

Expensive Multi-Objective Bayesian Optimization Based on Diffusion Models

Bingdong Li^{1,2}, Zixiang Di^{1,2}, Yongfan Lu^{1,2}, Hong Qian¹,
Feng Wang^{3*}, Peng Yang^{4,5}, Ke Tang⁵, Aimin Zhou^{1*}

¹Shanghai Frontiers Science Center of Molecule Intelligent Syntheses, Shanghai Institute of AI for Education, and School of Computer Science and Technology, East China Normal University, Shanghai 200062, China

²Key Laboratory of Advanced Theory and Application in Statistics and Data Science, Ministry of Education

³School of Computer Science, Wuhan University, Wuhan 430072, China

⁴Department of Statistics and Data Science, Southern University of Science and Technology, Shenzhen 518055, China

⁵Guangdong Provincial Key Laboratory of Brain-Inspired Intelligent Computation, Department of Computer Science and Engineering, Southern University of Science and Technology, Shenzhen 518055, China

bdli@cs.ecnu.edu.cn, 51265901113@stu.ecnu.edu.cn, 51255901053@stu.ecnu.edu.cn, hqian@cs.ecnu.edu.cn, fengwang@whu.edu.cn, yangp@sustech.edu.cn, tangk3@sustech.edu.cn, amzhou@cs.ecnu.edu.cn

Abstract

Multi-objective Bayesian optimization (MOBO) has shown promising performance on various expensive multi-objective optimization problems (EMOPs). However, effectively modeling complex distributions of the Pareto optimal solutions is difficult with limited function evaluations. Existing Pareto set learning algorithms may exhibit considerable instability in such expensive scenarios, leading to significant deviations between the obtained solution set and the Pareto set (PS). In this paper, we propose a novel Composite Diffusion Model based Pareto Set Learning algorithm (CDM-PSL) for expensive MOBO. CDM-PSL includes both unconditional and conditional diffusion model for generating high-quality samples efficiently. Besides, we introduce a weighting method based on information entropy to balance different objectives. This method is integrated with a guiding strategy to appropriately balancing different objectives during the optimization process. Experimental results on both synthetic and real-world problems demonstrates that CDM-PSL attains superior performance compared with state-of-the-art MOBO algorithms.

Code — <https://github.com/ilog-ecnu/CDM-PSL>

Introduction

Expensive multi-objective optimization problems are commonly seen in various fields, such as antenna structure design (Ding et al. 2019), financial cloud services (Yang et al. 2024), and clinical drug trials (Yu, Ramakrishnan, and Meinzer 2019). Handling EMOPs involves optimizing multiple (often conflicting) objectives simultaneously with limited function evaluations due to time or financial reasons.

To meet these challenges, multi-objective Bayesian optimization (MOBO) (Laumanns and Ocenasek 2002), an extension of single-objective Bayesian Optimization (BO) (Moćkus 1975) for expensive multi-objective optimization problems, has emerged as a promising paradigm. BO itself is recognized as an exceedingly effective strategy for

global optimization, particularly noted for its success in addressing black-box optimization issues (Snoek, Larochelle, and Adams 2012; Qian, Xiong, and Xue 2020; Huang et al. 2024; Wang et al. 2024; Li et al. 2024a). The core principle of BO involves creating probabilistic surrogate models, typically Gaussian processes (GP), that closely represent the black-box functions. These models are utilized in conjunction with acquisition functions to seek out globally optimal solutions. BO has been widely used in a variety of fields, including hyperparameter tuning (Bergstra et al. 2011), A/B testing (Letham et al. 2019), combinatorial optimization (Zhang et al. 2015), among others. MOBO represents a fusion of Bayesian optimization with multi-objective optimization. A widely adopted MOBO approach is the random scalarization technique, which effectively translates a multi-objective optimization problem into several single-objective optimization problems. Another noteworthy strategy in MOBO involves the use of sophisticated acquisition functions, such as the expected hypervolume improvement (EHVI) (Couckuyt, Deschrijver, and Dhaene 2014) and predictive entropy search (PESMO) (Hoffman and Ghahramani 2015; Hernández-Lobato et al. 2016). Among them, Pareto set learning (PSL) based methods (e.g. (Lin et al. 2022)), which aim to modeling the Pareto set via machine learning techniques, have shown promising performance.

However, effectively capturing and modeling distributions of limited samples is difficult when faced with expensive multi-objective Bayesian optimization problems (EMOPs). Existing PSL algorithms may exhibit considerable instability in expensive scenarios. This instability can lead to substantial deviations between the obtained solution set and the true Pareto set (PS). In other words, the quality of the resulting solution set is highly influenced by the performance of the PSL model, which is largely limited on EMOPs. On the other hand, Diffusion model (DM), inspired by the natural diffusion of gases, has been successfully applied in various domains such as computer vision (Dhariwal and Nichol 2021; Baranchuk et al. 2021; Yu et al. 2022), natural language processing (Austin et al. 2021a), and wave-

*Corresponding authors: Feng Wang and Aimin Zhou.
Copyright © 2025, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

form signal processing (Leng et al. 2022) etc, due to its advantage in distribution coverage, stationary training objective, and effortless scalability. Moreover, the methodology of DM is particularly effective in scenarios with limited sample sizes, as it can efficiently extract substantial information from each individual sample, thereby amplifying the learning impact (Yang et al. 2023). These features of DM show promise for its application in PSL of EMOPs.

In this paper, we propose a novel Composite Diffusion Model based Pareto Set Learning algorithm named CDM-PSL for expensive MOBO, where DM progressively learns the inherent distribution of the high-quality solution set. The major contributions of this paper are summarized as follows:

1) We introduce a composite diffusion model based Pareto set learning method for offspring generation for expensive MOBO, which includes both unconditional and conditional sample generation.

2) We devise a guided sampling process to improve the quality of solutions generated by the diffusion model, resulting in a conditional diffusion model;

3) We introduce a guiding strategy based on information entropy weighting to balance the importance of different objectives of EMOPs. This method ensures that all objectives are appropriately balanced and given due consideration during the optimization process;

4) We have conducted extensive experiments on both synthetic benchmarks and real-world problems, clearly demonstrating that CDM-PSL obtains superior performance compared with various state-of-the-art MOBO algorithms.

