Machine Learning (2024) 113:6511–6538
https://doi.org/10.1007/s10994-024-06575-2

PROUD: PaRetO-gUided diffusion model for multi-objective generation

Yinghua Yao[1,2] · Yuangang Pan[1,2] · Jing Li[1,2] · Ivor Tsang[1,2] · Xin Yao[3,4]

Received: 6 December 2023 / Revised: 23 May 2024 / Accepted: 30 May 2024 /
Published online: 2 July 2024

Abstract

Recent advancements in the realm of deep generative models focus on generating samples that satisfy multiple desired properties. However, prevalent approaches optimize these property functions independently, thus omitting the trade-offs among them. In addition, the property optimization is often improperly integrated into the generative models, resulting in an unnecessary compromise on generation quality (i.e., the quality of generated samples). To address these issues, we formulate a constrained optimization problem. It seeks to optimize generation quality while ensuring that generated samples reside at the Pareto front of multiple property objectives. Such a formulation enables the generation of samples that cannot be further improved simultaneously on the conflicting property functions and preserves good quality of generated samples.Building upon this formulation, we introduce the ParetO-gUided Diffusion model (PROUD), wherein the gradients in the denoising process are dynamically adjusted to enhance generation quality while the generated samples adhere to Pareto optimality. Experimental evaluations on image generation and protein generation tasks demonstrate that our PROUD consistently maintains superior generation quality while approaching Pareto optimality across multiple property functions compared to various baselines

Keywords Multi-objective generation · Diffusion model · Pareto optimality · Generative model

1 Introduction

Deep generative models have been developing prosperously over the last decade, with advances in variational autoencoders (Kingma and Welling, 2014), generative adversarial networks (Goodfellow et al., 2014; Zhang et al., 2023), normalizing flows (Papamakarios et al., 2021), energy-based models (Song and Kingma, 2021), and diffusion models (Song and Ermon, 2019; Ho et al., 2020). Particularly, controllable generative models can generate samples that satisfy multiple properties of interest, showing great promise in various applications, such as material design (Jin et al., 2020; Tagasovska et al., 2022) and

Editor: Myra Spiliopoulou.

Extended author information available on the last page of the article

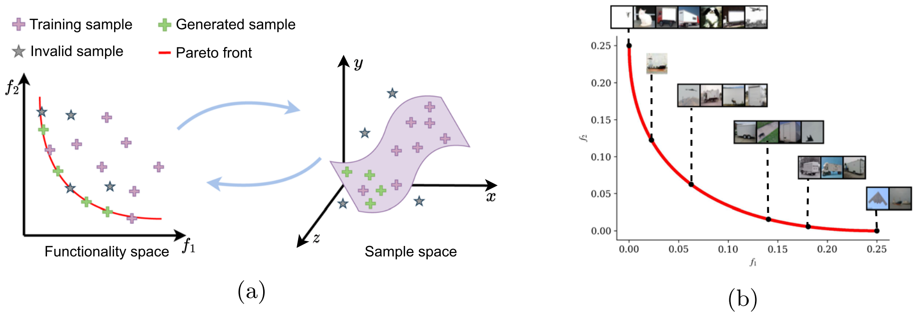


Fig. 1 **a** Diagram of multi-objective generation (best viewed in color). Our multi-objective generation aims to produce samples that simultaneously lie on the Pareto front in the functionality space (Left Panel) and remain within the manifold $\mathcal{X}$ in the sample space (Right Panel), i.e., the green cross. **b** Visualization of the image generation task optimized with two objectives on CIFAR10. Images are directly taken from the original CIFAR10 dataset (see full resolution images in Fig. 12), whose objective values lie on the Pareto front, namely, $\{x|x \in X, F(x) = [f_1^*, f_2^*] \in F^*\}$, where F^* denotes the points on the Pareto front

controlled text/image generation (Dathathri et al., 2020; Liao et al., 2020). These properties of interest vary depending on the specific application domains. For example, in protein design, the properties can refer to specified structural or functional characteristics, such as solubility or binding affinity (Watson et al., 2023). In image generation, the properties can refer to certain attributes or features, such as specified hairstyle & makeup (Wang et al., 2023), or specified color patches (Liu et al., 2021b). In addition, it is considered imperative that generated samples should reside in the same data manifold[1] as training samples for data naturalness concerns (Gruver et al., 2023).

Before delving into details, we first establish the problem setting. Given a dataset $X \subseteq \mathcal{X}$, where $\mathcal{X} \subset \mathbb{R}^d$ denotes a low-dimensional manifold in the high-dimensional space $\mathbb{R}^d$. Suppose we have m objective functions $F(x) = [f_1(x), f_2(x), \dots, f_m(x)]$, each of which returns a property value for the sample $x \in \mathcal{X}$. The aim of multi-objective generation is to learn a generative model that produces samples optimized to achieve the best values across these functions while ensuring the generated samples remain within the manifold $\mathcal{X}$ (green cross in Fig. 1a, namely, ensuring that the quality of generated samples (dubbed as *generation quality*) is good[2].

The multi-objective generation problem introduced above inherently requires reconciling the optimization challenges in two spaces: the functionality space and the sample space as shown in Fig. 1a. Given the need to deal with multiple conflicting objectives in order to achieve the generation with desired properties, one challenge is how to produce samples that cannot be further improved simultaneously across the objectives, a.k.a. *Pareto optimality* (Chinchuluun and Pardalos, 2007) (the Pareto front in Fig. 1a). The second challenge arises from the manifold assumption that the generated samples should lie within the data manifold $\mathcal{X}$, namely, generated samples are supposed to be of good quality (Sanchez-Lengeling and Aspuru-Guzik, 2018). Optimizing multiple objectives without considering generation quality could result in Pareto solutions outside of the data manifold (i.e., invalid samples on

[1] This relates to the manifold hypothesis that many real-world high-dimensional datasets lie on low-dimensional latent manifolds in the high-dimensional space (Fefferman et al., 2016).

[2] In other words, the generated samples is as realistic as samples in the given dataset X.

We have checked all citations and DOIs and ensured that they are existent, true and duplicate-free.

the Pareto front of Fig. 1a). The third challenge relates to the coordination of generation quality and multi-property optimization. To guarantee generation quality, generative models typically define a divergence between the distribution of generated data and that of real training data X (Yang et al., 2023; Goodfellow et al., 2014), which tends to disperse the generated data throughout the whole data manifold $\mathcal{X}$ (the purple plane in Fig. 1a). However, since only a limited fraction of the samples on the data manifold lie on the Pareto front, there inevitably exists some distribution gap between the generated data and the training data, leading to *compromise of generation quality*, when achieving Pareto optimality.

