

# High-Dimensional Dueling Optimization with Preference Embedding

Yangwenhui Zhang, Hong Qian\*, Xiang Shu, Aimin Zhou

Shanghai Institute of AI for Education and School of Computer Science and Technology,  
East China Normal University, Shanghai 200062, China  
{51215901086, 51255901138}@stu.ecnu.edu.cn, {hqian, amzhou}@cs.ecnu.edu.cn

## Abstract

In many scenarios of black-box optimization, evaluating the objective function values of solutions is expensive, while comparing a pair of solutions is relatively cheap, which yields the dueling black-box optimization. The side effect of dueling optimization is that it doubles the dimension of solution space and exacerbates the dimensionality scalability issue of black-box optimization, e.g., Bayesian optimization. To address this issue, the existing dueling optimization methods fix one solution when dueling throughout the optimization process, but it may reduce their efficacy. Fortunately, it has been observed that, in recommendation systems, the dueling results are mainly determined by the latent human preferences. In this paper, we abstract this phenomenon as the preferential intrinsic dimension and inject it into the dueling Bayesian optimization, resulting in the preferential embedding dueling Bayesian optimization (PE-DBO). PE-DBO decouples optimization and pairwise comparison via the preferential embedding matrix. Optimization is performed in the preferential intrinsic subspace with much lower dimensionality, while pairwise comparison is completed in the original dueling solution space. Theoretically, we disclose that the preference function can be approximately preserved in the lower-dimensional preferential intrinsic subspace. Experiment results verify that, on molecule discovery and web page recommendation dueling optimization tasks, the preferential intrinsic dimension exists and PE-DBO is superior in scalability compared with that of the state-of-the-art (SOTA) methods.

## Introduction

Black-box optimization (Conn, Scheinberg, and Vicente 2009; Liu et al. 2022), also termed as derivative-free optimization, is of significance in machine learning (Snoek, Larochelle, and Adams 2012), reinforcement learning (Qian and Yu 2021) and scientific computing (Shields et al. 2021). It regards the objective function as black-box and optimization is performed only with point-wise evaluation. Other information such as gradient is assumed to be inaccessible. In many scenarios of black-box optimization, directly evaluating the objective function values of solutions is expensive (Jones, Schonlau, and Welch 1998; Snoek, Larochelle,

and Adams 2012; Shields et al. 2021). Thus, the function evaluation budget is strictly limited and sample-efficient methods, e.g., Bayesian optimization (Shahriari et al. 2016; Song et al. 2022), are leveraged.

In contrast to direct evaluating the objective function value on a given solution, comparing a pair of solutions and assessing which one is better is relatively cheap. Kahneman and Tversky (1979) have pointed out that it is easier and cheaper for humans to give their preference on two items compared with directly rating two items, e.g., ranking all the dishes according to one’s taste. This result leads to the  $K$ -armed dueling bandit (Yue et al. 2012; Sui et al. 2018) in discrete domains and the *dueling or preferential optimization* (González et al. 2017; Sui et al. 2017) in continuous domains.

**Problem.** In dueling optimization, the side effect of pairwise comparison is that it doubles the dimension of solution space and exacerbates the dimensionality scalability issue of black-box optimization methods, e.g., Bayesian optimization.

**Related Work.** González et al. (2017) propose the preferential Bayesian optimization (PBO) that makes the two solutions flexible when comparing them. PBO uses the dueling results to fit a Gaussian process (GP) over the preference function space. Based on GP, PBO introduces dueling-Thompson sampling (DTS) that respectively determines the potential best and uncertain solutions for dueling in the next round. Then, these two solutions are concatenated to update GP. Restricted by the dimensionality scalability issue, PBO is qualified only for the low-dimensional dueling solution space ( $\leq 6$ ). Benavoli, Azzimonti, and Piga (2021) improve the optimization performance of PBO by using SkewGP model to fit the preference function but still face the scalability issue.

Sui et al. (2017) and Xu et al. (2020) propose to fix one solution when dueling throughout the optimization process, i.e., kernel-self-sparring (KSS) and comp-GP-UCB. In (Sui et al. 2017), KSS substitutes the preference function in PBO with the function whose value is the probability of one solution beating the optimal solution. The optimal solution is obviously fixed. This function is modeled via GP and the dueling results are used to fit GP. KSS acquires a batch of potential best solutions for dueling in the next round. Then, these solutions are used to sequentially update GP. In (Xu et al. 2020), comp-GP-UCB substitutes the preference function in PBO with the Borda function which is inspired by the Borda score (Sui et al. 2018) in the dueling bandit. The

\*Corresponding Author: hqian@cs.ecnu.edu.cn.

Copyright © 2023, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

value of the Borda function is the probability of one solution beating the mean performance of all solutions. The mean performance of all solutions can be regarded as the performance of one fixed solution. The Borda function is modeled by GP and the dueling results are applied to fit GP. Comp-GP-UCB acquires the potential best solution and a random solution whose expectation performance is the mean performance of all solutions for dueling in the next round. Then, this potential best solution is used to update GP. Although fixing one solution in dueling optimization can reduce the dimensionality, i.e., from dueling solution space to solution space, it may reduce their efficacy. Selecting a fixed solution that is proved to be suitable is difficult in general. If selecting an unsuitable solution to fix, it becomes hard to tell which solution is better among the flexible solutions. Furthermore, fixing one solution may restrict the scope of dueling exploration and thus the high-quality solution may be excluded.

Inspired by LINEBO (Kirschner et al. 2019), Mikkola et al. (2020) and Tucker et al. (2020) only consider the preference across 1-dimensional subspace in each iteration. However, this kind of methods is a little greedy and lacks theoretical analysis to guarantee its efficacy.

**Our Contributions.** In this paper, we aim to make the high-dimensional dueling optimization tractable while retaining the flexibility of two solutions when dueling. To this end, the paper introduces the *preferential intrinsic dimension*, and focuses on the high-dimensional dueling optimization problems with low preferential intrinsic dimension. In these problems, the dueling result of two solutions is mainly determined by a small number of approximately effective variables.

The introduced preferential intrinsic dimension is inspired by the latent human preference in the recommendation systems (Wang et al. 2016a; Canal et al. 2019). In preference-based recommendation systems, which item is preferred by a user is mainly determined by only a few ingredients that are called the latent human preference of a user. Wang et al. (2016a) and Canal et al. (2019) embed all items as well as the user’s favorite item into a low-dimensional subspace and use a distance metric to represent the user’s preference on items.

This paper abstracts the aforementioned phenomenon as the preferential intrinsic dimension and injects it into the dueling Bayesian optimization, resulting in the preferential embedding dueling Bayesian optimization (PE-DBO). PE-DBO decouples optimization and pairwise comparison by a randomly generated preferential embedding matrix. Via preference embedding, optimization is performed in the preferential intrinsic subspace with much lower dimensionality and two solutions are both flexible when dueling therein, while the pairwise comparison is completed in the original dueling solution space. Theoretically, we reveal that the preference function can be approximately preserved in the lower-dimensional preferential intrinsic subspace for both unbounded and bounded domains. Experiment results show that, on molecule discovery and web page recommendation dueling optimization tasks, the preferential intrinsic dimension exists and PE-DBO is superior in scalability compared with that of the SOTA methods. To the best of our knowledge, PE-DBO is the first method that can address more than 100-dimensional dueling solution space, and notably up to

nearly 300 dimension on the real-world dataset.

