Pareto Front-Diverse Batch Multi-Objective Bayesian Optimization

Alaleh Ahmadianshalchi[*1]**, Syrine Belakaria**[*2]**, Janardhan Rao Doppa**[1]

[1] School of EECS, Washington State University
[2] Computer Science Department, Stanford University
a.ahmadianshalchi@wsu.edu, syrineb@stanford.edu, jana.doppa@wsu.edu

Abstract

We consider the problem of multi-objective optimization (MOO) of expensive black-box functions with the goal of discovering high-quality and diverse Pareto fronts where we are allowed to evaluate a batch of inputs. This problem arises in many real-world applications including penicillin production where diversity of solutions is critical. We solve this problem in the framework of Bayesian optimization (BO) and propose a novel approach referred to as **P**areto front-**D**iverse Batch Multi-Objective **BO** (PDBO). PDBO tackles two important challenges: 1) How to automatically select the best acquisition function in each BO iteration, and 2) How to select a diverse batch of inputs by considering multiple objectives. We propose principled solutions to address these two challenges. First, PDBO employs a multi-armed bandit approach to select one acquisition function from a given library. We solve a cheap MOO problem by assigning the selected acquisition function for each expensive objective function to obtain a candidate set of inputs for evaluation. Second, it utilizes Determinantal Point Processes (DPPs) to choose a Pareto-front-diverse batch of inputs for evaluation from the candidate set obtained from the first step. The key parameters for the methods behind these two steps are updated after each round of function evaluations. Experiments on multiple MOO benchmarks demonstrate that PDBO outperforms prior methods in terms of both the quality and diversity of Pareto solutions.

1 Introduction

A wide range of science and engineering applications, including materials design (Ashby 2000), biological sequence design (Taneda 2015), and drug/vaccine design (Nicolaou and Brown 2013) involves optimizing multiple *expensive-to-evaluate* objective functions. For example, in nanoporous materials design (Deshwal, Simon, and Doppa 2021), the goal is to optimize the adsorption property and cost of synthesis guided by physical lab experiments. Since the experiments are expensive in terms of the consumed resources, our goal is to approximate the *optimal Pareto set* of solutions. In many of the aforementioned applications, there are two important considerations. First, we can perform multiple parallel experiments which should be leveraged to accelerate the discovery of high-quality solutions. Second, practitioners care about

diversity in solutions and their outcomes. For example, in penicillin production application, diverse solutions for objectives including penicillin production, the time to ferment, and the CO_2 byproduct (Birol, Ündey, and Cinar 2002).

We consider the problem of multi-objective optimization (MOO) over expensive-to-evaluate functions to find high-quality and diverse Pareto fronts when we are allowed to perform a batch of experiments. We solve this problem using Bayesian optimization (BO) (Shahriari et al. 2015) which has been shown to be highly effective for such problems. The key idea behind BO is to learn a surrogate model (e.g., Gaussian process) from past experiments and use it to intelligently select the sequence of experiments guided by an acquisition function (e.g., expected improvement). There is prior BO work on selecting a batch of experiments to find diverse high-performing solutions for single-objective optimization. However, there is very limited work on batch BO for MOO problems and to produce diverse MOO solutions. A key drawback of the existing BO methods for MOO is that they evaluate diversity in terms of input space, which is not appropriate for MOO (diversity in input space $\nRightarrow$ diversity in output space). To overcome this drawback and to measure the diversity of MOO solution in the output space, we define a new metric referred to as *Diversity of the Pareto Front (DPF)*.

We propose a novel approach referred to as **P**areto front-**D**iverse Batch Multi-Objective **BO** (PDBO). PDBO selects a batch B of inputs for evaluations in each iteration using two main steps. First, it employs a principled multi-arm bandit strategy to dynamically select one acquisition function (AF) from a given library of AFs within the single-objective BO literature. A cheap MOO problem is solved by assigning the selected AF for each expensive objective function to obtain a Pareto set. Second, a principled configuration of determinantal point processes (DPPs) (Borodin 2009; Kulesza, Taskar et al. 2012) for multiple objectives is used to select B inputs for evaluation from the Pareto set of the first step to improve the diversity of the uncovered Pareto front. PDBO updates the key parameters of the algorithms for these two steps (dynamic selection of AF and selecting B Pareto-front diverse inputs from a candidate Pareto set) after obtaining the objective function evaluations. Our experiments on multiple benchmarks with varying input dimensions and number of objective functions demonstrate that PDBO outperforms prior methods in finding high-quality and diverse Pareto fronts.

*These authors contributed equally.

Contributions. The key contribution of this paper is developing and evaluating the PDBO approach for solving MOO problems to find high-quality and diverse Pareto fronts.

Specific contributions are as follows:

- A multi-arm bandit strategy with a novel reward function to dynamically select one acquisition function from a given library for MOO problems.

- A novel DPP method for selecting a batch of inputs from a given Pareto set to maximize Pareto front diversity using a new mechanism to generate the DPP kernel for MOO.

- To demonstrate the effectiveness of PDBO and study the Pareto front diversity, we propose a new metric to measure the diversity of *Pareto fronts*. To the best of our knowledge, this is the first attempt at extensive experimental evaluation of Pareto front diversity within BO.

- Theoretical analysis of our PDBO algorithm in terms of asymptotic regret bounds.

- Experimental evaluation of PDBO and baselines on multiple benchmark MOO problems. The code for PDBO is publicly available at https://github.com/Alaleh/PDBO.

2 Problem Setup and Background

We first formally define the batch MOO problem along with the metrics to evaluate the quality and diversity of solutions. Next, we provide an overview of the BO framework.

Batch Multi-Objective Optimization. We consider a MOO problem where the goal is to optimize multiple conflicting functions. Let $\mathfrak{X} \subset \mathbb{R}^d$ be the input space of d design variables, where each candidate input $\mathbf{x} \in \mathfrak{X}$ is a d-dimensional input vector. And let $\{f_1, \cdots, f_K\}$ with $K \geq 2$ be the objective functions defined over the input space $\mathfrak{X}$ where $f_1(\mathbf{x}), \cdots, f_K(\mathbf{x}) : \mathfrak{X} \to \mathbb{R}$. We denote the functions evaluation at an input $\mathbf{x}$ as $\mathbf{y} = [y_1, \cdots, y_K]$, where $y_i = f_i(\mathbf{x})$ for all $i \in \{1, \cdots, K\}$. Without loss of generality, we assume minimization for all K objective functions. The optimal solution of the MOO problem is a set of inputs $\mathcal{X}^* \subset \mathfrak{X}$ such that no input $\mathbf{x}' \in \mathfrak{X} \setminus \mathcal{X}^*$ Pareto-dominates another input $\mathbf{x} \in \mathcal{X}^*$. An input $\mathbf{x}$ Pareto-dominates another point $\mathbf{x}'$ if and only if $\forall j : f_j(\mathbf{x}) \leq f_j(\mathbf{x}')$ and $\exists j : f_j(\mathbf{x}) < f_j(\mathbf{x}')$. The set of input solutions $\mathcal{X}^*$ is called the optimal *Pareto set* and the corresponding set of function values $\mathcal{Y}^*$ is called the optimal *Pareto front*. We can select B inputs for parallel evaluation in each iteration, and our goal is to uncover a high-quality and diverse Pareto front while minimizing the total number of expensive function evaluations.

Metrics for Quality and Diversity of Pareto Front. Our goal is to find high-quality and diverse Pareto fronts. The diversity of the Pareto front has not been formally *evaluated* in any previous work. We introduce an appropriate evaluation metric to measure the diversity of the Pareto front and discuss an existing measure of Pareto front quality below.

