Accelerating Look-ahead in Bayesian Optimization: Multilevel Monte Carlo is All You Need

Shangda Yang [*1] **Vitaly Zankin** [1] **Maximilian Balandat** [2] **Stefan Scherer** [2]
Kevin Carlberg [2] **Neil Walton** [*3 1] **Kody J. H. Law** [*2 1]

Abstract

We leverage multilevel Monte Carlo (MLMC) to improve the performance of multi-step look-ahead Bayesian optimization (BO) methods that involve nested expectations and maximizations. Often these expectations must be computed by Monte Carlo (MC). The complexity rate of naive MC degrades for nested operations, whereas MLMC is capable of achieving the canonical MC convergence rate for this type of problem, independently of dimension and without any smoothness assumptions. Our theoretical study focuses on the approximation improvements for two- and three-step look-ahead acquisition functions, but, as we discuss, the approach is generalizable in various ways, including beyond the context of BO. Our findings are verified numerically and the benefits of MLMC for BO are illustrated on several benchmark examples. Code is available at https://github.com/Shangda-Yang/MLMCBO.

1. Introduction

Bayesian optimization (BO) is a global optimization method for expensive-to-evaluate black-box functions that generally have unknown structures. In this setting, only the function value is observed for a given input value. BO works by constructing a probabilistic surrogate model, often a Gaussian process (GP), for the black-box function and then iteratively updating the surrogate model, guided by the optimization of an acquisition function, until some stopping criterion is achieved. See Frazier (2018); Shahriari et al. (2015) for a detailed review of BO.

[*]Equal contribution [1]Department of Mathematics, University of Manchester [2]Meta Platforms, Inc. [3]Durham University Business School. Correspondence to: Shangda Yang <shangda.yang@manchester.ac.uk>.

Proceedings of the 41st International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024. Copyright 2024 by the author(s).

The choice of acquisition function is crucial in the design of BO algorithms. Typical myopic acquisition functions such as upper confidence bound (Auer et al., 2002; Auer, 2002; Srinivas et al., 2009), expected improvement (EI) (Močkus, 1975; Jones et al., 1998), entropy search (Hennig & Schuler, 2012), and predictive entropy search (Hernández-Lobato et al., 2014) only consider the immediate reward of the decision. They do not trade off the depth of decision-making with the computational budget. By contrast, a multi-step acquisition function looks into the future and will optimize over the planning horizon by formulating the problem as a Markov decision process (MDP). This can substantially reduce the number of required function evaluations (González et al., 2016; Wu & Frazier, 2019; Yue & Kontar, 2020; Lee et al., 2020; Jiang et al., 2020a;b). However, the complexity of approximating such acquisition functions increases exponentially with the number of look-ahead steps for current methods, which limits their use.

The goal of this paper is to improve the computational complexity of approximating look-ahead acquisition functions.

Multi-step look-ahead acquisition functions have traditionally been optimized by nesting optimization and Monte Carlo (MC) estimations, whereas we leverage a nested Sample Average Approximation (SAA) (Balandat et al., 2020), which facilitates the application of deterministic higher-order optimization methods. Standard MC (without nested operations) requires $O(\varepsilon^{-2})$ samples to achieve mean-squared-error (MSE) of ε. For just one nested operation, the sample complexity of MC with SAA is at least $O(\varepsilon^{-3})$ and $O(\varepsilon^{-4})$ in the worst case, depending on the smoothness of the integrand. For k nested operations, the cost for MC grows exponentially in k to $O(\varepsilon^{-2(k+1)})$, leading to a *curse of dimensionality*.

This paper is the first to apply the **MLMC** framework to Bayesian Optimization, thereby *improving the performance of look-ahead methods*. MLMC for expectations of approximations, such as nested MC, works by constructing a telescoping sum of estimators from low accuracy to high accuracy (Giles, 2015). Computational complexity is reduced by performing most simulations with low-accuracy

and low-cost, and only very few high-accuracy and high-cost simulations (see Figure 1). Investigations of MLMC with nested MC appeared in Giles & Haji-Ali (2019) for efficient risk estimation inside a Heaviside function, Hironaka et al. (2020); Giles & Goda (2019) for the expected value of sample information with the feasible set being finite, Goda et al. (2020) for nested MC inside a log function, and Giles (2018) for more general problems. Hu et al. (2021) use MLMC gradient methods for computing the nested MC problems, and Beck et al. (2020) use MLMC for approximating the expected information gain in a Bayesian optimal experimental design. MLMC is still a nascent technology within Machine Learning, with results only now beginning to appear in the context of Markov chain Monte Carlo (Lyne et al., 2015; Strathmann et al., 2015; Chada et al., 2022; Cai & Adams, 2022) and stochastic gradient descent (Fujisawa & Sato, 2021; Goda et al., 2022).

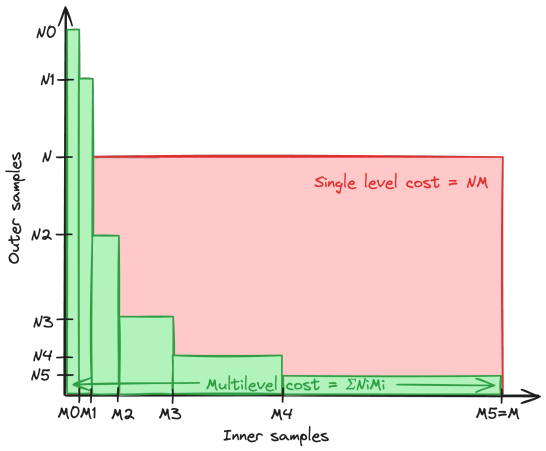


Figure 1. A graphical description of MLMC complexity (green area) improvement over nested MC (red area).

Variance reduction techniques are crucial for look-ahead acquisition functions in BO. Recent work using variance reduction techniques for myopic acquisition functions include Bogunovic et al. (2016); Balandat et al. (2020); Lee et al. (2020); Nguyen et al. (2017). In general, for smooth and relatively low-dimensional integrals, quadrature methods rule (Heath, 2018; James, 1980). However, these methods suffer from the curse of dimensionality, i.e., the *rate* of convergence degrades exponentially with dimension. Sparse grid approaches can mitigate this effect in moderate dimensions (Bungartz & Griebel, 2004). However, the only methods that deliver dimension-independent convergence rates for individual integrals are MC methods (James, 1980). For smooth functions, quasi-MC (QMC) methods can improve the canonical MC rate; however, for non-smooth functions, the rate returns to the canonical MC rate (Caflisch, 1998). Importance sampling can reduce variance, but the variance *constant* often grows exponentially in the dimension (Chatterjee & Diaconis, 2018). Furthermore, the critical dimension-dependence of the convergence rate *reappears along the nesting axis*, i.e. here the look-ahead direction, where one has expectation of function of expectation of... To our knowledge, only MLMC methods, or the closely related *multifidelity* methods (Peherstorfer et al., 2018), are capable of delivering the MC rate of convergence (independently of dimension) for nested MC approximation of non-smooth functions. Look-ahead acquisition functions in BO are indeed non-smooth, hence their approximation is a prime candidate for leveraging MLMC in Machine Learning.

We apply MLMC to compute a multi-step look-ahead acquisition function. The feasible set is infinite, and the integrand may contain nested optimization functions due to the MDP formulation. We prove that the error of the SAA with nested MC approximations can be decomposed into "variance" and "bias" terms, which allows us to apply MLMC for improved efficiency. Through its ability to improve complexity, our results suggest that MLMC is the state-of-the-art method for accelerating current Bayesian optimization frameworks.

The narrative is summarized concisely as follows:

(i) The primary bottleneck in BO is quantified by the number of black-box *function evaluations* (BB), which can be substantially reduced by look-ahead acquisition functions (AF).

(ii) These look-ahead acquisition functions are *themselves expensive to approximate*. The cost of (BB), B, implies a cost constraint for (AF), say between $(B/10, B)$.

(iii) We deliver a method which can be tuned to get designs which are either the best for a given budget or cheapest for a target accuracy, in terms of *order of complexity*, i.e. *the bigger the budget or the higher the target accuracy, the more gain there is to be had.*

This article is structured as follows. Section 2.1 illustrates the intuition and benefits of applying the MLMC method in BO. Section 2.2 introduces the general BO settings for this paper. Section 3 discusses SAA, standard MC, and nested MC. Basic MLMC and MLMC for BO is introduced in Section 4. Numerical tests are conducted in Section 5, where we illustrate the benefits of the method on several benchmark examples from Balandat et al. (2020).

2. Bayesian Optimization

Suppose we want to maximize an expensive-to-evaluate black-box function $g : \mathcal{X} \to \mathbb{R}$, for $\mathcal{X} \subseteq \mathbb{R}^d$ where d is the dimension of the input space. Here, "black box" means that we do not know the structure or the derivatives of the function. We can only observe the output of the function given an input value. Mathematically, for an input value

$x^{(i)}$, we observe

$$y^{(i)} = g(x^{(i)}) + \epsilon_i$$

where the ϵ_i are i.i.d. zero mean Gaussian random variables.

Suppose we have collected n observations $\mathcal{D}_n := \{(x^{(i)}, y^{(i)})\}_{i=1}^n$. We can solve the optimization problem sequentially using BO, which involves emulating $g(x)$ using a surrogate model, specifically a Gaussian Process, selecting $x^{(n+1)}$ given $\mathcal{D}_n$ by maximizing an acquisition function, incorporating $\{(x^{(n+1)}, y^{(n+1)})\}$ into the surrogate, and iterating. The standard setting for BO is low/moderate dimension d and small sample size n.

2.1. Brief overview of intuition and result

We illustrate the intuition and result concisely with a simple 1D toy example first, where $\mathcal{X} = [-10, 10]$ and

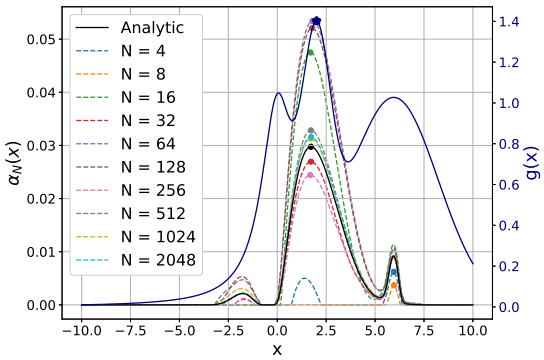
$$g(x) = e^{-(x-2)^2} + e^{-\frac{(x-6)^2}{10}} + \frac{1}{x^2+1}, \qquad (1)$$

which has one global maximum $g(x^* = 2.0087) = 1.4019$, shown in Figure 2(a).

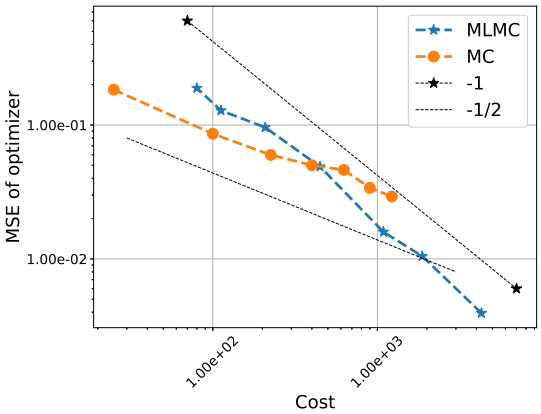
Figure 2(a) illustrates the need to approximate the acquisition functions with high accuracy, because a low accuracy function approximation is susceptible to finding the wrong mode. Multi-modality can lead to a higher inaccuracy than one would expect in a uni-modal setting, and ultimately drive the BO trajectory off course, thus wasting costly function evaluations and delaying convergence.

In the following, we introduce a technique that improves accuracy or reduces computational complexity compared to the standard MC method while balancing exploration and exploitation found with BO. In particular, we leverage the MLMC method to efficiently approximate analytically intractable look-ahead acquisition functions.

We use mean-squared error (MSE) as the error metric for acquisition function approximation. Figure 2(b) shows the MSE from approximating one acquisition function with MC and MLMC in BO. It shows that by applying MLMC, we can approximate the acquisition function's maximizer with the canonical convergence rate, i.e., Cost $\propto$ MSE^{-1} rather than the sub-optimal Cost $\propto$ MSE^{-2} rate of nested MC. Intuitively, this means that for a given computational cost, the maximizer can be approximated with higher accuracy, thus improving the performance of the whole BO algorithm. Alternatively, we can achieve the same accuracy as the MC method with less computational cost, reducing the computational cost of the BO algorithm. The benefits of MLMC are more significant if high accuracy is required or if the approximation is very costly. Later, we will see that MLMC achieves a better normalized MSE for the same computational cost for the full outer BO problem across a range of examples (Figure 3).



(a)



(b)

Figure 2. Panel (a): (i) The blue solid line is the objective function g (with axis on the right). The function has a unique global maximizer and maximum. (ii) The black solid line is the analytical EI acquisition function, and the black dot is the reference solution. (iii) The dashed colored lines are the Monte Carlo approximation of the acquisition function with varying N, and the corresponding dots are the respective maximums. Low-accuracy (small N) approximations can result in maximizer of the approximation to be far from the true maximizer. *Panel (b):* Complexity diagram of MLMC and nested MC approximation of two-step look-ahead EI with the cost measured by the number of operations. The reference solution for MSE is computed with high accuracy. Each curve is computed with 200 realizations.

2.2. Detailed Construction

We briefly recall standard results on Gaussian Process regression and introduce look-ahead acquisition functions.

2.2.1. Gaussian Process Regression

We use f to denote the GP surrogate of the objective g, which indicates $f(x^{(i)}) = y^{(i)}$ at observed point $x^{(i)}$. Initially, in BO we construct a GP prior and then update the posterior sequentially with new observations using GP regression, as we now describe.

Assuming the data are collected randomly from a multivariate normal prior distribution with a specific mean and covariance matrix, we have

$$f(x^{(1:n)}) \sim N\left(\mu_0(x^{(1:n)}), \Sigma_0(x^{(1:n)}, x^{(1:n)})\right).$$

Assuming further that $y^{(i)}$ is observed without noise (which can be easily relaxed), we can derive the posterior distribution as

$$f(x)|\mathcal{D}_n \sim N(\mu_n(x), \sigma_n^2(x)),$$

where

$$\begin{aligned}\mu_n(x) &= \mu_0(x) \\ &\quad + \Sigma_0^{(1:n)}(x)(\Sigma_0^{(1:n)})^{-1}(f(x^{(1:n)}) - \mu_0(x^{(1:n)})), \\ \sigma_n^2(x) &= \Sigma_0(x,x) - \Sigma_0^{(1:n)}(x)(\Sigma_0^{(1:n)})^{-1}\Sigma_0^{(1:n)}(x)^T,\end{aligned}$$

with $\Sigma_0^{(1:n)}(x) = \Sigma_0(x, x^{(1:n)})$, $\Sigma_0^{(1:n)} = \Sigma_0^{(1:n)}(x^{(1:n)})$.