The consequent sections respectively give the necessary preliminaries, introduce the preferential intrinsic dimension and present the proposed PE-DBO method, show the theoretical and empirical results, and finally conclude the paper.

## Preliminaries

**Dueling Optimization.** Let  $f: \mathcal{X} \rightarrow \mathbb{R}$ , where  $\mathcal{X} \subset \mathbb{R}^D$ , be a black-box function that is expensive to evaluate. The goal of continuous global optimization is to find out  $\mathbf{x}^* = \operatorname{argmax}_{\mathbf{x} \in \mathcal{X}} f(\mathbf{x})$  on the *solution space* with dimension  $D$ . Since directly evaluating the function value is very expensive, dueling optimization methods aim to evaluate the preference between pairs of solutions  $(\mathbf{x}, \mathbf{x}')$ , i.e., dueling to optimize the objective function. Each duel has a feedback (0 or 1) shows which solution in the duel that the oracle prefers ( $\mathbf{x}$  or  $\mathbf{x}'$ ), and this is all data that dueling optimization methods can acquire. In the rest of this paper, each vector is a column vector and we use  $[\mathbf{x}; \mathbf{x}']$  to denote a column vector concatenated by two column vectors  $\mathbf{x}$  and  $\mathbf{x}'$ . We use  $[\mathbf{x}; \mathbf{x}'] \in \mathbb{R}^{2D}$  to represent the duel  $(\mathbf{x}, \mathbf{x}')$ . The space with dimension  $2D$  is called the *dueling solution space* because each point in this space represents a duel.

**Preference Function and Copeland Score.** In dueling optimization, the feedback of a duel  $[\mathbf{x}; \mathbf{x}']$  given by human oracle is modeled as a stochastic process. The feedback is sampled from a Bernoulli distribution and the corresponding probability shows how likely the solution  $\mathbf{x}$  is preferred to  $\mathbf{x}'$ . This probability is assumed to be positively related to the difference of objective function values, i.e.,  $P(\mathbf{x} \succ \mathbf{x}') \propto f(\mathbf{x}) - f(\mathbf{x}')$ . A straight way to convert the difference of function value to probability is using the logistic function, thus the preference function on the dueling solution space is

$$\pi_f([\mathbf{x}; \mathbf{x}']) = P(\mathbf{x} \succ \mathbf{x}') = \frac{1}{1 + e^{-[f(\mathbf{x}) - f(\mathbf{x}')]}}.$$

The preference function value represents the probability of  $\mathbf{x}$  being preferred to  $\mathbf{x}'$  in the dueling solution space.

The preference function only shows the probability of each duel and cannot reveal which solution is  $\mathbf{x}^*$ . Fortunately, the Copeland winner can be extended from the  $K$ -armed dueling bandit problem to the dueling optimization task to determine which solution is the best across all solutions. Because the objective function is continuous, the normalized Copeland score (González et al. 2017) can be defined as

$$S(\mathbf{x}) = \operatorname{Vol}(\mathcal{X})^{-1} \int_{\mathcal{X}} \mathbb{I}_{\{\pi_f([\mathbf{x}; \mathbf{x}']) \geq 0.5\}} d\mathbf{x}',$$

where  $\operatorname{Vol}(\mathcal{X})^{-1} = \int_{\mathcal{X}} 1 d\mathbf{x}'$  is a normalizing constant such that  $S(\mathbf{x})$  is in the interval  $[0, 1]$ , and  $\mathbb{I}_{\{\cdot\}}$  is the indicator function, i.e., it equals to 1 if  $\pi_f([\mathbf{x}; \mathbf{x}']) \geq 0.5$ , and 0 otherwise. This score represents the proportion of duels that  $\mathbf{x}$  beats  $\mathbf{x}'$  with a probability larger than 0.5. Considering the target  $\mathbf{x}^*$ , for every solution  $\pi_f([\mathbf{x}^*; \mathbf{x}']) \geq 0.5$  holds and thus  $S(\mathbf{x}^*) = \operatorname{Vol}(\mathcal{X})^{-1} \int_{\mathcal{X}} 1 d\mathbf{x}' = 1$  which is the upper bound of the normalized Copeland score. Hence, we can get  $\mathbf{x}^*$  by optimizing  $S(\mathbf{x})$ .

However, the indicator function makes the computation of the normalized Copeland score more difficult, and a soft

version (González et al. 2017) that has the empirically same maximum as the normalized Copeland score is considered. The soft-Copeland score is defined as

$$C(\mathbf{x}) = \text{Vol}(\mathcal{X})^{-1} \int_{\mathcal{X}} \pi_f([\mathbf{x}; \mathbf{x}']) \, d\mathbf{x}'.$$

**Preferential Bayesian Optimization (PBO).** PBO is a typical dueling optimization method (González et al. 2017). It uses Gaussian process (GP) to fit the preference function with dimension  $2D$ , optimizes the acquisition function to decide the next duel, asks the oracle the preference of this duel and updates the dataset to fit a GP. It repeats the above process until the number of iterations is exhausted. The most important contribution of PBO is its acquisition function DTS. DTS consists of two acquisition functions with dimension  $D$  that represent exploitation and exploration respectively. The first solution is chosen with the highest soft-Copeland score and represents the exploitation, and the second solution is chosen with the highest variance of GP and represents the exploration. DTS balances the trade-off between exploitation and exploration, and guarantees the result of PBO. However, since it uses GP to fit the preference function which is a function with dimension  $2D$ , the precision of the GP model and the computation complexity are both significantly affected by the dimension. Although the acquisition function is well defined and easy to optimize on a  $D$  dimension space, its results become worse when  $D$  is high. Thus, the bottleneck of PBO is the dimension of the preference function.

## Proposed Method

The above section introduces the PBO method and shows its bottleneck. To address the issue of scalability of the dueling black-box optimization, one possible way is to reduce the dimension of the dueling solution space.

In fact, previous studies find out that, in many scenarios, although the dimension of the objective function is very high, there are only a few dimensions that affect the function value while the remaining dimensions only have a limited effect. This has been formally defined as the optimal effective dimension (i.e., the remaining dimensions have no effect on the function value) (Wang et al. 2016b) or the optimal  $\epsilon$ -effective dimension (i.e., the remaining dimensions only have an up to  $\epsilon$  effect) (Qian, Hu, and Yu 2016). The optimal  $\epsilon$ -effective dimension has been found in many real-world tasks, such as recommendation systems (Qin et al. 2010), chemical molecular design (Trabucco et al. 2022) and pre-trained large language models for natural language processing (Aghajanyan, Gupta, and Zettlemoyer 2021).