Diversity of Pareto Front. Diversity is an important criterion for many optimization problems. Prior work on batch BO, both in the single-objective and MO settings focused on evaluating diversity with respect to the input space (Jain et al. 2022). However, in most real-world MOO problems, diversity in the input space does not necessarily reflect diversity

in the output space. In MOO, practitioners might care more about the diversity of the Pareto front rather than the Pareto set. Yet, little work has gone into understanding, formalizing, and measuring Pareto front diversity in MOO. In most cases, finding a more diverse set of points in the output space leads to a higher hypervolume (Zitzler and Thiele 1999). However, a higher hypervolume does not necessarily correspond to a more diverse Pareto front. (Konakovic Lukovic, Tian, and Matusik 2020) is the only prior work that proposed a diversity-guided approach for batch MOO. However, the diversity of the produced Pareto front *was not evaluated*. To fill this gap, we propose an evaluation metric to fill this gap to assess the *Diversity of the Pareto Front (DPF)*. Given a Pareto front $\mathcal{Y}_t$, $DPF(\mathcal{Y}_t)$ is the average pairwise distance between points (i.e., output vectors) in Pareto front $\mathcal{Y}_t$. It is important to clarify that the pairwise distances are computed in the output space between different vector pairs $(\mathbf{y}, \mathbf{y}') \in \mathcal{Y}_t$, unlike previously used metrics to assess input space diversity in the single-objective setting (Angermueller et al. 2020).

$$DPF(\mathcal{Y}_t) = \frac{\sum_{(i,j) \in \mathcal{I}} ||\mathbf{y_i} - \mathbf{y_j}||}{|\mathcal{I}|}$$

$$\text{with } \mathcal{I} = \{(i,j) \forall i, j \in \{1 \cdots t\}, i < j\}$$

We provide a more detailed discussion of existing metrics and some illustrative results on previous metrics and their utility in evaluating diversity in the Appendix.

Hypervolume Indicator. The hypervolume indicator (Zitzler and Thiele 1999) is the most commonly employed measure to evaluate the quality of a given Pareto front. Given a set of functions evaluations $\mathcal{Y}_t = \{\mathbf{y}_0, \cdots, \mathbf{y}_t\}$, the Pareto hypervolume (PHV) indicator is the volume between a predefined reference point $\mathbf{r}$ and the given Pareto front.

2.1 Bayesian Optimization Framework

BO (Shahriari et al. 2015) is a general framework for solving expensive black-box optimization problems in a sample-efficient manner. BO algorithms iterate through three steps.

1) Build a probabilistic *surrogate model* of the true expensive objective function. Gaussian process (GPs) (Williams and Rasmussen 2006)) are widely employed in BO.

2) Define an *acquisition function (AF)* to score the utility of unevaluated inputs. It uses the surrogate model's predictions as a cheap substitute for expensive function evaluations and strategically trades off exploitation and exploration.

Some examples of widely used acquisition functions in single-objective optimization include expected improvement (EI) (Mockus, Tiesis, and Zilinskas 1978), upper confidence bound (UCB) (Auer 2002), Thompson sampling (TS) (Thompson 1933) and Identity (ID).

$$EI(\mathbf{x}) = \sigma(\mathbf{x})(\alpha\Phi(\alpha) + \phi(\alpha)), \alpha = (\tau - \frac{\mu(\mathbf{x})}{\sigma(\mathbf{x})}) \quad (1)$$

$$UCB(\mathbf{x}) = \mu(\mathbf{x}) + \sqrt{\beta}\sigma(\mathbf{x}) \quad (2)$$

$$TS(\mathbf{x}) = \hat{f}(\mathbf{x}) \; with \; \hat{f} \sim \text{GP} \quad (3)$$

$$ID(\mathbf{x}) = \mu(\mathbf{x}) \quad (4)$$

Here, β is a parameter to balance exploration and exploitation in the UCB acquisition function (Srinivas et al. 2009),

τ is the best-uncovered function value; and Φ and ϕ are the CDF and PDF of a standard normal distribution respectively.

3) Select the input with the highest utility score for function evaluation by optimizing the acquisition function.

3   Related Work

Multi-Objective BO. There is relatively less work on MOO in comparison with single-objective BO. Prior work builds on the insights from single-objective methods (Shahriari et al. 2015; Hernández-Lobato, Hoffman, and Ghahramani 2014; Wang and Jegelka 2017; Hvarfner, Hutter, and Nardi 2022) for MOO. Recent work on MOBO includes Predictive Entropy Search (Hernández-Lobato et al. 2016), Maxvalue Entropy Search (Belakaria, Deshwal, and Doppa 2019), Multi-Objective Regionalized BO (Daulton et al. 2022), Uncertainty-aware Search (Belakaria et al. 2020a), Pareto-Frontier Entropy Search (Suzuki et al. 2020), and Expected Hypervolume Improvement (Daulton, Balandat, and Bakshy 2020; Emmerich and Klinkenberg 2008). These methods are shown to perform well on a variety of MOO problems.

Batch Multi-objective BO. The Batch BO problem in the multi-objective setting is even much less studied. Diversity-Guided Efficient Multi-Objective Optimization (DGEMO) (Konakovic Lukovic, Tian, and Matusik 2020) approximates and analyzes a piecewise-continuous Pareto set representation which allows the algorithm to introduce a batch selection strategy that optimizes for both hypervolume improvement and diversity of selected samples. However, DGEMO work did not study or evaluate the Pareto-front diversity of the produced solutions. qEHVI (Daulton, Balandat, and Bakshy 2020) is an exact computation of the joint EHVI of q new candidate points (up to Monte-Carlo integration error).

qPAREGO is a novel extension of ParEGO (Knowles 2006; Daulton, Balandat, and Bakshy 2020) that supports parallel evaluation and constraints. More recent work (Lin et al. 2022) proposed an approach to address MOO problems with continuous/infinite Pareto fronts by approximating the whole Pareto set via a continuous manifold. This approach enables a better preference-based exploration strategy for practitioners compared to prior work (Abdolshah et al. 2019; Paria, Kandasamy, and Póczos 2020; Astudillo and Frazier 2020). However, it is typically unknown to the user if the Pareto front is dense/continuous, especially in expensive function settings where the data is limited. *It is not known whether any of these batch methods produce diverse Pareto fronts or not, as they were not evaluated on diversity metrics*. We perform an experimental evaluation to answer this question.

DPPs for Batch Single-Objective BO. DPPs are elegant probabilistic models (Borodin and Olshanski 2005; Borodin 2009) that characterize the property of repulsion in a set of vectors and are well-suited for the selection of a diverse subset of inputs from a predefined set. Prior work used DPPs for selecting batches for evaluation in the single-objective BO literature (Kathuria, Deshpande, and Kohli 2016; Nava, Mutny, and Krause 2022; Wang et al. 2017). In the context of MOO for cheap objective functions using evolutionary algorithms, DPP was deployed using non-learning-based kernels such as cosine function applied to points in the Pareto front while disregarding the input space (Wang et al. 2022; Zhang et al. 2020; Okoth et al. 2022). However, *to the best of our knowledge, there is no work on using DPPs for multi-objective BO to uncover diverse Pareto fronts using a learned kernel that captures the trade-off between multiple objectives, similarity in the input space, and diversity of the Pareto front*.

Adaptive Acquisition Function Selection. There has been a plethora of research on finding efficient and reliable acquisition functions (AFs). However, prior work has shown that no single acquisition function is universally efficient and consistently outperforms all others. GP-Hedge (Hoffman et al. 2011) proposed to use a portfolio of acquisition functions. The optimization of each AF will nominate an input, and the algorithm will select one of them for evaluation using the selection probabilities. The GP-Hedge method uses the Hedge strategy (Freund and Schapire 1997), a multi-arm bandit method designed to choose one action amongst a set of different possibilities using selection probabilities calculated based on the reward (performance given by function values) collected from previous evaluations. Vasconcelos et al. extended Hoffman et al. by proposing to use discounted cumulative reward and Vasconcelos et al. suggested using Thompson sampling to automatically set the hedge hyperparameter η.

4   Proposed PDBO Algorithm

We start by providing an overview of the proposed PDBO algorithm illustrated in Figure 1. Next, we explain our algorithms for two key components of PDBO, namely, adaptive acquisition function selection via a multi-arm bandit strategy and diverse batch selection via determinantal point processes for multi-objective output space diversity.

Overview of PDBO. PDBO is an iterative algorithm. It introduces novel methods for selecting varying acquisition functions and for bringing the diversity of inputs into the multi-objective BO setting. The method builds K independent Gaussian processes $\mathrm{GP}_1, \cdots, \mathrm{GP}_K$ as surrogates for each of the objective functions. Its three key steps at each iteration t to select B inputs for evaluation are:

1. Solving multiple cheap MOO problems: PDBO takes as input a portfolio of acquisition functions, $\mathcal{P} = \{\mathrm{AF}_1, \cdots, \mathrm{AF}_M\}$ for single-objective BO. It constructs M cheap MOO problems, each corresponding to one AF. The multiple objectives defining the cheap MOO problems are acquisition functions respectively corresponding to the K objective functions. Solving cheap MOO problems will generate M cheap Pareto-sets of solutions $\mathcal{X}_c^1 \cdots \mathcal{X}_c^M$.