2.2.2. Acquisition Functions

In this work, we focus on multi-step acquisition functions which are formulated in terms of the underlying Markov decision process (MDP), defined as follows (Ginsbourger & Le Riche, 2010; Lam et al., 2016; Jiang et al., 2020b; Astudillo et al., 2021; Garnett, 2023). The belief state of the MDP is the posterior $f_n(\cdot) = f(\cdot; \mathcal{D}_n) \in \mathcal{S}$, parameterized by mean $\mu_n(\cdot)$ and covariance kernel $\Sigma_n(\cdot,\cdot)$, where $\mathcal{S}$ is the space of state. The action is where we take observations $x \in \mathcal{X}$, and the stage-wise reward $r(f,x)$ characterizes the acquisition function:

$$\alpha(x; \mathcal{D}_n) := \mathbb{E}[r(f,x)|\mathcal{D}_n].$$

A simple reward function may look like $r(f_n(x))$, in which case $\mathbb{E}[r(f,x)|\mathcal{D}_n] = \mathbb{E}[r(f_n(x))]$. The input action of the MDP and its output updates the GP with a new observation $(x^{(n+1)}, y^{(n+1)})$ as described above, i.e. the dynamics of the MDP are given by

$$\begin{aligned}\mathcal{F} : \mathcal{S} \times \mathcal{X} \times \mathcal{Y} &\rightarrow \mathcal{S} \\ (f_n, x^{(n+1)}, y^{(n+1)}) &\mapsto f_{n+1}(\cdot) = f(\cdot; \mathcal{D}_{n+1}).\end{aligned}$$

The BO algorithm provides a greedy stage-wise solution to the MDP by repeating the following steps until the computational resources are exhausted: 1) starting from an initial state (a GP f_{n-1} with initial observations $\mathcal{D}_{n-1}$); 2) maximize the expected multi-step reward to determine the action x_n at which we should take an observation; 3) observe the environment, y_n; 4) update the state with the new observation $f_{n-1} \mapsto f_n$ (see Algorithm 1). Letting $\mathcal{D}$ denote all observations at the current state, the multi-step look-ahead acquisition functions are

$$\alpha_0(x; \mathcal{D}) := \mathbb{E}_{f(\cdot;\mathcal{D})}[r(f,x)]$$

Algorithm 1 Bayesian optimization

Inputs: $\mathcal{D}_0$, ϵ, Black Box objective "g", stopping criterion.
Outputs: $\hat{x}^* \approx \mathsf{argmin}_x g(x)$.
$n = 0$.
while stopping criterion is not met **do**
  Compute Single Design (Alg. 2) $\mathcal{D}_n \mapsto x_{n+1}$;
  Evaluate Black Box $y_{n+1} = g(x_{n+1})$;
  Augment $\mathcal{D}_{n+1} = \mathcal{D}_n \cup \{x_{n+1}, y_{n+1}\}$;
  $n \mapsto n+1$.
end while
$\hat{x}^* = \mathsf{argmax}_{\mathcal{D}_n} g(x)$.

$$\begin{aligned}\alpha_1(x;\mathcal{D}) &:= \mathbb{E}_{f(\cdot;\mathcal{D})}\left[r(f,x) + \max_{x_1} \mathbb{E}_{f(\cdot;\mathcal{D}_1(x))}[r(f,x_1)]\right] \\ \alpha_2(x;\mathcal{D}) &:= \mathbb{E}_{f(\cdot;\mathcal{D})}\Bigg[r(f,x) + \max_{x_1} \mathbb{E}_{f(\cdot;\mathcal{D}_1(x))}\Big[r(f,x_1) \\ &\qquad + \max_{x_2} \mathbb{E}_{f(\cdot;\mathcal{D}_2(x,x_1))}[r(f,x_2)]\Big]\Bigg] \\ &\vdots \end{aligned} \tag{2}$$

where $\mathcal{D}_1(x) = \mathcal{D} \cup \{(x, f(x;\mathcal{D}))\}$, $\mathcal{D}_2(x,x_1) = \mathcal{D}_1(x) \cup \{(x_1, f(x_1;\mathcal{D}_1(x)))\}, \ldots,$ and we use $\mathbb{E}_{f(\cdot;\mathcal{D})}$ to denote that the expectations above are taken over the Gaussian process f given data $\mathcal{D}$. Here, the α_0 is the standard one-step look-ahead acquisition function and α_k denotes the $(k+1)$-step look-ahead acquisition function.

Many reward functions lead to analytically intractable acquisition functions, so the 2-step look-ahead acquisition function α_1 requires nested MC approximation as follows

$$\begin{aligned}\alpha_{1,N,M}(x;\mathcal{D}) = \frac{1}{N}\sum_{i=1}^{N}\Bigg[&r(f^i(x;\mathcal{D})) \\ &+ \left(\max_{x_1^i} \frac{1}{M}\sum_{j=1}^{M} r(f^{ij}(x_1^i; \mathcal{D}_1^i(x)))\right)\Bigg].\end{aligned} \tag{3}$$

Additional details are provided in Appendix A. In general, n-step look-ahead acquisition functions can be approximated with n-nested MC approximations. In the present work, we will focus on deriving theories and showing numerical results for **single-nested MC**, which is amenable to MLMC. This includes 2-step look-ahead q-EI (Wang et al., 2020) and 3-step look-ahead EI. As we will see, the 2-step look ahead formulation is sufficient to gain substantial benefits in terms of MSE when compared with SOTA methods, which are either single MC, e.g. 1-step qEI or 2-step EI (Wu & Frazier, 2019; Ginsbourger & Le Riche, 2010; Chevalier & Ginsbourger, 2013; Renganathan et al., 2020; Balandat

et al., 2020), or rely on very rough approximations for more than 2 steps (Lam et al., 2016; Jiang et al., 2020b; Astudillo et al., 2021; Garnett, 2023). There is an extension of the MLMC technology that is applicable to the general n-nested case, which is called multi-index MC (Haji-Ali et al., 2016; Jasra et al., 2018; 2023), and that will be the topic of future investigation.

Numerous reward functions can be used in BO (see e.g., Frazier (2018); Balandat et al. (2020)), but for the present exposition, we focus on two of the most ubiquitous ones.

Expected improvement (EI) (Močkus, 1975) is obtained by setting

$$r(f, x) := (f(x; \mathcal{D}) - f^*(\mathcal{D}))_+,$$

with $(\cdot)_+ = \max\{0, \cdot\}$ and $f^*(\mathcal{D}) = \max_{(x, f(x)) \in \mathcal{D}} f(x)$. Expected improvement can be computed analytically as

$$\begin{aligned}\mathsf{EI}(x|\mathcal{D}) &:= \mathbb{E}\left[(f(x; \mathcal{D}) - f^*(\mathcal{D}))_+ | \mathcal{D}\right] \\ &= \sigma(x) z(x) \Phi\left(z(x)\right) + \sigma(x) \phi\left(z(x)\right)\end{aligned}$$

where $z(x) := (\mu(x) - f^*(\mathcal{D}))/\sigma(x)$ and where $\phi(\cdot)$ and $\Phi(\cdot)$ are the PDF and CDF of the standard normal distribution, respectively.

q-Expected improvement (qEI) (Wang et al., 2020) maximizes over a batch $x = (x_1, \ldots, x_q)$ of $q > 1$ points jointly, i.e., $r(f, x) = \max_{j=1,\ldots,q}(f(x_j; \mathcal{D}) - f^*(\mathcal{D}))_+$ so that

$$\mathsf{qEI}(x|\mathcal{D}) = \mathbb{E}\left[\max_{j=1,\ldots,q} (f(x_j; \mathcal{D}) - f^*(\mathcal{D}))_+ | \mathcal{D}\right]. \quad (4)$$

This is analytically intractable but can be approximated by MC estimation

$$\mathsf{qEI}(x|\mathcal{D}) \approx \frac{1}{N} \sum_{i=1}^{N} \max_{j=1,\ldots,q} (f^i(x_j; \mathcal{D}) - f^*(\mathcal{D}))_+.$$

Examples of two-step look-ahead EI, two-step look-ahead qEI, and three-step look-ahead EI are explicitly given in Appendix A.1. Noting that the latter two both result in a single-nested MC objective, we henceforth denote it by $\alpha(x; \mathcal{D})$ without subscripts, and the corresponding optimizers by x^*.

3. Sample Average Approximation

Sample average approximation (SAA) (Kleywegt et al., 2002; Balandat et al., 2020) is constructed from i.i.d. samples as

$$x_N := \operatorname*{argmax}_{x \in \mathcal{X}} \frac{1}{N} \sum_{i=1}^{N} r(f^i(x; \mathcal{D})) \quad (5)$$

$$\approx \operatorname*{argmax}_{x \in \mathcal{X}} \alpha(x; \mathcal{D}) =: x^*. \quad (6)$$

Below and hereafter, $\|\cdot\|$ denotes the Euclidean distance. Proposition 1 below guarantees the rate of convergence of the maximizer under standard assumptions: essentially, $x \mapsto \alpha(x; \mathcal{D})$ has a compact domain with Lipschitz derivatives and is locally quadratic at its optimum (see Appendix B for a more formal statement and further discussion).

Proposition 1 (Theorem 12 of (Kim et al., 2015))**.** *Given a unique optimizer and Assumption 1*

$$\mathbb{E}[\|x_N - x^*\|] = O(N^{-1/2}).$$

In practice, we assume that our optimization algorithm can find the global optimizer, and we attempt to achieve this with a committee of multiple initializations of a local optimization algorithm. For given realizations, the function $\alpha_N(\cdot)$ is deterministic, and we can apply deterministic optimization algorithms such as L-BFGS with multi-start initialization. We now show the decomposition of approximation error of AF (3), omitting the subscript 1 henceforth.

Proposition 2. *If Assumption 1 holds and* $\forall x, x_1 \in \mathcal{X}, \mathsf{Var}\left(r(f, x) + \mathbb{E}_{f(\cdot; \mathcal{D}_1(x))}\left[r(f, x_1)\right]\right) \leq \sigma_{\max}$ *and* $\mathsf{Var}(r(f, x_1)) \leq \sigma_{\max}$, *for some* $\sigma_{\max} > 0$, *then the following holds, for some* $C > 0$

$$\sup_{x \in \mathcal{X}} \mathbb{E}\left[|\alpha_{N,M}(x; \mathcal{D}) - \alpha(x; \mathcal{D})|^2\right] \leq C\left(\frac{1}{N} + \frac{1}{M}\right). \quad (7)$$

The proof is provided in Appendix C. The decomposition for three-step look-ahead EI (16) follows similarly. The rate with respect to the outer and inner MC is numerically verified in Appendix H.1.

Given the decomposition of MSE as $O(1/N + 1/M)$, if we want to achieve an MSE of ε^2 for $\varepsilon > 0$, then we require $N = O(\varepsilon^{-2})$ and $M = O(\varepsilon^{-2})$, hence a computational complexity of $MN = O(\varepsilon^{-4})$. If there is no nested MC, such as α_0 from (2), the computational complexity to achieve an MSE of ε^2 is $O(\varepsilon^{-2})$, which is called the canonical rate of MC. We can leverage MLMC to reduce the complexity of α_1 from sub-canonical $O(\varepsilon^{-4})$ if (3) is used, to canonical $O(\varepsilon^{-2})$.

4. Multilevel Monte Carlo

The idea behind MLMC is relatively simple (Giles, 2015). We first introduce the general concept and then discuss the specific construction for BO. Suppose we want to estimate the expectation of some quantity of interest φ, which must itself be approximated. Here the quantity of interest is the given look-ahead stage-wise reward and the approximation refers to the inner MC. We leverage a sequence of approximations to φ with increasing accuracy and cost, where an initial estimate using a large number of samples with low accuracy and cost is corrected with progressively fewer-sample estimates of *increments* with higher accuracy and cost. We make this precise in the following.

Denote the sequence of approximations with increasing accuracy and cost over levels l by $\varphi_{M_0}, \varphi_{M_1}, \ldots, \varphi_{M_L}$. Let $M_l = 2^l$. The MLMC approximation to $\mathbb{E}[\varphi]$ follows from the telescoping sum identity

$$\mathbb{E}[\varphi] \approx \mathbb{E}[\varphi_{M_L}] = \sum_{l=0}^{L} \mathbb{E}[\Delta\varphi_{M_l}],$$

where $\Delta\varphi_{M_l} = \varphi_{M_l} - \varphi_{M_{l-1}}$ is the increment and $\varphi_{M_{-1}} = 0$. Now we let

$$Z_{N_l,M_l} = \frac{1}{N_l}\sum_{i=1}^{N_l}(\varphi^i_{M_l} - \varphi^i_{M_{l-1}})$$

be an unbiased MC estimator of $\mathbb{E}[\Delta\varphi_{M_l}]$ for $l = 0, 1, ..., L$, where $\varphi^i_{M_l}$ and $\varphi^i_{M_{l-1}}$ are computed using the same input samples, i.i.d for each i and l. If $|\mathbb{E}[\varphi_{M_l} - \varphi]|$ and the variance $\mathsf{Var}(\Delta\varphi_{M_l})$ decay exponentially as l increases with specific rates, we can achieve computational benefits over the standard MC method. The following theorem is the standard MLMC theorem similar to (Giles, 2015).

Proposition 3. *Assume there exists positive constants s, w, γ with $s \geq \frac{1}{2}\min\{\beta, \gamma\}$ and C (where C may vary line to line) such that*

- $B_l := |\mathbb{E}[\varphi_{M_l} - \varphi]| \leq CM_l^{-s} = C2^{-sl}$
- $V_l := \mathsf{Var}(\Delta\varphi_{M_l}) \leq CM_l^{-\beta} = C2^{-\beta l}$
- $\mathsf{COST}_l := \mathsf{COST}(\Delta\varphi_{M_l}) \leq CM_l^{\gamma} = C2^{\gamma l}$,

then there are values L and N_l such that the multilevel estimator

$$Z = \sum_{l=0}^{L} Z_{N_l,M_l}$$

can be approximated with an accuracy ε^2 measured by the mean square error (MSE) with a computational complexity (COST) for which

$$\mathbb{E}[\mathsf{COST}] \leq \begin{cases} C\varepsilon^{-2}, & \beta > \gamma, \\ C\varepsilon^{-2}(\log\varepsilon)^2, & \beta = \gamma, \\ C\varepsilon^{-2-(\gamma-\beta)/s}, & \beta < \gamma. \end{cases}$$

MLMC requires a regularity assumption on bias and incremental variance convergence. For the canonical situation, we require the variance to decay faster than the cost increase such that the cost at the lower level dominates; for the borderline situation, the cost is equally spread among levels; for the worst situation, the cost at the finer level dominates that corresponds to the MC rate. A detailed proof can be found in Giles (2008). The main components are provided in Lemma 3 in Appendix G.

4.1. Multilevel formulation of the maximizer of the acquisition function

We construct a multilevel estimator for the two-step look-ahead acquisition function

$$\alpha(x;\mathcal{D}) := \mathbb{E}_{f(\cdot;\mathcal{D})}\left[r(f,x) + \max_{x_1}\mathbb{E}_{f(\cdot;\mathcal{D}_1(x))}\left[r(f,x_1)\right]\right]. \tag{8}$$

Since our target is to find the next observation point guided by the acquisition function, we formulate the multilevel estimator for the maximizer as

$$x^*_{\mathsf{ML}} := \sum_{l=0}^{L} z_{N_l,M_l} - z_{N_l,M_{l-1}} = \sum_{l=0}^{L}\Delta z_{N_l,M_l}\,, \tag{9}$$

where

$$z_{N_l,M_l} = \operatorname*{argmax}_{x\in\mathcal{X}} \frac{1}{N_l}\sum_{i=1}^{N_l}\Bigg[r(f^i(x;\mathcal{D})) + \left(\max_{x_1^i}\frac{1}{M_l}\sum_{j=1}^{M_l} r(f^{ij}(x_1^i;\mathcal{D}_1^i(x)))\right)\Bigg], \tag{10}$$

$$z_{N_l,M_{l-1}} = \operatorname*{argmax}_{x\in\mathcal{X}} \frac{1}{N_l}\sum_{i=1}^{N_l}\Bigg[r(f^i(x;\mathcal{D})) + \left(\max_{x_1^i}\frac{1}{M_{l-1}}\sum_{j=1}^{M_{l-1}} r(f^{ij}(x_1^i;\mathcal{D}_1^i(x)))\right)\Bigg], \tag{11}$$

where $z_{N_0,M_{-1}} \equiv 0$. To leverage MLMC, we first simulate M_l samples of $f(x_1;\mathcal{D}_1(x))$ and construct the z_{N_l,M_l} and then subsample M_{l-1} samples from the M_l samples to construct $z_{N_l,M_{l-1}}$. We remark that the terms $\Delta z_{N_l,M_l}(x)$ for $l = 0, 1, ..., L$ are mutually independent. Algorithm 2 shows a general implementation of the MLMC approximation, where Φ_l^l and Φ_l^{l-1} denote the fine and coarse approximation of AF at level l and Φ_0 is the single AF at level 0.