Since the existence of effective dimension of an objective function is common in many tasks, trying to use this property is a straightforward way to address the high-dimensional optimization problems. If the objective function has the optimal  $\epsilon$ -effective dimension, optimization can be conducted in the low-dimensional subspace whereas function value evaluation is still performed in the original high-dimensional space. This is realized via embedding the subspace into the original space, and the optimization performance can be guaranteed (Wang et al. 2016b; Qian, Hu, and Yu 2016).

---

## Algorithm 1: PE-DBO

---

### Input:

Initial dataset  $\mathcal{D}_M = \{[\mathbf{y}_i; \mathbf{y}'_i], p_i\}_{i=1}^M$ , number of available duels  $N$ , original dimension  $D$ , low dimension  $d$  and the boundary of low dimension subspace  $\mathcal{Y}$ .

### Procedure:

- 1: Generate random matrix  $\mathbf{A} \in \mathbb{R}^{D \times d}$  and  $\mathbf{A}_p = \begin{bmatrix} \mathbf{A} & \mathbf{0} \\ \mathbf{0} & \mathbf{A} \end{bmatrix}$ .
  - 2: **for**  $j = M$  **to**  $M + N - 1$  **do**
  - 3:   Fit a  $\mathcal{GP}$  to  $\mathcal{D}_j$  and learn  $\pi_{f_p, j}([\mathbf{y}; \mathbf{y}'])$ .
  - 4:   Sample a function  $\pi_{f_p}$  from  $\mathcal{GP}$ .
  - 5:    $\mathbf{y}_{\text{next}} = \arg\max_{\mathbf{y} \in \mathcal{Y}} \int_{\mathcal{Y}} \pi_{f_p}([\mathbf{y}; \mathbf{y}']; \mathcal{D}_j) \, d\mathcal{Y}$ .
  - 6:    $\mathbf{y}'_{\text{next}} = \arg\max_{\mathbf{y}' \in \mathcal{Y}} \sigma(\mathcal{GP} | \mathbf{y} = \mathbf{y}_{\text{next}}, \mathcal{D}_j)$ .
  - 7:   Get  $[\mathbf{x}_{\text{next}}; \mathbf{x}'_{\text{next}}] = \mathbf{A}_p [\mathbf{y}_{\text{next}}; \mathbf{y}'_{\text{next}}]$ .
  - 8:   Run duel  $[\mathbf{x}_{\text{next}}; \mathbf{x}'_{\text{next}}]$  and obtain  $p_{j+1}$ .
  - 9:   Augment  $\mathcal{D}_{j+1} = \{\mathcal{D}_j \cup ([\mathbf{y}_{\text{next}}; \mathbf{y}'_{\text{next}}], p_{j+1})\}$ .
  - 10: **end for**
  - 11: Fit a  $\mathcal{GP}$  to  $\mathcal{D}_{M+N}$  and find the solution  $\mathbf{y}^*$  with highest soft-Copeland score.
  - 12: **return**  $\mathbf{x}^* = \mathbf{A}\mathbf{y}^*$ .
- 

For dueling optimization, this paper shows that the effective dimension can be extended to the dueling case. Actually, in the preference-based recommendation systems, which item is preferred by a user is mainly determined by only a few ingredients. Wang et al. (2016a) and Canal et al. (2019) embed all items as well as the user’s favorite item into a low-dimensional subspace and use a distance metric to represent the user’s preference on items. This inspires us that there may exist a subspace with a lower dimension than the dueling solution space (we call this subspace as preferential intrinsic subspace), where the dueling result is almost the same as the corresponding dueling result in the dueling solution space. We formally introduce it as Definition 1.

**Definition 1** (Preferential Intrinsic Dimension). *A preference function  $\pi_f : \mathbb{R}^{2D} \rightarrow \mathbb{R}$  is said to have **preferential intrinsic dimension**  $d_p$ , with  $d_p \leq 2D$ , if*

1. *there exists a preferential intrinsic subspace  $\mathcal{T} = \mathcal{X}_\epsilon \times \mathcal{X}'_\epsilon$ ,  $\mathcal{X}_\epsilon \subset \mathbb{R}^D$ , s.t., for all  $[\mathbf{x}; \mathbf{x}'] \in \mathbb{R}^{2D}$  and for any  $\epsilon > 0$ , we have  $|\pi_f([\mathbf{x}; \mathbf{x}']) - \pi_f([\mathbf{x}_\epsilon; \mathbf{x}'_\epsilon])| \leq \epsilon$ , where  $\mathbf{x}_\epsilon$  and  $\mathbf{x}'_\epsilon \in \mathcal{X}_\epsilon$  are orthogonal projection of  $\mathbf{x}$  and  $\mathbf{x}'$  onto  $\mathcal{X}_\epsilon$ ;*
2. *for all preferential intrinsic subspaces  $\mathbb{T}_\epsilon$  of  $\pi_f$ ,  $d_p$  is the smallest dimension of  $\mathcal{T}$ , i.e.,  $d_p = \min_{\mathcal{T} \in \mathbb{T}_\epsilon} \dim(\mathcal{T})$  where  $\dim(\mathcal{T})$  denote the dimension of  $\mathcal{T}$ .*

$\mathcal{T}^\perp$  denotes the orthogonal complement of  $\mathcal{T}$ .  $[\mathbf{x}; \mathbf{x}']$  can be decomposed into two part that is  $[\mathbf{x}; \mathbf{x}'] = [\mathbf{x}_\epsilon; \mathbf{x}'_\epsilon] + [\mathbf{x}_\perp; \mathbf{x}'_\perp]$ , where  $[\mathbf{x}_\epsilon; \mathbf{x}'_\epsilon] \in \mathcal{T}$  and  $[\mathbf{x}_\perp; \mathbf{x}'_\perp] \in \mathcal{T}^\perp$ .

Definition 1 implies that there exist some important dimensions in the dueling solution space that significantly affect the preference function and other dimensions only have little effects on the preference probability (no more than  $\epsilon$ ).

If the preference function has the preferential intrinsic dimension  $d_p$ , how to embed the preferential intrinsic subspace into the dueling solution space is a crucial step.  $\mathbf{x}$  and  $\mathbf{x}'$  in a duel are the same since they are both in the same solution space and are the variables for the same objective function.

Hence, each of the two solutions should have the same embedding operation. To realize this, the preferential embedding matrix  $\mathbf{A}_p$  can be represented as the union of a random matrix, i.e.,  $\mathbf{A}_p = \begin{bmatrix} \mathbf{A} & \mathbf{O} \\ \mathbf{O} & \mathbf{A} \end{bmatrix}$  where  $\mathbf{A} \in \mathbb{R}_{D \times d}$  is a random matrix with independent entries sampled from  $\mathcal{N}(0, d^{-1})$  and  $d_p \leq 2d \leq 2D$ . With preference function  $\pi_f$  and preferential embedding matrix  $\mathbf{A}_p$ , we can construct a preference function  $\pi_{f_p}(\mathbf{y}; \mathbf{y}') = \pi_f(\mathbf{A}_p[\mathbf{y}; \mathbf{y}'])$  with only  $2d$  dimension. Fitting  $\pi_{f_p}$  with GP is nearly the same as fitting  $\pi_f$ .