2. Diverse batch selection: From each cheap Pareto set $\mathcal{X}_c^j$, a batch $X_t^{B_j} \subset \mathcal{X}_c^j$ of B inputs is selected using a diversity-aware approach based on DPPs. Importantly, the adapted DPP is configured to favor the diversity in the output space and to handle multiple objective settings by using a principally fitted convex combination of the kernels of the K Gaussian processes. The convex combination scalars are strategically set to maximize the likelihood of selecting a diverse subset of inputs with respect to the Pareto front.

3. Acquisition function selection: From $\{X_t^{B_j}; \ \forall \ j \in [1, \cdots, M]\}$, only one nominated subset would be selected
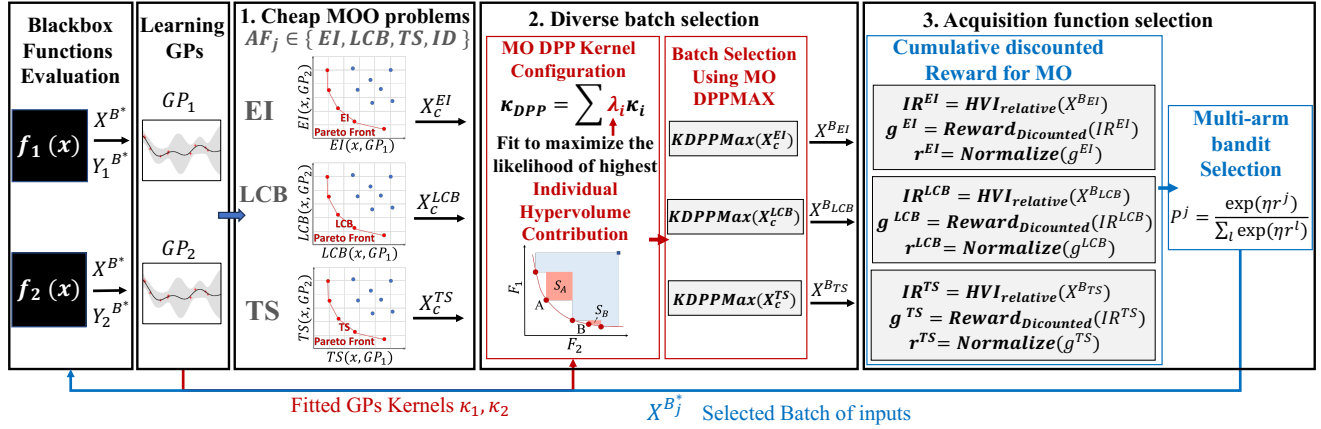
Figure 1: High-level overview of PDBO algorithm illustrating its three key components.

using a multi-arm bandit strategy. The keys to this selection are probabilities $p^1, \cdots, p^j$, one for each acquisition function to capture their performance based on past iterations. p^j is the probability of selecting the batch generated by the acquisition function AF_j, defined in equation 9. These probabilities are updated based on the discounted cumulative reward r^j of each of the respective acquisition functions. The reward values r^j are updated based on the quality of the batches nominated by the respective acquisition functions.

Algorithm 1 provides a pseudocode with high-level steps of the PDBO approach. The DPP-SELECT and ADAPTIVE-AF-SELECT represent the second and third key steps. The details of these methods and their corresponding pseudocodes are provided in Sections 4.1 and 4.2, respectively.

4.1 Multi-arm Bandit Strategy for Adaptive Acquisition Function Selection

In this section, we propose a multi-arm bandit (MAB) approach to *adaptively* select one acquisition function (AF) from a given library of AFs in each iteration of PDBO.

Multi-arm Bandit Formulation. We are given a portfolio of M acquisition functions $\mathcal{P} = \{\mathrm{AF}_1, \cdots, \mathrm{AF}_M\}$ and our goal is to adaptively select one AF in each iteration of PDBO. Each acquisition function in $\mathcal{P}$ corresponds to one arm, and we need to select an arm based on the performance of past selections for solving the MOO problem. Inspired by previous work on AF selection and algorithm selection in the single objective setting (Hoffman et al. 2011; Vasconcelos et al. 2019, 2022), we propose an adaptive acquisition function selection approach for the MOO setting (see Algorithm 2). We explain the two main steps of this approach below.

Nominating Promising Candidates via Cheap MOO. In each PDBO iteration, we employ the updated statistical models $\{\mathrm{GP}_1 \cdots \mathrm{GP}_K\}$ and the portfolio $\mathcal{P}$ to generate M sets of candidate points. For each AF_j, the algorithm constructs a cheap MOO problem with the objectives defined as $\mathrm{AF}_j(\mathrm{GP}_1, \mathbf{x}) \cdots \mathrm{AF}_j(\mathrm{GP}_K, \mathbf{x})$. Assuming minimization, the cheap MOO generate M Pareto-sets of solutions $\mathcal{X}_c^1 \cdots \mathcal{X}_c^M$

Algorithm 1: Pareto front-Diverse Batch Multi-Objective BO

Input: $\mathfrak{X}$ input space; $\{f_1, \cdots, f_K\}$, K black-box objective functions; $\mathcal{P} = \{\mathrm{AF}_1, \cdots, \mathrm{AF}_M\}$ portfolio of acquisition functions; B batch size; and T_{max} number of iterations

1: Initialize data $\mathcal{D}_0 = \{\mathcal{X}_0, \mathcal{Y}_0\}$ with N_0 initial points
2: **for** each iteration $t \in [1, T_{max}]$ **do**
3:     Fit statistical models $\mathrm{GP}_1, \cdots, \mathrm{GP}_k$ using $\mathcal{D}_{t-1}$
4:     **for** each acquisition function $\mathrm{AF}_j \in \mathcal{P}$ **do**
5:         $\mathcal{X}_c^j \leftarrow \arg min_{\mathbf{x} \in \mathfrak{X}} \ (\mathrm{AF}_j(\mathrm{GP}_1, \mathbf{x}), \cdots, \mathrm{AF}_j(\mathrm{GP}_k, \mathbf{x}))$
        // Solve cheap MOO problem
6:         $X_t^{B_j} \leftarrow$ DPP-SELECT$(\mathcal{X}_c^j, \{\mathrm{GP}_1, \cdots, \mathrm{GP}_k\}, \mathcal{D}_{t-1})$
        // Select a batch of inputs from $\mathcal{X}_c^j$ using DPPs
7:     **end for**
8:     $\mathrm{AF}_{j^*} \leftarrow$ ADAPTIVE-AF-SELECT$(\mathcal{D}_{t-1}, \{X_{t-1}^{B_j}\}_{j=1}^M)$
    // Select an AF using previously aggregated data
9:     $X_t^B = X_t^{B_{j^*}}$ // Choose the batch nominated by AF_{j^*}
10:     $Y_t^B \leftarrow \{[f_1(\mathbf{x}), \cdots, f_k(\mathbf{x})]; \ \forall \ \mathbf{x} \in X_t^B\}$
    // Evaluate objective functions for batch of inputs X_t^B
11:     $\mathcal{D}_t = \{\mathcal{X}_t, \mathcal{Y}_t\} \leftarrow \{\mathcal{X}_{t-1}, \mathcal{Y}_{t-1}\} \cup \{(X_t^B, Y_t^B)\}$
12: **end for**
13: **return** Pareto set $\mathcal{X}_{T_{max}}$ and Pareto front $\mathcal{Y}_{T_{max}}$

(one for each acquisition function) defined as:

$$\mathcal{X}_c^j \leftarrow \arg \min_{\mathbf{x} \in \mathfrak{X}} \{\mathrm{AF}_j(\mathrm{GP}_1, \mathbf{x}), \cdots, \mathrm{AF}_j(\mathrm{GP}_K, \mathbf{x})\} \quad (5)$$

We employ the algorithm proposed by Deb et al. to solve cheap MOO problems defined in 5. From each $\mathcal{X}_c^j$, a batch $X_t^{B_j} \subset \mathcal{X}_c^j$ of B points is selected in iteration t using a diversity-aware approach described in Section 4.2. We denote the function evaluations of inputs in $X_t^{B_j}$ by $Y_t^{B_j}$.