Algorithm 2 Single Design with MLMC acquisition

Inputs: $\mathcal{D}_n$, ϵ
Outputs: x_{n+1}
Define L, $N_0, \ldots, N_L$, and $M_0, \ldots, M_L$ (as in Thm 1)
Compute $z_0 = \mathsf{argmin}_z \Phi_0(z)$
for $l = 1, \ldots, L$ **do**
  $z_l^l = \mathsf{argmin}_z \Phi_l^l(z)$ and $z_l^{l-1} = \mathsf{argmin}_z \Phi_l^{l-1}(z)$;
  $\Delta_l = z_l^l - z_l^{l-1}$.
end for
$x_{n+1} = z_0 + \sum_{l=1}^{L}\Delta_l$.

An *antithetic coupling* approach can improve the rate of convergence. It involves replacing the coarse estimator (11)

with

$$z^A_{N_l,M_{l-1}} = \operatorname*{argmax}_x \frac{1}{N_l}\sum_{i=1}^{N_l}\Bigg[r(f^i(x;\mathcal{D})) + \frac{1}{2}\Bigg(\max_{x_1^i}\frac{1}{M_{l-1}}\sum_{j=1}^{M_{l-1}} r(f^{ij}(x_1^i;\mathcal{D}_1^i(x))) + \max_{x_1^i}\frac{1}{M_{l-1}}\sum_{j=M_{l-1}+1}^{M_l} r(f^{ij}(x_1^i;\mathcal{D}_1^i(x)))\Bigg)\Bigg], \tag{12}$$

where $z^A_{N_0,M_{-1}} \equiv 0$ and the samples used for $z^A_{N_l,M_{l-1}}$ are taken by splitting the M_l samples from z_{N_l,M_l}. In the coarser estimator $z^A_{N_l,M_{l-1}}$, one averages the maxima associated with two batches of M_{l-1} samples rather than maximizing the average. It can easily be shown using Taylor expansion for smooth functions (Blanchet et al., 2015) that this can double the rate, but it can also yield improvement even for non-smooth functions (Giles, 2018; Giles & Goda, 2019), as in our situation. The multilevel formulation for the three-step look-ahead AF is given in Appendix D.2.

The complexity is determined by the cost to evaluate the terms above, which is $O(d(\sum_{l=0}^{L} N_l(M_l+1))$ for both the antithetic and regular cases. Without loss of generality, we assume that a fast super-linear solver is available, so computing (10) can be done with a linear cost in the evaluation of the objective function, which corresponds to $\gamma = 1$.

Note the maximizer increments of (9) are *not unbiased*, as is common in Bayesian MLMC (Beskos et al., 2017). In the present work, we provide theory for a simplified estimator, and introduce strong assumptions for (9). The more subtle and challenging complete theory is deferred to future work.

4.2. MLMC approximation of the acquisition function

We first present a theoretical analysis of a procedure for MLMC BO using the easier-to-analyse AF results. We combine MLMC ideas with SAA techniques on AF instead of optimizers, i.e. we directly construct an MLMC estimator for the AF.

Similarly to (10) and (11), we construct $\Delta\alpha_{N_l,M_l}(x) = \alpha_{N_l,M_l}(x) - \alpha_{N_l,M_{l-1}}(x)$, with $\alpha_{N_0,M_{-1}}(x) \equiv 0$, as incremental approximations of (8). From this, we form the estimator

$$\alpha_{\mathsf{ML}}(x) := \sum_{l=0}^{L}\Delta\alpha_{N_l,M_l}(x)\,.$$

We now argue that α_{ML} is a computationally efficient proxy for the intractable limit objective $\alpha(x;\mathcal{D})$, which leads to computational benefits for approximating optimizer x^* (see Appendix D.1 for some further details).

Before proceeding with the theoretical analysis, we note again that the above estimator $\alpha_{\mathsf{ML}}(x)$ and the corresponding $\operatorname{argmax}_{x\in\mathcal{X}}\alpha_{\mathsf{ML}}(x)$ is just one estimator that leverages the MLMC methodology. It is particularly tractable from a theoretical perspective due to unbiased increments. We require an additional technical Assumption 2, which holds for GP with standard kernels. This and supporting Lemmas are presented in Appendix F.

Theorem 1 (Q-Function Convergence)**.** *Suppose Assumptions 1 and 2 hold. Let $M_l = 2^l$. For $x \in \mathcal{X}$, the MLMC estimator $\alpha_{\mathsf{ML}}(x)$ is such that*

$$\mathbb{E}\Big[\sup_x |\alpha_{\mathsf{ML}}(x) - \alpha(x;\mathcal{D})|^2\Big] \le C(L+1)\left(\frac{V_0}{N_0} + \sum_{l=1}^{L}\frac{1}{N_l M_l}\right) + \frac{C}{M_L}\,,$$

$$\mathbb{E}\Big[\sup_x |\nabla\alpha_{\mathsf{ML}}(x) - \nabla\alpha(x;\mathcal{D})|^2\Big] \le C(L+1)\left(\frac{V_0}{N_0} + \sum_{l=1}^{L}\frac{1}{N_l M_l}\right) + \frac{C}{M_L}\,,$$

for some constant C. Consequently taking[1] $L = O(|\log\varepsilon|)$,

$$N_0 = \varepsilon^{-2}K_L\left(\frac{V_0}{M_0}\right)^{1/2} \quad \textit{and} \quad N_l = \varepsilon^{-2}K_L\frac{1}{M_l}, \tag{13}$$

for $l \ge 1$ with $K_L = (V_0M_0)^{1/2} + L$, the MLMC estimator $\alpha_{\mathsf{ML}}(x)$ gives the following error estimate, for $x \in \mathcal{X}$,

$$\mathbb{E}\big[\sup_x |\alpha_{\mathsf{ML}}(x) - \alpha(x;\mathcal{D})|^2\big] \le C\varepsilon^2$$

for a sample complexity of $O(\varepsilon^{-2}|\log\varepsilon|^3)$.

The proof of Theorem 1 is in Appendix G.2. Appendix E provides corollaries for the value function and the maximizer. Here, we consider the inner sup with a loose bound; up to logarithmic terms, we achieve the same rate as a standard MC stimulation (without nested summations and maximizations).

4.3. Main Result

We now establish the main MLMC complexity result, which requires yet another additional Assumption 3, given in Appendix F. This final assumption essentially states that the strong rate of convergence of increments of the acquisition function $\Delta\alpha_{N_l,M_l}$ implies the corresponding rate for the maximizer $\Delta z_{N_l,M_l}$. As indicated earlier, we expect the result to hold for the maximizer, however the proof of this is deferred to future work.

[1] The ceiling function provides integer values of L, N_l, and M_l.

Theorem 2. *Suppose Assumptions 1, 2 and 3 hold. Let $M_l = 2^l$. Then one can choose $L, \{N_l\}_{l=0}^{L}$, as in Theorem 1, such that*

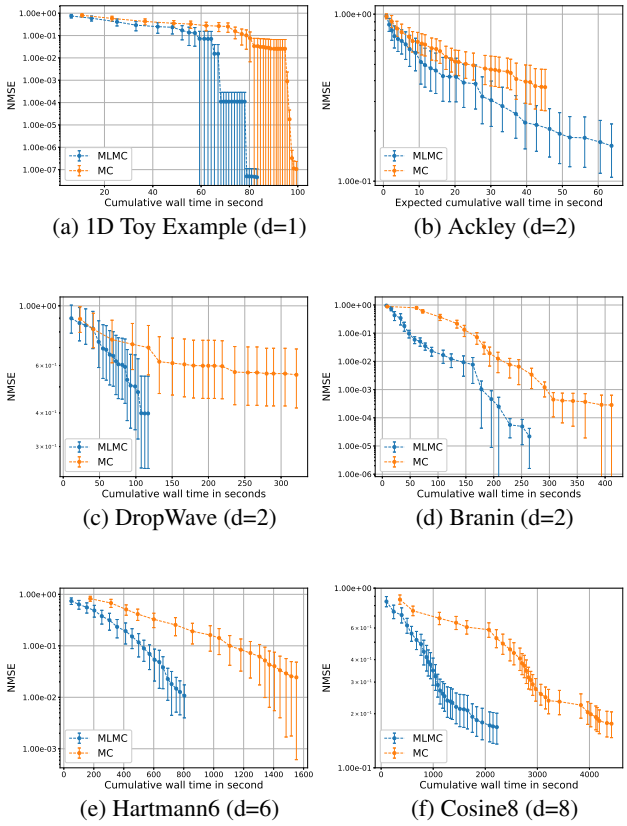
$$\mathbb{E}[\|x^*_{\mathsf{ML}} - x^*\|^2] \leq C\varepsilon^2 \,,$$

for a complexity of $O(\varepsilon^{-2}(\log \varepsilon)^2)$.

The proof is given in Appendix G.5. We verify $\beta = 1$ numerically with the difference of optimizer in Appendix H.3. This corresponds to the borderline scenario of Proposition 3 with $\beta = \gamma = 1$, which still yields an improvement compared with the standard MC. We have the following remark, which can further improve the computational complexity.

Remark 1. *The antithetic construction (12) gives $\beta \approx 1.5$ leading to an overall canonical computational complexity $O(\varepsilon^{-2})$. This is similar to the work by Giles & Goda (2019), but here, we are interested in constructing a multilevel estimator of the optimizer. The rates are numerically verified with the 1D toy example 1. See Appendix H.3 for the verification of $\beta \approx 1.5$ and Figure 2(b) for the complexity.*

5. Numerical Results



(a) 1D Toy Example (d=1) (b) Ackley (d=2)

(c) DropWave (d=2) (d) Branin (d=2)

(e) Hartmann6 (d=6) (f) Cosine8 (d=8)

Figure 3. Convergence of the BO algorithm with respect to the cumulative wall time in seconds, with error bars (computed with 20 realizations). The Matérn kernel is applied. The initial BO run starts with $2 \times d$ observations.

In this section, we run the whole BO algorithm on various synthetic functions from BoTorch (Balandat et al., 2020) and show the benefits of using MLMC (9) with the antithetic approach over standard MC (3) in AF approximation. We apply the normalized MSE, defined as

$$\text{NMSE} = \mathbb{E}\left[\frac{\|\max_{\mathcal{D}_k} g(x) - g(x^*)\|^2}{\|\max_{\mathcal{D}_0} g(x) - g(x^*)\|^2}\right], \qquad (14)$$

where $\mathcal{D}_0$ and $\mathcal{D}_k$ are the initial observation set and the observation set after k iterations, as the error metric. It measures the algorithm's expected relative improvement over the initial values. It normalizes the value to $[0, 1]$, which can make the performance on different problems with different scales of domain comparable and remains the decaying feature of error. The AF we applied is the two-step look-ahead 1-EI + 2-EI defined in (19). According to Figure 3, in expectation, compared with the MC method, the MLMC method can achieve the same NMSE with less computational cost to find the optimum of test functions. We primarily focus on the method from Section 4.1 as our initial study. Comparisons between the multilevel optimizer method from Section 4.1 and the multilevel function method from Section 4.2 are performed in Section H.4.

The number of samples in MC is $N = M = 1/\varepsilon^2$. The number of samples in MLMC for N_l and M_l, related to ε, is chosen according to the formula in Theorem 1. For comparison, the ε is set to be the same for any given problem. Here, we use $\varepsilon = 0.2$ for Figure 3(a), 3(b), 3(c), 3(d) and 3(f), and $\varepsilon = 0.15$ for Figure 3(e). It is useful to note that MLMC requires balancing the "variance" and "bias" to achieve the canonical rate. This requires careful tuning in selecting the finest level and the number of samples at each level. From the previously derived formulas, the level and the number of samples depend on the variance of the increments. In practice, since we need to approximate the AF interactively, it is not realistic nor efficient to compute the variance every time before we evaluate the functions. A practical way to deal with that is to use $C2^{-\beta l}$, which is due to the proportionality of the variance, with the constant C empirically. It turns out that the worst situation is that MLMC degenerates to MC, and we have an MC rate of convergence in a single AF approximation. Overall, MLMC with this pragmatic choice performs much better in practice.

Because L-BFGS is a local optimization method, we need to run it with multiple initial conditions to get an approximation of the global maximizer. An additional complication in the present context is that we need to match the coarse and fine maximizers at each level l in order for (9) to deliver a good approximation. The simplest way to achieve this is by matching all levels with the best one from level 0 (or L), which is the strategy adopted in the present work. However, we want the estimator (9) which delivers the maximum value, rather than being anchored to any individual

Function Name		PI	EI	UCB	PES	GLASSES	R-4-9	R-4-10	R-5-9	R-5-10	Ours
Branin-Hoo	Mean	0.847	0.818	0.848	0.861	0.846	0.904	0.898	0.887	0.903	**0.992**
	Median	0.922	0.909	0.910	0.983	0.909	0.959	0.943	0.921	0.950	**0.997**
Goldstein-Price	Mean	0.873	0.866	0.733	0.819	0.782	0.895	0.784	0.861	0.743	**0.987**
	Median	0.983	0.981	0.899	0.987	0.919	0.991	0.985	0.989	0.928	**0.998**
Griewank	Mean	0.827	0.884	0.913	0.972	1.0[2]	0.882	0.885	0.930	0.867	**0.973**
	Median	0.904	0.953	0.970	**0.987**	1.0[2]	0.967	0.962	0.960	0.954	0.975
6-hump Camel	Mean	0.850	0.887	0.817	0.664	0.776	0.860	0.825	0.793	0.803	**0.942**
	Median	0.893	0.970	0.915	0.801	0.941	0.926	0.900	0.941	0.907	**0.979**

Table 1. [Reproduced with permission from (Lam et al., 2016)]. Performance of our 2-step look-ahead acquisition function (1EI + 2EI) on benchmark functions. The rollout algorithms of (Lam et al., 2016) are denoted R-.... The other algorithms are commonly used and all are myopic except GLASSES. The error metric is GAP $= (g(x_0) - \max_{\mathcal{D}_k} g(x))/(g(x_0) - g(x^*))$ after a fixed budget of $k = 15$ function evaluations, and statistics are from 40 realizations randomly initialized with a single function evaluation.

level. This can be achieved with backward (or forward) matching, where we essentially prune a tree whose branches start at level 0 (or L) and branch forward (or backward) in levels. This approach is more robust but also slightly more complicated and costly, and we have found that the former strategy is satisfactory in practice. The approaches are illustrated in Figures 10(a) and 10(b) in Appendix H.7. A global strategy which explores all combinations of the local optimizers of the $2L$ intermediate problems would incur a non-trivial cost of $O(\varepsilon^{-2})$.

In Table 1 we replicate Table 2 of Lam et al. (2016) and include a comparison with our 2-step look-ahead AF (1EI + 2EI) computed using MLMC. We observe that the 2-step look-ahead AF (1EI + 2EI) is the best in 7 out of 8 cases and the second best for the median of the Griewank function. Wu & Frazier (2019); Jiang et al. (2020a;b) provide further comparisons of look-ahead methods with myopic competitors, illustrating the advantage of look-ahead in general. Further discussion is provided in Appendix H.5.