Injecting the preferential embedding matrix in the PBO method can get the PE-DBO method. The flow of PE-DBO is to fit a GP in the low dimension subspace, optimize the acquisition function to choose the next duel  $[\mathbf{y}_{\text{next}}; \mathbf{y}'_{\text{next}}]$ , embed it into dueling solution space by the preferential embedding matrix, get the feedback of the corresponding duels  $[\mathbf{x}_{\text{next}}; \mathbf{x}'_{\text{next}}]$ , augment the dataset and begin the next loop until the termination. The full procedure of PE-DBO is depicted in Algorithm 1. The acquisition function used is the same as PBO which selects the solutions with highest soft-Copeland score and highest variance of GP respectively, where  $\sigma(\mathcal{GP}|\mathbf{y} = \mathbf{y}_{\text{next}}, \mathcal{D}_j)$  means the variance of GP with dataset  $\mathcal{D}_j$  and  $\mathbf{y} = \mathbf{y}_{\text{next}}$ . PE-DBO fits a GP on a low dimension subspace with dimension  $2d$  which is lower than  $2D$ , and  $2d$  is only affected by the preferential intrinsic dimension of the preference function. Thus, as long as the preferential intrinsic dimension is low, PE-DBO only needs to fit a GP on the low dimension subspace and it does not affect by  $2D$  dramatically like PBO. Notably, preference embedding can improve the scalability of the PBO framework since we prove that  $\pi_{f_p}$  is a good approximation of  $\pi_f$  in this paper. Thus, we only combine our method with the original version of PBO. One can also use the method in Benavoli, Azzimonti, and Piga (2021) and get the same improvement.

## Theoretical Analysis

This section presents some theoretical analyses on the proposed PE-DBO. Due to page limitation, please cf. Appendix A-C from <https://github.com/Zhangywh/PE-DBO> to get more details. Identifying whether the preference functions have the preferential intrinsic dimension is the first step to determine the feasibility of PE-DBO. However, it may be hard to identify in real-world tasks. The optimal  $\epsilon$ -effective dimension of a function has been proposed and investigated (Qian, Hu, and Yu 2016), and finding the optimal  $\epsilon$ -effective dimension of a function is relatively easy. We recall the definition of it.

**Definition 2** (Optimal  $\epsilon$ -Effective Dimension). *For any  $\epsilon > 0$ , a function  $f: \mathbb{R}^D \rightarrow \mathbb{R}$  is said to have an  $\epsilon$ -effective subspace  $\mathcal{V}_\epsilon$ , if there exists a linear subspace  $\mathcal{V}_\epsilon \subseteq \mathbb{R}^D$ , s.t. for all  $\mathbf{x} \in \mathbb{R}^D$ , we have  $|f(\mathbf{x}) - f(\mathbf{x}_\epsilon)| \leq \epsilon$ , where  $\mathbf{x}_\epsilon \in \mathcal{V}_\epsilon$  is the orthogonal projection of  $\mathbf{x}$  onto  $\mathcal{V}_\epsilon$ . Let  $\mathbb{V}_\epsilon$  denote the collection of all the  $\epsilon$ -effective subspaces of  $f$ , and  $\dim(\mathcal{V})$  denote the dimension of  $\mathcal{V}$ . We define the optimal  $\epsilon$ -effective dimension of  $f$  as  $d_\epsilon = \min_{\mathcal{V}_\epsilon \in \mathbb{V}_\epsilon} \dim(\mathcal{V}_\epsilon)$ .*

It is expected that one can identify the preferential intrinsic dimension through whether the objective function has the optimal  $\epsilon$ -effective dimension. To this end, we prove Lemma 1 that provides a sufficient condition to judge the existence of the preferential intrinsic dimension.

**Lemma 1** (Sufficient Condition). *If an objective function  $f$  has the optimal  $\epsilon$ -effective dimension  $d_\epsilon$ , then the corresponding preference function  $\pi_f$  has the preferential intrinsic dimension  $d_p \leq 2d_\epsilon$ .*

By Lemma 1, if an objective function has the optimal  $\epsilon$ -effective dimension, then the corresponding preference function also has the preferential intrinsic dimension. Therefore, one only needs to focus on whether the objective function has the optimal  $\epsilon$ -effective dimension  $d_\epsilon$  and the upper bound of the preferential intrinsic dimension is  $2d_\epsilon$ .

Then we need to confirm the feasibility of using  $\pi_{f_p}$  instead of the preference function  $\pi_f$ . Previously, random embedding is only known to be effective on the function value based methods (Wang et al. 2016b; Qian, Hu, and Yu 2016), and this work effectively extends it to the preference based ones. The extension is divided into two categories: unbounded and bounded domains. We first present the following theorem for the unbounded domains.

**Theorem 1** (Effectiveness on Unbounded Domains). *Suppose the preference function  $\pi_f: \mathbb{R}^{2D} \rightarrow [0, 1]$  has preferential intrinsic dimension  $d_p \leq 2d_\epsilon$ . Then, with probability 1, for any  $[\mathbf{x}; \mathbf{x}'] \in \mathbb{R}^{2D}$ , there exists a  $[\mathbf{y}; \mathbf{y}'] \in \mathbb{R}^{2d}$ , s.t.  $|\pi_f([\mathbf{x}; \mathbf{x}']) - \pi_{f_p}([\mathbf{y}; \mathbf{y}'])| \leq 2\epsilon$ .*

When the domain of the objective function is unbounded, i.e., the domain is  $\mathbb{R}^D$ . Theorem 1 shows that for any duel  $[\mathbf{x}; \mathbf{x}'] \in \mathbb{R}^{2D}$ , there exists a duel  $[\mathbf{y}; \mathbf{y}'] \in \mathbb{R}^{2d}$ , s.t., the difference between  $\pi_f$  and  $\pi_{f_p}$  is no more than  $2\epsilon$ . This confirms we can use  $\pi_{f_p}$  to substitute the  $\pi_f$  with a small loss of accuracy. Since the values of the preference function at each point in the low dimension subspace and the dueling solution space are nearly the same, the soft-Copeland score  $C(\mathbf{y}^*)$  is close to  $C(\mathbf{x}^*)$ , i.e., the optimal solution in the low dimension subspace is very closed to  $\mathbf{x}^*$  after embedding. Therefore, optimize  $\pi_{f_p}$  in the low dimension subspace which has the same effect as optimizing  $\pi_f$ .

Practical problems are often bounded by domain  $\mathcal{X} \subset \mathbb{R}^D$ , and for the bounded domain case, we need to consider two questions. How to deal with the situation when the embedded duel is out of the domain of  $\pi_f$  and how to set the boundary of low dimension subspace to guarantee the validity of fitting  $\pi_{f_p}$ . The first question is simple, one can use a projection operator based on the distance to find the nearest duel inside the domain to substitute the illegal duel. That is, suppose the domain of objective function is  $\mathcal{X}$ , when  $\mathbf{A}_p[\mathbf{y}; \mathbf{y}'] \notin \mathcal{X} \times \mathcal{X}$ , we use a duel  $[\mathbf{x}_\mathcal{X}; \mathbf{x}'_\mathcal{X}] = \operatorname{argmin}_{[\mathbf{x}; \mathbf{x}'] \in \mathcal{X} \times \mathcal{X}} \|\mathbf{x}; \mathbf{x}' - \mathbf{A}_p[\mathbf{y}; \mathbf{y}']\|_2$  inside the domain to substitute the duel out of  $\mathcal{X} \times \mathcal{X}$ . The second problem can be explained by the following theorem.