Multi-Objective Reward Update. We employ the relative hypervolume improvement as the quality metric to define our reward. The Pareto hypervolume captures the quality of nominated batches from a Pareto-dominance perspective and carries information about the represented trade-off between the multiple objectives. Defining the immediate reward as the raw Pareto hypervolume of the nominated batch $IR_t^j = HV(Y_t^j)$ can lead to an undesirable assessment of

the suitable acquisition function since a batch of points can have a large hypervolume value at iteration t but does not provide a significant improvement over the previous Pareto front $\mathcal{Y}_{t-1}$ while another batch nominated in iteration $t-1$ may have a smaller hypervolume $HV(Y_{t-1}^j)$ yet provides a higher improvement over $\mathcal{Y}_{t-2}$. Additionally, initial iterations might provide drastic hypervolume improvements even if the selected points are not optimal. To mitigate these issues, we use the relative hypervolume improvement as the immediate reward instead of the hypervolume. In each BO iteration t, the immediate reward IR_t^j for each acquisition function AF_j; $\forall j \in \{1 \cdots M\}$ is defined as follows.

$$IR_t^j = \frac{HV(\tilde{\mathcal{Y}}_{t-1} \cup \tilde{Y}_t^j) - HV(\tilde{\mathcal{Y}}_{t-1})}{HV(\tilde{\mathcal{Y}}_{t-1})} \quad (6)$$

where $\tilde{\mathcal{Y}}_{t-1}$ is the Pareto front at iteration $t-1$ and $\tilde{Y}_t^j$ is the evaluation of the batch of points X_t^j nominated by AF_j computed using the *predictive mean of the updated GP based statistical models*. As optimization progresses, statistical models provide a better representation of the objective functions, and the batches nominated by each AF become more informative about the quality of its selections. Hence, the impact of the early iterations may become irrelevant later. Consequently, we employ a *discounted cumulative reward* for each acquisition function AF_j at iteration t (denoted g_t^j).

$$g_t^j = \gamma g_{t-1}^j + IR_t^j = \sum_{t' \le t} \gamma^{t'-1} IR_{t'}^j \quad (7)$$

Where γ is a decay rate that trades off past and recent improvements. The use of the decay rate can lead to equal or comparable rewards in advanced iterations, causing the algorithm to select the acquisition function randomly. To address this problem, the discounted cumulative reward g_t^j should be normalized (Vasconcelos et al. 2019). The rewards at the first iteration are all initialized to zero and then updated at each iteration t using the following expression:

$$r_t^j = \frac{g_t^j - g_{max}^j}{g_{max}^j - g_{min}^j} \quad (8)$$

where $g_{max}^j = max(\{g_{t'}^j, t' \in [1, t]\})$ and $g_{min}^j = min(\{g_{t'}^j, t' \in [1, t]\})$. Finally, the probability of selection of each AF at each iteration t is calculated using equation 9.

$$p_t^j = \frac{exp(\eta r_t^j)}{\sum_{l=1}^M exp(\eta r_t^l)} \; for \; j = 1 \ldots M \quad (9)$$

The proposed MAB approach is an extension of the Hedge algorithm but is fundamentally distinct in its methodology and applicability. While it does generalize certain aspects of Hedge, it introduces critical variations that make it unique within the context of this study. Our approach diverges from the original Hedge algorithm by incorporating two key modifications: **1)** The use of discounted rewards and the application of normalization. Unlike the conventional Hedge, where rewards are typically used without any discounting or normalization, our strategy accounts for these factors, enhancing

---

Algorithm 2: ADAPTIVE-AF-SELECT

Input: training data $\mathcal{D}_t$; Batches nominated by different AFs $\{X_t^{B_j}, \forall j \in \{1 \cdots M\}\}$

1: **if** $t == 0$ **then**
2: $\quad$ $r_t^j = 0$ for $j = 1 \cdots M$
3: **else**
4: $\quad$ Compute rewards: r_t^j for $j = 1 \cdots M$ using Equation 8
5: $\quad$ Update probabilities: $p_t^j = \frac{exp(\eta r_t^j)}{\sum_{l=1}^M exp(\eta r_t^l)}$ for $j = 1 \cdots M$
6: **end if**
7: Select AF_{j*} according to the probabilities $\{p_t^j\}_{j=1}^M$
8: **return** acquisition function AF_{j*}

---

its adaptability to the specific problem domain. **2)** Another significant departure lies in the problem setting itself. Hedge was initially designed for single-objective optimization while our proposed approach solves the more challenging problem of multi-objective optimization. This shift in focus has substantial implications, as it requires an entirely different set of considerations and techniques to address the complexities introduced by multiple conflicting objectives.

Remark. It is important to note that we are using a full information multi-arm bandit strategy that requires the reward to be updated for all possible actions (i.e., for all acquisition functions) at each iteration. Since we evaluate only the batch nominated by the selected AF, we achieve this by computing the reward using the predictive mean functions of the *updated* surrogate models. For this reason, we solve a cheap MOO problem for each $\text{AF}_j \in \mathcal{P}$ even though the acquisition function is selected based on the data from the previous iterations. Algorithm 2 provides the pseudocode of the adaptive AF selection based on the estimated rewards and probabilities.

4.2 DPPs for Batch Selection

We explain our approach to select a batch of diverse inputs by configuring DPPs to promote *output space diversity*.

Determinantal Point Processes. (DPPs) (Kulesza, Taskar et al. 2012) are well-suited to model samples of a diverse subset of k points from a predefined set of n of points. Given a similarity function over a pair of points, DPPs assign a high probability of selection to the most diverse subsets according to the similarity function. The similarity function is typically defined as a kernel. Formally, given a DPP kernel defined over a set S of n elements, the k-DPP distribution is defined as selecting a subset S' of size k with $S' \subset S$ with probability proportional to the determinant of the kernel:

$$Pr(S') = \frac{det(\kappa(S'))}{\sum_{|s|=k} det(\kappa(s))} \quad (10)$$

(Kathuria, Deshpande, and Kohli 2016) introduced the use of DPP for batch BO in the single-objective setting. Given the surrogate GP of the objective function, the covariance of the GP is used as the similarity function for the DPP. The approach selects the first point in the batch by maximizing the UCB acquisition function. Next, it creates a set of points referred to as relevance region by bounding the search space with the maximizer of the LCB acquisition function and manually discretizing the bounded space into a grid of n points.

DPP selects the remaining $(k-1)$ points out of the n points in the relevance region. (Wang et al. 2017; Oh et al. 2021; Nava, Mutny, and Krause 2022) used similar techniques to apply DPPs to high-dimensional and discrete spaces.

There exist two approaches to selecting a diverse subset with a fixed size via DPP: 1) Choosing the subset that maximizes the determinant referred to as *DPP-max*; and 2) Sampling with the determinantal probability measure referred to as *DPP-sample*. In this paper, we will focus on DPP-max. Although selecting the subset that maximizes the determinant is an NP-Hard problem, several approximations were proposed (Nikolov 2015). A greedy strategy (Kathuria, Deshpande, and Kohli 2016) provides an approximate solution and was adopted in several BO papers (Wang et al. 2017).

Limitations of Prior Work and Challenges for MOO. We list the key limitations of prior methods for DPP-based batch selection in the single-objective setting as they are applicable to the multi-objective setting too. **L1)** How can we overcome the limitation of selecting the first point separately regardless of the DPP diversity? **L2)** How can we prevent the potential limitation of under-explored search space caused by the discretization of the space to create the relevance region set?

The key challenges to employing DPPs for batch selection in the MOO setting include **C1)** How to define a kernel that captures the diversity for multiple objectives given that we have K separate surrogate models and their corresponding kernel? **C2)** How can the DPP kernel capture the Pareto front diversity and the trade-off between the objectives without compromising the Pareto quality of selected points?

DPPs for Multi-objective BO. We propose principled methods to address the limitations of prior work on DPPs for batch BO (**L1 and L2**) and the challenges **C1** and **C2** for MOO.

Multi-objective Relevance Region. Our proposed algorithm naturally mitigates the two limitations of the single-objective DPP approach. Recall that the first step of PDBO algorithm (Section 4.1) proposes to generate cheap approximate Pareto-sets which capture the trade-offs between the objectives in the utility space and might include, with high probability, optimal points (Belakaria et al. 2020a; Konakovic Lukovic, Tian, and Matusik 2020). We consider the cheap Pareto sets as the multi-objective relevance region. Our approach allows for generating the relevance region without manually discretizing the search space. Also, the full batch is selected from the multi-objective relevance region leading to a better diversity among all the points in the batch.