A large number of studies (Klys et al., 2018; Deng et al., 2020; Wang et al., 2024; Li et al., 2022) attempt to design controllable generative models with multiple properties by simply assuming that these properties are independent and aggregating the multiple property objectives into a single one $\sum_{i=1}^{m} f_i$ for controlled generation. Notably, a very recent study (Gruver et al., 2023) takes into consideration the trade-offs between multiple properties by incorporating the multi-objective optimization techniques into the generative models. It modified the gradient of sampling in vanilla diffusion models as a linear combination of the original diffusion gradient and the gradient solved by the multi-objective Bayesian optimization. However, the adopted fixed coefficient is challenging to effectively coordinate the generation quality and the optimization of multiple property objectives. This results in an unnecessary loss of generation quality while achieving Pareto optimality for the property objectives.

In this work, we propose PaRetO-gUided Diffusion model (PROUD) for multi-objective generation. PROUD is formulated as a constrained optimization that minimizes the Kullback–Leibler (KL) divergence between the distribution of the generated data and that of the training data, where the distribution of the generated data is also constrained to be close to the distribution of Pareto solutions under the KL divergence. This guarantees that generated samples are moved towards the Pareto set and then the quality of these generated samples is optimized to the best within a neighborhood of the Pareto set. Specifically, constrained optimization is implemented during the generative process of a pre-trained unconditional diffusion model. Multiple gradient descents for the multiple objectives and the original diffusion gradient are adaptively weighted to denoise samples. The contributions of this work are summarized as follows:

- We propose a novel constrained optimization formulation for controllable generation adhering to multiple properties, defined as multi-objective generation, which can better coordinate the generation quality and the optimization for multi-objectives.
- A new controllable diffusion model (PROUD) is introduced to solve the constrained optimization formulation. The guidance of multiple objectives is adaptively integrated with that of data likelihood, which can reduce the needless comprise of generation quality while achieving Pareto optimality in terms of multiple property objectives.
- We apply our PROUD to optimizing multiple objectives in the tasks of controllable image generation and protein design. Additionally, we establish various baselines based on diffusion models to demonstrate the superiority of our PROUD.

2 Related work

In the section, we summarize the related works based on their strategies for integrating the optimization of multiple property objectives into deep generative models.

Single-objective generation (SOG) refers to approaches that simply combine multiple objectives into a single one to guide the generation. Extensive efforts have been devoted to controllable generation with multiple properties independent of each other (Klys et al., 2018; Guo et al., 2020; Jin et al., 2020; Deng et al., 2020; Wang et al., 2024; Li et al., 2022). Nevertheless, these methods fail to capture the correlation between properties and ignore the conflicting nature among properties, leading to an insufficient exploration of the solution space.

Multi-objective Generation (MOG) refers to approaches that introduce multi-objective optimization techniques into generative models. Wang et al. (2022) adopted a weighted-sum strategy to deal with the trade-offs between properties, which can only work in cases of convex Pareto fronts and a uniformly distributed grid of weighting cannot guarantee uniform points on the Pareto front (Sener and Koltun, 2018; Liu et al., 2021a). Stanton et al. (2022) proposed LaMBO (Latent Multi-objective Bayesian Optimization), which applies multi-objective Bayesian optimization in the latent space of denoising autoencoder to optimize the generated samples with multiple black-box objectives. Although it can characterize the Pareto front, the data generated by denoising autoencoder is of inferior quality. Gruver et al. (2023) further applied LaMBO to the latent space of discrete diffusion models. It generalized classifier-guided diffusion models (Dhariwal and Nichol, 2021) by replacing the classifier gradient with the gradient obtained by LaMBO. The combination of the score estimate of a diffusion model and the classifier gradient necessitates manual tuning of the combination coefficient, which is theoretically inappropriate for non-convex functions (Gong et al., 2021). Tagasovska et al. (2022) proposed to use multiple gradient descent (Désidéri, 2012) for sampling within compositional energy-based models (EBMs) where each EBM is conditioned on one specific property, but training multiple conditional EBMs requires much more supervision than training discriminative models. Moreover, this kind of paradigm cannot enjoy post-hoc controls upon the pre-trained unconditional generative models. Multi-objective generative flow networks (GFlowNets) (Jain et al., 2023) fully integrated guidance from multiple objectives into the training process. So, they must be retrained whenever the objectives change and are also not suitable for use with pre-trained generative models. In addition, this kind of models are usually difficult to train (Shen et al., 2023).

Diffusion models (Ho et al., 2020; Sohl-Dickstein et al., 2015; Song and Ermon, 2019; Song et al., 2021b) represent the state-of-the-art (SOTA) in deep generative models. Therefore, we build our multiple-objective generation model based on diffusion models. While most related works design their methods based on other deep generative models, we apply their ideas to the diffusion model as much as possible for the sake of comparison. Please refer to Sect. 5 for more details.

3 Preliminaries

Before delving into our method, we introduce the technical background about multi-objective optimization in Sect. 3.1 and diffusion models in Sect. 3.2, respectively.

3.1 Multi-objective optimization

Let $x \in \mathbb{R}^d$ be a decision variable. Assuming that $F(x) = \left[f_1(x), f_2(x), \dots, f_m(x)\right]$ be a set of m objective functions, each of which represents a property and is preferred to have a smaller value. The multi-objective optimization problem (Chinchuluun and Pardalos, 2007; Deb, 2001) can be conventionally expressed as:

$$\min_{x\in\mathbb{R}^d} F(x) = \min_{x\in\mathbb{R}^d} \left[f_1(x), f_2(x), \dots, f_m(x)\right]. \tag{1}$$

In this context, for $x_1, x_2 \in \mathbb{R}^d$, x_1 is said to dominate x_2, i.e., $x_1 \prec x_2$, iff $f_i(x_1) \leq f_i(x_2), \forall i = 1, 2, \dots, m$, and $F(x_1) \neq F(x_2)$.

Definition 1 (*Pareto optimality*) A point $x^* \in \mathbb{R}^d$ is called Pareto optimal iff there exists no any other $x' \in \mathbb{R}^d$ such that $x' \prec x^*$. The collection of Pareto optimal points are called *Pareto set*, denoted as $\mathcal{P}^*$. The collection of function values $F(x^*)$ of the Pareto set is called the *Pareto front* (Van Veldhuizen and Lamont, 1998; Borghi et al., 2023).

Definition 2 (*Pareto stationarity*) Pareto stationarity is a necessary condition for Pareto optimality. A point x is called Pareto stationary if there exists a set of scalar $\omega_i, i = 1, 2, \dots, m$, such that $\sum_{i=1}^m \omega_i \nabla f_i(x) = \mathbf{0}$, $\sum_{i=1}^m \omega_i = 1, \omega_i > 0, \forall i = 1, 2 \dots, m$.