Backgrounds

Multi-objective Optimization

A multi-objective optimization problem (MOP) can be defined as follows:

$$\begin{aligned} & \text{minimize } \mathbf{f}(\mathbf{x}) = (f_1(\mathbf{x}), f_2(\mathbf{x}), \dots, f_M(\mathbf{x}))^T \\ & \text{subject to } \mathbf{x} \in \Omega, \end{aligned} \quad (1)$$

where $\mathbf{x} = (x_1, x_2, \dots, x_d)$ represents the decision vector, $\mathbf{f}(\cdot): \Omega \rightarrow \Lambda$ includes M ($M \geq 2$) objectives, Ω symbolizes the non-empty *decision space*, and Λ is the *objective space*. An MOP is considered expensive when the evaluation of $\mathbf{f}(\mathbf{x})$ involves either time-intensive computations or high-cost experimental procedures.

Definition 1 (Pareto dominance) *Considering two solutions \mathbf{x} and $\mathbf{y} \in \Omega$, \mathbf{x} is said to dominate \mathbf{y} (expressed as $\mathbf{x} \prec \mathbf{y}$) if and only if the following conditions are met: 1) $\forall i \in \{1, 2, \dots, M\}, f_i(\mathbf{x}) \leq f_i(\mathbf{y})$; 2) $\exists j \in \{1, 2, \dots, M\}, f_j(\mathbf{x}) < f_j(\mathbf{y})$. This definition encapsulates the essential criterion for determining the quality of solutions in terms of meeting multiple objectives simultaneously. (Yu 1974)*

Definition 2 (Pareto optimal) *A solution $\mathbf{x}^* \in \Omega$ is Pareto optimal if there exists no other solution $\mathbf{x} \in \Omega$ that can dominate it. This implies that within the feasible region Ω , \mathbf{x}^* is considered to be Pareto optimal if no alternative solution offers better outcomes across all objectives without being worse in at least one of them. The collection of all the Pareto optimal solutions and their objective vectors are the Pareto Set (PS) and Pareto Front (PF) respectively.*

Diffusion Models

Diffusion models are a specialized form of probabilistic generative models that operate by learning to reverse a forward process that gradually increases noise in the training data (Sohl-Dickstein et al. 2015; Ho, Jain, and Abbeel 2020). They have demonstrated remarkable performance on a wide variety of tasks, such as image generation (Austin et al. 2021b; Bao et al. 2022), voice synthesis (Liu et al. 2022), video generation (Ho et al. 2022) and inpainting (Lugmayr et al. 2022). Training a DM involves two processes: the forward diffusion process and the backward denoising process.

Forward Process. In the forward phase, Gaussian noise is added to the input data step by step until a pure Gaussian noise is produced, which is a Markovian process. Given an initial data distribution $\mathbf{x}_0 \sim q(\mathbf{x})$, the noised $x_1, x_2 \dots, x_T$ can be obtained from the following equation:

$$q(x_t|x_{t-1}) = \mathcal{N}\left(x_t; \sqrt{1 - \beta_t} \cdot x_{t-1}, \beta_t \cdot \mathbf{I}\right), \forall t \in \{1, \dots, T\},$$

where T is the number of diffusion steps and the step sizes are controlled by a variance schedule $\{\beta_t \in (0, 1)\}_{t=1}^T$. Moreover, the properties of this recursive formula make it possible to obtain $q(x_t)$ directly from x_0 by the following equation:

$$q(x_t|x_0) = \mathcal{N}\left(x_t; \sqrt{\hat{\beta}_t} \cdot x_0, (1 - \hat{\beta}_t) \cdot \mathbf{I}\right), \forall t \in \{1, \dots, T\},$$

where $\hat{\beta}_t = \prod_{i=1}^t \alpha_i$ and $\alpha_t = 1 - \beta_t$. Thus, x_t can be sampled from $q(x_t|x_0)$ as follows:

$$x_t = \sqrt{\hat{\beta}_t} \cdot x_0 + \sqrt{(1 - \hat{\beta}_t)} \cdot z_t,$$

where $z_t \sim \mathcal{N}(0, \mathbf{I})$.

Reverse Process. The reverse process recreates the true sample from a Gaussian noise input $x_T \sim \mathcal{N}(0, \mathbf{I})$:

$$q(x_{t-1}|x_t) = \mathcal{N}(x_{t-1}; \mu(x_t, t), \Sigma(x_t, t))$$

However, $q(x_{t-1}|x_t)$ cannot easily be evaluated due to the reverse process lacking a complete dataset and therefore we need to train a neural network $p_\theta(x_{t-1}|x_t) = \mathcal{N}(x_{t-1}; \mu_\theta(x_t, t), \Sigma_\theta(x_t, t))$ to approximate these conditional probabilities. Specifically, The model takes in the noisy data x_t and the corresponding embedding at time step t as input, and is trained to predict the mean $\mu_\theta(x_t, t)$ and the covariance $\Sigma_\theta(x_t, t)$. Based on this, Ho (Ho, Jain, and Abbeel 2020) proposed to fix the covariance $\Sigma_\theta(x_t, t)$ to a constant value and reformulating the mean $\mu_\theta(x_t, t)$ as a function dependent on noise, as follows:

$$\mu_\theta = \frac{1}{\sqrt{\alpha_t}} \cdot \left(x_t - \frac{1 - \alpha_t}{\sqrt{1 - \hat{\beta}_t}} \cdot z_\theta(x_t, t) \right).$$

This enables DM to predict the noise of the data rather than directly the mean and covariance.

Diffusion Model Based Optimization Algorithms

Krishnamoorthy proposed a black-box optimization algorithm named DDOM (Krishnamoorthy, Mashkaria, and Grover 2023), based on the diffusion model. This algorithm converts a single-objective optimization problem into a continuous diffusion process, leveraging the inverse process of the diffusion model to efficiently address complex problems. Subsequently, Yan and Jin’s EmoDM (Yan and Jin 2024) extended the application of the diffusion model to multi-objective optimization. By learning the noise distribution in the previous evolutionary search task, a set of non-dominated solutions can be generated for the new multi-objective optimization problem without further evolutionary search. Fang’s DMO (Fang et al. 2024) further demonstrates the diffusion model’s efficacy by applying it to create a gasoline hybrid scheduling scheme, highlighting its capability in solving practical multi-objective optimization challenges. Building upon these contributions, this paper’s CDM-PSL advances the field by optimizing the balance between solution convergence and diversity through the integration of conditional and unconditional diffusion models. Moreover, CDM-PSL incorporates gradient information, weighted by information entropy (Zhan et al. 2024), into the process of generating solutions, significantly enhancing convergence performance during the early-stage iterations. The combination of these strategies makes CDM-PSL have competitive performance in solving EMOPs.

Our Method

Overview

We present a composite diffusion model based Pareto set learning method for EMOBO, denoted as CDM-PSL (Algorithm 1 and Figure 1). CDM-PSL contains three components to generate offspring: data extraction, diffusion model training and conditional generation.