Numerics related to 2-step look-ahead 2-EI (18) are performed in Appendix H.6.

6. Future Directions

Evaluating multi-step look-ahead acquisition functions is computationally expensive. In this work we introduced MLMC into this area and showed the benefits of using MLMC over MC. Further theoretical analysis remains to be carried out. This paper mainly focuses on the two- and three-step look-ahead settings, but it can be extended to additional steps. MLMC may not be able to achieve the canonical rate with an increasing number of look-ahead step due to the curse of dimensionality, which leads to a violation of regularity assumptions. Under mixed regularity assumptions, this can be solved by using multi-index Monte Carlo (MIMC) (Haji-Ali et al., 2016; Jasra et al., 2023), which works by considering increments of increments (of increments ...) over a grid of levels/indices instead of a simple increment at each level. In this case, the regularity conditions required for canonical complexity are only required on each nested MC individually.

Randomized MLMC (Rhee & Glynn, 2012; 2015) and randomized MIMC can eliminate the bias and costly tuning ordinarily required to balance with variance, at the cost of a potentially larger constant (Liang et al., 2023). It may be possible to further reduce computational complexity to MSE$^{-1/2}$ or $O(\varepsilon^{-1})$ for small $\varepsilon > 0$ by introducing multi-level quasi-Monte Carlo or multi-index quasi-Monte Carlo developed based on MLMC or MIMC and quasi-Monte Carlo (Caflisch, 1998; L'Ecuyer et al., 2009), which generates a sequence of low discrepancy samples for variance reduction. However, the improved complexity relies on the smoothness of the integrand, and performance degenerates to that of standard MC, $O(\varepsilon^{-2})$, in the non-smooth case.

[2]According to (Lam et al., 2016), the GAP = 1 results of GLASSES arise from an arbitrary choice by one optimizer to evaluate the origin, which coincides with the minimizer of the Griewank function. Those results are excluded from the analysis.

Impact Statement

This paper presents work whose goal is to advance the field of Machine Learning. There are many potential societal consequences of our work, none which we feel must be specifically highlighted here.

References

Ament, S., Daulton, S., Eriksson, D., Balandat, M., and Bakshy, E. Unexpected improvements to expected improvement for Bayesian optimization. *Advances in Neural Information Processing Systems*, 36, 2024.

Astudillo, R., Jiang, D., Balandat, M., Bakshy, E., and Frazier, P. Multi-step budgeted Bayesian optimization with unknown evaluation costs. *Advances in Neural Information Processing Systems*, 34:20197–20209, 2021.

Auer, P. Using confidence bounds for exploitation-exploration trade-offs. *Journal of Machine Learning Research*, 3(Nov):397–422, 2002.

Auer, P., Cesa-Bianchi, N., and Fischer, P. Finite-time analysis of the multiarmed bandit problem. *Machine learning*, 47:235–256, 2002.

Balandat, M., Karrer, B., Jiang, D., Daulton, S., Letham, B., Wilson, A. G., and Bakshy, E. BoTorch: A framework for efficient Monte-Carlo Bayesian optimization. *Advances in neural information processing systems*, 33: 21524–21538, 2020.

Beck, J., Mansour Dia, B., Espath, L., and Tempone, R. Multilevel double loop Monte Carlo and stochastic collocation methods with importance sampling for Bayesian optimal experimental design. *International Journal for Numerical Methods in Engineering*, 121(15):3482–3503, 2020.

Beskos, A., Jasra, A., Law, K., Tempone, R., and Zhou, Y. Multilevel sequential Monte Carlo samplers. *Stochastic Processes and their Applications*, 127(5):1417–1440, 2017.

Blanchet, J. H., Chen, N., and Glynn, P. W. Unbiased Monte Carlo computation of smooth functions of expectations via Taylor expansions. In *2015 Winter Simulation Conference (WSC)*, pp. 360–367. IEEE, 2015.

Bogunovic, I., Scarlett, J., Krause, A., and Cevher, V. Truncated variance reduction: A unified approach to Bayesian optimization and level-set estimation. *Advances in neural information processing systems*, 29, 2016.

Borovkov, A. A. *Probability theory*. CRC Press, 1999.

Bungartz, H.-J. and Griebel, M. Sparse grids. *Acta numerica*, 13:147–269, 2004.

Caflisch, R. E. Monte Carlo and quasi-Monte Carlo methods. *Acta numerica*, 7:1–49, 1998.

Cai, D. and Adams, R. P. Multi-fidelity Monte Carlo: A pseudo-marginal approach. *Advances in Neural Information Processing Systems*, 35:21654–21667, 2022.

Chada, N. K., Jasra, A., Law, K. J., and Singh, S. S. Multilevel Bayesian deep neural networks. *arXiv preprint arXiv:2203.12961*, 2022.

Chatterjee, S. and Diaconis, P. The sample size required in importance sampling. *The Annals of Applied Probability*, 28(2):1099–1135, 2018.

Chevalier, C. and Ginsbourger, D. Fast computation of the multi-points expected improvement with applications in batch selection. In *International conference on learning and intelligent optimization*, pp. 59–69. Springer, 2013.

Frazier, P. I. A tutorial on Bayesian optimization. *arXiv preprint arXiv:1807.02811*, 2018.

Fujisawa, M. and Sato, I. Multilevel Monte Carlo variational inference. *The Journal of Machine Learning Research*, 22(1):12741–12784, 2021.

Garnett, R. *Bayesian optimization*. Cambridge University Press, 2023.

Giles, M. B. Multilevel Monte Carlo path simulation. *Operations research*, 56(3):607–617, 2008.

Giles, M. B. Multilevel Monte Carlo methods. *Acta numerica*, 24:259–328, 2015.

Giles, M. B. MLMC for nested expectations. In *Contemporary Computational Mathematics-A Celebration of the 80th Birthday of Ian Sloan*, pp. 425–442. Springer, 2018.

Giles, M. B. and Goda, T. Decision-making under uncertainty: Using MLMC for efficient estimation of EVPPI. *Statistics and Computing*, 29(4):739–751, 2019.

Giles, M. B. and Haji-Ali, A.-L. Multilevel nested simulation for efficient risk estimation. *SIAM/ASA Journal on Uncertainty Quantification*, 7(2):497–525, 2019.

Ginsbourger, D. and Le Riche, R. Towards Gaussian process-based optimization with finite time horizon. In *mODa 9–Advances in Model-Oriented Design and Analysis: Proceedings of the 9th International Workshop in Model-Oriented Design and Analysis held in Bertinoro, Italy, June 14-18, 2010*, pp. 89–96. Springer, 2010.

Goda, T., Hironaka, T., and Iwamoto, T. Multilevel Monte Carlo estimation of expected information gains. *Stochastic Analysis and Applications*, 38(4):581–600, 2020.

Goda, T., Hironaka, T., Kitade, W., and Foster, A. Unbiased MLMC stochastic gradient-based optimization of Bayesian experimental designs. *SIAM Journal on Scientific Computing*, 44(1):A286–A311, 2022.

González, J., Osborne, M., and Lawrence, N. GLASSES: Relieving the myopia of Bayesian optimisation. In *Artificial Intelligence and Statistics*, pp. 790–799. PMLR, 2016.

Haji-Ali, A.-L., Nobile, F., and Tempone, R. Multi-index Monte Carlo: When sparsity meets sampling. *Numerische Mathematik*, 132:767–806, 2016.

Heath, M. T. *Scientific computing: An introductory survey, revised second edition*. SIAM, 2018.

Hennig, P. and Schuler, C. J. Entropy search for information-efficient global optimization. *Journal of Machine Learning Research*, 13(6), 2012.

Hernández-Lobato, J. M., Hoffman, M. W., and Ghahramani, Z. Predictive entropy search for efficient global optimization of black-box functions. *Advances in neural information processing systems*, 27, 2014.

Hironaka, T., Giles, M. B., Goda, T., and Thom, H. Multilevel Monte Carlo estimation of the expected value of sample information. *SIAM/ASA Journal on Uncertainty Quantification*, 8(3):1236–1259, 2020.

Hu, Y., Chen, X., and He, N. On the bias-variance-cost tradeoff of stochastic optimization. *Advances in Neural Information Processing Systems*, 34:22119–22131, 2021.

James, F. Monte Carlo theory and practice. *Reports on progress in Physics*, 43(9):1145, 1980.

Jasra, A., Kamatani, K., Law, K. J. H., and Zhou, Y. A multi-index Markov chain Monte Carlo method. *International Journal for Uncertainty Quantification*, 8(1), 2018.

Jasra, A., Law, K. J., Walton, N., and Yang, S. Multi-index sequential Monte Carlo ratio estimators for Bayesian inverse problems. *Foundations of Computational Mathematics*, pp. 1–56, 2023.

Jiang, S., Chai, H., Gonzalez, J., and Garnett, R. BINOCULARS for efficient, nonmyopic sequential experimental design. In *International Conference on Machine Learning*, pp. 4794–4803. PMLR, 2020a.

Jiang, S., Jiang, D., Balandat, M., Karrer, B., Gardner, J., and Garnett, R. Efficient nonmyopic Bayesian optimization via one-shot multi-step trees. *Advances in Neural Information Processing Systems*, 33:18039–18049, 2020b.

Jones, D. R., Schonlau, M., and Welch, W. J. Efficient global optimization of expensive black-box functions. *Journal of Global optimization*, 13(4):455, 1998.

Kim, S., Pasupathy, R., and Henderson, S. G. A guide to sample average approximation. *Handbook of simulation optimization*, pp. 207–243, 2015.

Kleywegt, A. J., Shapiro, A., and Homem-de Mello, T. The sample average approximation method for stochastic discrete optimization. *SIAM Journal on Optimization*, 12 (2):479–502, 2002.

Lam, R. and Willcox, K. Lookahead Bayesian optimization with inequality constraints. *Advances in neural information processing systems*, 30, 2017.

Lam, R., Willcox, K., and Wolpert, D. H. Bayesian optimization with a finite budget: An approximate dynamic programming approach. *Advances in Neural Information Processing Systems*, 29, 2016.

L'Ecuyer, P., Owen, A. B., et al. *Monte Carlo and Quasi-Monte Carlo Methods 2008*. Springer, 2009.

Lee, E., Eriksson, D., Bindel, D., Cheng, B., and Mccourt, M. Efficient rollout strategies for Bayesian optimization. In *Conference on Uncertainty in Artificial Intelligence*, pp. 260–269. PMLR, 2020.

Liang, X., Yang, S., Cotter, S. L., and Law, K. J. A randomized Multi-index sequential Monte Carlo method. *Statistics and Computing*, 33(5):97, 2023.

Lyne, A.-M., Girolami, M., Atchadé, Y., Strathmann, H., and Simpson, D. On Russian Roulette estimates for Bayesian inference with doubly-intractable likelihoods. *Statistical Science*, 30(4):443–467, 2015. ISSN 08834237, 21688745. URL http://www.jstor.org/stable/24780815.

Močkus, J. On Bayesian methods for seeking the extremum. In *Optimization Techniques IFIP Technical Conference: Novosibirsk, July 1–7, 1974*, pp. 400–404. Springer, 1975.

Nguyen, V., Gupta, S., Rana, S., Li, C., and Venkatesh, S. Predictive variance reduction search. In *NIPS Workshop on Bayesian Optimization*, volume 12, 2017.

Peherstorfer, B., Willcox, K., and Gunzburger, M. Survey of multifidelity methods in uncertainty propagation, inference, and optimization. *Siam Review*, 60(3):550–591, 2018.

Renganathan, S. A., Larson, J., and Wild, S. Recursive two-step lookahead expected payoff for time-dependent Bayesian optimization. *arXiv preprint arXiv:2006.08037*, 2020.

Rhee, C.-h. and Glynn, P. W. A new approach to unbiased estimation for SDE's. In *Proceedings of the 2012 Winter Simulation Conference (WSC)*, pp. 1–7. IEEE, 2012.

Rhee, C.-h. and Glynn, P. W. Unbiased estimation with square root convergence for SDE models. *Operations Research*, 63(5):1026–1043, 2015.

Shahriari, B., Swersky, K., Wang, Z., Adams, R. P., and De Freitas, N. Taking the human out of the loop: A review of Bayesian optimization. *Proceedings of the IEEE*, 104(1):148–175, 2015.

Shapiro, A., Dentcheva, D., and Ruszczynski, A. *Lectures on stochastic programming: Modeling and theory*. SIAM, 2021.

Srinivas, N., Krause, A., Kakade, S. M., and Seeger, M. Gaussian process optimization in the bandit setting: No regret and experimental design. *arXiv preprint arXiv:0912.3995*, 2009.

Strathmann, H., Sejdinovic, D., and Girolami, M. Unbiased Bayes for big data: Paths of partial posteriors. *arXiv preprint arXiv:1501.03326*, 2015.

Wang, J., Clark, S. C., Liu, E., and Frazier, P. I. Parallel bayesian global optimization of expensive functions. *Operations Research*, 68(6):1850–1865, 2020.

Wu, J. and Frazier, P. Practical two-step lookahead Bayesian optimization. *Advances in neural information processing systems*, 32, 2019.

Yue, X. and Kontar, R. A. Why non-myopic Bayesian optimization is promising and how far should we look-ahead? A study via rollout. In *International Conference on Artificial Intelligence and Statistics*, pp. 2808–2818. PMLR, 2020.

Zhang, Y., Zhang, X., and Frazier, P. I. Two-step lookahead Bayesian optimization with inequality constraints. In Ranzato, M., Beygelzimer, A., Dauphin, Y. N., Liang, P., and Vaughan, J. W. (eds.), *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pp. 12563–12575, 2021.

A. Acquisition Functions

Some one-step look-ahead acquisition functions, such as expected improvement (EI), are analytically tractable, but the functions are generally intractable for the look-ahead steps greater than one. Monte Carlo (MC) estimation is a standard method for approximating the values. Using the re-parameterization trick, we assume the following

$$f(x,\xi;\mathcal{D}) = N(\mu(x),\sigma^2(x)) = \mu(x)+\sigma(x)\xi,$$
$$f(x_1,\xi_1;\mathcal{D}_1(x;\xi)) = N(\mu(x_1;x),\sigma^2(x_1;x)) = \mu(x_1;x)+\sigma(x_1;x)\xi_1,$$

where $\xi,\xi_1 \sim N(0,1)$, $\mu(\cdot;x)$ and $\sigma^2(\cdot;x)$ are the posterior mean and variance functions derived with $\mathcal{D}_1(x;\xi) = \{\mathcal{D},(x,f(x,\xi;\mathcal{D}))\}$. For simplicity, we do not carry on the reparameterization trick explicitly and keep using the notation $f(x;\mathcal{D})$ when the dependence is on ξ and $f(x_1;\mathcal{D}_1(x))$ when the dependence is on ξ and ξ_1. The superscript notations are used for realization driven by a random source. Specifically, the notation

$$f^i(x;\mathcal{D}) = f(x,\xi^i;\mathcal{D})$$
$$f^j(x_1;\mathcal{D}_1(x)) = f(x_1,\xi_1^j;\mathcal{D}_1(x;\xi))$$
$$f^{ij}(x_1;\mathcal{D}_1(x)) = f(x_1,\xi_1^j;\mathcal{D}_1(x;\xi^i))$$

for $i=1,2,...$ and $j=1,2,...$ is used to stand for a realization of the function. Here, ξ^i, ξ_1^j and ξ_1^{ij} are i.i.d. The data set follows the superscript notation as well.

