediate regret bound of DEC-HBO. As a consequence, the immediate regret of DuMBO is bounded from above by the regret bound of DEC-HBO. This allows us to rely on proofs in Hoang et al. (2018) to establish some properties of DuMBO. In particular, DEC-HBO is provably asymptotically optimal whether the domain $\mathcal{D}$ is discrete or continuous (see Theorems 2 and 3 in Hoang et al. (2018)). These results rely on their immediate regret bound over a finite, discrete domain. Hence, they directly apply to DuMBO as well and yield the following corollary.

Corollary 5.3. *Let $\delta \in (0, 1)$ and $R_t = \sum_{k=1}^{t} r_k$ denote the cumulative regret of DuMBO. Then, with probability at least $1 - \delta$, there exists a monotonically increasing sequence $\{\beta_t\}_t$ such that $\beta_t \in \mathcal{O}(\log t)$ and $\lim_{t \to +\infty} R_t / t = 0$.*

6 Performance Experiments

In this section, we detail the experiments carried out to evaluate the empirical performance of DuMBO. An open-source implementation of DuMBO, based on BoTorch (Balandat et al., 2020), is available on GitHub[1].

[1] `https://github.com/abardou/dumbo`

Table 2: Comparison of eight state-of-the-art solutions against two different versions of DuMBO on synthetic and real-world problems. Decomposing BO algorithms can be identified with the prefix "(+)". The reported metrics are the minimal regret attained for the synthetic functions, and the average negative reward for the real-world problems. The significantly best performance metrics among all the strategies are written in **bold text**, and the significantly best among the strategies that do not have access to the additive decomposition are underlined.

Algorithm	Synthetic Functions (d-$\bar{d}$)				Real-World Problems (d-$\bar{d}$)		
Unknown Add. Dec.	SHC (2-2)	Hartmann (6-6)	Powell (24-4)	Rastrigin (100-5)	Cosmo (9-)	WLAN (12-6)	Rover (60-)
SAASBO	0.013	0.89	3,901	1,073	16.55	-116.40	10.82
TuRBO	0.322	1.89	667	1,109	**5.82**	-118.39	**7.01**
LineBO	0.016	0.69	4,830	1,388	**5.90**	-118.68	8.24
MS-UCB	0.012	0.80	22,271	1,455	**5.87**	-117.95	7.65
(+) ADD-GPUCB	0.102	1.29	11,760	N/A	7.46	-119.05	26.57
(+) DEC-HBO	**0.005**	1.47	7,937	N/A	14.90	-116.58	10.07
(+) DuMBO	**0.006**	**0.54**	496	986	**5.86**	-120.67	**6.38**
Known Add. Dec.							
(+) ADD-DuMBO	0.009	**0.53**	**469**	**678**	N/A	**-121.11**	N/A

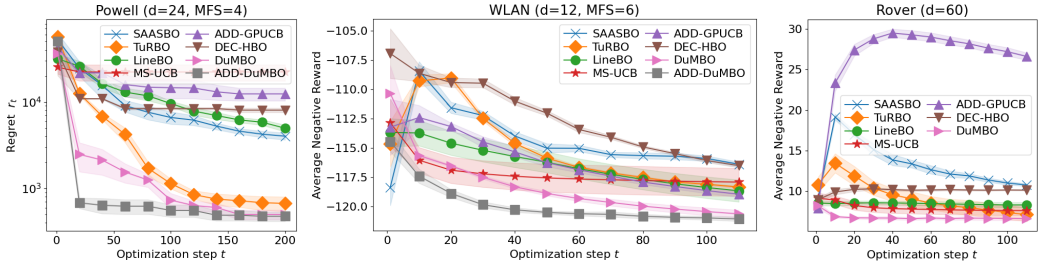
Our benchmark comprises four synthetic functions and three real-world experiments. We consider two state-of-the-art decomposing BO algorithms: ADD-GPUCB (Kandasamy et al., 2015) that assumes $\bar{d} = 1$, and DEC-HBO (Hoang et al., 2018) for which, similarly to its authors in their empirical evaluation, we assume $\bar{d} \leq 3$. We also consider four state-of-the-art BO algorithms that do not assume an additive decomposition of the objective function: TuRBO (Eriksson et al., 2019), SAASBO (Eriksson & Jankowiak, 2021), LineBO (Kirschner et al., 2019) and MS-UCB (Gupta et al., 2020) with its hyperparameter $\alpha = 0$. We compare these eight algorithms with two versions of the proposed algorithm: DuMBO that must systematically infer an additive decomposition of f (see Appendix B) and ADD-DuMBO that, conversely, can observe the true decomposition of f if it exists (see Appendix C). Finally, note that we chose a Matérn kernel (with its hyperparameter $\nu = 5/2$) for each GP involved in these experiments.