Multi-objective DPP Kernel Fitting. To overcome the challenges of using DPPs in the MOO setting, we build a new kernel κ_{dpp} that is defined as a convex combination of the K kernels of the statistical models (GPs) representing each of the black-box objective functions. Let $\Lambda = [\lambda_1, \cdots, \lambda_K]$ be a vector of size K where each λ_i corresponds to the convex combination scalar associated with kernel κ_i of the objective function f_i. The DPP kernel κ_{DPP} is defined as:

$$\kappa_{DPP} = \sum_{i=1}^{K} \lambda_i \cdot \kappa_i \;\; st. \sum_{i=1}^{i=K} \lambda_i = 1 \qquad (11)$$

The hyperparameters of the kernels $\kappa_i \; \forall\, i \in \{1, 2, \cdots K\}$ are

---

Algorithm 3: DPP-SELECT

Input: cheap Pareto set $\mathcal{X}_c$; surrogate models $\mathrm{GP}_1, \cdots, \mathrm{GP}_k$; data $\mathcal{D}_t$.

1: $\mathbf{C}_t = [HVC(\mathbf{y}), \quad \forall\ \mathbf{y} \in \mathcal{Y}_t]$ // Calculate the individual hypervolume contribution for each input $\in \mathcal{D}_t$
2: $\kappa_{DPP} = \sum_{i=1}^{K} \lambda_i \cdot \kappa_i \; st. \sum_{i=1}^{K} \lambda_i = 1$
  // Construct κ_{DPP} as a convex combination of function kernels
3: $\Lambda^* = \arg\min_{\Lambda \in [0,1]^K} \log p(\mathbf{C}_t | \mathcal{X}_t) \quad s.t \sum_{i=1}^{K} \lambda_i = 1$
  // Select λ_i values by maximizing the LML
4: Use the fitted κ_{DPP} kernel to select the most diverse points X_t^B from the cheap Pareto set $\mathcal{X}_c$ via DPP-max
5: **return** the selected B inputs, X_t^B

---

fixed during the fitting of the GPs. To set the convex combination scalars Λ in a principled manner that promotes diverse batch selection, we propose to set the Λ by maximizing the log marginal likelihood of selecting points with the highest individual hypervolume contribution.

The individual hypervolume contribution (HVC) of each point in the evaluated Pareto front $\mathcal{Y}_t$ (via evaluated training data D_t) is the reduction in hypervolume if the point is removed from the Pareto front. HVC is considered a Pareto front (PF) diversity indicator (Daulton et al. 2022). Points in crowded regions of the Pareto front have smaller HVC values. Therefore, more Pareto front points with high HVC indicate more output space coverage and consequently, higher PF diversity.

$$HVC(\mathbf{y}) = HV(\mathcal{Y}_t) - HV(\mathcal{Y}_t \setminus \{\mathbf{y}\}) \, \forall\, \mathbf{y} \in \mathcal{Y}_t \quad (12)$$

Given equation 12, we can construct the training set for the fitting of Λ. Let $\mathbf{C}_t = [HVC(\mathbf{y}); \; \forall\, \mathbf{y} \in \mathcal{Y}_t]$ be a vector of the individual HV contributions of evaluated points $\mathcal{X}_t \in \mathcal{D}_t$.

$$\Lambda^* = \operatorname{argmin}_{\Lambda \in [0,1]^K} \log p(\mathbf{C}_t | \mathcal{X}_t) \quad \text{s.t.} \sum_{i=1}^{K} \lambda_i = 1 \quad (13)$$

$$\text{where } \log p(\mathbf{C}_t | \mathcal{X}_t) = -\frac{1}{2}\mathbf{C}_t^T \mathcal{K_{DPP}}^{-1} \mathbf{C}_t$$
$$-\frac{1}{2}\log|\mathcal{K_{DPP}}| - \frac{n}{2}\log 2\pi$$

Algorithm 3 provides the pseudo-code for selecting the diverse batch from a given candidate/cheap Pareto set $\mathcal{X}_c$.

5 Theoretical Analysis

Prior work developed regret bound for different single objective acquisition functions including UCB (Srinivas et al. 2009; Belakaria et al. 2020a). However, the theoretical analysis of our proposed MAB algorithm is more challenging as it involves input selection based on different acquisition functions at each iteration t. The choices made at any given iteration t influence the state and subsequent rewards of all future iterations (Hoffman et al. 2011) and therefore there is a need to adapt prior theoretical analysis. Additionally, regret bounds for Hedge MAB strategies have been developed independently outside the context of acquisition function selection (Cesa-Bianchi and Lugosi 2006). We follow similar

steps suggested by Hoffman et al. (2011) to derive a suitable regret bound for our MOO setting.

We assume maximization of objectives and further assume that the UCB acquisition function is in the portfolio of acquisition functions used by PDBO. We make this choice for the sake of clarity and ease of readability as we build our theoretical analysis on prior seminal work (Srinivas et al. 2009; Hoffman et al. 2011). Notably, this is not a restrictive assumption, and with minimal mathematical manipulations, the same derived regret bound holds for the case of minimization with the LCB acquisition function being in the portfolio instead.

To simplify the proof and solely for the sake of theoretical regret bound, we consider the instant reward at iteration t to be the sum of predictive means of the Gaussian processes

$$IR_t = \sum_{i=1}^{k} \mu_{i,t-1}(\mathbf{x}_t) \tag{14}$$

where $\mu_{i,t-1}$ is the posterior mean of function i. The cumulative reward over T_{max} iterations that would have been obtained using acquisition function AF_j is defined as:

$$r_{T_{max}}^j = \sum_{t=1}^{T_{max}} IR_t = \sum_{t=1}^{T_{max}} \sum_{i=1}^{k} \mu_{i,t-1}(\mathbf{x}_t^j). \tag{15}$$

It is important to note that in our proposed PDBO algorithm, we use a different and better instant reward IR_t and cumulative reward $r_{T_{max}}^i$. The rewards in equation 14 is a design choice to achieve the following regret bound. In Section 5.1, we provide a discussion accompanied by an experimental ablation study comparing the reward function used in theory to the reward function used in PDBO.

Theorem 5.1. *Let $\mathbf{x}^*$ be a point in the optimal Pareto set $\mathcal{X}^*$. Let $\mathbf{x}$ be a point in the Pareto set $\mathcal{X}_t$ estimated by PDBO via solving cheap MOO problem at the tth iteration. Let the cumulative regret for the multiple objectives be defined as $R_{T_{max}}(\mathbf{x}^*) = \sum_{t=1}^{T_{max}} \sum_{i=1}^{k} f_i(\mathbf{x}^*) - f_i(\mathbf{x}_t)$*

Assuming maximization of objectives and that UCB is in the portfolio of acquisition functions, let β_t be the UCB parameter and $\gamma_{T_{max}}^i$ be the bound on the information gained for function i at points selected by PDBO after T_{max} iterations, then with probability at least $1 - \delta$ the cumulative regret is bounded by

$$R_{T_{max}}(\mathbf{x}^*) \leq \mathcal{O}(\sqrt{T_{max}}) + \sum_{t=1}^{T_{max}} \sum_{i=1}^{k} \sqrt{\beta_t} \sigma_{i,t-1}(\mathbf{x}_t^{\text{UCB}})$$
$$+ \sqrt{C_i T_{max} \beta_{T_{max}} \gamma_{T_{max}}^i}.$$

We provide complete proof in the Appendix. The theorem suggests that our regret is bounded by two sublinear terms and another term that might include points suggested by UCB but not necessarily selected by the Hedge strategy. Additionally, the theoretical proof accounts only for sequential input selection. Extension to batch selection using DPP is possible by carefully accounting for results introduced by Kathuria, Deshpande, and Kohli (2016).

5.1 Analysis and Ablation Study

To simplify the proof for regret bound, we defined a new instant reward and cumulative reward in equations 14 and 15 that are different from the rewards we use for PDBO in equations 6, 7, and 8. The goal of this section was to define a reward that allows tractable theoretical analysis. However, this reward is not intuitive and has several practical issues: **1)** It is defined as a summation over predictive means of the functions and does not provide any insight on the quality of the selected points; **2)** It does not account for improvement with respect to previous iterations which is uninformative in terms of the quality of points selected by different acquisition functions at different iterations; **3)** It is non-discounted and does not account for the importance of the iterative progress of the input selection; and **4)** It is not normalized and therefore can lead to random selection in the advanced iterations. All stated issues have been addressed by our proposed reward function which we carefully designed to be intuitive and informative about the different acquisition strategies and to mitigate potential numerical issues. We performed an ablation study to compare the performance of PDBO when using our proposed reward function and the reward function in the theoretical proof. Our results in the Appendix show superior performance for our proposed reward strategy.