Désidéri (2012) proposed Multiple Gradient Descent (MGD) to find the Pareto optimal solutions of Eq.(1). To be specific, given any initial point $x \in \mathbb{R}^d$, we can iteratively update x according to:

$$x_{t+1} = x_t - \eta v_t, \tag{2}$$

where t is the iteration step. The update direction v_t is expected to be close to each gradient $\nabla f_i(x)\, \forall i = 1, 2, \dots, m$ as much as possible, which is therefore formulated into the following problem:

$$\max_{v\in\mathbb{R}^d} \left\{ \min_i \nabla f_i(x)^\top v - \frac{1}{2}\|v\|^2 \right\}. \tag{3}$$

Through Lagrange strong duality, the solution to Eq.(3) can be framed into

$$v(x) = \nabla F(x) = \sum_{i=1}^m \omega_i^* \nabla f_i(x), \tag{4}$$

where $\{\omega_i^*\}_{i=1}^m = \arg\min_{\{\omega_i\}_{i=1}^m} \|\sum_{i=1}^m \omega_i \nabla f_i(x)\|^2$ under the constraint that $\sum_{i=1}^m \omega_i = 1, \omega_i > 0, \forall i = 1, 2 \dots, m$.

3.2 Diffusion models

The idea of Diffusion models is to progressively diffuse data to noise, and then learn to reverse this process for sample generation. Considering a sequence of prescribed noise scales $0 < \beta_1 < \beta_2 < \dots < \beta_T < 1$, Denoising Diffusion Probabilistic Model (DDPM) (Ho et al., 2020) diffuses data $x_0 \sim q_{\text{data}}(x)$ to noise via constructing a discrete Markov chain $\{x_0, x_1, \dots, x_T\}$, where $q(x_t|x_{t-1}) = \mathcal{N}(x_t; \sqrt{1-\beta_t}x_{t-1}, \beta_t\mathbf{I}), x_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. This process is called the forwarded process or diffusion process. In particular, $q(x_t|x_0) = \mathcal{N}(x_t; \sqrt{\alpha_t}x_0, (1-\alpha_t)\mathbf{I})$, where $\alpha_t = \prod_{i=1}^t \left(1-\beta_t\right)$.

The key of diffusion-based generative models is to train a reverse Markov chain so that we can generate data starting from a Gaussian noise $p(x_T) \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. The training loss of the reverse diffusion process, a.k.a. generative process, is to minimize a simplified variational bound of negative log likelihood. Namely,

$$\mathbb{E}_{x_0 \sim q_{\text{data}}(x), \epsilon \sim \mathcal{N}(\mathbf{0},\mathbf{I})}\left[\|\epsilon - \epsilon_\theta\left(\sqrt{\alpha_t}x_0 + \sqrt{1-\alpha_t}\epsilon, t\right)\|^2 \right], \tag{5}$$

where $\epsilon_\theta(x_t, t)$ is a neural network-based approximator to predict the noise ϵ from $x_t = \sqrt{\alpha_t}x_0 + \sqrt{1-\alpha_t}\epsilon$.

After training the neural network parameterized by θ to obtain the optimal $\epsilon^*_\theta(x_t, t)$, samples can be generated by starting from $x_T \sim \mathcal{N}(\mathbf{0},\mathbf{I})$ and reversing the Markov chain:

$$x_{t-1} = \frac{1}{\sqrt{1-\beta_t}}\left(x_t - \frac{\beta_t}{\sqrt{1-\alpha_t}}\epsilon^*_\theta\left(x_t, t\right) \right) + \sqrt{\beta_t}z_t, \tag{6}$$

where $z_t \sim \mathcal{N}(\mathbf{0},\mathbf{I})$ and $t = T, T-1, \dots, 1$. More variants of diffusion models can be seen in Yang et al. (2023).

Existing attempts for incorporating multiple desired properties into the diffusion model (Gruver et al., 2023) can be straightforwardly adding the derived MGD $\nabla F(x)$ in Eq.(4) to the noise predictor $\epsilon^*_\theta(x_t, t)$ at each denoising step, namely,

$$x_{t-1} = \frac{1}{\sqrt{1-\beta_t}}\left(x_t - \frac{\beta_t}{\sqrt{1-\alpha_t}}\left(\epsilon^*_\theta\left(x_t, t\right) + \lambda\nabla F(x)\right) \right) + \sqrt{\beta_t}z_t, \tag{7}$$

where $t = T, T-1, \dots, 1$. λ is a trade-off hyper-parameter which balances the generation quality (i.e., the noise predictor $\epsilon^*_\theta(x_t, t)$) and multiple-objectives (i.e., the MGD $\nabla F(x)$). Note that an inappropriate λ may lead to unsatisfied samples which either suffer from low quality or fail to possess required properties (Refer to experimental observations in Sect. 5).

4 Multi-objective generation

As discussed above, optimizing generative models in terms of m objectives aims to produce samples that cannot be simultaneously improved for all objectives, namely, *Pareto optimality* (see Definition 1). Meanwhile, the generated samples are required to be as realistic as the training samples, which is usually achieved by enforcing distribution alignment between the generated samples and the training samples.

4.1 MOG compared with MOO

As shown in Table 1, both the MOO and MOG share the same objectives $F(x)$ but differ in the space that x resides in, which is termed as "decision space" or "solution space" in the MOO problem (Chinchuluun and Pardalos, 2007) and is termed as "data space" in the MOG problem (Gruver et al., 2023; Wang et al., 2024). To be specific, the decision space of the MOO problem is defined as the whole space of $\mathbb{R}^d$ (Cheng et al., 2017), while the data space of the MOG problem only resides in a low-dimensional manifold $\mathcal{X}$ embedded in $\mathbb{R}^d$ (a.k.a. the ambient space) (Fefferman et al., 2016; Roweis and Saul, 2000; McInnes et al., 2018). Such a difference highlights that the objectives

Table 1 The MOO problem versus the MOG problem

	Objectives	Decision/data space	Generation quality
MOO	$F(x) = [f_1(x), f_2(x), \dots, f_m(x)]$	$x \in \mathbb{R}^d$	×
MOG		$x \in \mathcal{X}, \mathcal{X} \subset \mathbb{R}^d$	✓

The generation quality in MOG is usually modeled based on the given dataset $X \subset \mathcal{X}$, where $\mathcal{X}$ denotes a low-dimensional manifold embedded in the high dimensional space $\mathbb{R}^d$

to be optimized for MOG are only meaningful within the data manifold. When simply applying MOO algorithms to search for solutions in the high-dimensional sample space, the obtained solutions cannot guarantee residing within the data manifold, thus resulting in very low data quality (i.e., invalid samples in Fig. 1a) and a loss of practicability (Sanchez-Lengeling and Aspuru-Guzik, 2018).

To sum up, the necessity to concurrently consider generation quality distinguishes the MOG problem from the MOO problem. Specifically, a dataset with real samples is required to define the data manifold on which the generated samples are expected to reside (Eq.(8)).