**Theorem 2** (Effectiveness on Bounded Domains). *Suppose the preference function  $\pi_f$  has preferential intrinsic dimension  $d_p$ , the domain of  $f$  is  $\mathcal{X} \subset \mathbb{R}^D$  and  $\mathcal{X}$  is centered around  $\mathbf{0}$ . Let  $\mathbf{x}^* = \operatorname{argmax}_{\mathbf{x} \in \mathcal{X}} f(\mathbf{x})$  be an optimizer of  $f$  with the  $\mathcal{X}$  and  $\mathbf{x}_\epsilon^* \in \mathcal{X} \cap \mathcal{X}_\epsilon$  be the optimizer of  $f$  inside  $\mathcal{X}_\epsilon$ . For any  $\mathbf{x}' \in \mathcal{X}$ ,  $\mathbf{x}'_\epsilon \in \mathcal{X} \cap \mathcal{X}_\epsilon$ , there exists a duel  $[\mathbf{y}^*; \mathbf{y}'] \in \mathbb{R}^{2d}$ , s.t.  $|\pi_{f_p}([\mathbf{y}^*; \mathbf{y}']) - \pi_f([\mathbf{x}_\epsilon^*; \mathbf{x}'_\epsilon])| \leq 2\epsilon$  and  $\|\mathbf{y}^*; \mathbf{y}'\|_2 \leq \sqrt{\frac{dd_p}{2\epsilon^2}} \|\mathbf{x}_\epsilon^*; \mathbf{x}'_\epsilon\|_2$  w.p. at least  $1 - \epsilon$ .*

Theorem 2 shows the substitute is also effective on the bounded domain and the theoretical boundary of the low dimension subspace. Therefore, if  $\mathcal{X} = [-1, 1]^D$ , with probability (w.p.) at least  $1 - \epsilon$ , as long as the boundary of the low dimension subspace contains a ball with radius  $(dd_p^2/2\epsilon^2)^{-\frac{1}{2}}$  and centered at the origin, the substitute  $\pi_f$  with  $\pi_{f_p}$  is effective. But in the experiment, for tasks whose optimal is near around  $\mathbf{0}$ , setting the domain of  $\pi_{f_p}$  by  $[-1, 1]^{2d}$  can get well result. This is easy to understand. We have sampled a preferential embedding matrix  $\mathbf{A}_p$ . For a duel  $[\mathbf{y}; \mathbf{y}'] = [1; \dots; 1]$  at the boundary of the low dimension subspace, by multiply the random matrix  $\mathbf{A}_p$ , we can get an vector  $[\mathbf{x}; \mathbf{x}'] = [x_1; \dots; x_{2D}]$ , for any  $x_i$ , it obeys the  $\mathcal{N}(0, 1)$ . Thus, this setting is big enough to cover the domain of  $\pi_f$ , i.e.,  $\mathcal{X} \times \mathcal{X} = [-1, 1]^{2D}$ .

## Experiments

PE-DBO is implemented by BoTorch (Balandat et al. 2020). The code is available at <https://github.com/Zhangywh/PE-DBO>. To study the effectiveness of preference embedding rather than the improvements on GP model, we use the default setting of GP model in BoTorch. CMA-ES (Hansen, Müller, and Koumoutsakos 2003) is adopted as the optimizer of the acquisition function. PE-DBO is deployed on both synthetic testing functions and real-world tasks. There are three preferential optimization algorithms for us to compare, i.e., PBO (González et al. 2017), KSS (Sui et al. 2017) and a pure compare version of comp-GP-UCB (Xu et al. 2020) which removes the second optimization part that uses the function value. The experiments aim to answer four questions.

1. Effectiveness: Can PE-DBO handle the high-dimensional dueling optimization tasks?
2. Scalability boundary: What is the maximum tractable dimension of PE-DBO?
3. Existence and superiority: Does the preferential intrinsic dimension exist, and can PE-DBO beat other methods on the real-world datasets?
4. The benefit of dueling optimization: When the total budget is fixed, whether preference-based methods can beat function value based methods?

The four questions are answered in order in the next sections. Since question 3 deals with the existence of the preferential intrinsic dimension and the superiority of PE-DBO, it corresponds to two parts in the following sections.

For the tasks that the optimal value of the objective function is known, the simple regret is used to evaluate the performance of each method. The simple regret is the gap between the optimal value and the best function value found by the algorithm. A smaller simple regret indicates the algorithm has found a better solution. For the tasks that the optimal function value is unknown, the best function value found by the algorithm is used as the evaluation criteria.

### On Synthetic Testing Functions

For a testing function  $f : \mathbb{R}^{d_f} \rightarrow \mathbb{R}$  (the domain has been re-scaling to  $[-1, 1]^{d_f}$ ) used to maximize, the synthetic testing function  $F_c : \mathbb{R}^D \rightarrow \mathbb{R}$  can be constructed as  $F_c(\mathbf{x}) =$

$f(\mathbf{x}_{[1:d_f]} - \mathbf{c}) - K^{-1} \sum_{i=d_f+1}^D (x_i - c)^2$ , where  $\mathbf{x} \in \mathbb{R}^D$  is the input of  $F_c$  and  $\mathbf{x}_{[1:d_f]}$  is the first  $d_f$  dimensions of  $\mathbf{x}$ .  $\mathbf{c} = [c; \dots; c] \in \mathbb{R}^d$  is a constant vector in order to avoid the optimal solution located at the origin which is in all linear subspaces.  $K$  is a constant to control the effects of the dimensions except the first  $d_f$  ones. It's obvious that  $F_c$  has optimal  $\epsilon$ -effective dimension  $d_e = d_f$  and  $\epsilon \leq K^{-1}$ . The effective dimensions here are the first  $d_f$  dimensions and randomly choosing these effective dimensions is similar.

**Effectiveness.** Four synthetic testing functions with  $D = 200$  and  $d_e = 10$  are constructed by Ackley, Dixon-Price, Levy and Sphere functions.<sup>1</sup> In the synthetic testing function experiments,  $I = 500$  points are used to estimate the integration of the soft-Copeland score,  $M = 30$  duels are used to initialize the GP model and  $N = 50$  duels are used for optimizing. The dimension of the low dimension subspace is 24, i.e.,  $2d = 24$  and the boundary is  $[-1, 1]^{24}$ . All experiments are repeated 20 times and the results are shown in Figure 1.