Data Extraction

To prepare training data for Pareto set learning (Algorithm 2), we introduce the shift-based density estimation (SDE) from (Li, Yang, and Liu 2013) as fitness function, which is mathematically represented as:

$$Fitness(\mathbf{p}) = \min_{\mathbf{q} \in \mathbf{Y}_k \setminus \mathbf{p}} \sqrt{\sum_{i=1}^M (\max\{0, f_i(\mathbf{q}) - f_i(\mathbf{p})\})^2}. \quad (2)$$

In this formula, \mathbf{p} and \mathbf{q} are solutions within the set \mathbf{Y}_k , and $f_i(\mathbf{p})$ indicates the i -th objective value of solution \mathbf{p} . The SDE methodology assesses the quality of samples based on their convergence and diversity characteristics. From \mathbf{X}_k , a total of T candidate solutions ($T = \frac{|\mathbf{X}_k|}{3}$) with superior SDE values are identified as Pareto optimal samples.

Diffusion Model Training

Diffusion Process. Given set of samples X_k^* and a specified step t , the diffusion process involves gradually introducing Gaussian noise $\epsilon \sim \mathcal{N}(0, I)$ to X_k^* over t steps:

$$X_{k,t}^* = \sqrt{1 - \beta_t} X_{k,t-1}^* + \sqrt{\beta_t} \epsilon. \quad (3)$$

Algorithm 1: The framework of MOBO with CDM-PSL

- 1: **Input:** black-box function $f(x)$, number of iterations K , batch size B , number of initial solutions N
- 2: **Output:** final solutions $\{\mathbf{X}_K, \mathbf{Y}_K\}$, Pareto front \mathcal{P}_f
- 3: Initialize N solutions $\{\mathbf{X}_0, \mathbf{Y}_0\}$ by LHS
- 4: **for** $k = 0$ **to** $K - 1$ **do**
- 5: Train surrogate model (Gaussian process) GP_i^k based on $\{\mathbf{X}_k, \mathbf{Y}_k\}$ for each objective $f_i, i = 1, \dots, M$
- 6: $\mathbf{X}_k^* \leftarrow$ Data extraction based on $\{\mathbf{X}_k, \mathbf{Y}_k\}$ (**Algorithm 2**)
- 7: **if** $Flag_{CDM}$ is True **then**
- 8: $\mathbf{S} \leftarrow$ Pareto set learning based on \mathbf{X}_k^* by CDM-PSL (**Algorithm 3**)
- 9: **else**
- 10: $\mathbf{S} \leftarrow$ Generate offspring using *Operator*
- 11: **end if**
- 12: $\mathbf{X}_k^B \leftarrow$ Batch selection based on GP^k, \mathbf{S} and $\{\mathbf{X}_k, \mathbf{Y}_k\}$
- 13: Evaluate and update $\mathbf{X}_{k+1} \leftarrow \mathbf{X}_k \cup \mathbf{X}_k^B, \mathbf{Y}_{k+1} \leftarrow \mathbf{Y}_k \cup f(\mathbf{X}_k^B)$
- 14: Decide whether to invert $Flag_{CDM}$ based on HV
- 15: **end for**
- 16: Approximate the Pareto front \mathcal{P}_f by non-dominated solutions in \mathbf{Y}_K

Algorithm 2: Data Extraction

- 1: **Input:** all already-evaluated solutions in the k -th iteration $\{\mathbf{X}_k, \mathbf{Y}_k\}$, number of real samples T
- 2: **Output:** solutions after data extraction \mathbf{X}_k^*
- 3: $Fitness \leftarrow$ Calculate the fitness of each solution in $\{\mathbf{X}_k, \mathbf{Y}_k\}$ by Eq. 2
- 4: $\mathbf{X}_k^* \leftarrow$ Select top T candidate solutions with $Fitness$

In this equation, $X_{k,t}^*$ denotes the data at step t , and $\beta_t \in [1e - 5, 5e - 2]$ represents the noise level at step t . The DM learning process for PSL operates as a Markov chain. This stepwise approach simplifies the learning task compared to direct Pareto set learning and effectively captures the distribution characteristics of optimal samples.

Noise Prediction. The noise prediction phase involves reconstructing the samples $X_{k,t}^*$, which have undergone the diffusion process, back to their original state, X_k^* . This reconstruction is achieved through a model \mathcal{M} that predicts the noise added at each step, thereby reversing the diffusion process. This process follows the equation 4:

$$\tilde{X}_{k,t-1}^* = \frac{1}{\sqrt{1 - \beta_t}} \left(X_{k,t}^* - \frac{\beta_t}{\sqrt{1 - \sum_{s=1}^t \beta_s}} \epsilon_\theta(X_{k,t}^*, t) \right). \quad (4)$$

In this equation, $\tilde{X}_{k,t-1}^*$ represents the data after reconstruction, and θ denotes the parameters of the model. The term $\epsilon_\theta(X_{k,t}^*, t)$ is the predicted noise by model \mathcal{M} at step t .