4.2 Constrained optimization for MOG

A straightforward solution of MOG is to take consideration of generation quality as an additional objective and formulate it into a $m + 1$ objectives problem. However, the heterogeneity of multiple objective optimization (usually defined w.r.t. a single sample) and the distribution alignment (defined w.r.t. a dataset) would bring out the optimization difficulty for the resultant MOO. Although it is feasible to simplify the distribution divergence w.r.t. a dataset as quality scores for individual samples in some deep generative models (Arjovsky et al., 2017), it is still challenging to obtain desired solutions that achieve Pareto optimality on m objectives from the optimization of $m + 1$ objectives which explore a much larger space, as empirically verified in the experiments. In addition, the complexity of multi-objective optimization increases significantly with the number of objectives (Ishibuchi et al., 2008).

Instead of formulating a complex and ineffective $m + 1$ objective problem, we implement the multi-objective generation through a tailor-designed constrained optimization problem upon m property objectives. Such a formulation also allows us to stress respective significance of data generation and m-objective optimization, instead of treating them equally important. Specifically, let $p_\theta(x)$ denote the target data distribution parameterized by θ, and p_0 denote the distribution of the solution samples on the Pareto front, our constrained optimization problem can be formulated as follows

$$\min_{\theta} D\big[q_{\text{data}}(x)||p_\theta(x)\big] \quad s.t.\ D\big[p_0(x)||p_\theta(x)\big] \le \varepsilon. \tag{8}$$

where $D(\cdot,\cdot)$ denotes the distribution divergence and ε is a small positive value.

The loss in Eq.(8) controls the generation quality, which ensures the quality of the generated data as realistic as possible. The constraint in Eq.(8) ensures the generated data $x \sim p_\theta(x)$ to be Pareto optimal (with a small bearable error). Overall, Eq.(8) provides certain quality assurance while obtaining samples that can approach Pareto optimality of multiple property objectives.

4.3 Langevin dynamics for data distribution approximation

It is difficult to directly solve Eq.(8) when both $q_{\text{data}}(x)$ and $p_0(x)$ are unknown. Motivated by those widely-developed techniques of sampling algorithms for approximating data distribution (Andrieu et al., 2003; Song and Ermon, 2019; Liu et al., 2021a), we develop Langevin dynamic-based sampling techniques to solve Eq.(8). Specifically, Langevin dynamics are capable of generating samples from a given probability distribution $q(x)$ solely by utilizing its score function $\nabla \log q(x)$. Given an initial value $x_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, the Langevin method recursively computes the following:

$$x_{t-1} = x_t - \kappa g(x_t) + \sqrt{2\kappa} z, \quad t = T, T-1, \dots, 0, \tag{9}$$

where κ is the step size and can be fixed or dynamic, z is sampled from the standard normal distribution $\mathcal{N}(\mathbf{0}, \mathbf{I})$ and $g(x_t)$ is the update direction for x_t, equal to $\nabla \log q(x_t)$. The distribution of x_0 will be close to the given data distribution $q(x)$ when $\kappa \to 0$ and $T \to \infty$ under some regularity conditions (Welling and Teh, 2011).

Before deriving the proper gradient $g(x_t)$ to approximate the distribution optimized in Eq.(8) as a whole, we investigate the gradient-based strategies to optimize $D\left[q_{\text{data}}(x) || p_\theta(x)\right]$ and $D\left[p_0(x) || p_\theta(x)\right]$ via Langevin dynamics, separately.

Optimization of $D\left[q_{data}(x) || p_\theta(x)\right]$ *in Eq.*(8). Actually, various generative models are deduced to approximate the minimization of the KL divergence between the data distribution $q_{\text{data}}(x)$ and the model distribution $p_\theta(x)$ (Kingma and Welling, 2014; Song et al., 2021a; Papamakarios et al., 2021). Here, we choose diffusion models as the representative for optimizing $D\left[q_{\text{data}}(x) || p_\theta(x)\right]$ given their equivalent form to Eq. (9) (Ho et al., 2020; Song et al., 2021b). Particularly, the time-dependent predicted noise $\epsilon_\theta^*\left(x_t, t\right)$ in Eq. (6) is the update direction $g(x_t)$ in anneal Langevin dynamics with a dynamic step size η_t:

$$x_{t-1} = x_t - \eta_t \epsilon_\theta^*\left(x_t, t\right) + \sqrt{2\eta_t} z. \tag{10}$$

Consequently, the distribution of $p_\theta(x_0)$ will approach $q_{\text{data}}(x)$ (Song et al., 2021a).

Optimization of $D\left[p_0(x) || p_\theta(x)\right]$ *in Eq.*(8). On the other hand, we can integrate MGD (Eq. (4)) into Langevin dynamics to optimize $D\left[p_0(x) || p_\theta(x)\right]$, aiming to approximate the distribution of the Pareto set $p_0(x)$ upon convergence. Namely,

$$x_{t-1} = x_t - \eta \nabla F(x_t) + \sqrt{2\eta} z, \tag{11}$$

where η is a fixed step size. The distribution of x_0 will converge to $p_0(x)$, as demonstrated in Theorem 3.3 of Liu et al. (2021a).

4.4 Pareto-guided diffusion model

Based on the above analysis, the key to solving the constrained optimization problem (Eq. (8)) is to design a proper strategy for unifying the optimization of $D\left[q_{\text{data}}(x) || p_\theta(x)\right]$ and $D\left[p_0(x) || p_\theta(x)\right]$ within the framework of Langevin dynamic sampling. Therefore, we can indirectly solve Eq. (8) by designing the following strategies to update the gradient $g(x_t)$ in Eq. (9):

1. If the sample x_t is far away from the Pareto front (*constraint violation*), $g(x_t)$ is chosen to assure Pareto improvement (i.e., decreasing all the m objectives) to x_t. The amount of Pareto improvement is determinant by the distance of x_t to the Pareto front.
2. If there are multiple directions that can yield Pareto improvement (*constraint violation*), the direction of Pareto improvement that decreases $D\left[q_{\text{data}}(x)||p_\theta(x)\right]$ most (*reducing loss*) is chosen as $g(x_t)$.
3. If x_t is close to the Pareto front (*constraint satisfaction*), i.e., having a small $\|\nabla F\left(x_t\right)\|$ according to Definition 2, $g(x_t)$ is chosen to fully optimize $D\left[q_{\text{data}}(x)||p_\theta(x)\right]$ (*reducing loss*).