The horizontal axis shows the number of duels that are used for optimization and the vertical axis is the simple regret. Across all functions, PE-DBO gets the best result. The preferential intrinsic dimension of each synthetic testing function is  $d_p = 20$ , hence setting  $2d \geq d_p$  can reduce the effect of high dimension. The result in Figure 1 confirms that PE-DBO can deal with tasks with a very high input dimension but with a lower preferential intrinsic dimension. The performance of comp-GP-UCB is the same as KSS because we removed the second optimization part of comp-GP-UCB that use the function value. Furthermore, the objective functions of the two methods are nearly the same (is the probability of beating the average solution or the optimal solution), hence this result. PBO method often gets the worst performance. That's because the dimension of the preference function of PBO is  $2D$  while that for comp-GP-UCB and KSS is  $D$  hence PBO dramatically affected by the dimension of the dueling solution space. PE-DBO gets a better result in the first iteration than other baselines. Since lower dimensionality means better coverage with fewer points, 30 duels can lead to lower simple regret in the embedded space and make the following optimization perform well. PE-DBO has the fastest coverage rate than other methods because PE-DBO retains the flexibility of two solutions. The final results show PE-DBO handles the high-dimensional dueling optimization tasks well which confirms the validity of our motivation.

**Scalability.** To study the scalability of PE-DBO, we change the dimension of the synthetic testing functions  $D$  and analyze the results on different  $D$ . The experiments deploy on Levy and Ackley. All settings are the same as the experiments in the above section except  $D$  for the synthetic testing functions. We choose  $D = \{50, 100, 150, 200, 500\}$  and the experiment results are in Figure 2.

From Figure 2, the performances of different methods decrease when  $D$  increases. Since the search space of each method expands when  $D$  increases, which needs more effort to explore the solution space but the effort is limited. The result shows that the performance of PBO is significantly affected by the dimension  $D$  which confirms our motivation.

<sup>1</sup><http://www.sfu.ca/~ssurjano/optimization.html>

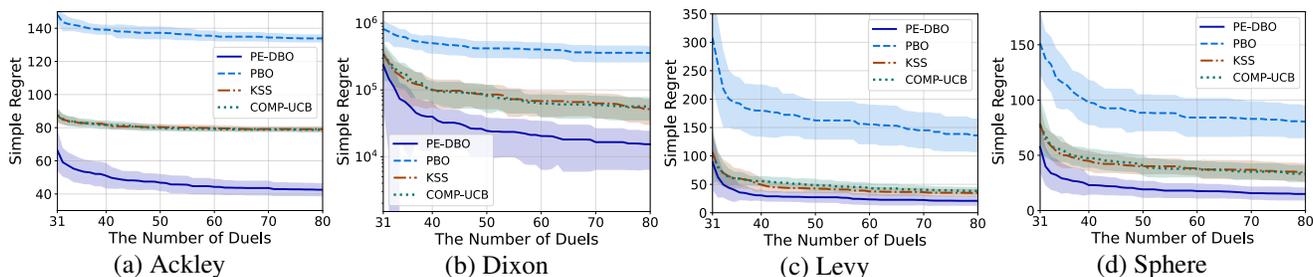


Figure 1: The convergence performance on the synthetic testing functions. KSS stands for kernel-self-sparring and COMP-UCB stands for comp-GP-UCB. All experiments have 50 iterations and repeat 20 times. We draw the mean and the deviation of simple regret. The horizontal axis is the number of duels and it begins from 31 since we use 30 duels to initialize each algorithm.

The methods that fix one solution are not dramatically affected by  $D$  but still get worse result when  $D$  increases. Since the preference intrinsic dimension is not changed across different  $D$ , the result of PE-DBO is the best and virtually unaffected by  $D$ . Thus, we can speculate that the convergence rate of PE-DBO is only depending on  $d$  but PBO depends on  $D$ , which is a significant improvement.

## On Real-world Tasks

**The Existence of Preferential Intrinsic Dimension on Real-world Datasets.** We verify PE-DBO on two real-world datasets. The first dataset is the Microsoft learning to rank (MSLR) dataset (Qin et al. 2010) and we use the MSLR-WEB10K version which has more than 10,000 queries. Each data has 136 different features to describe a website page. For each website page, there is a relevance judgment which is an integer value from 0 to 4 that indicates whether this page is irrelevant or perfectly relevant. The second dataset is ChEMBL dataset in design-bench (Trabucco et al. 2022), and design-bench is a benchmark for black-box model-based optimization problems. ChEMBL is to find a molecule with the highest MCHC value. Each molecule is encoded by 31 features by using the SMILES (Weininger 1988) method.

We use a neural network (NN) to fit MSLR dataset as the objective function, and judge the preference via the output of the NN since directly using it as an exact oracle is difficult. The NN has 3 hidden layers with 128, 64 and 32 units respectively and the Sigmoid function is used as the activation function. Random forest oracle provided by design-bench is

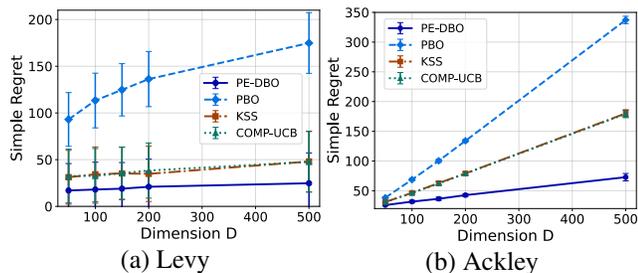


Figure 2: The scalability on the synthetic testing functions with different  $D$ . All settings are the same as Figure 1.

used as the objective function for ChEMBL dataset. We need to test whether those datasets fit the setting, i.e., having preferential intrinsic dimension. From Lemma 1 we only need to test the  $\epsilon$ -effective dimension of the objective function. Suppose the domain of the objective function  $f$  is  $\mathcal{X} \subset \mathbb{R}^D$  and  $\Delta_f$  denotes the maximal variation of  $f$  over  $\mathcal{X}$ . By restricting  $f$  on a subspace with dimension  $k$ , i.e.,  $\mathcal{X}^k \subset \mathcal{X}$ , we can assess the effect of  $\mathcal{X}^k$  on  $f$  by  $\mathcal{E} = \Delta_f / \Delta_f^k$ , where  $\Delta_f^k$  is the maximal variation of  $f$  over  $\mathcal{X}^k$ .

The result of MSLR dataset is shown in Figure 3(a). According to Figure 3(a), a subspace with dimension 50 can cover the total function value change and other dimensions are useless, i.e., the MSLR dataset has  $\epsilon$ -effective dimension no more than 50. Thus, we set  $2d = 100$  for the following experiment. By the same analysis with MSLR dataset, we found that the  $\epsilon$ -effective dimension of ChEMBL dataset is no more than 25, hence we set  $2d = 50$  for PE-DBO. We also find that in the ChEMBL dataset, certain features are completely invariant for different data. Therefore, since our objective function is a re-scale version of the oracle, the corresponding features become fixed values passed to oracle after the transformation, so it has no effect on the function value, and therefore ChEMBL does has the  $\epsilon$ -effective dimensionality.