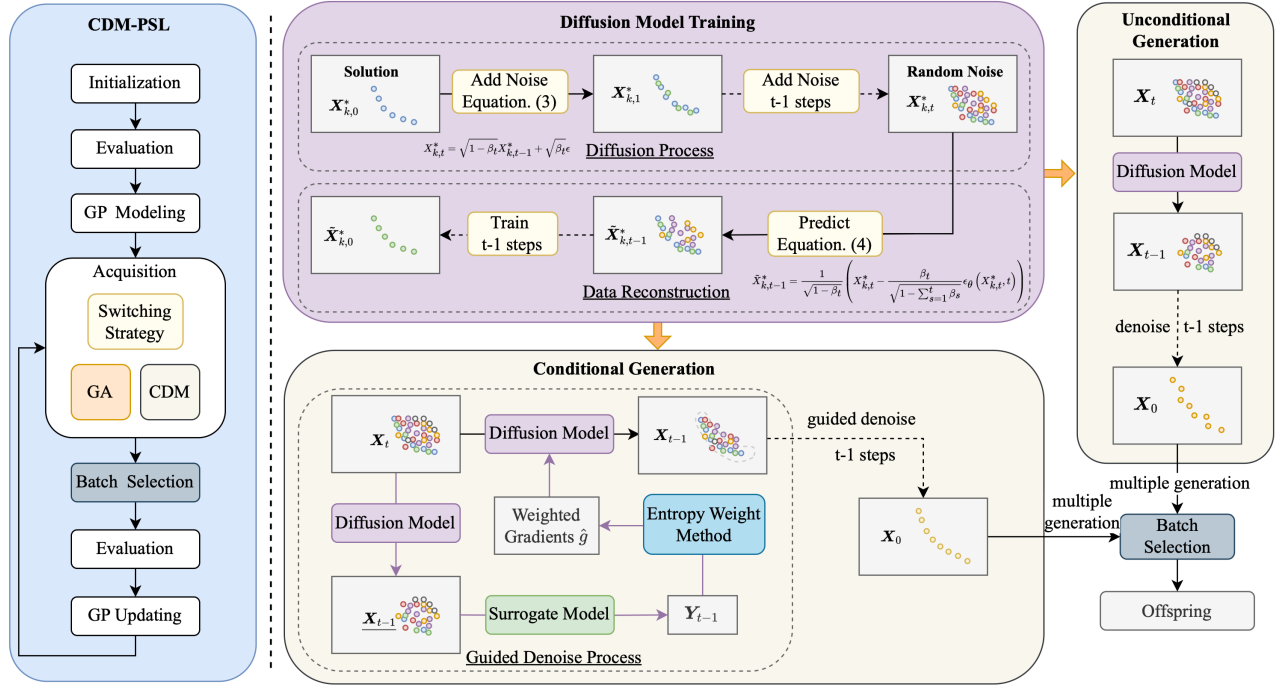


Figure 1: (left) The framework of CDM-PSL. (right) Diffusion Model Training (DMT), Conditional Generation (CG), and Unconditional Generation (UG). DMT involves learning from the selected samples through multiple steps; CG is designed to create high-quality samples with an optimized distribution; UG is used to generate diverse samples with high efficiency.

Algorithm 3: Composite Diffusion Model based Generation

```

1: Input: trained model  $\mathcal{M}$ 
2: Parameter: steps  $T$ , number of generation  $N_1, N_2$ 
3: Output: solutions generated by CDM  $\mathcal{S}$ 
4:  $\mathcal{S} \leftarrow \emptyset$ 
5: for  $i = 1$  to  $N_1$  do
6:   for  $t = T$  to  $1$  do
7:      $z \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ 
8:      $\hat{g} \leftarrow$  Calculate weighted gradient by entropy weight method and surrogate model
9:      $x_{t-1} \leftarrow \frac{1}{\sqrt{\alpha_t}}(x_t - \frac{1-\alpha_t}{\sqrt{1-\alpha_t}}\epsilon_\theta(x_t, t)) + \sigma_t^2\hat{g} + \sigma_t z$ 
10:   end for
11:    $\mathcal{S} \leftarrow \mathcal{S} \cup \{x_0\}$ 
12: end for
13: for  $i = 1$  to  $N_2$  do
14:   for  $t = T$  to  $1$  do
15:      $z \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ 
16:      $x_{t-1} \leftarrow \frac{1}{\sqrt{\alpha_t}}(x_t - \frac{1-\alpha_t}{\sqrt{1-\alpha_t}}\epsilon_\theta(x_t, t)) + \sigma_t z$ 
17:   end for
18:    $\mathcal{S} \leftarrow \mathcal{S} \cup \{x_0\}$ 
19: end for

```

The loss function \mathcal{L} for model \mathcal{M} is defined as:

$$\mathcal{L} = \frac{1}{\mathcal{H}} \sum_{i=1}^{\mathcal{H}} (\epsilon - \epsilon_\theta(x_{k,i,t}^*, t)). \quad (5)$$

This equation takes \mathcal{H} into account, the total number of optimal samples, where each $x_{k,i,t}^*$ is an instance from $X_{k,t}^*$, and $\epsilon_\theta(x_{k,i,t}^*, t)$ represents the predicted noise. The training of \mathcal{M} , using Equation 5, aims to minimize the loss \mathcal{L} , thereby enhancing the accuracy of noise prediction.

The entire DM training process, comprising both the diffusion process and noise prediction, plays a crucial role in effective Pareto Set Learning for EMOPs. This method presents a novel approach to learning high-quality solutions, striking a balancing between exploration and exploitation in the search space. Also, it might be interesting to study hyperbolic search space in the future (Li et al. 2024b).

Conditional Generation

Conditional Generation (CG) module includes two components: guided denoising process and weighted gradient. The CG module of CDM-PSL adaptively adjusts the weights of different objectives at each step of denoising based on the current estimated objective values. This is in contrast to using pre-defined hyperparameters or average weights determined by human experience.

Guided Denoising Process. Given step t and sample x_t , the guided denoising process can be implemented as

$$x_{t-1} = \frac{1}{\sqrt{\alpha_t}}(x_t - \frac{1-\alpha_t}{\sqrt{1-\alpha_t}}\epsilon_\theta(x_t, t)) + \sigma_t^2\hat{g} + \sigma_t z, \quad (6)$$

where α_t represents $1 - \beta_t$, $\epsilon_\theta(x_t, t)$ is the predicted noise by the trained DM, σ_t is the standard deviation of the t -th

step, and \hat{g} denotes the weighted gradients for guiding the denoising process.

To procure the gradients essential for guiding the model in sample generation, we establish separate Gaussian Process (GP) models for each objective, as proposed by Balandat (Balandat et al. 2020). These models are utilized to compute the objective values for all generated samples, thus obtaining gradients g for each objective, facilitating the realization of conditional generation.

Weighted Gradients. Employing weighted gradients is essential to address EMOPs and it requires the determination of appropriate weights for each objective. Therefore, we use a weighting methodology grounded in information entropy, facilitating the derivation of these weighted gradients \hat{g} .

In order to obtain weights based on information entropy, it is necessary to first normalize the objective values. Let y_{ij} represent the j -th objective value of the i -th individual. The normalized objective value \tilde{y}_{ij} is calculated by min-max normalization. Subsequent to this, for each objective j ($j = 1, 2, \dots, M$), the probability matrix P_{ij} is calculated using Equation 7, where $k = 1, 2, \dots, N$ and N represents the number of individuals in the population.

$$P_{ij} = \frac{\tilde{y}_{ij}}{\sum_{k=1}^N \tilde{y}_{kj}}. \quad (7)$$

Subsequently, for each objective j ($j = 1, 2, \dots, M$), the information entropy E_j is computed. The method of calculation is as outlined in Equation 8:

$$E_j = -\frac{1}{\ln(N)} \sum_{i=1}^N (P_{ij} \times \ln(P_{ij} + \eta)). \quad (8)$$

Herein, $i = 1, 2, \dots, N$ and $j = 1, 2, \dots, M$, where N signifies the total number of samples, and M represents the quantity of objectives. To avoid the occurrence of $\ln(0)$, a small positive number, η , is introduced. The computation of information entropy relies on Shannon entropy (Shannon 1948), expressed as $H(X) = -\sum (P(x) \times \log(P(x)))$. Additionally, the coefficient $\frac{1}{\ln(N)}$ is employed to guarantee that the values of information entropy are confined within the range of 0 to 1.