Following Ye and Liu (2022), we design a new objective based on the gradients to achieve the above conditions. To be specific, since $\epsilon^*_\theta\left(x_t, t\right)$ is the gradient for optimizing $D\left[q_{\text{data}}(x)||p_\theta(x)\right]$, and $\nabla F(x)$ is the gradient for optimizing $D\left[p_0(x)||p_\theta(x)\right]$, the integrated gradient $g(x_t)$ can be solved by the following objective:

$$\begin{aligned} & g(x_t) = \arg\min_g \frac{1}{2}\|g - \epsilon^*_\theta\left(x_t, t\right)\|^2 \\ & s.t. \quad \nabla f_i(x)^T g \geq \phi_t, \quad \forall i = 1, 2, \dots, m, \\ & \qquad \phi_t = \begin{cases} \alpha\|\nabla F\left(x_t\right)\| & \text{if } \|\nabla F\left(x_t\right)\| > e \\ -\infty & \text{otherwise} \end{cases}, \end{aligned} \tag{12}$$

where α and e are positive hyper-parameters. The constraint in Eq.(8) can be approximated by the small gradient norm $\nabla F(x)$ due to Pareto stationarity (Definition 2). In particular, when $\|\nabla F\left(x_t\right)\| > e$, ϕ_t is set to be proportionate to $\|\nabla F\left(x_t\right)\|$. This will encourage the gradient $g(x_t)$ to have positive inner products with all $\nabla f_i(x)$, approximating $\nabla F(x)$. Meanwhile, the amount of Pareto improvement is based on the distance of x_t to the Pareto front. If $\|\nabla F\left(x_t\right)\|$ has a very small norm, which means that the sample x_t is close to the Pareto front, we will have $g_t(x) = \epsilon^*_\theta\left(x_t, t\right)$ with $\phi_t = -\infty$. Therefore, samples will be updated with a pure gradient descent on $D\left[q_{\text{data}}(x)||p_\theta(x)\right]$ without taking into account the m objectives $\{f_i(x)\}_{i=1}^m$, namely, $\lambda_{i,t} = 0, \forall i \in [m]$.

At the situation of $\|\nabla F\left(x_t\right)\| > e$, the solution $g(x_t)$ of Eq.(12) is expressed as:

$$g(x_t) = \epsilon^*_\theta\left(x_t, t\right) + \sum_{i=1}^m \lambda_{i,t}\nabla f_i(x_t), \tag{13}$$

where $\{\lambda_{i,t}\}_{i=1}^m$ is the solution of the following dual problem:

$$\max_{\lambda_{i,t}\in\mathbb{R}^m_+} -\frac{1}{2}\|\epsilon^*_\theta\left(x_t, t\right) + \sum_{i=1}^m \lambda_{i,t}\nabla f_i(x_t)\|^2 + \sum_{i=1}^m \lambda_{i,t}\phi_t. \tag{14}$$

Substituting the derived gradient $g(x_t)$ (Eq.(13)) into Eq.(9) and adopting a dynamic step size η_t, we can obtain a new kind of controllable diffusion modeling, which is named as PaRetO-gUided Diffusion model (PROUD):

$$x_{t-1} = x_t - \eta_t\left(\epsilon_\theta^*\left(x_t, t\right) + \sum_{i=1}^{m} \lambda_{i,t}\nabla f_i(x_t)\right) + \sqrt{2\eta_t}z. \tag{15}$$

PROUD does not modify the training process of diffusion models but only updates gradients during the generative process, as summarized in Algorithm 1. Therefore, our PROUD can be plugged into any pre-trained diffusion model to gain post-hoc control during the generative process.

In contrast to existing methods that crudely combine generative models with multi-objective optimization techniques using a predefined balance coefficient, our constrained optimization formulation (Eq.(8)) allows to dynamically infer the balance coefficient (Eq. (14)), prioritizing the guarantee of Pareto optimality.

Algorithm 1 Pareto-guided Reverse Diffusion Process for a Single Sample

```
Input: a pre-trained unconditional diffusion model $\epsilon_\theta^*$, the dynamic step
size $\{\eta_t\}_{t=1}^{T}$, multiple property objectives $\{f_i\}_{i=1}^{m}$.
Hyper-parameters: $\alpha$ and $e$ in Eq.(12).
Initialize: $x_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$.
for t $= T, T-1, \dots, 0$ do
    calculate the multiple gradient descent: $\nabla F(x_t)$ based on Eq.(4);
    if $\|\nabla F(x_t)\| > e$ then # calculate the weight coefficients
        $\{\lambda_{i,t}\}_{i=1}^{m}$ takes the solution of Eq.(14) with $\phi_t = \alpha\|\nabla F(x)\|$;
    else
        $\lambda_{i,t} = 0, \forall i \in [m]$;
    end if
    calculate the denoising gradient: $g(x_t) = \epsilon_\theta^*(x_t, t) + \sum_{i=1}^{m} \lambda_{i,t}\nabla f_i(x_t)$ as Eq.(13);
    sample $z \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$;
    denoise the sample: $x_{t-1} = x_t - \eta_t g(x_t) + \sqrt{2\eta_t}z$;
end for
Output: the sample $x_0$ which meets Pareto optimality of $m$ objectives.
```

4.5 Diversity regularization for diversified pareto solutions

In practice, MGD integrated with Langevin dynamics fails to obtain diversified Pareto solutions although it can be guaranteed to obtain solutions on the Pareto front (Liu et al., 2021a). To make the solutions be evenly distributed on the Pareto front, we consider adding a diversity regularization, which can be enforced either in the sample space or the functionality space. Because we are interested in high-dimensional data generation, imposing larger distances between samples can be challenging. Furthermore, a significant separation between samples does not necessarily ensure a substantial distinction between their respective functionalities. Therefore, we define the diversity regularization based on the objective values.

Suppose there are N particles $\{x^1, x^2, \dots, x^N\}$ in each step of our PROUD. We omit the subscript t of the time step for simplicity. The diversity loss is defined to encourage the dissimilarity of the objective values:

$$l(x^1, x^2, \dots, x^N) = \sum_{i \neq j} \frac{1}{\|F(x^i) - F(x^j)\|^2}. \tag{16}$$

The diversity loss Eq.(16) is added to the main objective in Eq.(8) with a weight coefficient γ.

5 Experiments

In this section, we evaluate the effectiveness of our PROUD in optimizing image generation and protein generation with multiple conflicting objectives. We study white-box multi-objectives in this work and particularly focus on using MGD as the MOO technique to obtain the gradient from multi-objectives. The exploration of the black-box setting, as mentioned in Stanton et al. (2022), is discussed in the conclusion and remains for future work.

Dataset. In the task of image generation, we use the CIFAR10 (Krizhevsky and Hinton, 2009) dataset, which consists of 60,000 color images, each with a size of $3 \times 32 \times 32$, distributed across 10 classes. Regarding protein generation, following Gruver et al. (2023), the experiment was conducted on the paired Observed Antibody Space (pOAS) dataset (Olsen et al., 2022), which comprises 90, 990 antibody sequences, each processed to a fixed length of 300.

Baselines. First, we include the most closely-related and SOTA work in MOG that applies the MOO technique to the deep generative model (Gruver et al., 2023). This baseline is termed as "DM+m-MGD", where the MGD of m objectives is used to guide the generation of diffusion models (DM). We also include the baseline regarding single-objective generation, termed as "DM+single". It fuses multiple objectives into a single objective and uses the gradient of the obtained single objective to guide the generation of diffusion models. Another considered baseline is "$m+1$-MGD". It treats the objective of the diffusion model as an additional objective and formulates multi-objective generation as the optimization of $m+1$ objectives. MGD is then applied directly for the resultant $m+1$ objectives. To stress the necessity of quality assurance in the generation problem, which is the core difference between MOG and MOO, we include the MGD of m objectives as the baseline, called "m-MGD".