Since BO is often used in the optimization of expensive black-box functions, we are interested in the ability of each algorithm at obtaining good performance in a small number of iterations. Also, to strengthen our results, each experiment is replicated 5 independent times. Table 2 gathers the averaged results that were obtained. Additionally, we made wall-clock time measurements on some experiments and we discuss them in Appendix J.

6.1 Optimizing Synthetic Functions

In this section, we compare the six BO algorithms mentioned above using four synthetic functions: the 2d Six-Hump Camel (SHC), the 6d Hartmann, the 24d Powell and the 100d Rastrigin. A detailed description of the synthetic functions, as well as the complete set of figures depicting the performance of the BO algorithms can be found in Appendix H.

Figure 1(a) reports the minimal regrets of the algorithms on the Powell function, where $d = 24$ and the MFS $\bar{d} = 4$. Observe that the two decomposing algorithms, ADD-GPUCB and DEC-HBO, obtain the worst minimal regrets. This is because they infer an additive decomposition of f based on an assumption on the MFS, that is $\bar{d} \leq 3$ when actually $\bar{d} = 4$. Conversely, DuMBO, which does not make any restrictive assumption on $\bar{d}$, manages to rapidly achieve a low regret by inferring an efficient additive decomposition of f. DuMBO also outperforms SAASBO, TuRBO, LineBO and MS-UCB. Finally, Figure 1(a) shows that, when given access to the true additive decomposition of f, ADD-DuMBO achieves its lowest regret in a lower number of iterations. Similar results were obtained with the optimization of other synthetic functions (see Appendix H). Finally, note that among all the BO algorithms tested in the experiments, the two versions of DuMBO are the only ones able to properly infer and/or exploit the additive decomposition of f given its large MFS.



(a) (b) (c)

Figure 1: Performance achieved by the BO algorithms listed in Section 6 for (a) the 24d Powell synthetic function, (b) the optimization of the Shannon capacity in a WLAN and (c) the trajectory planning of a rover. The shaded areas indicate the standard error intervals.

6.2 Solving Real-World Problems

We consider three real-world problems: (a) fine-tuning some cosmological constants to maximize the likelihood of observed astronomical data (Cosmo), (b) controlling the power of devices in a Wireless Local Area Network (WLAN) to maximize its Shannon capacity (Kemperman, 1974) and (c) the trajectory planning of a rover (Rover). The problems, along with a complete set of figures depicting the performance of the tested BO algorithms, are discussed in details in Appendix I.

Figures 1(b) and 1(c) depict the performance of the BO algorithms on problems (b) and (c), where $d = 12$ and 60, respectively. Figure 1(b) shows that DuMBO is able to significantly outperform every other state-of-the-art BO algorithm. Additionally, and similarly to what was observed with Figure 1(a), Figure 1(b) suggests that having access, and being able to handle additive decompositions with large MFS, is a significant advantage. As a matter of fact, this allows to outperform BO algorithms that are unable to exploit this additional information. Figure 1(c) exhibits patterns similar to Figure 1(a): ADD-GPUCB and DEC-HBO fail to infer an adequate additive decomposition because of their restrictive MFS assumptions. In contrast, DuMBO, which does not make such an assumption on the size of the MFS, achieves the best performance along with TuRBO. Note that ADD-DuMBO is not evaluated on problem (c) since its objective function is not additive.

7 Conclusion

In this article, we showed that it is possible to completely relax the restrictive assumptions of low-MFS in the additive decomposition of f without weakening the asymptotic optimality guarantees of decomposing BO algorithms. This allows BO algorithms to simultaneously keep their no-regret property and infer a complex additive decomposition of the objective function f, or directly exploit it when it is available. To illustrate the effectiveness of such design choices, we proposed DuMBO, an asymptotically optimal decentralized BO algorithm that optimizes f using a tighter decentralized approximation of GP-UCB that requires less exploration than the previously proposed approximations. As demonstrated by Sections 5 and 6, DuMBO is a no-regret, competitive alternative to state-of-the-art BO algorithms, able to optimize complex objective functions in a small number of iterations. Compared to other decomposing algorithms, such as ADD-GPUCB and DEC-HBO, DuMBO brings a significant improvement, particularly when the decomposition of f has a large MFS with numerous factors.