**Superiority.** In the experiment on MSLR dataset, we set the boundary of low dimension subspace as  $[-\sqrt{d}, \sqrt{d}]^{2d}$  (other different boundaries have also been tested and please cf. Appendix E for more details). Because according to our analysis of the MSLR-WEB10K dataset, most pages with a relevance judgment of 4 tend to have a maximum or minimum value on each feature, i.e., the maximum value of the objective function is near the boundary of the domain. We got a bad performance when we set the boundary as  $[-1, 1]^{2d}$ , that because this boundary of low dimension subspace after embedding did not cover the boundary of the domain of the objective function. But setting a much bigger boundary possibly causes more effort on a useless domain, thus the setting of boundary is a trade-off, and how to automatically set the boundary is a promising direction to discuss. Since the optimal point is not at the boundary of the domain in ChEMBL dataset, the boundary of the low dimension subspace of ChEMBL dataset is  $[-1, 1]^{2d}$ . All experiments are repeated 20 times with  $N = 80$  to optimize,  $M = 25$ ,  $I = 600$  for MSLR and  $M = 15$ ,  $I = 300$  for ChEMBL.

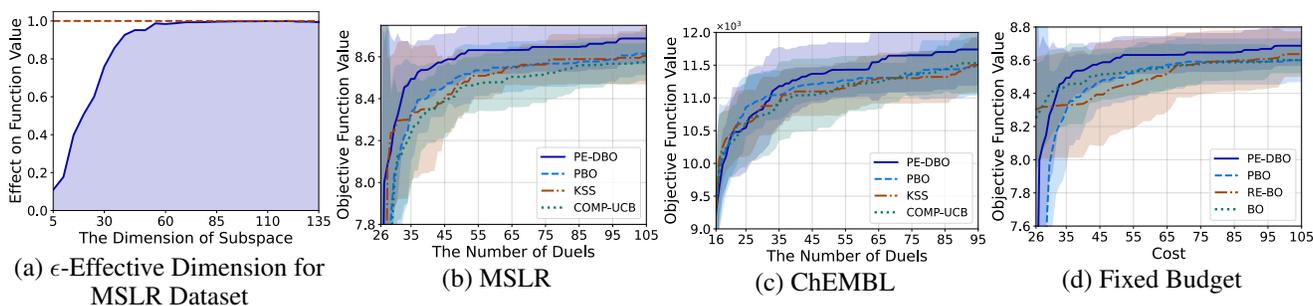


Figure 3: Results on real-world tasks. (a) is to verify the existence of  $\epsilon$ -effective dimension on the MSLR dataset. The horizontal axis is the dimension of the tested subspace and the vertical axis is the value of  $\mathcal{E}$ . (b) and (c) show the convergence results on MSLR and ChEMBL respectively. The vertical axis is the objective function value and the horizontal axis is the number of duels that the algorithm used. 25 duels and 15 duels are used to initialize the algorithm on MSLR and ChEMBL respectively. Thus, the horizontal axis begins from 26 and 16. (d) is the result with fixed budget on MSLR. In this task, evaluating the function value is twice as expensive as comparing pairs of solutions. The vertical axis is the function value and the horizontal axis is the cost that the algorithm has used. The preference-based methods use 25 duels to initialize GP, and the function value based methods can use 12.5 solutions (round up to 13) to initialize. Thus, the horizontal axis begins from 26.

Figure 3(b) and Figure 3(c) show that PE-DBO has the best result. From Figure 3(b), PE-DBO has the fastest convergence rate and the best result, since PE-DBO optimizes  $\pi_{f_p}$  which is a good approximation of  $\pi_f$  by using the preferential embedding matrix. The low dimension subspace is much smaller than the dueling solution space, and thus PE-DBO can get the global landscape of the objective function and leads to finding the global optima as fast as possible. Figure 3(c) shows the result on MSLR dataset. The result of PBO is worse since  $D = 31$  is still too high for PBO to optimize. From experiments in Appendix D, when the dimension of synthetic testing function is 10, PBO can beat the methods that fix one solution. This result shows that PE-DBO improves the scalability of PBO. On real-world tasks that have  $\epsilon$ -effective dimension, the preference embedding method makes optimization work better by optimizing  $\pi_{f_p}$ .

**The Cost of Dueling Optimization.** To verify the strength of the preference-based methods, we aim to show that pairwise comparison is cheaper but has a comparable efficacy to evaluating the function. This experiment fixes the total budget for two kinds of methods and compares their result. The setting is that the cost of evaluating the function value is twice expensive as that of making a pairwise comparison. Other settings are acceptable as long as function evaluation is more expensive than comparison. The function value based methods are the Bayesian optimization with upper confidence bound acquisition function (BO) (Srinivas et al. 2010) and Bayesian optimization with random embedding (RE-BO) (Wang et al. 2016b), the preference-based methods are PBO and PE-DBO. All methods are verified on the MSLR dataset and repeated 20 times. The preference-based methods use 25 duels to initialize, hence the function value based methods can use 12.5 solutions (round up to 13) to initialize.

The results are shown in Figure 3(d). From the result we can find that, after the initialize process, function value based methods start with a higher function value than preference-based methods. This may result from the optimization space of function value based methods is smaller than preference-

based methods and function value allows the algorithm to fully explore the global landscape hence function value based methods can find a better solution than preference-based methods. Since directly evaluating the function is much more expensive, the optimization process is slower than preference-based methods and is caught up at the end of this experiment. PBO method has a comparable result with BO, which means that with a limited budget, pairwise comparison result has the same efficacy as function value. After the embedding processes for PBO and BO, PE-DBO exhibiting a better result than RE-BO confirms the effectiveness of our method and implies the strength of preference-based methods that can use cheaper evaluation. Thus, preference-based methods are more effective on expensive tasks. We analyze the effect of hyper-parameters in Appendix E, and the result shows that PE-DBO is hyper-parameter insensitivity in some situations.

## Conclusion

This paper aims to make the high-dimensional dueling optimization tractable while retaining the flexibility of two solutions when dueling. We define the preferential intrinsic dimension and inject it into the dueling Bayesian optimization, resulting in PE-DBO. PE-DBO decouples optimization and pairwise comparison via the preferential embedding matrix, and it suffices to perform optimization within a much lower-dimensional subspace. We theoretically disclose that the preference function can be approximately preserved in the lower-dimensional preferential intrinsic subspace. Experiment results verify that molecule discovery and web page recommendation dueling optimization tasks have the preferential intrinsic dimension, and PE-DBO is superior in scalability compared with that of the SOTA methods. The paper implies that the scalability of PE-DBO only depends on the dimension of the low dimension subspace. Therefore, PE-DBO can be deployed on real-world dueling tasks with higher dimension. The future work includes adaptively choosing the boundary of the low dimension subspace and equipping the dueling optimization with other derivative-free algorithms.

## Acknowledgments

The authors would like to thank the anonymous reviewers for their constructive and valuable comments. The authors also would like to thank Yu-Peng Wu and Hua-Kang Lu for the helpful discussion. This work is supported by the National Natural Science Foundation of China (No. 62106076), the “Chenguang Program” sponsored by Shanghai Education Development Foundation and Shanghai Municipal Education Commission (No. 21CGA32), and the Natural Science Foundation of Shanghai (No. 21ZR1420300).