For all methods equipped with MGD, the diversity regularization (Eq.(16)) is included except for $m+1$-MGD since its extra objective $f_{m+1}(x)$, i.e., data likelihood, is not accessible for the diffusion models.

Metrics. In terms of generation quality, the Frechet Inception Distance (FID) (Heusel et al., 2017) is adopted as the metric for image quality, while the log-likelihood assigned by ProtGPT (Ferruz et al., 2022) is considered as the metric for the quality of protein sequences following Gruver et al. (2023). Concerning Pareto optimality, Hypervolume (HV) (Zitzler and Thiele, 1999) is adopted to measure how well the methods approximate the Pareto set.

5.1 Image generation

We follow Liu et al. (2021b)[3] to optimize CIFAR10 images with the objectives that force the middle of an image to be a specified color square.

(1) Controllable generation on CIFAR10 with two objectives (Fig. 1b):

- $f_1(x) = \|x_\Omega - 1_\Omega\|_2^2$, where x represents the entire image, and $x_\Omega \subseteq x$ is an image patch in the region Ω, corresponding to the square at the center of the image. Similar to the practical relevance shown in Liu et al. (2021b), this objective is to restrict the center of the generated images to be a white square, which is to sample CIFAR10 images that exhibit white color in their middle. The patch size is set to $3 \times 8 \times 8$ in the experiment.
- $f_2(x) = \|x_\Omega - 0.5_\Omega\|_2^2$ with the similar setting. This objective is to constrain the center to be a grey square.

The desired generation for these two objectives would be those CIFAR10-like images with patches in normalized RGB color values[4] between [0.5, 0.5, 0.5] (grey) and [1, 1, 1] (white), in the middle, according to Ishibuchi et al. (2013); Li et al. (2017). Please refer to "Appendix B" for more details.

(2) Controllable generation on CIFAR10 with three objectives:

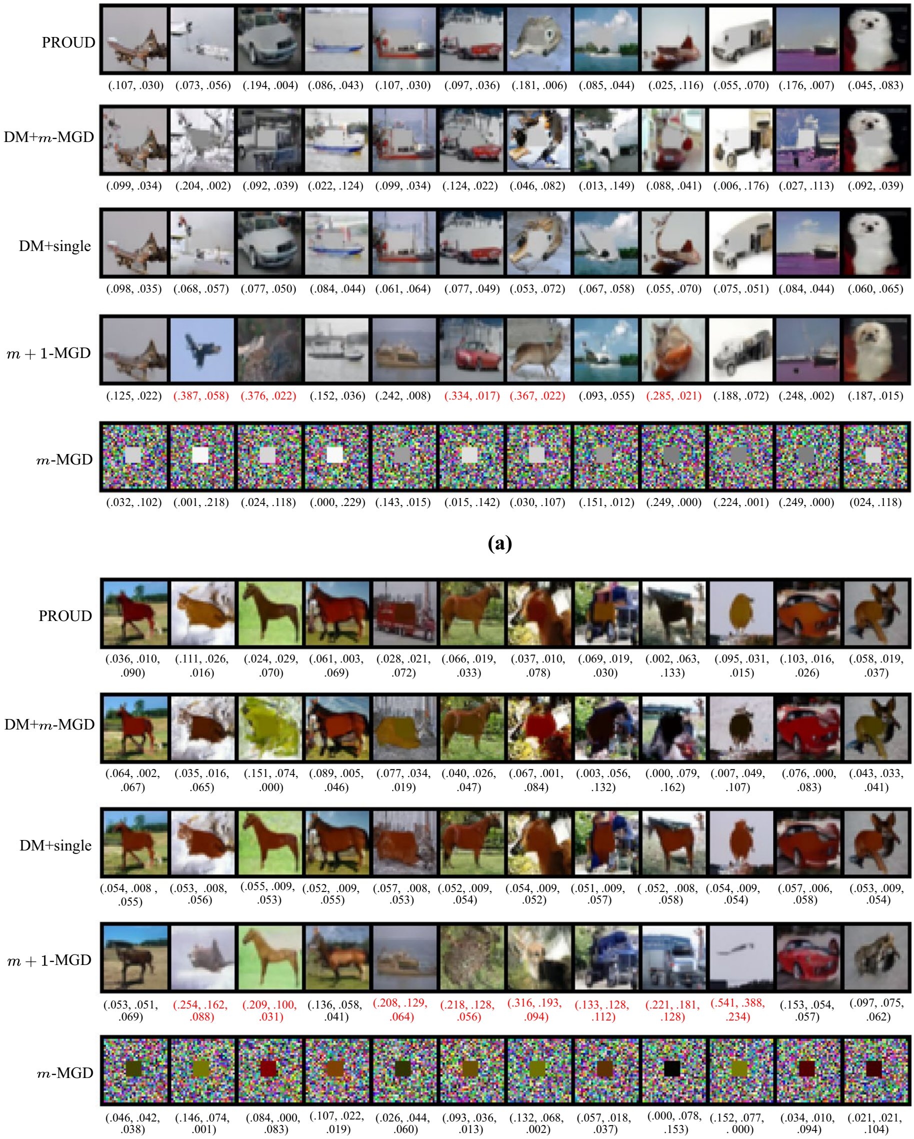
- $f_1(x) = \|x_\Omega - a_\Omega\|_2^2$, where x represents the entire image, and $x_\Omega \subseteq x$ is an image patch in the region Ω, corresponding to the square at the center of the image. This objective is to restrict the center of the generated images to be a black square. The patch size is set to $3 \times 8 \times 8$ in the experiment. $a_\Omega = [0, 0, 0]_{8\times 8}$.
- $f_2(x) = \|x_\Omega - b_\Omega\|_2^2$ with the similar setting. This objective is to constrain the center to be a deep red square. $b_\Omega = [0.5, 0, 0]_{8\times 8}$.
- $f_3(x) = \|x_\Omega - c_\Omega\|_2^2$ with the similar setting. This objective is to constrain the center to be a deep yellow square. $c_\Omega = [0.5, 0.5, 0]_{8\times 8}$.

The desired generation for these three objectives would be those CIFAR10-like images with patches in normalized RGB color values belonging to the convex triangle formed by the points [0, 0, 0] (black), [0.5, 0, 0] (deep red) and [0.5, 0.5, 0] (deep yellow). Please refer to "Appendix B" for more details. We adopt the diffusion model used in Song and Ermon (2020) as the backbone for CIFAR10 image generation.