Finally, for each objective $j = 1, 2, \dots, M$, the weight W_j is computed using Equation 9, where $k = 1, 2, \dots, M$, and M denotes the total number of objectives.

$$W_j = \frac{1 - E_j}{\sum_{k=1}^M (1 - E_k)}. \quad (9)$$

Using the calculated weights W , we can derive the entropy weighted gradients (EWG) \hat{g} as:

$$\hat{g} = \sum_j W_j (\nabla F_j - \text{coef_lcb} \cdot \nabla S_j), \quad (10)$$

where W_j is the entropy-based weight for the j -th objective; ∇F_j is the gradient of the mean F_j of the surrogate model (Gaussian process) for the j -th objective; ∇S_j is the gradient of the standard deviation S_j of the surrogate model for the j -th objective; coef_lcb is a coefficient (default setting is

0.1) used to balance the mean and standard deviation in the Lower Confidence Bound (LCB) strategy.

The entropy weighting method, serving as an adaptive weight allocation approach, assigns weights based on the information entropy of objectives. This method reduces subjectivity in weighting gradients for different objective values, and ensures that the algorithm places more emphasis on objectives with rich information content, thereby achieving better performance on EMOPs.

Selection Strategy

Batch Selection. After obtaining the solutions sampled by the CDM-PSL, we employ the batch selection strategy of PSL-MOBO (Lin et al. 2022) to select a small subset $\mathbf{X}_k^B = \{x_b | b = 1, \dots, B\}$. Specifically, this strategy uses the Hypervolume (HV) indicator (Zitzler and Thiele 1999) as the selection criteria, which is defined as follows:

$$\text{HV}(S) = \Lambda(\{q \in R^d | \exists p \in S : p \leq q \text{ and } q \leq r\}), \quad (11)$$

where S denotes a solution set, r is a reference vector, and $\Lambda(\cdot)$ denotes the Lebesgue measure.

Operator Selection. At the end of each iteration round, we have devised operator switching strategy based on the growth rate of the HV indicator. In algorithm 1, $Flag_{CDM}$ is a flag used to determine if CDM is currently being used to generate offspring. After passing through an iteration window (e.g. three evaluation rounds), if the HV indicator's growth rate falls below a predefined threshold (default setting is 5%), we switch the offspring generation operator to another one (e.g. from CDM-PSL to Genetic Algorithm (GA)). This is to prevent the algorithm from becoming trapped in local optima specific to the current operator.

Experimental Study

Experimental Settings

Instances and Baselines: To comprehensively validate the performance of CDM-PSL, experiments were conducted on 9 benchmark problems (2- and 3-objective ZDT1-3 (Zitzler, Deb, and Thiele 2000) and DTLZ2-7 (Deb et al. 2005)) and 7 real-world problems (Tanabe and Ishibuchi 2020). Moreover, we have compared CDM-PSL with 9 state-of-the-art and classical algorithms, including NSGA-II (Deb et al. 2002), MOEA/D-EGO (Zhang et al. 2009), TSEMO (Bradford, Schweidtmann, and Lapkin 2018), USEMO-EI (Belakaria et al. 2020), DGEMO (Konakovic Lukovic, Tian, and Matusik 2020), PSL-MOBO (Lin et al. 2022), qNparEGO (Knowles 2006), qEHVI (Daulton, Balandat, and Bakshy 2020) and qNEHVI (Daulton, Balandat, and Bakshy 2021).

Parameter Settings: For fair comparison, the population size N was initialized to 100 for all the compared algorithms. Bayesian optimization algorithms were executed for 20 batches, each with a batch size of 5, across all algorithms. Each method was randomly run 10 times. For CDM-PSL, the hyperparameter t was set to 25, the number of CG N_1 was 10 and number of UG N_2 was 100, the batch size m was 1024, the learning rate γ was 0.001, with training spanning 4000 epochs. The configurations for other methods were aligned with those in their original publications.

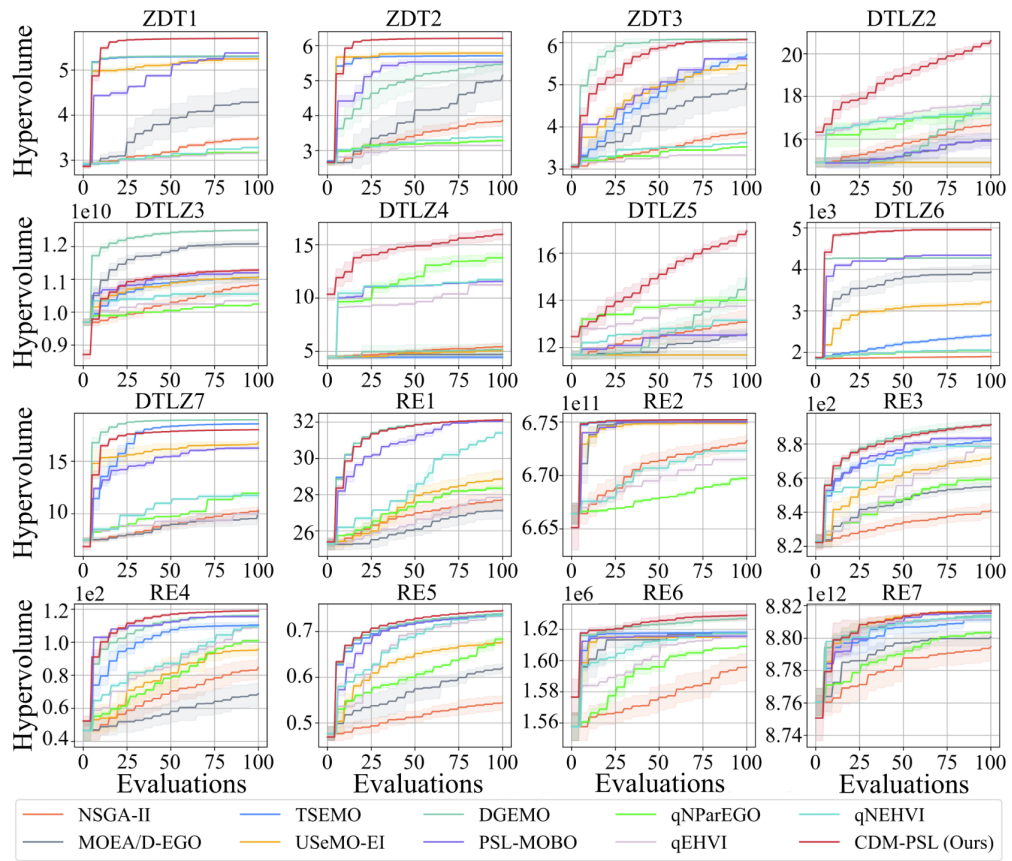


Figure 2: The HV results of 10 algorithms on synthetic test functions and real-world problems ($d = 20$). The horizontal axis denotes the FEs after the initialization phase, similarly hereinafter.