Even if the one-step AF is analytically tractable, we must approximate the two-step and three-step look-ahead AFs, which can be done with MC as follows

$$\alpha_{1,N}(x;\mathcal{D}) = \mathbb{E}_{f(\cdot;\mathcal{D})}[r(f,x)] + \frac{1}{N}\sum_{i=1}^{N}\max_{x_1^i}\mathbb{E}_{f(\cdot;\mathcal{D}_1^i(x))}\left[r(f,x_1^i)\right] , \tag{15}$$

$$\alpha_{2,N,M}(x;\mathcal{D}) = \mathbb{E}_{f(\cdot;\mathcal{D})}[r(f,x)] + \frac{1}{N}\sum_{i=1}^{N}\max_{x_1^i}\left\{\mathbb{E}_{f(\cdot;\mathcal{D}_1^i(x))}\left[r(f,x_1^i)\right] + \frac{1}{M}\sum_{j=1}^{M}\max_{x_2^{ij}}\mathbb{E}_{f(\cdot;\mathcal{D}_2^{ij}(x,x_1^i))}[r(f,x_2^{ij})]\right\}, \tag{16}$$

where N and M are number of samples.

A.1. Explicit 2- and 3-step functions

Two-step look-ahead expected improvement can be written as

$$\alpha_1(x;\mathcal{D}) := \mathbb{E}_{f(\cdot;\mathcal{D})}\left[r(f,x)+\max_{x_1}\mathbb{E}_{f(\cdot;\mathcal{D}_1(x))}\left[r(f,x_1)\right]\right], \tag{17}$$

with $r(f,x) = (f(x;\mathcal{D})-f^*(\mathcal{D}))_+$, $r(f,x_1) = (f(x_1;\mathcal{D}_1(x))-f^*(\mathcal{D}_1(x))_+$, where $f^*(\mathcal{D}_1(x)) = \max\{f^*(\mathcal{D}),f(x;\mathcal{D})\}$. The two-step look-ahead EI (17) is analytically intractable, which can be approximated by referring to (15).

Two-step look-ahead q-expected improvement can be written as

$$\alpha_1(x;\mathcal{D}) := \mathbb{E}_{f(\cdot;\mathcal{D})}\left[r(f,x)+\max_{x_1}\mathbb{E}_{f(\cdot;\mathcal{D}_1(x))}\left[r(f,x_1)\right]\right], \tag{18}$$

with $r(f,x) = \max_{j=1,...,q}(f(x_j;\mathcal{D})-f^*(\mathcal{D}))_+$ and $r(f,x_1) = \max_{j=1,...,q}(f(x_{1,j};\mathcal{D}_1(x))-f^*(\mathcal{D}_1(x))_+$. The two-step look-ahead qEI (18) is analytically intractable and introduces a nested MC due to the qEI, which can be evaluated by referring to (3).

In the numerical section 5 of this article, we let the zero-step reward be EI and the two-step reward be qEI for convenience, i.e.

$$\alpha_1(x;\mathcal{D}) := \mathbb{E}_{f(\cdot;\mathcal{D})}\left[r_0(f,x)+\max_{x_1}\mathbb{E}_{f(\cdot;\mathcal{D}_1(x))}\left[r_1(f,x_1)\right]\right], \tag{19}$$

with $r_0(f,x) = (f(x;\mathcal{D}) - f^*(\mathcal{D}))_+$ and $r_1(f,x_1) = \max_{j=1,\dots,q}(f(x_{1,j};\mathcal{D}_1(x)) - f^*(\mathcal{D}_1(x))_+$.

Three-step look-ahead expected improvement can be written as

$$\alpha_2(x;\mathcal{D}) := \mathbb{E}_{f(\cdot;\mathcal{D})}\left[r(f,x) + \max_{x_1}\mathbb{E}_{f(\cdot;\mathcal{D}_1(x))}\left[r(f,x_1) + \max_{x_2}\mathbb{E}_{f(\cdot;\mathcal{D}_2(x,x_1))}[r(f,x_2)]\right]\right] \tag{20}$$

with $r(f,x) = (f(x;\mathcal{D}) - f^*(\mathcal{D}))_+$, $r(f,x_1) = (f(x_1;\mathcal{D}_1(x)) - f^*(\mathcal{D}_1(x))_+$ and $r(f,x_2) = (f(x_2;\mathcal{D}_2(x,x_1)) - f^*(\mathcal{D}_2(x,x_1)))_+$, where $f^*(\mathcal{D}_2(x,x_1)) = \max\{f^*(\mathcal{D}_1(x)), f(x_1;\mathcal{D}_1(x))\}$. Referring to equation (16), we obtain the approximation of the Three-step look-ahead EI (20).

B. Assumptions and discussion related to SAA

The following is a set of sufficient conditions for basic SAA convergence results to hold, which will be made throughout.

Assumption 1.

1. **(Quadratic Growth)** *For the optimum x^*, there exist a constant $K > 0$ and an open neighborhood V of x^* such that for all $x \in \Omega \cap V$,*
$$\alpha(x;\mathcal{D}) \le \alpha(x^*;\mathcal{D}) - K\|x^* - x\|^2.$$
2. *The input space $\mathcal{X}$ is compact, and $\alpha(x;\mathcal{D}) \in C^1(U)$, where U is a neighbourhood of S^*.*
3. $\mathbb{E}[\alpha(x;\mathcal{D})^2|\mathcal{D}],\ \mathbb{E}[\|\nabla_x\alpha(x';\mathcal{D})\|^2] < \infty$ *for all $x \in \mathcal{X}$ and $x' \in U$.*
4. *Both $\alpha(\cdot;\mathcal{D})$ and $\nabla_x\alpha(\cdot;\mathcal{D})$ are Lipschitz continuous with Lipschitz constant $L(\xi)$ on the input space, which is finite in expectation.*

Theorem 5.3 of (Shapiro et al., 2021) implies that the limit point (in N) of any solution from S^N lies in S^*, where S^N is the set of solutions of the SAA (5), by assuming 1(3, 4). If the function is convex, then a local optimizer is a global optimizer. The rate of convergence of optimizers is based on more factors. We require an essential regularity condition called quadratic growth condition, defined in 1(1). Otherwise, if the function r increases linearly near x^*, the optimizer may converge at a faster rate (Kim et al., 2015) while the rate may be slower if it behaves like a higher-order polynomial.

C. Proof of Proposition 2

Proof. Recall $\alpha(x;\mathcal{D})$, (15) and (3) as follows

$$\begin{aligned}
\alpha(x;\mathcal{D}) &:= \mathbb{E}_{f(\cdot;\mathcal{D})}\left[r(f,x) + \max_{x_1}\mathbb{E}_{f(\cdot;\mathcal{D}_1(x))}\left[r(f,x_1)\right]\right]\\
\alpha_N(x;\mathcal{D}) &= \frac{1}{N}\sum_{i=1}^{N}\left[r(f^i(x;\mathcal{D})) + \max_{x_1^i}\mathbb{E}_{f(\cdot;\mathcal{D}_1^i(x))}\left[r(f,x_1^i)\right]\right]\\
\alpha_{N,M}(x;\mathcal{D}) &= \frac{1}{N}\sum_{i=1}^{N_l}\left[r(f^i(x;\mathcal{D})) + \max_{x_1^i}\frac{1}{M}\sum_{j=1}^{M} r(f^{ij}(x_1^i;\mathcal{D}_1^i(x)))\right].
\end{aligned}$$

We first prove equation (7). Note that

$$\mathbb{E}[|\alpha_{N,M}(x;\mathcal{D}) - \alpha(x;\mathcal{D})|^2] \le 2\mathbb{E}[|\alpha_N(x;\mathcal{D}) - \alpha(x;\mathcal{D})|^2] \tag{21}$$
$$+ 2\mathbb{E}[|\alpha_{N,M}(x;\mathcal{D}) - \alpha_N(x;\mathcal{D})|^2], \tag{22}$$

due to $(a+b)^2 \le 2a^2 + 2b^2$. We deal with the two terms (21) and (22) separately.

The first term (21) is simply a standard MC error bounded by the following

$$\mathbb{E}[|\alpha_N(x;\mathcal{D}) - \alpha(x;\mathcal{D})|^2] = \mathbb{E}\left[\left(\frac{1}{N}\sum_{i=1}^{N}\left[r(f^i(x;\mathcal{D})) + \max_{x_1^i}\mathbb{E}_{f(\cdot;\mathcal{D}_1^i(x))}\left[r(f,x_1^i)\right]\right]\right.\right. \tag{23}$$

$$\left.\left. - \mathbb{E}_{f(\cdot;\mathcal{D})}\left[r(f,x) + \max_{x_1}\mathbb{E}_{f(\cdot;\mathcal{D}_1(x))}\left[r(f,x_1)\right]\right]\right)^2\right]$$

$$= \frac{\mathsf{Var}\Big(r(f,x) + \max_{x_1}\mathbb{E}_{f(\cdot;\mathcal{D}_1(x))}\left[r(f,x_1)\right]\Big)}{N}$$

$$\leq \frac{\sigma_{\max}}{N}, \tag{24}$$

where the last inequality is due to the bounded variance assumption.

For the second term (22), we have

$$\mathbb{E}[|\alpha_{N,M}(x;\mathcal{D}) - \alpha_N(x;\mathcal{D})|^2] = \mathbb{E}\left[\left|\frac{1}{N}\sum_{i=1}^{N_l}\left[r(f^i(x;\mathcal{D})) + \max_{x_1^i}\frac{1}{M}\sum_{j=1}^{M} r(f^{ij}(x_1^i;\mathcal{D}_1^i(x)))\right]\right.\right.$$

$$\left.\left. - \frac{1}{N}\sum_{i=1}^{N}\left[r(f^i(x;\mathcal{D})) + \max_{x_1^i}\mathbb{E}_{f(\cdot;\mathcal{D}_1^i(x))}\left[r(f,x_1^i)\right]\right]\right|^2\right]$$

$$= \mathbb{E}\left[\left|\frac{1}{N}\sum_{i=1}^{N_l}\max_{x_1^i}\frac{1}{M}\sum_{j=1}^{M} r(f^{ij}(x_1^i;\mathcal{D}_1^i(x))) - \frac{1}{N}\sum_{i=1}^{N}\max_{x_1^i}\mathbb{E}_{f(\cdot;\mathcal{D}_1^i(x))}\left[r(f,x_1)\right]\right|^2\right]$$

Define

$$x_{1,M}^{i,*} = \operatorname*{argmax}_{x_1^i}\frac{1}{M}\sum_{j=1}^{M} r(f^{ij}(x_1;\mathcal{D}_1^i(x))).$$

Then, we have

$$\max_{x_1^i}\mathbb{E}_{f(\cdot;\mathcal{D}_1^i(x))}\left[r(f,x_1^i)\right] \geq \mathbb{E}_{f(\cdot;\mathcal{D}_1^i(x))}\left[r(f,x_{1,M}^{i,*})\right].$$

Thus,

$$\begin{aligned}
&\mathbb{E}[|\alpha_{N,M}(x;\mathcal{D}) - \alpha_N(x;\mathcal{D})|^2] \\
\leq&\mathbb{E}\left[\left|\frac{1}{N}\sum_{i=1}^{N}\frac{1}{M}\sum_{j=1}^{M} r(f^{ij}(x_{1,M}^{i,*};\mathcal{D}_1^i(x))) - \frac{1}{N}\sum_{i=1}^{N}\mathbb{E}_{f(\cdot;\mathcal{D}_1^i(x))}\left[r_1(f,x_{1,M}^{i,*})\right]\right|^2\right] \\
\leq&\frac{N}{N^2}\sum_{i=1}^{N}\mathbb{E}\left[\left|\frac{1}{M}\sum_{j=1}^{M} r(f^{ij}(x_{1,M}^{i,*};\mathcal{D}_1^i(x))) - \mathbb{E}_{f(\cdot;\mathcal{D}_1^i(x))}\left[r(f,x_{1,M}^{i,*})\right]\right|^2\right] \\
\leq&\frac{1}{N}\sum_{i=1}^{N}\mathsf{Var}\left(\frac{1}{M}\sum_{j=1}^{M} r(f^{ij}(x_{1,M}^{i,*};\mathcal{D}_1^i(x)))\right) \\
\leq&\frac{\sigma_{\max}}{M},
\end{aligned} \tag{25}$$

the first inequality is by taking the maximizer, the second inequality is by Cauchy-Schwarz inequality, and the last follows from standard SAA arguments.

Combining (24) and (25), we have

$$\mathbb{E}|\alpha_{N,M}(x) - \alpha(x)|^2 \leq C_1(1/N + 1/M),$$

for some constant C_1.

□

D. MLMC estimators

D.1. MLMC Acquisition Function

We define $\Delta\alpha_{N_l,M_l}(x) = \alpha_{N_l,M_l}(x) - \alpha_{N_l,M_{l-1}}(x)$ and adapt the convention that $\alpha_{N_0,M_{-1}}(x) \equiv 0$. From this, we form the estimator

$$\alpha_{\mathsf{ML}}(x) := \sum_{l=0}^{L} \Delta\alpha_{N_l,M_l}(x)\,,$$

where

$$\alpha_{N_l,M_l}(x) = \frac{1}{N_l}\sum_{i=1}^{N_l}\left[r(f^i(x;\mathcal{D})) + \left(\max_{x_1^{ij}} \frac{1}{M_l}\sum_{j=1}^{M_l} r(f^{ij}(x_1^{ij};\mathcal{D}_1^i(x)))\right)\right],$$

$$\alpha_{N_l,M_{l-1}}(x) = \frac{1}{N_l}\sum_{i=1}^{N_l}\left[r(f^i(x;\mathcal{D})) + \left(\max_{x_1^{ij}} \frac{1}{M_{l-1}}\sum_{j=1}^{M_{l-1}} r(f^{ij}(x_1^{ij};\mathcal{D}_1^i(x)))\right)\right].$$

We remark that the terms $\Delta\alpha_{N_l,M_l}(x)$ for $l = 0, 1, ..., L$ are mutually independent. We now argue that α_{ML} is a computationally efficient proxy for the intractable limit objective $\alpha(x;\mathcal{D})$, which leads to computational benefits for approximating optimizer x^*.