## References

- Aghajanyan, A.; Gupta, S.; and Zettlemoyer, L. 2021. Intrinsic Dimensionality Explains the Effectiveness of Language Model Fine-Tuning. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing*, 7319–7328. Virtual.
- Balandat, M.; Karrer, B.; Jiang, D. R.; Daulton, S.; Letham, B.; Wilson, A. G.; and Bakshy, E. 2020. BoTorch: A Framework for Efficient Monte-Carlo Bayesian Optimization. In *Advances in Neural Information Processing Systems 33*, 21524–21538. Virtual.
- Benavoli, A.; Azzimonti, D.; and Piga, D. 2021. Preferential Bayesian Optimisation with Skew Gaussian Processes. In *Proceedings of the Genetic and Evolutionary Computation Conference Companion*, 1842–1850. Lille, France.
- Canal, G.; Massimino, A. K.; Davenport, M. A.; and Rozell, C. J. 2019. Active Embedding Search via Noisy Paired Comparisons. In *Proceedings of the 36th International Conference on Machine Learning*, 902–911. Long Beach, CA.
- Conn, A. R.; Scheinberg, K.; and Vicente, L. N. 2009. *Introduction to Derivative-Free Optimization*. Philadelphia, PA: SIAM.
- González, J.; Dai, Z.; Damianou, A. C.; and Lawrence, N. D. 2017. Preferential Bayesian Optimization. In *Proceedings of the 34th International Conference on Machine Learning*, 1282–1291. Sydney, Australia.
- Hansen, N.; Müller, S. D.; and Koumoutsakos, P. 2003. Reducing the Time Complexity of the Derandomized Evolution Strategy with Covariance Matrix Adaptation (CMA-ES). *Evolutionary computation*, 11(1): 1–18.
- Jones, D. R.; Schonlau, M.; and Welch, W. J. 1998. Efficient Global Optimization of Expensive Black-Box Functions. *Journal of Global optimization*, 13(4): 455–492.
- Kahneman, D.; and Tversky, A. 1979. On the Interpretation of Intuitive Probability: A Reply to Jonathan Cohen. *Cognition*.
- Kirschner, J.; Mutny, M.; Hiller, N.; Ischebeck, R.; and Krause, A. 2019. Adaptive and Safe Bayesian Optimization in High Dimensions via One-Dimensional Subspaces. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, 3429–3438. Long Beach, CA.
- Liu, Y.; Hu, Y.; Qian, H.; Yu, Y.; and Qian, C. 2022. ZOOpt: A Toolbox for Derivative-Free Optimization. *Science China Information Sciences*, 65: 207101.
- Mikkola, P.; Todorovic, M.; Järvi, J.; Rinke, P.; and Kaski, S. 2020. Projective Preferential Bayesian Optimization. In *Proceedings of the 37th International Conference on Machine Learning*, 6884–6892. Virtual.
- Qian, H.; Hu, Y.-Q.; and Yu, Y. 2016. Derivative-Free Optimization of High-Dimensional Non-Convex Functions by Sequential Random Embeddings. In *Proceedings of the 25th International Joint Conference on Artificial Intelligence*, 1946–1952. New York, NY.
- Qian, H.; and Yu, Y. 2021. Derivative-Free Reinforcement Learning: A Review. *Frontiers of Computer Science*, 15(6): 156336.
- Qin, T.; Liu, T.; Xu, J.; and Li, H. 2010. LETOR: A Benchmark Collection for Research on Learning to Rank for Information Retrieval. *Information Retrieval*, 13(4): 346–374.
- Shahriari, B.; Swersky, K.; Wang, Z.; Adams, R. P.; and de Freitas, N. 2016. Taking the Human Out of the Loop: A Review of Bayesian Optimization. *Proceedings of the IEEE*, 104(1): 148–175.
- Shields, B. J.; Stevens, J.; Li, J.; Parasram, M.; Damani, F.; Alvarado, J. I. M.; Janey, J. M.; Adams, R. P.; and Doyle, A. G. 2021. Bayesian Reaction Optimization as a Tool for Chemical Synthesis. *Nature*, 590(7844): 89–96.
- Snoek, J.; Larochelle, H.; and Adams, R. P. 2012. Practical Bayesian Optimization of Machine Learning Algorithms. In *Advances in Neural Information Processing Systems 25*, 2960–2968. Lake Tahoe, NV.
- Song, L.; Xue, K.; Huang, X.; and Qian, C. 2022. Monte Carlo Tree Search based Variable Selection for High Dimensional Bayesian Optimization. In *Advances in Neural Information Processing Systems 35*. New Orleans, LA.
- Srinivas, N.; Krause, A.; Kakade, S. M.; and Seeger, M. W. 2010. Gaussian Process Optimization in the Bandit Setting: No Regret and Experimental Design. In *Proceedings of the 27th International Conference on Machine Learning*, 1015–1022. Haifa, Israel.
- Sui, Y.; Zhuang, V.; Burdick, J. W.; and Yue, Y. 2017. Multi-Dueling Bandits with Dependent Arms. In *Proceedings of the 33rd Conference on Uncertainty in Artificial Intelligence*. Sydney, Australia.
- Sui, Y.; Zoghi, M.; Hofmann, K.; and Yue, Y. 2018. Advancements in Dueling Bandits. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, 5502–5510. Stockholm, Sweden.
- Trabucco, B.; Geng, X.; Kumar, A.; and Levine, S. 2022. Design-Bench: Benchmarks for Data-Driven Offline Model-Based Optimization. In *Proceedings of the 39th International Conference on Machine Learning*, 21658–21676. Baltimore, MD.
- Tucker, M.; Cheng, M.; Novoseller, E. R.; Cheng, R.; Yue, Y.; Burdick, J. W.; and Ames, A. D. 2020. Human Preference-Based Learning for High-Dimensional Optimization of Exoskeleton Walking Gaits. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 3423–3430. Las Vegas, NV.

Wang, X.; Xu, C.; Guo, Y.; and Qian, H. 2016a. Constrained Preference Embedding for Item Recommendation. In *Proceedings of the 25th International Joint Conference on Artificial Intelligence*, 2139–2145. New York, NY.

Wang, Z.; Hutter, F.; Zoghi, M.; Matheson, D.; and de Feitas, N. 2016b. Bayesian Optimization in a Billion Dimensions via Random Embeddings. *Journal of Artificial Intelligence Research*, 55: 361–387.

Weininger, D. 1988. SMILES, a Chemical Language and Information System. 1. Introduction to Methodology and Encoding Rules. *Journal of Chemical Information and Computer Sciences*, 28(1): 31–36.

Xu, Y.; Joshi, A.; Singh, A.; and Dubrawski, A. 2020. Zeroth Order Non-Convex Optimization with Dueling-Choice Bandits. In *Proceedings of the 36th Conference on Uncertainty in Artificial Intelligence*, 899–908. Virtual.

Yue, Y.; Broder, J.; Kleinberg, R.; and Joachims, T. 2012. The K-Armed Dueling Bandits Problem. *Journal of Computer and System Science*, 78(5): 1538–1556.