For future work, we plan to extend DuMBO to batch mode (Li et al., 2016; Daxberger & Low, 2017) and to apply it to suitable technological contexts such as computer networks (Bardou & Begin, 2022), UAVs (Xie et al., 2018) or within a robots team (Chen et al., 2013).

Acknowledgments

This work was supported in part by the LABEX MILYON (ANR-10-LABX-0070) of Université de Lyon, within the program "Investissements d'Avenir" (ANR-11-IDEX- 0007) operated by the French National Research Agency (ANR).

REFERENCES

Hédy Attouch, Jérôme Bolte, Patrick Redont, and Antoine Soubeyran. Proximal alternating minimization and projection methods for nonconvex problems: An approach based on the kurdyka-łojasiewicz inequality. *Mathematics of operations research*, 35(2):438–457, 2010.

Peter Auer. Using confidence bounds for exploitation-exploration trade-offs. *J. Mach. Learn. Res.*, 3:397–422, mar 2003. ISSN 1532-4435.

Maximilian Balandat, Brian Karrer, Daniel R. Jiang, Samuel Daulton, Benjamin Letham, Andrew Gordon Wilson, and Eytan Bakshy. BoTorch: A Framework for Efficient Monte-Carlo Bayesian Optimization. In *Advances in Neural Information Processing Systems 33*, 2020. URL http://arxiv.org/abs/1910.06403.

Anthony Bardou and Thomas Begin. Inspire: Distributed bayesian optimization for improving spatial reuse in dense wlans. In *Proceedings of the 25th International ACM Conference on Modeling Analysis and Simulation of Wireless and Mobile Systems*, pp. 133–142, 2022.

James Bergstra, Daniel Yamins, and David Cox. Making a science of model search: Hyperparameter optimization in hundreds of dimensions for vision architectures. In *International conference on machine learning*, pp. 115–123. PMLR, 2013.

Mathieu Bourdeau, Xiao qiang Zhai, Elyes Nefzaoui, Xiaofeng Guo, and Patrice Chatellier. Modeling and forecasting building energy consumption: A review of data-driven techniques. *Sustainable Cities and Society*, 48:101533, 2019.

Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, and Jonathan Eckstein. *Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers*. Now Foundations and Trends, 2011.

Rick Chartrand and Brendt Wohlberg. A nonconvex admm algorithm for group sparsity with sparse groups. In *2013 IEEE international conference on acoustics, speech and signal processing*, pp. 6009–6013. IEEE, 2013.

Jie Chen, Kian Hsiang Low, and Colin Keng-Yan Tan. Gaussian process-based decentralized data fusion and active sensing for mobility-on-demand system. *arXiv preprint arXiv:1306.1491*, 2013.

C.-H. Chuang, F. Prada, A. J. Cuesta, D. J. Eisenstein, E. Kazin, N. Padmanabhan, A. G. Sanchez, X. Xu, F. Beutler, M. Manera, D. J. Schlegel, D. P. Schneider, D. H. Weinberg, J. Brinkmann, J. R. Brownstein, and D. Thomas. The clustering of galaxies in the SDSS-III baryon oscillation spectroscopic survey: single-probe measurements and the strong power of f(z) 8(z) on constraining dark energy. *Monthly Notices of the Royal Astronomical Society*, 433(4):3559–3571, jul 2013. doi: 10.1093/mnras/stt988. URL https://doi.org/10.1093%2Fmnras%2Fstt988.

Erik A. Daxberger and Bryan Kian Hsiang Low. Distributed batch Gaussian process optimization. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 951–960. PMLR, 06–11 Aug 2017.

Nicolas Durrande, David Ginsbourger, and Olivier Roustant. Additive covariance kernels for high-dimensional gaussian process modeling. In *Annales de la Faculté des sciences de Toulouse: Mathématiques*, volume 21, pp. 481–499, 2012.

David K Duvenaud, Hannes Nickisch, and Carl Rasmussen. Additive gaussian processes. *Advances in neural information processing systems*, 24, 2011.

David Eriksson and Martin Jankowiak. High-dimensional Bayesian optimization with sparse axis-aligned subspaces. In *Proceedings of the Thirty-Seventh Conference on Uncertainty in Artificial Intelligence*, volume 161 of *Proceedings of Machine Learning Research*, pp. 493–503. PMLR, 27–30 Jul 2021.

David Eriksson, Michael Pearce, Jacob Gardner, Ryan D Turner, and Matthias Poloczek. Scalable global optimization via local bayesian optimization. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

Daniel Gabay and Bertrand Mercier. A dual algorithm for the solution of nonlinear variational problems via finite element approximation. *Computers & Mathematics with Applications*, 2(1): 17–40, 1976. ISSN 0898-1221. doi: https://doi.org/10.1016/0898-1221(76)90003-1.