D.2. Multilevel formulation of three-step look-ahead acquisition function

The multilevel estimation for the optimizer of the three-step look-ahead AF $\alpha_2(x;\mathcal{D})$ can be formulated as

$$x^*_{\mathsf{ML}} := \sum_{l=0}^{L} z_{N_l,M_l} - z_{N_l,M_{l-1}} = \sum_{l=0}^{L} \Delta z_{N_l}\,,$$

with

$$z_{N_l,M_l} = \operatorname*{argmax}_x \frac{1}{N_l}\sum_{i=1}^{N_l}\left[r(f^i(x;\mathcal{D})) + \max_{x_1^i}\frac{1}{M_l}\sum_{j=1}^{M_l}\left(r(f^{ij}(x_1^i;\mathcal{D}_1^i(x))) + \max_{x_2^{ij}} \mathbb{E}_{f(\cdot;\mathcal{D}_2^{ij}(x,x_1))}[r(f,x_2^{ij})]\right)\right], \tag{26}$$

$$z_{N_l,M_{l-1}} = \operatorname*{argmax}_x \frac{1}{N_l}\sum_{i=1}^{N_l}\left[r(f^i(x;\mathcal{D})) + \max_{x_1^i}\frac{1}{M_{l-1}}\sum_{j=1}^{M_{l-1}}\left(r(f^{ij}(x_1^i;\mathcal{D}_1^i(x))) + \max_{x_2^{ij}} \mathbb{E}_{f(\cdot;\mathcal{D}_2^{ij}(x,x_1))}[r(f,x_2^{ij})]\right)\right]. \tag{27}$$

Here we use the convention $z_{N_0,M_{-1}} \equiv 0$. To leverage the MLMC, we also require the subsampling of M_{l-1} samples from M_l similar to the two-step look-ahead one. The antithetic approach of the three-step function involves replacing the coarse estimator (27) with

$$\begin{aligned} z^A_{N_l,M_{l-1}} = \operatorname*{argmax}_x \frac{1}{N_l}\sum_{i=1}^{N_l}\Bigg[& r(f^i(x;\mathcal{D})) \\ & + \frac{1}{2}\Bigg(\max_{x_1^i}\frac{1}{M_{l-1}}\sum_{j=1}^{M_{l-1}}\left(r(f^{ij}(x_1^i;\mathcal{D}_1^i(x))) + \max_{x_2^{ij}} \mathbb{E}_{f(\cdot;\mathcal{D}_2^{ij}(x,x_1))}[r(f,x_2^{ij})]\right) \\ & + \max_{x_1^i}\frac{1}{M_{l-1}}\sum_{j=M_{l-1}+1}^{M_l}\left(r(f^{ij}(x_1^i;\mathcal{D}_1^i(x))) + \max_{x_2^{ij}} \mathbb{E}_{f(\cdot;\mathcal{D}_2^{ij}(x,x_1))}[r(f,x_2^{ij})]\right)\Bigg)\Bigg], \end{aligned}$$

where $z^A_{N_0,M_{-1}} \equiv 0$. The sampling step is the same as the two-step look-ahead one.

E. Value function and maximizer corollaries

Theorem 1 provides a convergence rate for q-functions. We can then use this result to establish the convergence of value functions:

Corollary 1 (Value Function Convergence)**.** *For* $v^\star(\mathcal{D}) := \max_x \alpha(x;\mathcal{D})$ *and* $v_{\mathsf{ML}} := \max_x \alpha_{\mathsf{ML}}(x)$

$$\mathbb{E}\big[|v_{\mathsf{ML}} - v^\star(\mathcal{D})|^2\big] \leq C(L+1)\left(\frac{V_0}{N_0} + \sum_{l=1}^{L}\frac{1}{N_l M_l}\right) + \frac{C}{M_L}$$

and thus under the parameter choices (13)*, a root-mean-square error of* ε^2 *can be achieved with a sample complexity of* $O(\varepsilon^{-2}|\log\varepsilon|^3)$.

The proof of Corollary 1 is given in Appendix G.3. Assuming quadratic growth at the optimizer, we can prove convergence to the optimal decision point.

Corollary 2 (Maximizer Convergence)**.** *Assume 1 and 2. For* $\tilde{x}^*_{\mathsf{ML}} \in \operatorname{argmax}_x \alpha_{\mathsf{ML}}(x)$ *and* $x^* = \operatorname{argmax}_x \alpha(x;\mathcal{D})$

$$\mathbb{E}[\|\tilde{x}^*_{\mathsf{ML}} - x^*\|^2] \leq C(L+1)\left(\frac{V_0}{N_0} + \sum_{l=1}^{L}\frac{1}{N_l M_l}\right) + \frac{1}{M_L}$$

and thus under the parameter choices (13)*, a mean-square error of* ε^2 *can be achieved with a sample complexity of* $O(\varepsilon^{-2}|\log\varepsilon|^3)$.

The proof of Corollary 2 is given in Appendix G.4.

It is difficult to numerically verify Theorem 1 due to the inner sup. Numerical verification of complexity for Corollary 1 and 2 are in Appendix H.2.

F. Assumptions and discussion related to MLMC

We let

$$\Delta_M(x) = \max_{x_1}\frac{1}{M}\sum_{j=1}^{M} r(f^j(x_1;\mathcal{D}_1(x))) - \max_{x_1}\mathbb{E}_{f(\cdot;\mathcal{D}_1(x))}\left[r(f,x_1)\right], \tag{28}$$

$$\nabla\Delta_M(x) = \nabla_x \max_{x_1}\frac{1}{M}\sum_{j=1}^{M} r(f^j(x_1;\mathcal{D}_1(x))) - \nabla_x \max_{x_1}\mathbb{E}_{f(\cdot;\mathcal{D}_1(x))}\left[r(f,x_1)\right]. \tag{29}$$

Notice that $\Delta_M(x)$ and $\nabla\Delta_M(x)$ summarize the error in the sample average approximation of the inner expectation and its derivative, respectively. Given the existence of comparable results on SAA, we make the following assumption:

Assumption 2. *For some constant* $C > 0$ *and* $\forall x \in \mathcal{X}$,

$$\sup_x \big|\mathbb{E}\big[\Delta_M(x)\big]\big| \leq \frac{C}{M^{\frac{1}{2}}} \qquad \textit{and} \qquad \mathbb{E}\left[\sup_x \Big|\Delta_M(x) - \mathbb{E}[\Delta_M(x)]\Big|^2\right] \leq \frac{C}{M},$$

$$\sup_x \big|\mathbb{E}\big[\nabla\Delta_M(x)\big]\big| \leq \frac{C}{M^{\frac{1}{2}}} \qquad \textit{and} \qquad \mathbb{E}\left[\sup_x \Big|\nabla\Delta_M(x) - \mathbb{E}[\nabla\Delta_M(x)]\Big|^2\right] \leq \frac{C}{M}.$$

Remark 2. *The above Assumption is found to hold for Sample Average Approximation for any given value of* x*, See Page 165 of (Shapiro et al., 2021). Here we ask for standard SAA approximation results to hold over all values of* x *taken in our outer sample. Also, if* $\max_{x_1}\frac{1}{M}\sum_{j=1}^{M} r(f^j(x_1;\mathcal{D}_1(x)))$ *is square integrable, Assumption 2 is satisfied by CLT. The square integrability of* $\max_{x_1}\frac{1}{M}\sum_{j=1}^{M} r(f^j(x_1;\mathcal{D}_1(x)))$ *can be obtained by the square integrability of underlying Gaussian random variables, following the similar argument of showing uniform integrability as Theorem 6.1.6 of (Borovkov, 1999). The underlying GP is square integrable by choosing a proper integrable kernel function.*

We state an intermediate result in Lemma 1, which shows the variance of the increments satisfies the MLMC assumption and leads to Theorem 1.

Lemma 1. *Assume 2 and $M_l = cM_{l-1}$ for some positive integer c. Then, for some constant $C > 0$,*

$$\mathbb{E}[\sup_x |\Delta\alpha_{N_l,M_l}(x) - \mathbb{E}[\Delta\alpha_{N_l,M_l}(x)]|^2] \leq \frac{C}{N_l M_l}.$$

The proof of Lemma 1 is in Appendix G.1.

F.1. Additional assumption for Theorem 2

The additional assumption required for the implementable algorithm is now given.

Assumption 3. *Assume for $\beta > 0$ and $x \in \mathcal{X}$, the bound*

$$\mathbb{E}[|\Delta\alpha_{N_l,M_l}(x) - \Delta\alpha_{\infty,M_l}(x)|^2] \leq \frac{C}{N_l M_l^\beta},$$

implies

$$\mathbb{E}[\|\Delta z_{N_l,M_l} - \Delta z_{\infty,M_l}\|^2] \leq \frac{C}{N_l M_l^\beta}.$$

Remark 3. *The implication Assumption 3 holds for differences, e.g. $\mathbb{E}[\|\Delta z_{N_l,M_l}\|] \leq \sup_{x\in\mathcal{X}} \mathbb{E}[|\nabla\Delta\alpha_{N_l,M_l}(x)|]$, under suitable assumptions, following the argument in the proof of Corollary 2. We have validated it numerically for differences of differences in Appendix H.3, but it remains to be proven.*

G. Proof of lemmas and theorems

Recall

$$\alpha(x;\mathcal{D}) = \mathbb{E}_{f(\cdot;\mathcal{D})}\left[r(f,x) + \max_{x_1} \mathbb{E}_{f(\cdot;\mathcal{D}_1(x))}\left[r(f,x_1)\right]\right],$$
$$\alpha_{N,M}(x;\mathcal{D}) = \frac{1}{N}\sum_{i=1}^{N}\left[r(f^i(x;\mathcal{D})) + \left(\max_{x_1^i} \frac{1}{M}\sum_{j=1}^{M} r(f^{ij}(x_1^i;\mathcal{D}_1^i(x)))\right)\right].$$

Define

$$\alpha_{N,\infty}(x;\mathcal{D}) = \frac{1}{N}\sum_{i=1}^{N}\left(r(f^i(x;\mathcal{D})) + \max_{x_1^i} \mathbb{E}_{f(\cdot;\mathcal{D}_1^i(x))}\left[r(f,x_1^i)\right]\right),$$
$$\alpha_{\infty,M}(x;\mathcal{D}) = \mathbb{E}_{f(\cdot;\mathcal{D})}\left[r(f,x) + \max_{x_1} \frac{1}{M_l}\sum_{j=1}^{M_l} r(f^j(x_1;\mathcal{D}_1(x)))\right].$$

We first state a lemma.

Lemma 2. *Assume 2. For some constant $C > 0$,*

$$\mathbb{E}\left[\sup_x \left|\alpha_{N,M}(x) - \alpha_{N,\infty}(x) - \mathbb{E}[\alpha_{N,M}(x) - \alpha_{N,\infty}(x)]\right|^2\right] \leq \frac{C}{NM}.$$

Proof. We have

$$\mathbb{E}\left[\sup_x \left|\alpha_{N,M}(x) - \alpha_{N,\infty}(x) - \mathbb{E}[\alpha_{N,M}(x) - \alpha_{N,\infty}(x)]\right|^2\right]$$
$$= \mathbb{E}\left[\sup_x \left|\frac{1}{N}\sum_{i=1}^{N}\left(\max_{x_1^i} \frac{1}{M}\sum_{j=1}^{M} r(f^{ij}(x_1^i;\mathcal{D}_1^i(x))) - \max_{x_1^i} \mathbb{E}_{f(\cdot;\mathcal{D}_1^i(x))}\left[r(f,x_1^i)\right]\right.\right.$$

$$
\begin{aligned}
&- \mathbb{E}\left[\max_{x_1^i} \frac{1}{M}\sum_{j=1}^{M} r(f^{ij}(x_1^i;\mathcal{D}_1^i(x))) - \max_{x_1^i} \mathbb{E}_{f(\cdot;\mathcal{D}_1^i(x))}\left[r(f,x_1^i)\right]\right]\Bigg)\Bigg|^2\Bigg] \\
=&\frac{1}{N}\mathbb{E}\Bigg[\sup_x\Bigg(\max_{x_1^i} \frac{1}{M}\sum_{j=1}^{M} r(f^{ij}(x_1^i;\mathcal{D}_1^i(x))) - \max_{x_1^i} \mathbb{E}_{f(\cdot;\mathcal{D}_1^i(x))}\left[r(f,x_1^i)\right] \\
&- \mathbb{E}\left[\max_{x_1^i} \frac{1}{M}\sum_{j=1}^{M} r(f^{ij}(x_1^i;\mathcal{D}_1^i(x))) - \max_{x_1^i} \mathbb{E}_{f(\cdot;\mathcal{D}_1^i(x))}\left[r(f,x_1^i)\right]\right]\Bigg)^2\Bigg] \\
=&\frac{1}{N}\mathbb{E}\left[\sup_x \left|\Delta_M(x) - \mathbb{E}[\Delta_M(x)]\right|^2\right] \\
\leq&\frac{C}{NM}.
\end{aligned}
$$

In the final inequality, we apply Assumption 2. □

G.1. Proof of Lemma 1

Now, we prove Lemma 1, which is a direct result of Lemma 2.

Proof. Note that

$$
\begin{aligned}
\Delta\alpha_{N_l,M_l}(x) &= \alpha_{N_l,M_l}(x) - \alpha_{N_l,M_{l-1}}(x) \\
&= \alpha_{N_l,M_l}(x) - \alpha_{N_l,\infty}(x) - \alpha_{N_l,M_{l-1}}(x) + \alpha_{N_l,\infty}(x).
\end{aligned}
$$

Thus,

$$
\begin{aligned}
\mathbb{E}[\sup_x |\Delta\alpha_{N_l,M_l}(x) - \mathbb{E}[\Delta\alpha_{N_l,M_l}(x)]|^2] &= \mathbb{E}\big[\sup_x \big|\alpha_{N_l,M_l}(x) - \alpha_{N_l,\infty}(x) - \mathbb{E}[\alpha_{N_l,M_l}(x) - \alpha_{N_l,\infty}(x)] \\
&\quad - \alpha_{N_l,M_{l-1}}(x) + \alpha_{N_l,\infty}(x) + \mathbb{E}[\alpha_{N_l,M_{l-1}}(x) - \alpha_{N_l,\infty}(x)]\big|^2\big] \\
&\leq 2\mathbb{E}\big[\sup_x \big|\alpha_{N_l,M_l}(x) - \alpha_{N_l,\infty}(x) - \mathbb{E}[\alpha_{N_l,M_l}(x) - \alpha_{N_l,\infty}(x)]\big|^2\big] \\
&\quad + 2\mathbb{E}\big[\sup_x \big|\alpha_{N_l,M_{l-1}}(x) - \alpha_{N_l,\infty}(x) - \mathbb{E}[\alpha_{N_l,M_{l-1}}(x) - \alpha_{N_l,\infty}(x)]\big|^2\big] \\
&\leq \frac{2C}{N_l M_l} + \frac{2C}{N_l M_{l-1}} \\
&\leq \frac{C}{N_l M_l},
\end{aligned}
$$

where the second inequality is obtained by applying Lemma 2 and the last inequality is by $M_l = cM_{l-1}$ for some positive integer c and relabeling the constant C. □

G.2. Proof of Theorem 1

The proof of Theorem 1 is given below. The proof is only given for α, however, given the assumptions the proof for $\nabla\alpha$ follows in exactly the same fashion.

Proof. To begin, the error can be divided into its bias and variance components as follows:

$$
\mathbb{E}[\sup_x \|\alpha_{\mathsf{ML}}(x) - \alpha(x;\mathcal{D})\|^2] \leq 2\mathbb{E}[\sup_x \|\alpha_{\mathsf{ML}}(x) - \mathbb{E}[\alpha_{\mathsf{ML}}(x)]\|^2] \tag{30}
$$

$$
+ 2\sup_x |\mathbb{E}[\alpha_{\mathsf{ML}}(x)] - \alpha(x;\mathcal{D})|^2 \tag{31}
$$

We will proceed to bound (30) and (31). We bound the term (30). Notice that, by Cauchy-Schwartz

$$
|\alpha_{\mathsf{ML}}(x) - \mathbb{E}[\alpha_{\mathsf{ML}}(x)]|^2 = \Big|\sum_{l=0}^{L} \Delta\alpha_{N_l,M_l}(x) - \mathbb{E}[\Delta\alpha_{N_l,M_l}(x)]\Big|^2 \leq (L+1)\sum_{l=0}^{L}\Big|\Delta\alpha_{N_l,M_l}(x) - \mathbb{E}[\Delta\alpha_{N_l,M_l}(x)]\Big|^2.
$$

Thus, we have

$$\mathbb{E}[\sup_x \|\alpha_{\mathsf{ML}}(x) - \mathbb{E}[\alpha_{\mathsf{ML}}(x)]\|^2] \leq (L+1)\sum_{l=0}^{L} \mathbb{E}[\sup_x \|\Delta\alpha_{N_l,M_l}(x) - \mathbb{E}[\Delta\alpha_{N_l,M_l}(x)]\|^2]$$
$$\leq C(L+1)\left(\frac{V_0}{N_0} + \sum_{l=1}^{L}\frac{1}{N_l M_l}\right). \tag{32}$$

The first line above is due to independence. For $l = 0$, we have the variance of α_{N_0,M_0} as $\mathbb{E}[\|\alpha_{N_0,M_0} - \mathbb{E}[\alpha_{N_0,M_0}]\|^2] = V_0/N_0$. The equation (32) follows from Lemma 1.