We sample images from our PROUD and other baselines using the same seeds for the sake of comparison. From Fig. 2, we can observe that: (1) our PROUD and two baselines, DM+m-MGD and m-MGD, can successfully generate harmonious images consistent with the patch-level constraints imposed by two conflicting objectives. Among them, the generated images of our PROUD exhibit better quality than DM+m-MGD in some instances, as the latter tends to sacrifice generation quality to excessively meet Pareto optimality of the

[3] As demonstrated in Sect. 3 and Fig. 3b of their study, an objective that forces the center of generated images to be a black square can be used for constrained sampling on CIFAR10. Accordingly, they obtain samples that lie on the CIFAR10 data manifold and exhibit the black square in the middle, such as "black plane" and "black dog" images which contain a black square (smaller size than the object) in the middle. This task can be considered as image outpainting (Yao et al., 2022), namely, extrapolating images based on specified color patches on CIFAR10.

[4] RGB values [0, 255] are divided by 255.

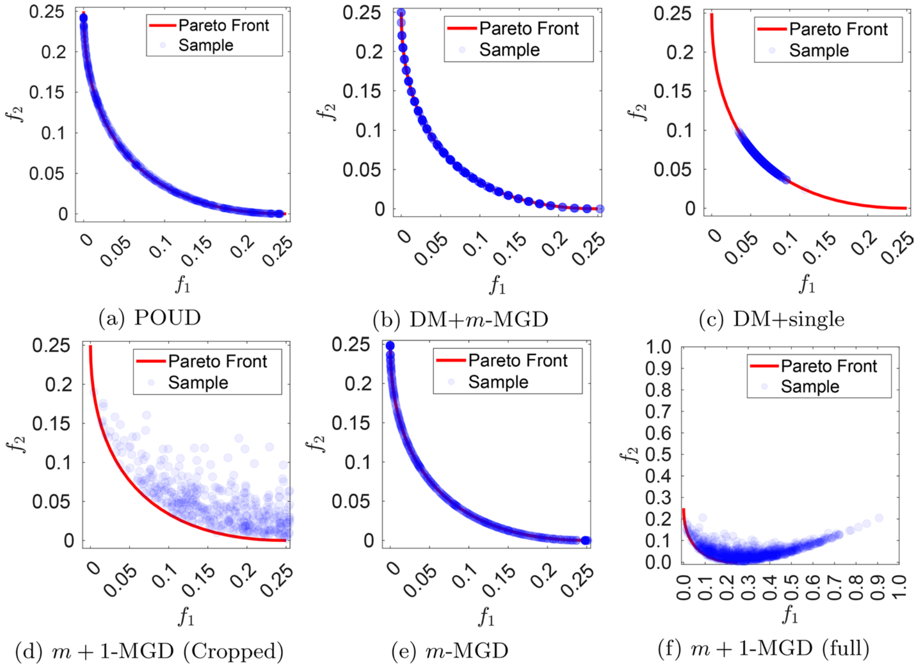


(a)

(b)

Fig. 2 Generated images from our PROUD and various baselines on CIFAR10 under two/three conflicting patch-based objectives. The scores under each image refer to its objective values $[f_1(x), f_2(x)]$ /$[f_1(x), f_2(x), f_3(x)]$, respectively, where those objective values do not reside on the Pareto front are marked in red

objectives due to the lack of a mechanism to emphasize the quality of generated samples. (2) $m+1$-MGD fails to generate satisfactory images consistent with the patch-level constraints, as the new objective (i.e, generation quality) biases the optimization of the original two objectives. Although the Pareto set of the original m-objectives resides within that of



(a) POUD

(b) DM+m-MGD

(c) DM+single

(d) $m+1$-MGD (Cropped)

(e) m-MGD

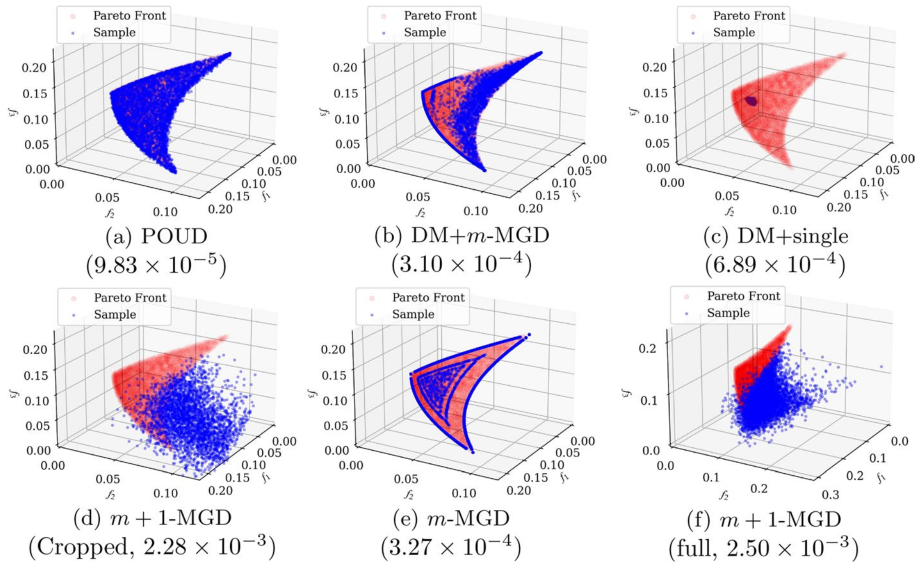
(f) $m+1$-MGD (full)

Fig. 3 Approximation of Pareto front of various methods on CIFAR10 optimized with two objectives. Each point denotes a generated sample, 1000 in total, where the coordinate corresponds to its objective values. The depth of color represents sample density, the deeper the higher

the $m+1$-objectives (Tanabe and Ishibuchi, 2020), the proportion is negligible even when sampling a large number of images. Refer to Figs. 3d, f and 4d, f for more details. (3) m-MGD, which does not consider generative quality in its optimization, generates meaningless images because the optimization of multiple objectives in the data generation task is only meaningful within the data manifold, as image data usually concentrate on low-dimensional manifolds embedded in a high-dimensional space.

For the MOG setting on CIFAR10 optimized with two objectives, we randomly select 1000 generated images for each method, and calculate their objective values $[f_1(x), f_2(x)]$, respectively. Figure 3 shows that: (1) our PROUD (Fig. 3a) and two baselines DM+m-MGD (Fig. 3b) and m-MGD (Fig. 3e) successfully generate samples which can cover the entire Pareto front. Among them, our PROUD and m-MGD spread more evenly over the Pareto front. (2) DM+single only covers a partial Pareto front as shown in Fig. 3c, because simply averaging multiple objectives into a single objective fails to explore the trade-off between multiple objectives and leads to insufficient solutions. (3) As discussed in Fig. 2, $m+1$-MGD explores a much larger solution space (Fig. 3f), while only a few of them are located at the Pareto front of the original m objectives (Fig. 3d).