6 Experiments and Results

We provide experimental details and compare PDBO to baseline methods on multiple MOO benchmarks and varying batch sizes. We evaluate all methods using the hypervolume indicator and diversity of Pareto front (DPF) measure.

Benchmarks. We conduct experiments on benchmarks with varying numbers of input and output dimensions to show the versatility and flexibility of our method. We use several synthetic problems: ZDT-1, ZDT-2, ZDT-3 (Zitzler, Deb, and Thiele 2000), DTLZ-1, DTLZ-3, DTLZ-5 (Deb et al. 2005) and three real wold problems: the gear train design problem (Deb and Srinivasan 2006; Konakovic Lukovic, Tian, and Matusik 2020), SWLLVM (Siegmund et al. 2012) and Unmanned aerial vehicle power system design (Belakaria et al. 2020b). More details and descriptions of problem settings are included in the Appendix. In the ablation studies, we provide additional experiments with synthetic benchmarks where we vary the input and output dimensions.

Baselines. We compare our PDBO method to state-of-the-art batch MOO methods: DGEMO, qEHVI, qPAREGO, and USEMO-EI. We also include NSGA-II as the evolutionary algorithm baseline and random input selection. We set the hyperparameters of PDBO to $\gamma = 0.7$ and $\tau = 4$ as recommended by (Hoffman et al. 2011; Vasconcelos et al. 2019). We define the AF portfolio as $\mathcal{P} = \{EI, TS, UCB, ID\}$.

Experimental Setup. All experiments are initialized with five random inputs/evaluations and run for at least 250 function evaluations. We conduct experiments with four different batch sizes $B \in \{2, 4, 8, 16\}$ and adjust the number of iterations accordingly. For instance, when using a batch size of two, we run the algorithm for 125 iterations. Each experiment is repeated 25 times, and we report the average and standard deviation of the hypervolume indicator and the DPF
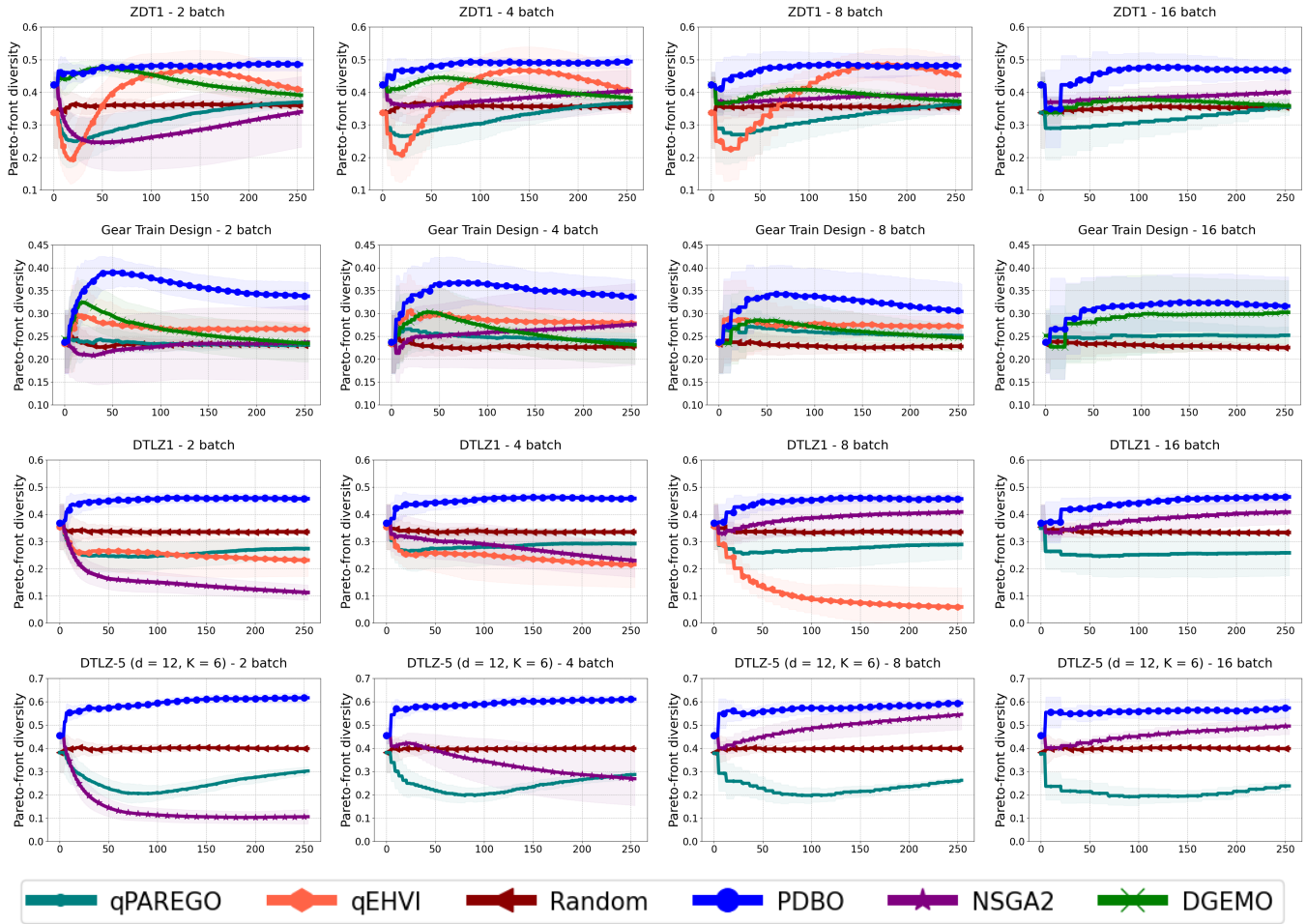
Figure 2: Diverse Pareto front (DPF) results evaluated on multiple benchmarks and batch sizes.

metric. To solve the constrained optimization problem in the DPP algorithm, we utilize an implementation of the SQP method (Lalee, Nocedal, and Plantenga 1998; Nocedal and Wright 2006) from the Python SciPy library (Virtanen et al. 2020). For baselines, we use the codes and hyperparameters provided in the open source repositories of DGEMO [1] and Botorch [2]. We provide the details for the NSGA-II baseline and cheap MO solver, and more details about the setup for fitting the hyperparameters of GP models in the Appendix.

Results and Discussion. Figure 2 demonstrates that PDBO outperforms other baselines with respect to the Pareto-front diversity metric. Additionally, Figure 3 demonstrates that PDBO outperforms all baseline methods in most experiments with respect to the Hypervolume indicator and provides a competitive performance on the others.

In the Appendix, we present a comprehensive set of additional results and analyses. This includes the evaluation of hypervolume and DPF on various benchmarks. We also introduce results using other metrics, notably the Inverted

Generational Distance (IGD) and a modified version of DPF, accompanied by a relevant discussion. Additionally, we compare the run-time of all baseline methods. For visual insight into the diversity of solutions, we include scatter plots representing the Pareto front for problems with two objectives. Lastly, we provide statistics on the selection of AF.