Finally, we deal with the term (31). First we note that $\alpha_{\mathsf{ML}}(x)$ is an unbiased estimator of $\alpha_{\infty,L}(x)$ since

$$\begin{aligned}
\mathbb{E}[\alpha_{\mathsf{ML}}(x)] &= \mathbb{E}\Big[\sum_{l=0}^{L}\Delta\alpha_{N_l,M_l}(x)\Big] \\
&= \mathbb{E}[\alpha_{N_0,M_0}(x)] + \sum_{l=1}^{L}(\mathbb{E}[\alpha_{N_l,M_l}(x)] - \mathbb{E}[\alpha_{N_l,M_{l-1}}(x)]) \\
&= \mathbb{E}[\alpha_{N_0,M_0}(x)] + \sum_{l=1}^{L}(\mathbb{E}[\alpha_{N_l,M_l}(x)] - \mathbb{E}[\alpha_{N_{l-1},M_{l-1}}(x)]) \\
&= \mathbb{E}[\alpha_{N_L,M_L}(x)] \\
&= \alpha_{\infty,M_L}(x).
\end{aligned}$$

For the third line, above, note that $\mathbb{E}[\alpha_{N_l,M_{l-1}}(x)] = \mathbb{E}[\alpha_{N_{l-1},M_{l-1}}(x)]$. In the fourth line, we cancel terms in the interpolating sum. Thus

$$|\mathbb{E}[\alpha_{\mathsf{ML}}(x)] - \alpha(x;\mathcal{D})| = |\alpha_{\infty,M_L}(x) - \alpha(x;\mathcal{D})|.$$

And so,

$$\begin{aligned}
&\sup_x |\alpha_{\infty,M_L}(x) - \alpha(x;\mathcal{D})|^2 \\
&= \sup_x \left|\mathbb{E}_{f(\cdot;\mathcal{D})}\left[r(f,x) + \max_{x_1}\frac{1}{M_L}\sum_{j=1}^{M_L} r(f^j(x_1;\mathcal{D}_1(x)))\right] - \mathbb{E}_{f(\cdot;\mathcal{D})}\left[r(f,x) + \max_{x_1}\mathbb{E}_{f(\cdot;\mathcal{D}_1(x))}\left[r(f,x_1)\right]\right]\right|^2 \\
&= \sup_x \left|\mathbb{E}_{f(\cdot;\mathcal{D})}\left[\max_{x_1}\frac{1}{M_L}\sum_{j=1}^{M_L} r(f^j(x_1;\mathcal{D}_1(x))) - \max_{x_1}\mathbb{E}_{f(\cdot;\mathcal{D}_1(x))}\left[r(f,x_1)\right]\right]\right|^2 \\
&= \sup_x |\mathbb{E}\big[\Delta_{M_L}(x)\big]|^2 \\
&\leq \frac{C}{M_L},
\end{aligned} \tag{33}$$

The final inequality follows by Assumption 2. Applying (32) and (33) to (30) and (31) gives

$$\mathbb{E}\big[\sup_x \|\alpha_{\mathsf{ML}}(x) - \alpha(x;\mathcal{D})\|^2\big] \leq C(L+1)\left(\frac{V_0}{N_0} + \sum_{l=1}^{L}\frac{1}{N_l M_l}\right) + \frac{C}{M_L}.$$

Thus the required the bound in 1 holds.

Setting $L = 2|\log\varepsilon|/\log 2$ gives the term C/M_L of $O(\varepsilon^2)$. Given the cost assumption, we have total computational cost Cost$= \sum_{l=0}^{L} M_l N_l$. We can minimise this cost for a fixed variance $\frac{V_0}{N_0} + \sum_{l=1}^{L}\frac{1}{N_l M_l} = \varepsilon^2/(L+1)$ using Lagrange multipliers. See Lemma 3 for a proof. Specifically, Lemma 3 (with $c_l = M_l$, $l = 0, ..., L$, $v_0 = V_0$ and $v_l = 1/M_l$, $l = 1, ..., L$) gives

$$N_0 = (L+1)\varepsilon^{-2}K_L\left(\frac{V_0}{M_0}\right)^{1/2} \quad \text{and} \quad N_l = (L+1)\varepsilon^{-2}K_L\frac{1}{M_l}, \quad \text{for} \quad l \geq 1,$$

for

$$K_L = (V_0 M_0)^{1/2} + L,$$

and hence COST $= (L+1)\varepsilon^{-2}K_L^2 = O(|\log\varepsilon|^3/\varepsilon^2)$ since $K_L = O(|\log\varepsilon|)$ and $L = O(|\log\varepsilon|)$. Finally, observe that if we round up terms, I.e. $N_0 = \left\lceil \varepsilon^{-2}K_L\left(\frac{V_0}{M_0}\right)^{1/2}\right\rceil$ and $N_l = \left\lceil \varepsilon^{-2}K_L\frac{1}{M_l}\right\rceil$, for $l \geq 1$, then the additional cost added is $\sum_{l=0}^{L} M_l \leq 2^{L+1} \leq \frac{2}{\epsilon^2}$. Thus the total cost of the algorithm is of order $O(\varepsilon^{-2}|\log\varepsilon|^3)$. □

Lemma 3. *The cost minimization*

$$\textit{minimize} \quad \sum_{l=0}^{L} n_l c_l \quad \textit{subject to} \quad \sum_{l=0}^{L} \frac{v_l}{n_l} \leq \epsilon^2 \quad \textit{over} \quad n_l \geq 0, \;\; l = 0, ..., L,$$

has optimal solution and the optimal cost

$$n_l^\star = \frac{1}{\epsilon^2}\sqrt{\frac{v_l}{c_l}}\left(\sum_{l'=0}^{L}\sqrt{v_{l'}c_{l'}}\right) \qquad \textit{and} \qquad C^\star = \frac{1}{\epsilon^2}\left(\sum_{l'=0}^{L}\sqrt{v_{l'}c_{l'}}\right)^2.$$

Proof. The Lagrangian of the above optimization problem is:

$$\mathcal{L}(n;\lambda) = \sum_{l=0}^{L} n_l c_l + \lambda\left(\sum_{l=0}^{L}\frac{v_l}{n_l} - \epsilon^2\right).$$

Thus

$$c_l - \frac{\lambda v_l}{n_l^2} = 0, \qquad \text{and thus} \qquad n_l = \sqrt{\frac{\lambda v_l}{c_l}}.$$

For primal feasibility, we require

$$\epsilon^2 = \frac{v_l}{n_l} = \sum_{l=0}^{L}\sqrt{\frac{c_l v_l}{\lambda}}, \qquad \text{thus} \qquad \lambda = \left(\frac{1}{\epsilon^2}\sum_{l=0}^{L}\sqrt{c_l v_l}\right)^2.$$

Thus we have, as required,

$$n_l = \left(\frac{1}{\epsilon^2}\sum_{l=0}^{L}\sqrt{c_l v_l}\right) \times \sqrt{\frac{v_l}{c_l}}, \qquad \text{and} \qquad \sum_{l=0}^{L} n_l c_l = \frac{1}{\epsilon^2}\left(\sum_{l=0}^{L}\sqrt{c_l v_l}\right)^2.$$

□

G.3. Proof of Corollary 1

The proof of Corollary 1 is a consequence of the following standard lemma.

Lemma 4. *For two functions $\alpha_1(x)$ and $\alpha_2(x)$*

$$|\max_x \alpha_1(x) - \max_x \alpha_2(x)| \leq \sup_x |\alpha_1(x) - \alpha_2(x)|$$

Proof. Let $x_1^\star \in \operatorname{argmax} \alpha_1(x)$ then

$$\max_x \alpha_1(x) - \max_x \alpha_2(x) = \alpha_1(x_1^\star) - \max_x \alpha_2(x) \leq \alpha_1(x_1^\star) - \alpha_2(x_1^\star) \leq \sup_x |\alpha_1(x) - \alpha_2(x)|$$

By a symmetric argument we have $\max_x \alpha_2(x) - \max_x \alpha_1(x) \leq \sup_x |\alpha_1(x) - \alpha_2(x)|$ and thus the result holds. □

Proof of Corollary 1. By Lemma 4 and Theorem 1

$$\mathbb{E}\big[|v_{\mathsf{ML}} - v^\star(\mathcal{D})|^2\big] \leq \mathbb{E}\Big[\sup_x |\alpha_{\mathsf{ML}}(x) - \alpha(x;\mathcal{D})|^2\Big] \leq C(L+1)\left(\frac{V_0}{N_0} + \sum_{l=1}^{L}\frac{1}{N_l M_l}\right) + \frac{C}{M_L}$$

□

G.4. Proof of Corollary 2.

Proof of Corollary 2. By the quadratic growth condition on $\alpha(x;\mathcal{D})$ in Assumption 1, we have for some $C > 0$,

$$\begin{aligned}
C\|\tilde{x}^*_{\mathsf{ML}} - x^*\|^2 &\leq \alpha(x^*;\mathcal{D}) - \alpha(\tilde{x}^*_{\mathsf{ML}};\mathcal{D}) \\
&= \alpha(x^*;\mathcal{D}) - \alpha(\tilde{x}^*_{\mathsf{ML}};\mathcal{D}) + \alpha_{\mathsf{ML}}(\tilde{x}^*_{\mathsf{ML}}) - \alpha_{\mathsf{ML}}(\tilde{x}^*_{\mathsf{ML}}) \\
&\leq \alpha(x^*;\mathcal{D}) - \alpha(\tilde{x}^*_{\mathsf{ML}};\mathcal{D}) + \alpha_{\mathsf{ML}}(\tilde{x}^*_{\mathsf{ML}}) - \alpha_{\mathsf{ML}}(x^*) \\
&= \delta_{\mathsf{ML}}(\tilde{x}^*_{\mathsf{ML}};\mathcal{D}) - \delta_{\mathsf{ML}}(x^*;\mathcal{D}) \\
&\leq \nabla\delta_{\mathsf{ML}}(y;\mathcal{D}) \cdot (\tilde{x}^*_{\mathsf{ML}} - x^*) \\
&\leq |\nabla\delta_{\mathsf{ML}}(y;\mathcal{D})|\|\tilde{x}^*_{\mathsf{ML}} - x^*\| \\
&= \|\nabla\alpha_{\mathsf{ML}}(y) - \nabla\alpha(y;\mathcal{D})\|\|\tilde{x}^*_{\mathsf{ML}} - x^*\|,
\end{aligned}$$

Dividing through by $\|x^*_{\mathsf{ML}} - x^*\|$, squaring and taking sup and expectation, we have

$$\begin{aligned}
C\mathbb{E}[\|x^*_{\mathsf{ML}} - x^*\|^2] &\leq \mathbb{E}[\sup_y \|\nabla\alpha_{\mathsf{ML}}(y) - \nabla\alpha(y;\mathcal{D})\|^2] \\
&\leq C(L+1)\left(\frac{V_0}{N_0} + \sum_{l=1}^{L}\frac{1}{N_l M_l}\right) + \frac{C}{M_L}\,.
\end{aligned}$$

Theorem 1 is applied directly. Applying the parameters, (13) we see that a mean-squared-error of ε^2 is reached with a sample complexity of $O(\varepsilon^{-2}|\log\varepsilon|^3)$. □

G.5. Proof of Theorem 2

We now provide the proof of the main MLMC Theorem 2, reproduced here for convenience.

Theorem 3. *Assume 1, 2 and 3. Let $M_l = 2^l$. The estimator x^*_{ML} is such that, for $x \in \mathcal{X}$,*

$$\mathbb{E}[\|x^*_{\mathsf{ML}} - x^*\|^2] \leq C\left(\frac{V_0}{N_0} + \sum_{l=1}^{L}\frac{1}{N_l M_l^\beta}\right) + \frac{C}{M_L}$$

for $\beta \geq 1$ and some constant C. Consequently taking $L = O(|\log\varepsilon|)$,

$$N_0 = \varepsilon^{-2}K_L\left(\frac{V_0}{M_0}\right)^{1/2} \text{ and } N_l = \varepsilon^{-2}K_L\frac{1}{M_l},$$

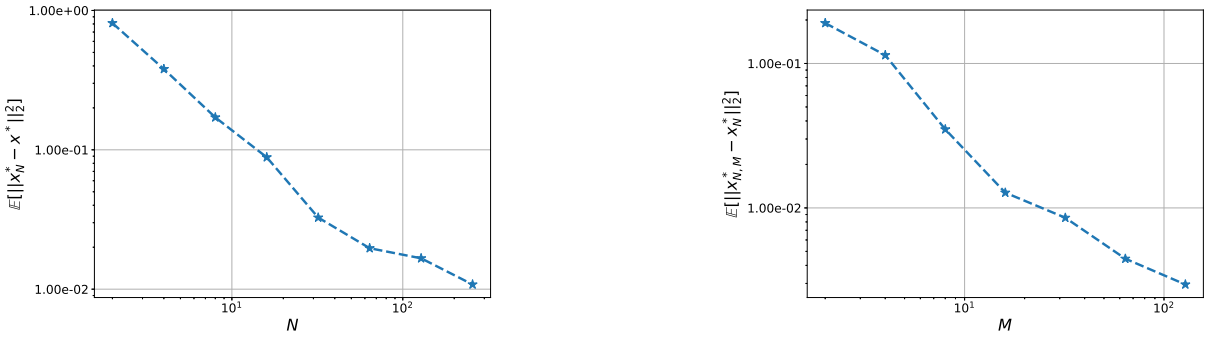
for $l \geq 1$ with $K_L = (V_0M_0)^{1/2} + L$, the MLMC estimator (9) gives the following error estimate, for some $x^* \in S^*$,

$$\mathbb{E}[\|x^*_{\mathsf{ML}} - x^*\|^2] \leq C\varepsilon^2\,,$$

for a complexity of $O(\varepsilon^{-2}(\log\varepsilon)^2)$.