Jacob Gardner, Chuan Guo, Kilian Weinberger, Roman Garnett, and Roger Grosse. Discovering and exploiting additive structure for bayesian optimization. In *Artificial Intelligence and Statistics*, pp. 1311–1319. PMLR, 2017.

Rafael Gómez-Bombarelli, Jennifer N Wei, David Duvenaud, José Miguel Hernández-Lobato, Benjamín Sánchez-Lengeling, Dennis Sheberla, Jorge Aguilera-Iparraguirre, Timothy D Hirzel, Ryan P Adams, and Alán Aspuru-Guzik. Automatic chemical design using a data-driven continuous representation of molecules. *ACS central science*, 4(2):268–276, 2018.

Javier González, Joseph Longworth, David C. James, and Neil D. Lawrence. Bayesian optimization for synthetic gene design. In *NIPS Workshop on Bayesian Optimization in Academia and Industry*, 2014.

Sunil Gupta, Santu Rana, Svetha Venkatesh, et al. Trading convergence rate with computational budget in high dimensional bayesian optimization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pp. 2425–2432, 2020.

Trong Nghia Hoang, Quang Minh Hoang, Ruofei Ouyang, and Kian Hsiang Low. Decentralized high-dimensional bayesian optimization with factor graphs. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence and Thirtieth Innovative Applications of Artificial Intelligence Conference and Eighth AAAI Symposium on Educational Advances in Artificial Intelligence*, AAAI'18/IAAI'18/EAAI'18. AAAI Press, 2018. ISBN 978-1-57735-800-8.

Gregory Hornby, Al Globus, Derek Linden, and Jason Lohn. Automated antenna design with evolutionary algorithms. In *American Institute of Aeronautics and Astronautics*, 2006. doi: 10.2514/6.2006-7242.

Donald R. Jones, Matthias Schonlau, and William J. Welch. Efficient global optimization of expensive black-box functions. *Journal of Global optimization*, 13(4):455–492, 1998.

Kirthevasan Kandasamy, Jeff Schneider, and Barnabas Poczos. High dimensional bayesian optimisation and bandits via additive models. In *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pp. 295–304, Lille, France, 07–09 Jul 2015. PMLR.

JHB Kemperman. On the shannon capacity of an arbitrary channel. In *Indagationes Mathematicae (Proceedings)*, volume 77, pp. 101–115. North-Holland, 1974.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015.

Johannes Kirschner, Mojmir Mutny, Nicole Hiller, Rasmus Ischebeck, and Andreas Krause. Adaptive and safe bayesian optimization in high dimensions via one-dimensional subspaces. In *International Conference on Machine Learning*, pp. 3429–3438. PMLR, 2019.

Krzysztof Kurdyka. On gradients of functions definable in o-minimal structures. In *Annales de l'institut Fourier*, volume 48, pp. 769–783, 1998.

Rongjie Lai and Stanley Osher. A splitting method for orthogonality constrained problems. *Journal of Scientific Computing*, 58(2):431–449, 2014.

Ben Letham, Roberto Calandra, Akshara Rai, and Eytan Bakshy. Re-examining linear embeddings for high-dimensional bayesian optimization. *Advances in neural information processing systems*, 33:1546–1558, 2020.

Cheng Li, Paul Resnick, and Qiaozhu Mei. Multiple queries as bandit arms. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*, pp. 1089–1098, 2016.

Cheng Li, Sunil Gupta, Santu Rana, Vu Nguyen, Svetha Venkatesh, and Alistair Shilton. High dimensional bayesian optimization using dropout. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI-17*, pp. 2096–2102, 2017. doi: 10.24963/ijcai.2017/291.

Athanasios P Liavas and Nicholas D Sidiropoulos. Parallel algorithms for constrained tensor factorization via alternating direction method of multipliers. *IEEE Transactions on Signal Processing*, 63(20):5450–5463, 2015.

Daniel Lizotte, Tao Wang, Michael Bowling, and Dale Schuurmans. Automatic gait optimization with gaussian process regression. In *Proceedings of the 20th International Joint Conference on Artifical Intelligence*, IJCAI'07, pp. 944–949, San Francisco, CA, USA, 2007. Morgan Kaufmann Publishers Inc.

Xiaoyu Lu, Alexis Boukouvalas, and James Hensman. Additive gaussian processes revisited. In *International Conference on Machine Learning*, pp. 14358–14383. PMLR, 2022.