PDBO Advantages. PDBO is fast and effective in producing high-quality and diverse Pareto fronts. While outperforming the baseline methods, it can also be used with any number of input and output dimensions as well as being flexible to run with any batch size. The two state-of-the-art methods are DGEMO and qEHVI. The DGEMO method fails to run for experiments with more than three objective functions as the graph cut algorithm consistently crashes (same observation was made by (Daulton, Balandat, and Bakshy 2021)). qEHVI fails to run with batch sizes higher than eight as the method becomes extremely memory-consuming even with GPUs. We provide a more detailed discussion about these limitations in the Appendix. Therefore, PDBO's ability to easily run with any input and output dimensions as well as any batch size is an advantage for practitioners. PDBO is capable of proactively creating a diverse Pareto front while improving

---

[1] https://github.com/yunshengtian/DGEMO
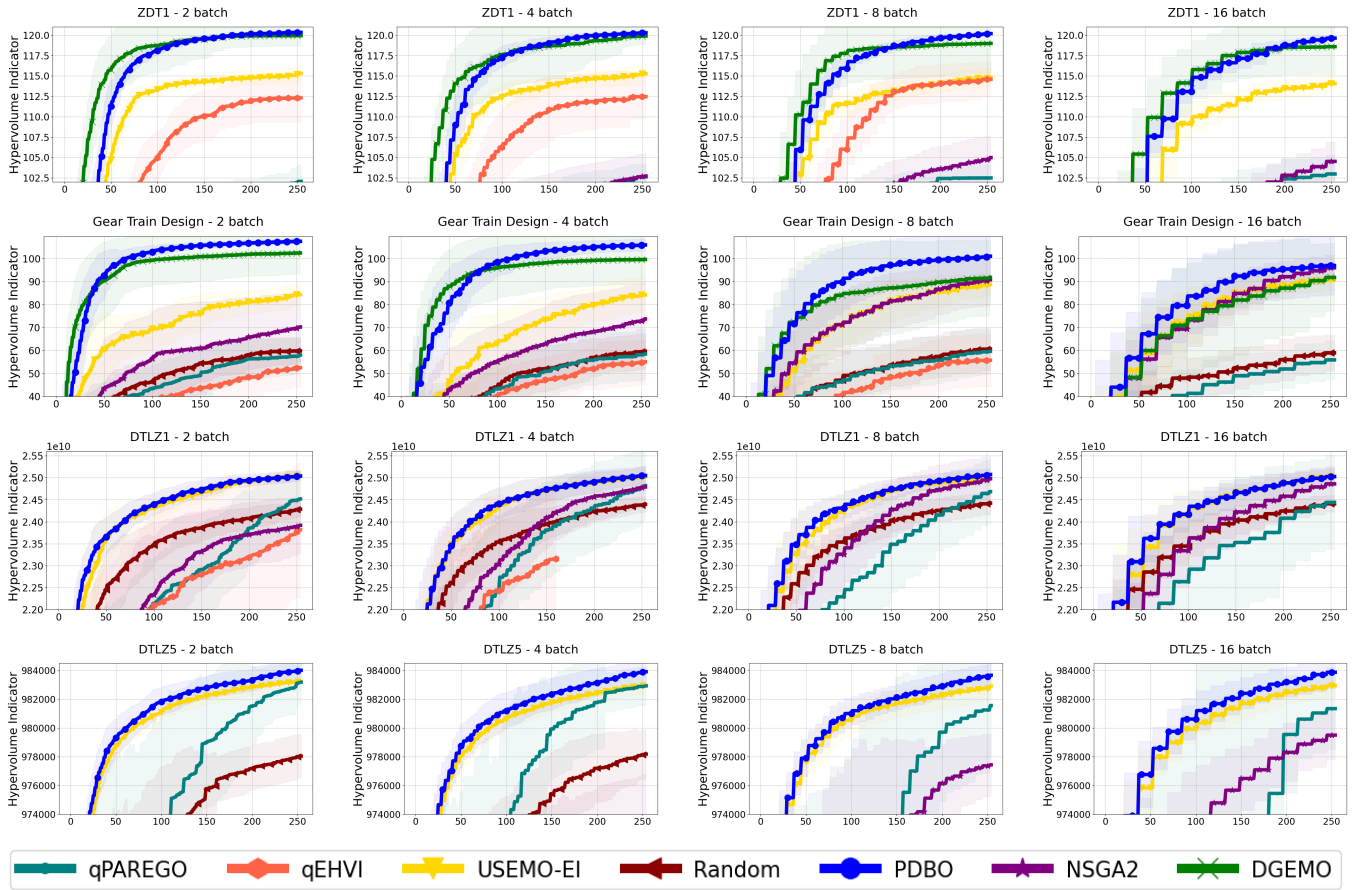
[2] https://github.com/pytorch/botorch

Figure 3: Hypervolume results evaluated on multiple benchmarks and batch sizes.

or maintaining the quality of the Pareto front.

Given that PDBO incorporates two key contributions, namely adaptive acquisition function selection and multi-objective batch selection using DPPs, we examine the individual contributions of each component to the overall performance by conducting ablation experiments.

Merits of Adaptive AF Selection. We demonstrate the superiority of the adaptive AF selection method, as outlined in Section 4.1, compared to using a static AF from the portfolio. To isolate the impact of this component from the batch selection process, we conduct an ablation study using the USEMO baseline. With a batch size of one, we consider USEMO with UCB, TS, ID, and EI as baselines. We then evaluate the efficacy of our MAB method by incorporating the adaptive AF selection approach into USEMO. Results shown in the Appendix consistently demonstrate the superior performance of the MAB strategy over using a static AF.

Merits of DPP-Based Batch Selection for MOO. Following a similar ablation approach, we employ the USEMO-EI baseline to examine the impact of the DPP-based batch selection. USEMO-EI selects the next input for evaluation from the cheap Pareto set based on an uncertainty metric. To perform this ablation, we replace the input selection mechanism

utilized in USEMO with our proposed DPP selection strategy and compare their performance. The ablation is conducted across different batch sizes $B \in \{2, 4, 8, 16\}$. The results presented in the Appendix reveal that the proposed DPP selection strategy, referred to as DPP-EI, surpasses the USEMO selection strategy in terms of diversity while simultaneously improving the quality of hypervolume.

7 Summary

We studied the Pareto front-Diverse Batch Multi-Objective BO (PDBO) method based on the BO framework. It employs a full information multi-arm bandit algorithm with discounted reward to adaptively select the most suitable acquisition function in each iteration. We also proposed an appropriate reward based on the relative hypervolume contribution of each acquisition function and a multi-objective DPP approach configured to select a batch of Pareto-diverse inputs for evaluation. Experiments on multiple benchmarks demonstrate that PDBO outperforms prior methods in terms of both diversity and quality of Pareto-front solutions.

Acknowledgements

References

Abdolshah, M.; Shilton, A.; Rana, S.; Gupta, S.; and Venkatesh, S. 2019. Multi-objective Bayesian optimisation with preferences over objectives. *Advances in neural information processing systems*, 32.

Angermueller, C.; Belanger, D.; Gane, A.; Mariet, Z.; Dohan, D.; Murphy, K.; Colwell, L.; and Sculley, D. 2020. Population-based black-box optimization for biological sequence design. In *International Conference on Machine Learning*, 324–334. PMLR.

Ashby, M. 2000. Multi-objective optimization in material design and selection. *Acta materialia*, 48(1): 359–369.

Astudillo, R.; and Frazier, P. 2020. Multi-attribute Bayesian optimization with interactive preference learning. In *International Conference on Artificial Intelligence and Statistics*, 4496–4507. PMLR.

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

Belakaria, S.; Deshwal, A.; and Doppa, J. R. 2019. Max-value Entropy Search for Multi-Objective Bayesian Optimization. In *Conference on Neural Information Processing Systems*.

Belakaria, S.; Deshwal, A.; Jayakodi, N. K.; and Doppa, J. R. 2020a. Uncertainty-aware search framework for multi-objective Bayesian optimization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34.

Belakaria, S.; Jackson, D.; Cao, Y.; Doppa, J. R.; and Lu, X. 2020b. Machine learning enabled fast multi-objective optimization for electrified aviation power system design. In *IEEE Energy Conversion Congress and Exposition (ECCE)*.

Birol, G.; Ündey, C.; and Cinar, A. 2002. A modular simulation package for fed-batch fermentation: penicillin production. *Computers & chemical engineering*, 26(11).

Borodin, A. 2009. Determinantal point processes. *arXiv preprint arXiv:0911.1153*.

Borodin, A.; and Olshanski, G. 2005. Harmonic analysis on the infinite-dimensional unitary group and determinantal point processes. *Annals of mathematics*, 1319–1422.

Cesa-Bianchi, N.; and Lugosi, G. 2006. *Prediction, learning, and games*. Cambridge university press.

Daulton, S.; Balandat, M.; and Bakshy, E. 2020. Differentiable expected hypervolume improvement for parallel multi-objective Bayesian optimization. *Advances in Neural Information Processing Systems*, 33: 9851–9864.

Daulton, S.; Balandat, M.; and Bakshy, E. 2021. Parallel Bayesian Optimization of Multiple Noisy Objectives with Expected Hypervolume Improvement. *NeurIPS*, 34.

Daulton, S.; Eriksson, D.; Balandat, M.; and Bakshy, E. 2022. Multi-Objective Bayesian Optimization over High-Dimensional Search Spaces. In *The 38th Conference on Uncertainty in Artificial Intelligence*.