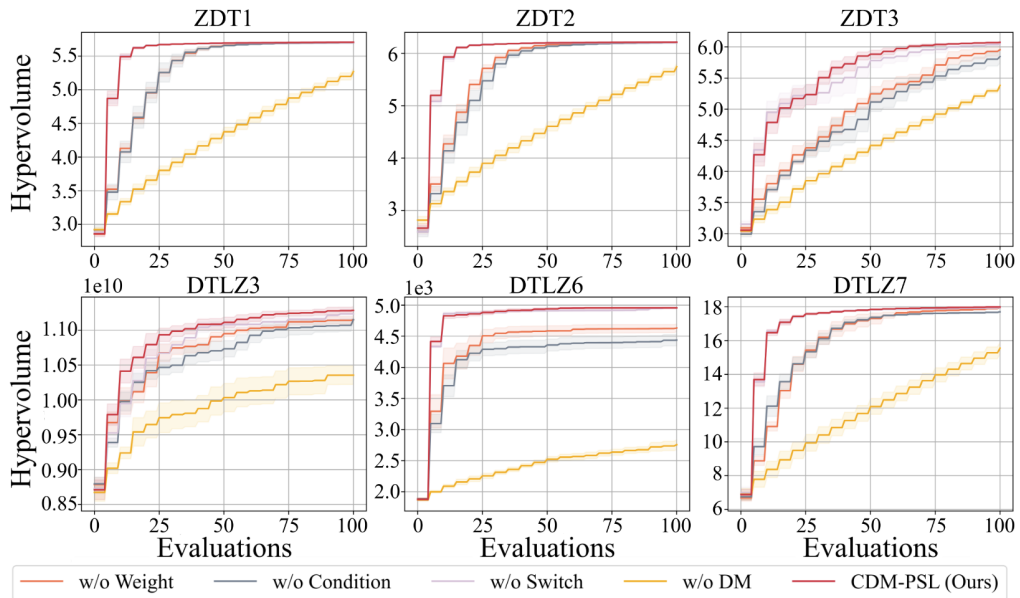


Figure 3: Ablation study results for different components.

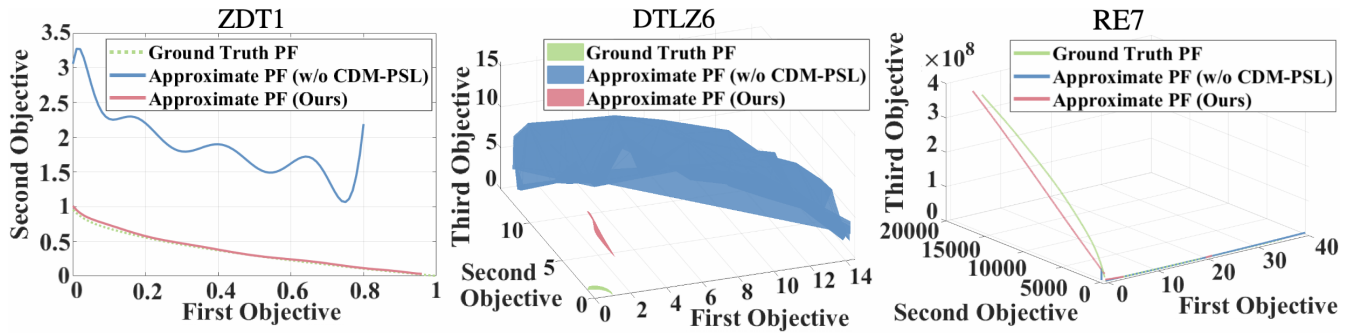


Figure 4: Approximate Pareto fronts obtained by CDM-PSL and MOBO w/o CDM-PSL.

Evaluation Metrics: The hypervolume (HV) in Equation 11, was employed to assess the quality of the solutions obtained. Higher HV values indicate better performance.

Experimental Results

In this section, we present the overall performance of CDM-PSL and the baselines, along with the results of ablation studies. Additionally, we provide a detailed analysis of the superiority of using DM and entropy weighting method in Appendix C.1 and C.2. Further details, including log HV difference, numerical results, running time, parameter sensitivity analysis and additional experiments and discussions are shown in Appendix C.3 to C.12.

Overall Performance. We conducted a series of experiments on a variety of widely recognized synthetic multi-objective benchmarks, including ZDT1-3 (Zitzler, Deb, and Thiele 2000) and DTLZ2-7 (Deb et al. 2005). The problems selected for the experiments featured 2 and 3 objectives, with the number of decision variables set at 20. We particularly highlight the results for a specific instance where $d = 20$, and further details are available in the supplementary materials. Additionally, the overall performance of CDM-PSL was fully assessed using 7 real-world problems.

Figure 2 shows a comparison of HV indicator relative to function evaluations (FE). CDM-PSL demonstrates outstanding performance across most synthetic benchmarks, excelling in both convergence speed and final values. Additionally, CDM-PSL exhibits ideal performance in real-world problems. These findings affirm the effectiveness and superiority of the proposed CDM-PSL.

Ablation Study. As shown in Figure 3, CDM-PSL w/o Weight uses mean weighted gradients to guide the sampling process instead of entropy weighted gradients (EWG). CDM-PSL demonstrates superior convergence performance compared to CDM-PSL w/o Weight on all the six tested problems and attains better final values in ZDT3, DTLZ3 and DLTZ6. This indicates that EWG can gauge the significance of objectives more accurately, thereby offering more effective guidance for the sampling process.

CDM-PSL w/o Condition refers to the variant which omits the conditional generation component. Similarly, CDM-PSL consistently and significantly outperforms

CDM-PSL w/o Condition across all the tested problems. This clearly validates the critical role and effectiveness of the conditional generation component in the CDM-PSL.

CDM-PSL w/o Switch denotes a CDM-PSL variant without the switching strategy. The results of the ablation experiments on ZDT3 and DTLZ3 demonstrate that the inclusion of this strategy is favor of enhancing convergence speed.