Jonas Mockus. Application of bayesian approach to numerical methods of global and stochastic optimization. *Journal of Global Optimization*, 4:347–365, 1994.

Riccardo Moriconi, Marc Peter Deisenroth, and KS Sesh Kumar. High-dimensional bayesian optimization using low-dimensional feature spaces. *Machine Learning*, 109(9):1925–1943, 2020.

Mojmir Mutny and Andreas Krause. Efficient high dimensional bayesian optimization with additivity and quadrature fourier features. In *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018.

Christian Robert, George Casella, Christian P Robert, and George Casella. Metropolis–hastings algorithms. *Introducing Monte Carlo Methods with R*, pp. 167–197, 2010.

Alex Rogers, Alessandro Farinelli, Ruben Stranders, and Nicholas R Jennings. Bounded approximate decentralised coordination via the max-sum algorithm. *Artificial Intelligence*, 175(2):730–759, 2011.

Lei Song, Ke Xue, Xiaobin Huang, and Chao Qian. Monte carlo tree search based variable selection for high dimensional bayesian optimization. In *Advances in Neural Information Processing Systems*, 2022.

Niranjan Srinivas, Andreas Krause, Sham M. Kakade, and Matthias W. Seeger. Information-theoretic regret bounds for gaussian process optimization in the bandit setting. *IEEE Transactions on Information Theory*, 58(5):3250–3265, 2012. doi: doi:10.1109/tit.2011.2182033.

The ns3 Project. The Network Simulator ns-3. `https://www.nsnam.org/`. Accessed: 2021-09-30.

Kai Wang, Bryan Wilder, Sze-chuan Suen, Bistra Dilkina, and Milind Tambe. Improving gp-ucb algorithm by harnessing decomposed feedback. In *Machine Learning and Knowledge Discovery in Databases*, pp. 555–569, Cham, 2020. Springer International Publishing. ISBN 978-3-030-43823-4.

Yu Wang, Wotao Yin, and Jinshan Zeng. Global convergence of admm in nonconvex nonsmooth optimization. *J. Sci. Comput.*, 78(1):29–63, jan 2019. ISSN 0885-7474. doi: 10.1007/s10915-018-0757-z.

Zi Wang and Stefanie Jegelka. Max-value entropy search for efficient Bayesian optimization. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 3627–3635. PMLR, 06–11 Aug 2017.

Zi Wang, Chengtao Li, Stefanie Jegelka, and Pushmeet Kohli. Batched high-dimensional Bayesian optimization via structural kernel learning. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 3656–3664. PMLR, 06–11 Aug 2017.

Zi Wang, Clement Gehring, Pushmeet Kohli, and Stefanie Jegelka. Batched large-scale bayesian optimization in high-dimensional spaces. In *International Conference on Artificial Intelligence and Statistics*, pp. 745–754. PMLR, 2018.

Ziyu Wang, Frank Hutter, Masrour Zoghi, David Matheson, and Nando de Freitas. Bayesian optimization in high-dimensions via random embeddings. In *Proceedings of the Twenty-Third International Joint Conference on Artificial Intelligence*, 2013.

Ziyu Wang, Frank Hutter, Masrour Zoghi, David Matheson, and Nando De Feitas. Bayesian optimization in a billion dimensions via random embeddings. *Journal of Artificial Intelligence Research*, 55:361–387, 2016.

Christopher KI Williams and Carl Edward Rasmussen. *Gaussian processes for machine learning*, volume 2. MIT press Cambridge, MA, 2006.

Lifeng Xie, Jie Xu, and Rui Zhang. Throughput maximization for uav-enabled wireless powered communication networks. *IEEE Internet of Things Journal*, 6(2):1690–1703, 2018.

Yuege Xie, Xiaoxia Wu, and Rachel Ward. Linear convergence of adaptive stochastic gradient descent. In *International conference on artificial intelligence and statistics*, pp. 1475–1485. PMLR, 2020.

Juliusz Krzysztof Ziomek and Haitham Bou Ammar. Are random decompositions all we need in high dimensional bayesian optimisation? In *International Conference on Machine Learning*, pp. 43347–43368. PMLR, 2023.

J. Zuntz, M. Paterno, E. Jennings, D. Rudd, A. Manzotti, S. Dodelson, S. Bridle, S. Sehrish, and J. Kowalkowski. CosmoSIS: Modular cosmological parameter estimation. *Astronomy and Computing*, 12:45–59, sep 2015. doi: 10.1016/j.ascom.2015.05.005. URL https://doi.org/10.1016%2Fj.ascom.2015.05.005.