Deb, K.; Pratap, A.; Agarwal, S.; Meyarivan, T.; and Fast, A. 2002. NSGA-II. *IEEE Transactions on Evolutionary Computation*, 6(2): 182–197.

Deb, K.; and Srinivasan, A. 2006. Innovization: Innovating design principles through optimization. In *Proceedings of the 8th annual conference on Genetic and evolutionary computation*, 1629–1636.

Deb, K.; Thiele, L.; Laumanns, M.; and Zitzler, E. 2005. Scalable test problems for evolutionary multiobjective optimization. In *Evolutionary multiobjective optimization*. Springer.

Deshwal, A.; Simon, C. M.; and Doppa, J. R. 2021. Bayesian optimization of nanoporous materials. *Molecular Systems Design & Engineering*, 6(12): 1066–1086.

Emmerich, M.; and Klinkenberg, J.-w. 2008. The computation of the expected improvement in dominated hypervolume of Pareto front approximations. *Technical Report, Leiden University*, 34.

Freund, Y.; and Schapire, R. E. 1997. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of computer and system sciences*, 55(1).

Hernández-Lobato, D.; Hernandez-Lobato, J.; Shah, A.; and Adams, R. 2016. Predictive entropy search for multi-objective Bayesian optimization. In *Proceedings of International Conference on Machine Learning (ICML)*, 1492–1501.

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

Hoffman, M.; Brochu, E.; De Freitas, N.; et al. 2011. Portfolio Allocation for Bayesian Optimization. In *UAI*, 327–336.

Hvarfner, C.; Hutter, F.; and Nardi, L. 2022. Joint entropy search for maximally-informed Bayesian optimization. *arXiv preprint arXiv:2206.04771*.

Jain, M.; Raparthy, S. C.; Hernandez-Garcia, A.; Rector-Brooks, J.; Bengio, Y.; Miret, S.; and Bengio, E. 2022. Multi-Objective GFlowNets. *arXiv preprint arXiv:2210.12765*.

Kathuria, T.; Deshpande, A.; and Kohli, P. 2016. Batched gaussian process bandit optimization via determinantal point processes. *NeurIPS*, 29.

Knowles, J. 2006. ParEGO: A hybrid algorithm with on-line landscape approximation for expensive multiobjective optimization problems. *IEEE Transactions on Evolutionary Computation*, 10(1): 50–66.

Konakovic Lukovic, M.; Tian, Y.; and Matusik, W. 2020. Diversity-guided multi-objective bayesian optimization with batch evaluations. *Advances in Neural Information Processing Systems*, 33: 17708–17720.

Kulesza, A.; Taskar, B.; et al. 2012. Determinantal point processes for machine learning. *Foundations and Trends® in Machine Learning*, 5(2–3): 123–286.

Lalee, M.; Nocedal, J.; and Plantenga, T. 1998. On the implementation of an algorithm for large-scale equality constrained optimization. *SIAM Journal on Optimization*, 8(3): 682–706.

Lin, X.; Yang, Z.; Zhang, X.; and Zhang, Q. 2022. Pareto Set Learning for Expensive Multi-Objective Optimization. *arXiv preprint arXiv:2210.08495*.

Mockus, J.; Tiesis, V.; and Zilinskas, A. 1978. The application of Bayesian methods for seeking the extremum. *Towards global optimization*, 2(117-129): 2.

Nava, E.; Mutny, M.; and Krause, A. 2022. Diversified Sampling for Batched Bayesian Optimization with Determinantal Point Processes. In *International Conference on Artificial Intelligence and Statistics*, 7031–7054. PMLR.

Nicolaou, C. A.; and Brown, N. 2013. Multi-objective optimization methods in drug design. *Drug Discovery Today: Technologies*, 10(3): e427–e435.

Nikolov, A. 2015. Randomized rounding for the largest simplex problem. In *Proceedings of the forty-seventh annual ACM symposium on Theory of computing*, 861–870.

Nocedal, J.; and Wright, S. 2006. Numerical Optimization (Springer, New York, 1999).

Oh, C.; Bondesan, R.; Gavves, E.; and Welling, M. 2021. Batch Bayesian Optimization on Permutations using Acquisition Weighted Kernels. *arXiv preprint arXiv:2102.13382*.

Okoth, M. A.; Shang, R.; Jiao, L.; Arshad, J.; Rehman, A. U.; and Hamam, H. 2022. A Large scale evolutionary algorithm based on determinantal point processes for Large scale multi-objective optimization problems. *Electronics*, 11(20): 3317.

Paria, B.; Kandasamy, K.; and Póczos, B. 2020. A flexible framework for multi-objective bayesian optimization using random scalarizations. In *Uncertainty in Artificial Intelligence*, 766–776. PMLR.

Shahriari, B.; Swersky, K.; Wang, Z.; Adams, R. P.; and De Freitas, N. 2015. Taking the human out of the loop: A review of Bayesian optimization. *Proceedings of the IEEE*.

Siegmund, N.; Kolesnikov, S. S.; Kästner, C.; Apel, S.; Batory, D.; Rosenmüller, M.; and Saake, G. 2012. Predicting performance via automated feature-interaction detection. In *Proceedings of the 34th International Conference on Software Engineering (ICSE)*, 167–177.

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

Suzuki, S.; Takeno, S.; Tamura, T.; Shitara, K.; and Karasuyama, M. 2020. Multi-objective Bayesian optimization using pareto-frontier entropy. In *International Conference on Machine Learning*, 9279–9288. PMLR.

Taneda, A. 2015. Multi-objective optimization for RNA design with multiple target secondary structures. *BMC bioinformatics*, 16: 1–20.

Thompson, W. R. 1933. On the likelihood that one unknown probability exceeds another in view of the evidence of two samples. *Biometrika*, 25: 285–294.

Vasconcelos, T. d. P.; de Souza, D. A.; Mattos, C. L.; and Gomes, J. P. 2019. No-PASt-BO: Normalized portfolio allocation strategy for Bayesian optimization. In *International Conference on Tools with Artificial Intelligence (ICTAI)*. IEEE.

Vasconcelos, T. d. P.; de Souza, D. A. R.; Virgolino, G. C. d. M.; Mattos, C. L.; and Gomes, J. P. 2022. Self-tuning portfolio-based Bayesian optimization. *Expert Systems with Applications*, 188: 115847.

Virtanen, P.; Gommers, R.; Oliphant, T. E.; Haberland, M.; Reddy, T.; Cournapeau, D.; Burovski, E.; Peterson, P.; Weckesser, W.; Bright, J.; van der Walt, S. J.; Brett, M.; Wilson, J.; Millman, K. J.; Mayorov, N.; Nelson, A. R. J.; Jones, E.; Kern, R.; Larson, E.; Carey, C. J.; Polat, İ.; Feng, Y.; Moore, E. W.; VanderPlas, J.; Laxalde, D.; Perktold, J.; Cimrman, R.; Henriksen, I.; Quintero, E. A.; Harris, C. R.; Archibald, A. M.; Ribeiro, A. H.; Pedregosa, F.; van Mulbregt, P.; and SciPy 1.0 Contributors. 2020. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, 17: 261–272.

Wang, M.; Ge, F.; Chen, D.; and Liu, H. 2022. A Many-objective Evolutionary Algorithm using Determinantal Point Process in Potential Region. In *Proceedings of the 6th International Conference on Control Engineering and Artificial Intelligence*, 83–91.

Wang, Z.; and Jegelka, S. 2017. Max-value Entropy Search for Efficient Bayesian Optimization. In *Proceedings of International Conference on Machine Learning (ICML)*.

Wang, Z.; Li, C.; Jegelka, S.; and Kohli, P. 2017. Batched high-dimensional Bayesian optimization via structural kernel learning. In *International Conference on Machine Learning*, 3656–3664. PMLR.

Williams, C. K.; and Rasmussen, C. E. 2006. *Gaussian processes for machine learning*. MIT press Cambridge, MA.

Zhang, P.; Li, J.; Li, T.; and Chen, H. 2020. A new many-objective evolutionary algorithm based on determinantal point processes. *IEEE Transactions on Evolutionary Computation*, 25(2): 334–345.

Zitzler, E.; Deb, K.; and Thiele, L. 2000. Comparison of multiobjective evolutionary algorithms: Empirical results. *Evolutionary computation*, 8(2): 173–195.

Zitzler, E.; and Thiele, L. 1999. Multiobjective evolutionary algorithms: a comparative case study and the strength Pareto approach. *IEEE transactions on Evolutionary Computation*, 3(4): 257–271.