CDM-PSL w/o DM refers to the variant that employs GA instead of the DM-based PSL model as its operator, utilizing Simulated Binary Crossover (SBX) to generate new solutions. This variant’s performance is markedly inferior compared to that of the default CDM-PSL. This disparity underscores the effectiveness of DM based Pareto set learning.

Figure 4 displays the solutions set obtained by CDM-PSL and by MOBO without CDM-PSL, based on the posterior mean. Clearly, CDM-PSL surpass MOBO without CDM-PSL on both synthetic benchmarks and real-world problems. For instance, MOBO without CDM-PSL is difficult to approach the true PF with given FE budget, yet CDM-PSL can effectively capture nearly all characteristics of the PF on ZDT1. On 3-objective DLTZ6, CDM-PSL can approximate the true PF faster in a limited number of FEs. Additionally, our methodology demonstrates commendable exploitation capabilities on complex problems, such as the rocket injector design (RE7) (Vaidyanathan et al. 2003), which is characterized by a complex PF where the Pareto optimal solutions being distributed across multiple regions. Please see Appendix C.9 for more PFs obtained by CDM-PSL.

Conclusion and Future work

In this paper, we introduce a composite diffusion model based Pareto set learning method, named CDM-PSL, for addressing EMOPs. CDM-PSL uses both unconditional and conditional diffusion model for generating high-quality samples. Besides, the quality of the generated solutions is significantly enhanced by employing entropy weighted gradients to guide the sampling process. Extensive experimental evaluations on 9 benchmark problems and 7 real-world problems verify the efficiency of CDM-PSL. For future research, there is an intention to incorporate Monte Carlo tree or dimensionality reduction approaches (Song et al. 2022; Qian et al. 2025; Qian and Yu 2017) into CDM-PSL to address higher-dimensional problems.

Acknowledgements

This work was supported by the National Natural Science Foundation of China (Grant No. 62173258, T2495254, 62272210), Shanghai Frontiers Science Center of Molecule Intelligent Syntheses, and the Open Research Fund of Key Laboratory of Advanced Theory and Application in Statistics and Data Science –MOE,ECNU.

References

- Austin, J.; Johnson, D. D.; Ho, J.; Tarlow, D.; and Van Den Berg, R. 2021a. Structured denoising diffusion models in discrete state-spaces. *Advances in Neural Information Processing Systems*, 34: 17981–17993.
- Austin, J.; Johnson, D. D.; Ho, J.; Tarlow, D.; and Van Den Berg, R. 2021b. Structured denoising diffusion models in discrete state-spaces. *Advances in Neural Information Processing Systems*, 34: 17981–17993.
- Balandat, M.; Karrer, B.; Jiang, D.; Daulton, S.; Letham, B.; Wilson, A. G.; and Bakshy, E. 2020. BoTorch: A framework for efficient Monte-Carlo Bayesian optimization. *Advances in neural information processing systems*, 33: 21524–21538.
- Bao, F.; Li, C.; Zhu, J.; and Zhang, B. 2022. Analytic-dpm: an analytic estimate of the optimal reverse variance in diffusion probabilistic models. *arXiv preprint arXiv:2201.06503*.
- Baranchuk, D.; Rubachev, I.; Voynov, A.; Khruikov, V.; and Babenko, A. 2021. Label-efficient semantic segmentation with diffusion models. *arXiv preprint arXiv:2112.03126*.
- Belakaria, S.; Deshwal, A.; Jayakodi, N. K.; and Doppa, J. R. 2020. Uncertainty-aware search framework for multi-objective Bayesian optimization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, 10044–10052.
- Bergstra, J.; Bardenet, R.; Bengio, Y.; and Kégl, B. 2011. Algorithms for hyper-parameter optimization. *Advances in neural information processing systems*, 24.
- Bradford, E.; Schweidtmann, A. M.; and Lapkin, A. 2018. Efficient multiobjective optimization employing Gaussian processes, spectral sampling and a genetic algorithm. *Journal of global optimization*, 71(2): 407–438.
- Couckuyt, I.; Deschrijver, D.; and Dhaene, T. 2014. Fast calculation of multiobjective probability of improvement and expected improvement criteria for Pareto optimization. *Journal of Global Optimization*, 60: 575–594.
- 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. *Advances in Neural Information Processing Systems*, 34: 2187–2200.
- Deb, K.; Pratap, A.; Agarwal, S.; and Meyarivan, T. 2002. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE transactions on evolutionary computation*, 6(2): 182–197.
- Deb, K.; Thiele, L.; Laumanns, M.; and Zitzler, E. 2005. *Scalable test problems for evolutionary multiobjective optimization*. Springer.
- Dhariwal, P.; and Nichol, A. 2021. Diffusion models beat gans on image synthesis. *Advances in Neural Information Processing Systems*, 34: 8780–8794.
- Ding, D.; Li, D.; Li, Z.; and Yang, L. 2019. Compact circularly-polarized microstrip antenna for hand-held RFID reader. In *2019 8th Asia-Pacific Conference on Antennas and Propagation (APCAP)*, 181–182. IEEE.
- Fang, W.; Du, W.; He, R.; Tang, Y.; Jin, Y.; and Yen, G. G. 2024. Diffusion Model-Based Multiobjective Optimization for Gasoline Blending Scheduling. *arXiv preprint arXiv:2402.14600*.
- Hernández-Lobato, D.; Hernandez-Lobato, J.; Shah, A.; and Adams, R. 2016. Predictive entropy search for multi-objective bayesian optimization. In *International conference on machine learning*, 1492–1501. PMLR.
- Ho, J.; Chan, W.; Saharia, C.; Whang, J.; Gao, R.; Gritsenko, A.; Kingma, D. P.; Poole, B.; Norouzi, M.; Fleet, D. J.; et al. 2022. Imagen video: High definition video generation with diffusion models. *arXiv preprint arXiv:2210.02303*.
- Ho, J.; Jain, A.; and Abbeel, P. 2020. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33: 6840–6851.
- Hoffman, M. W.; and Ghahramani, Z. 2015. Output-space predictive entropy search for flexible global optimization. In *NIPS workshop on Bayesian Optimization*, 1–5.
- Huang, X.; Song, L.; Xue, K.; and Qian, C. 2024. Stochastic Bayesian optimization with unknown continuous context distribution via kernel density estimation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 12635–12643.
- 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.
- Krishnamoorthy, S.; Mashkaria, S. M.; and Grover, A. 2023. Diffusion models for black-box optimization. In *International Conference on Machine Learning*, 17842–17857. PMLR.
- Laumanns, M.; and Ocenasek, J. 2002. Bayesian optimization algorithms for multi-objective optimization. In *Parallel Problem Solving from Nature—PPSN VII: 7th International Conference Granada, Spain, September 7–11, 2002 Proceedings* 7, 298–307. Springer.
- Leng, Y.; Chen, Z.; Guo, J.; Liu, H.; Chen, J.; Tan, X.; Mandic, D.; He, L.; Li, X.; Qin, T.; et al. 2022. Binauralgrad: A two-stage conditional diffusion probabilistic model for binaural audio synthesis. *Advances in Neural Information Processing Systems*, 35: 23689–23700.