Proof. We have

$$\begin{aligned}
\mathbb{E}[\|x^*_{\mathsf{ML}} - x^*\|^2] &\leq \mathbb{E}\left[\left\|\sum_{l=0}^{L}\Delta z_{N_l,M_l} - \sum_{l=0}^{L}(z_{\infty,M_l} - z_{\infty,M_{l-1}}) + \sum_{l=0}^{L}(z_{\infty,M_l} - z_{\infty,M_{l-1}}) - x^*\right\|^2\right] \\
&\leq C\sum_{l=0}^{L}\mathbb{E}\left[\left\|\Delta z_{N_l,M_l} - (z_{\infty,M_l} - z_{\infty,M_{l-1}})\right\|^2\right] + C\mathbb{E}[\|z_{\infty,M_L} - x^*\|^2] \\
&\leq C\left(\frac{V_0}{N_0} + \sum_{l=1}^{L}\frac{1}{N_l M_l^\beta}\right) + \frac{C}{M_L}
\end{aligned}$$



(a) (b)

Figure 4. Sample average approximation rate of convergence. The Matern kernel is applied with six observations. (a): convergence with respect to N. Rate of regression: -0.92. (b): convergence with respect to M with fixed $N_l = 2^5$. Rate of regression: -1.05. 100 realizations are used for both plots.

for $\beta = 1$, where the last line is due to Assumption 3, Lemma 1 and SAA results. The bias of $O(1/M_L) = O(2^{-L})$ requires one to choose $L = O(|\log \varepsilon|)$ for the desired MSE. We have total computational cost Cost$= \sum_{l=0}^{L} M_l N_l$. Minimising the cost for a fixed variance $\frac{V_0}{N_0} + \sum_{l=1}^{L} \frac{1}{N_l M_l} = \varepsilon^2$ using Lagrange multipliers as Lemma 3 gives

$$N_0 = \varepsilon^{-2} K_L (V_0 M_0)^{1/2} \quad \text{and} \quad N_l = \varepsilon^{-2} K_L M_l^{-(1+\gamma)/2}, \quad \text{for} \quad l \geq 1,$$

for

$$K_L = (V_0 M_0)^{1/2} + C \sum_{l=1}^{L} M_l^{-(1-1)/2} = (V_0 M_0)^{1/2} + CL,$$

and hence COST $= \varepsilon^{-2} K_L^2$ with $K_L = O(|\log \varepsilon|)$. We recover the borderline complexity. □

H. Numerical Results

H.1. Inner and outer Monte Carlo rates

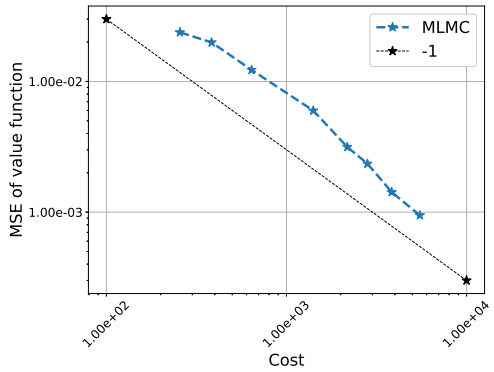
We test the inner and outer rates using the 1D toy example 1 introduced before. Here, we apply (15) with EIs and assume that we can only approximate the inner EI using MC estimation, i.e., we have

$$\alpha_{1,N,M}(x) = \mathsf{EI}(x|\mathcal{D}) + \frac{1}{N} \sum_{i=1}^{N} \max_{x_1} \frac{1}{M} \sum_{j=1}^{M} (f^j(x_1) - f^*(\{\mathcal{D}_n, (x, f^i(x))\}))_+. \tag{34}$$

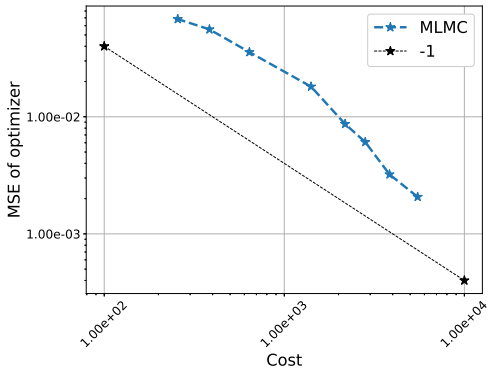
The reason for using (15) is that we have a benchmark for the inner expectation.

Rate with respect to N: We test the convergence with respect to N using equation (34) with inner MC replaced by the analytical solution of EI. The reference solution is computed with $N = 2^{12}$. According to Figure 4(a), the maximizer converges with $O(N^{-1})$.

Rate with respect to M: For the convergence with respect to M, we fix the number of sample $N = 2^5$ for the outer Monte Carlo of equation (34) and then let M varies. Note that the approximation is sensitive to the base sample, so we fixed the base samples (ξ^i) for the outer MC. Figure 4(b) illustrates that the maximizer converges with $O(M^{-1})$.



(a) Multilevel value function



(b) Optimizer of the multilevel value function

Figure 5. Complexity of MLMC value function and the corresponding optimizer. (a): a fitted slope of -1.08. (b): a fitted slope of -1.16. 200 realizations are used for both plots.

H.2. Corollary 1 and Corollary 2

Figure 5(a) and 5(b) verify the complexity of Corollary 1 and Corollary 2 where the rate MSE^{-1} occurs for large cost due to the log penalty. It is possible that the constant of the multilevel value function estimator is smaller than that of the multilevel optimizer estimator, see Figure 2(b) and 5(b), but this benefit is not practically significant due to the higher computational cost of the multilevel value function estimator, discussed in Appendix H.4, and the feature of BO which requires repeatedly solving for new observations.

H.3. MLMC variance assumptions

We now numerically verify the multilevel variance assumptions.

Multilevel construction without antithetic approach: Figure 6(a) shows $\beta = 1$ as we expected.

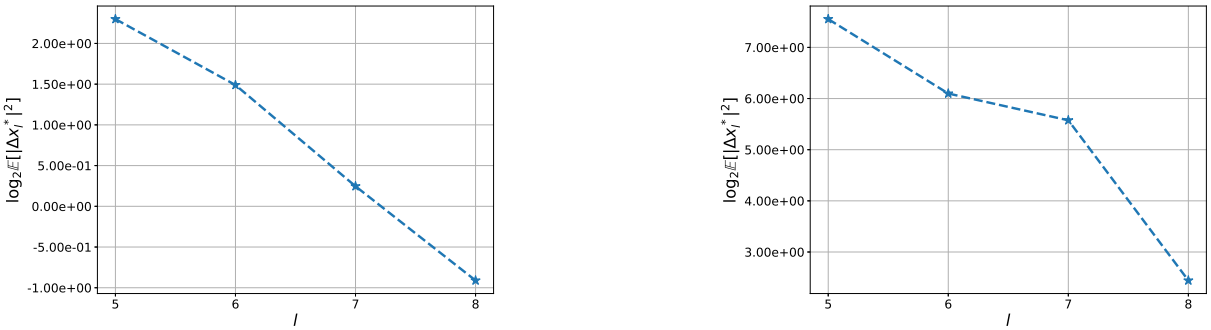
Multilevel construction with antithetic approach: Figure 6(b) shows $\beta \approx 1.5$ as we expected.

It is noted that the variance of the increments can be large. It is the case where some of the increments can be hundreds of times larger or smaller than others *for small sample sizes*. This effect, of course, vanishes asymptotically. In addition, the constant for the increments is smaller than that of the antithetic increments.

H.4. Results related to Theorem 1, and Corollaries 1, 2

Constructing the multilevel value function as Corollary 1 and seeking for optimizer requires solving a higher dimensional joint optimization problem which can be costly than solving multiple low dimensional problems as constructing the multilevel estimator for optimizer directly as Theorem 2 and Remark 1. See Figure 7(b), which indicates the cost for the multilevel value function is higher than the multilevel optimizer in expectation. Figure 7(c) shows computational times of multilevel value function and multilevel optimizer with different numbers of observations. This mimics the whole BO algorithm. A single estimation for the multilevel value function may be more computationally efficient than that for the multilevel optimizer, but in expectation that does not hold. Figure 7(a) shows that BO with the multilevel value function estimation even performs worse than BO with MC, but the purpose of introducing the value function estimation is for theoretical analysis and the numerical disadvantage is expected.

Figure 7(a) shows better performance. (Though use of the antithetic trick may improve performance for value functions.)



(a) Multilevel increments

(b) Antithetic multilevel increments

Figure 6. Rate of convergence of increments. (a): β of the multilevel increments with a fitted slope -1.09. (b): β of the multilevel antithetic increments with a fitted slope -1.59. A fixed $N_l = 2^5$ is applied for all levels, and 50 realisations are used for both plots.

H.5. Further Discussion of Multi-step Look-ahead

Figure 8 [reproduced from (Wu & Frazier, 2019)] illustrates the benefit of the look-ahead acquisition function 2-OPT in comparison to myopic alternatives. Follow-on work has demonstrated that the value of look-ahead is even greater for *constrained optimization* (Lam & Willcox, 2017; Zhang et al., 2021), which is a context we intend to explore in the future. See also (Jiang et al., 2020b) for comparisons between more multi-step look-ahead candidates and other acquisition functions, again showing the benefit of being non-myopic.

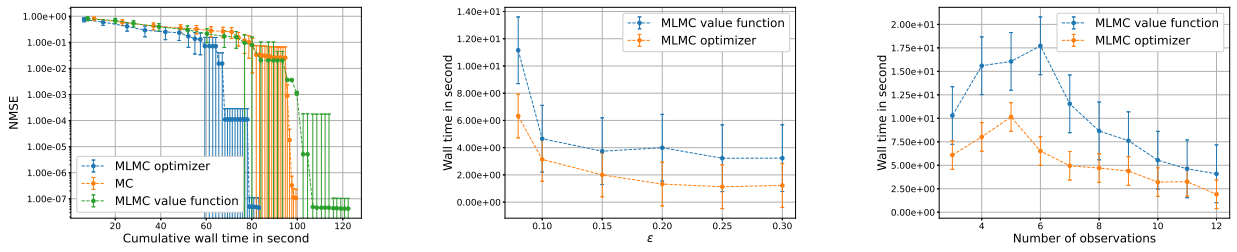
We emphasize here that the goal of this paper is to introduce a method that can improve the *computational efficiency* of multi-step look-ahead acquisition functions, either in terms of a wall time for a given accuracy or accuracy for a given wall time. And that gain is asymptotic, so that *the greater the accuracy the greater the gain in efficiency* (beware the converse that for very low accuracy there may not be any gain). We leave an exhaustive exploration of this space for the ideal acquisition function to future work. The recently introduced logEI family are compelling candidates (Ament et al., 2024).

H.6. Results with two-step look-ahead 2-expected improvement (18)

We now consider the two-step look-ahead 2-EI (18) where the one- and two-step is a 2-EI. Figure 9 shows the benefits of using MLMC over MC for 2D Ackley function, 2D DropWave and 2D Shekel function.

H.7. Matching strategies

Two greedy matching strategies are illustrated in Figures 10(a) and 10(b).



(a) (b) (c)

Figure 7. Computational times of multilevel value function and multilevel optimizer. Figure 7(a): convergence of the BO algorithm with respect to the cumulative wall time in seconds, with error bars (computed with 20 realizations). The Matern kernel is applied. The initial BO run starts with 2 observations. Figure 7(b): time spent by multilevel value function and multilevel optimizer with respect to the required accuracy of the approximation. Figure 7(c): time spent by multilevel value function and multilevel optimizer with respect to the different number of observations. The 1D toy example is applied. Each line is computed with 50 realizations

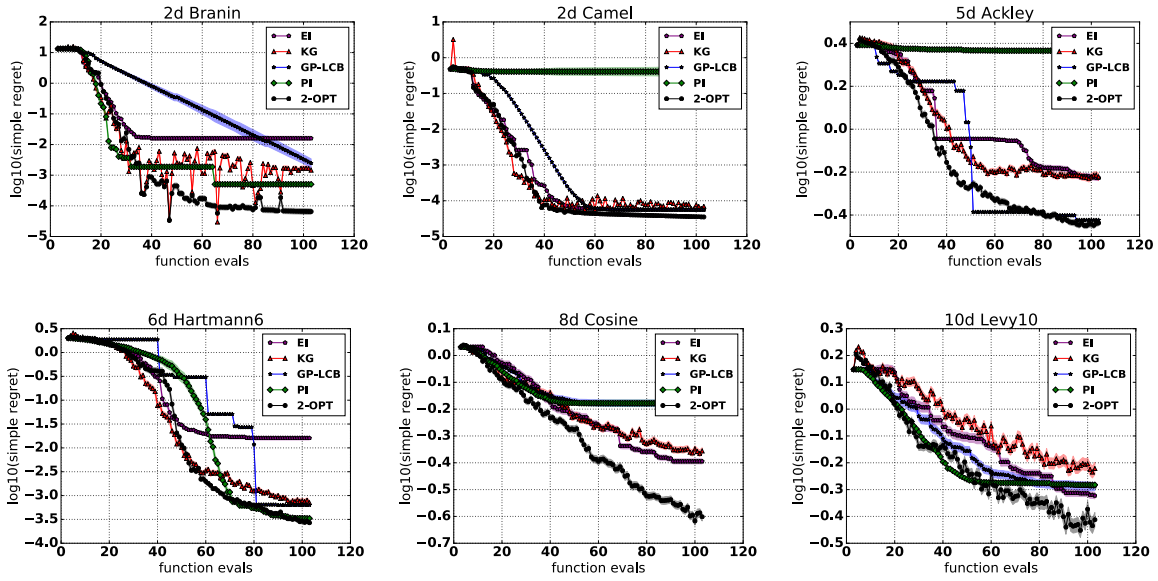
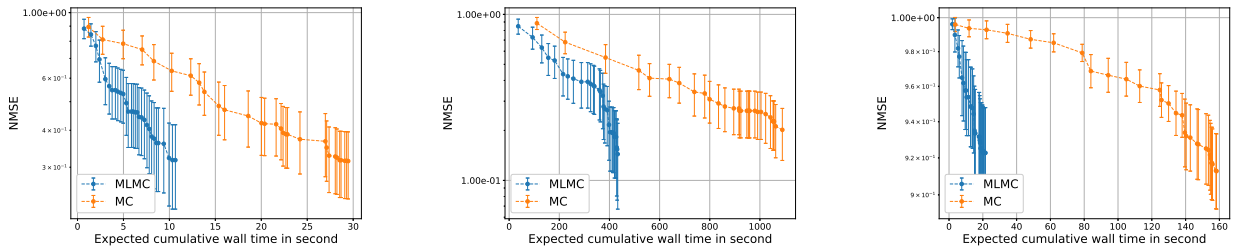


Figure 8. [Reproduced with permission from (Wu & Frazier, 2019)] Synthetic test functions, 90% quantile of log10 immediate regret compared with common one-step heuristics. 2-OPT[, 2-step lookead EI implemented with importance sampling and stochastic gradient descent,] provides substantially more robust performance.

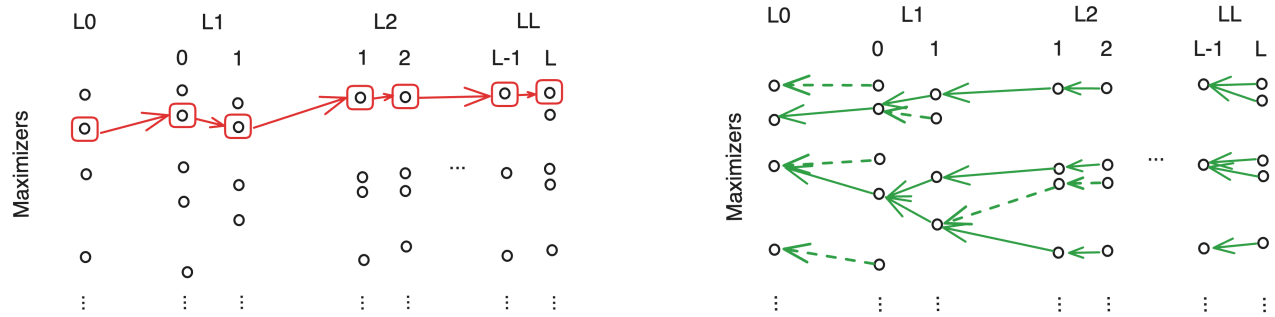


(a) Ackley (d=2)

(b) DropWave (d=2)

(c) Shekel (d=2)

Figure 9. Convergence of the BO algorithm with respect to the cumulative wall time in seconds for 2-step look-ahead 2-EI, with error bars (computed with 20 realizations). The Matérn kernel is applied. The initial BO run starts with $2 \times d$ observations.



(a) Point matching: all levels align with the best approximation of the global maximizer at level 0.

(b) Backward matching: a set of local maximizers are found at each subsequent level and pruned backwards (dashed arrows are pruned branches).

Figure 10. Greedy matching strategies. One candidate of the type (9) is considered in (a), whereas (b) results in an estimator of the type (9) for each local optimizer found at level L (or 0, if pruned in the opposite direction).