- Letham, B.; Karrer, B.; Ottoni, G.; and Bakshy, E. 2019. Constrained Bayesian optimization with noisy experiments.
- Li, B.; Di, Z.; Yang, Y.; Qian, H.; Yang, P.; Hao, H.; Tang, K.; and Zhou, A. 2024a. It's Morphing Time: Unleashing the Potential of Multiple LLMs via Multi-objective Optimization. *arXiv preprint arXiv:2407.00487*.
- Li, B.; Yang, Y.; Hong, W.; Yang, P.; and Zhou, A. 2024b. Hyperbolic Neural Network Based Preselection for Expensive Multi-Objective Optimization. *IEEE Transactions on Evolutionary Computation*, 1–1.
- Li, M.; Yang, S.; and Liu, X. 2013. Shift-based density estimation for Pareto-based algorithms in many-objective optimization. *IEEE Transactions on Evolutionary Computation*, 18(3): 348–365.
- Lin, X.; Yang, Z.; Zhang, X.; and Zhang, Q. 2022. Pareto Set Learning for Expensive Multi-Objective Optimization. In *36th Conference on Neural Information Processing Systems (NeurIPS 2022)*.
- Liu, J.; Li, C.; Ren, Y.; Chen, F.; and Zhao, Z. 2022. Diff-singer: Singing voice synthesis via shallow diffusion mechanism. In *Proceedings of the AAAI conference on artificial intelligence*, volume 36, 11020–11028.
- Lugmayr, A.; Danelljan, M.; Romero, A.; Yu, F.; Timofte, R.; and Van Gool, L. 2022. Repaint: Inpainting using denoising diffusion probabilistic models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 11461–11471.
- Moćkus, J. 1975. On Bayesian methods for seeking the extremum. In *Optimization Techniques IFIP Technical Conference: Novosibirsk, July 1–7, 1974*, 400–404. Springer.
- Qian, C.; Xiong, H.; and Xue, K. 2020. Bayesian Optimization using Pseudo-Points. In Bessiere, C., ed., *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI-20*, 3044–3050. International Joint Conferences on Artificial Intelligence Organization. Main track.
- Qian, H.; Wu, Y.-P.; Qin, R.-J.; An, X.; Chen, Y.; and Zhou, A. 2025. Provable space discretization based evolutionary search for scalable multi-objective security games. *Swarm and Evolutionary Computation*, 92: 101770.
- Qian, H.; and Yu, Y. 2017. Solving high-dimensional multi-objective optimization problems with low effective dimensions. In *Proceedings of the AAAI conference on artificial intelligence*, volume 31.
- Shannon, C. E. 1948. A mathematical theory of communication. *The Bell system technical journal*, 27(3): 379–423.
- Snoek, J.; Larochelle, H.; and Adams, R. P. 2012. Practical bayesian optimization of machine learning algorithms. *Advances in neural information processing systems*, 25.
- Sohl-Dickstein, J.; Weiss, E.; Maheswaranathan, N.; and Ganguli, S. 2015. Deep unsupervised learning using nonequilibrium thermodynamics. In *International conference on machine learning*, 2256–2265. PMLR.
- Song, L.; Xue, K.; Huang, X.; and Qian, C. 2022. Monte carlo tree search based variable selection for high dimensional bayesian optimization. *Advances in Neural Information Processing Systems*, 35: 28488–28501.
- Tanabe, R.; and Ishibuchi, H. 2020. An easy-to-use real-world multi-objective optimization problem suite. *Applied Soft Computing*, 89: 106078.
- Vaidyanathan, R.; Tucker, K.; Papila, N.; and Shyy, W. 2003. Cfd-based design optimization for single element rocket injector. In *41st Aerospace sciences meeting and exhibit*, 296.
- Wang, S.; Xue, K.; Song, L.; Huang, X.; and Qian, C. 2024. Monte Carlo Tree Search based Space Transfer for Black-box Optimization. *arXiv preprint arXiv:2412.07186*.
- Yan, X.; and Jin, Y. 2024. EmoDM: A Diffusion Model for Evolutionary Multi-objective Optimization. *arXiv preprint arXiv:2401.15931*.
- Yang, L.; Zhang, Z.; Song, Y.; Hong, S.; Xu, R.; Zhao, Y.; Zhang, W.; Cui, B.; and Yang, M.-H. 2023. Diffusion models: A comprehensive survey of methods and applications. *ACM Computing Surveys*, 56(4): 1–39.
- Yang, P.; Zhang, L.; Liu, H.; and Li, G. 2024. Reducing idleness in financial cloud services via multi-objective evolutionary reinforcement learning based load balancer. *Science China Information Sciences*, 67(2): 120102.
- Yu, P.-L. 1974. Cone convexity, cone extreme points, and nondominated solutions in decision problems with multiobjectives. *Journal of optimization Theory and Applications*, 14: 319–377.
- Yu, S.; Tack, J.; Mo, S.; Kim, H.; Kim, J.; Ha, J.-W.; and Shin, J. 2022. Generating videos with dynamics-aware implicit generative adversarial networks. *arXiv preprint arXiv:2202.10571*.
- Yu, Z.; Ramakrishnan, V.; and Meinzer, C. 2019. Simulation optimization for bayesian multi-arm multi-stage clinical trial with binary endpoints. *Journal of Biopharmaceutical Statistics*, 29(2): 306–317.
- Zhan, T.; Zhou, J.; Li, Z.; and Deng, Y. 2024. Generalized information entropy and generalized information dimension. *Chaos, Solitons & Fractals*, 184: 114976.
- Zhang, Q.; Liu, W.; Tsang, E.; and Virginas, B. 2009. Expensive multiobjective optimization by MOEA/D with Gaussian process model. *IEEE Transactions on Evolutionary Computation*, 14(3): 456–474.
- Zhang, Y.; Sohn, K.; Villegas, R.; Pan, G.; and Lee, H. 2015. Improving Object Detection With Deep Convolutional Networks via Bayesian Optimization and Structured Prediction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
- 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.