For the MOG setting on CIFAR10 optimized with three objectives, we randomly select 5000 generated images for each method and calculate their objective values $[f_1(x), f_2(x), f_3(x)]$, respectively. Figure 4 shows that our PROUD exhibits significant superiority in evenly covering the Pareto front under this more challenging setting. This is because our constrained optimization formulation can better coordinate the generation quality and the optimization for multi-objectives, while ensuring sample diversity



(a) POUD (9.83×10^{-5})

(b) DM+m-MGD (3.10×10^{-4})

(c) DM+single (6.89×10^{-4})

(d) $m+1$-MGD (Cropped, 2.28×10^{-3})

(e) m-MGD (3.27×10^{-4})

(f) $m+1$-MGD (full, 2.50×10^{-3})

Fig. 4 Approximation of Pareto front of various methods on CIFAR10 optimized with three objectives. Each point denotes each generated sample, 5000 in total, where the coordinate corresponds to its objective values. The depth of color represents sample density, the deeper the higher. The values in the brackets are earth mover distances between the generated samples and the ground-truth Pareto solutions. We add this measure to indicate that our generated samples are indeed close to the Pareto front given the 3D visualization

(Eq.(8), Eq.(16)). Although it is possible to force the two baselines DM+m-MGD and m-MGD to exhibit better diversity by setting a large diversity coefficient γ, but this would cause the samples they generate to violate Pareto optimality, as shown in Figs. 8 and 9 in the “Appendix”.

To further demonstrate the superiority of our PROUD on multi-objective generation, we collect the quantitative evaluation for Pareto approximation and image quality in the left part of Table 2 by sampling 50,000 images. It shows that: our PROUD achieves the best or the second best values in both two metrics, i.e., HV for Pareto approximation and FID for image quality. It demonstrates our claim that our PROUD can provide certain quality assurance for generated samples approaching the Pareto set of multiple properties. On the contrary, either single or multiple objective generation baselines, i.e., DM+single and DM+m-MGD, would inevitably sacrifice generation quality to excessively optimize the objectives.

5.2 Protein sequence generation

To further verify our model in more challenging applications, we design multiple-objective generation task on the pOAS dataset which aims to optimize two conflicting objectives for antibody sequences:

Table 2 Quantitative evaluation for Pareto approximation and generation quality

Method	CIFAR10 (2-obj)		CIFAR10 (3-obj)		pOAS	
	HV↑ (10^{-2})	FID↓	HV↑ (10^{-3})	FID↓	HV↑	ProtGPT↑
PROUD (ours)	**5.21±0.00**	31.39±0.05	**3.26±0.00**	44.22±0.13	**2472.55±60.15**	**− 645.93±0.99**
DM+m-MGD	**5.20±0.01**	38.72±0.36	**3.26±0.01**	49.90±0.14	2289.61±65.12	− 692.80±0.34
DM+single	4.77±0.01	36.35±0.47	2.21±0.00	57.77±0.05	2302.21±58.25	− 682.26±0.49
m + 1-MGD	5.17±0.00	**11.21±0.10**	2.87±0.03	**11.80±0.05**	838.74±14.08	− 662.86±0.76
m-MGD	**5.21±0.00**	–	**3.26±0.01**	–	–	–

Bolded values and underlined values indicate the best results and the second best results, respectively. The Friedman & Nemenyi test in "Appendix B" demonstrates that our PROUD is significantly better than other baselines. "–" denotes that the value is not available as no valid data are generated

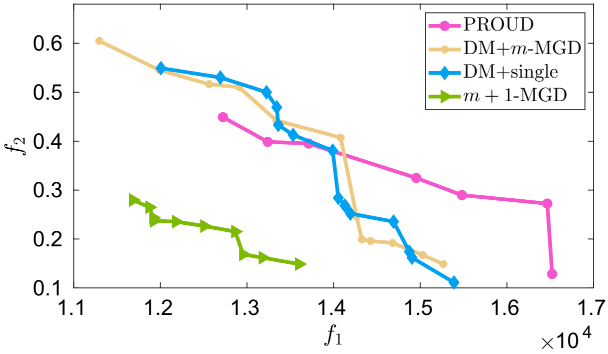


Fig. 5 The approximation of Pareto front (i.e., generated protein sequences) of various methods. We cannot visualize the results of m-MGD because all its generated protein sequences are invalid, resulting in nonexistent SASA evaluations (f_1)

- $f_1(x)$, the solvent accessible surface area (SASA) of the protein's predicted structure. Please refer to Ruffolo et al. (2023) for detailed procedures of calculating the SASA value using the protein sequences.
- $f_2(x)$, the percentage of beta sheets (%Sheets), which is measured on protein sequences directly (Cock et al., 2009).

The ground-truth Pareto front is not available due to the complexity of property objectives. Since the evaluation functions for SASA and %Sheet are not differentiable, we adopt the network predictors as differential surrogate functions for all methods. We apply the ground-truth evaluation functions for calculating the HV values on the generated samples. We adopt the discrete diffusion model in Gruver et al. (2023) as the backbone for protein sequence generation.

To demonstrate the superiority of our PROUD in multi-objective protein generation, we initially sample 5, 000 protein sequences for each method and collect the non-dominated samples based on their two target properties, as depicted in Fig. 5. The observations are as follows: (1) DM+single exhibits a wide coverage of the objective values. This could be attributed to the fact that the noise in discrete diffusion models can bring out large diversity (Gruver et al., 2023). By incorporating MGD into diffusion models, PROUD and DM+m-MGD achieve larger coverage of the objective values. This verifies the superiority of MOG over SOG. Our PROUD and DM+m-MGD emphasize respective Pareto improvement of the objectives. Nevertheless, Table 2 shows that our PROUD achieves a better HV.

Table 3 Sensitivity analysis on α and e in Eq. (12)

Metric	$\gamma = 0.2,\ e = 0.03$			$\gamma = 0.2,\ \alpha = 0.5$		
	$\alpha = 0.1$	$\alpha = 0.5$	$\alpha = 1$	$e = 0.01$	$e = 0.03$	$e = 0.05$
FID	31.58963073	31.48232218	31.5896311	31.58966697	31.48232218	31.58966696
HV	0.05211343	0.05211350	0.05211343	0.05211343	0.05211350	0.05211343

We retain more decimal places here to demonstrate the subtle differences between results

(2) Similar to the image generation task, $m + 1$-MGD demonstrates a much poorer approximation of the Pareto front for the original m objectives. Meanwhile, m-MGD even fails to generate any valid protein sequences, as the SASA evaluation ($f_1(x)$) for all its generated samples is nonexistent. This further highlights the difference between MOG and MOO.

Furthermore, we collect the quantitative evaluation for Pareto approximation and protein quality in the right part of Table 2 by sampling 5, 000 protein sequence.[5] Benefiting from our constrained-optimization formulation, our PROUD can avoid unnecessary loss of protein quality compared to other MOG/SOG counterparts, DM+m-MGD and DM+single. This improvement will greatly increase the practicality of its generated samples.

5.3 Hyper-parameter sensitivity study

We study PROUD with different configurations of the hyper-parameters, namely, α and e in Eq.(12) as well as the diversity coefficient γ in Eq.(16). The experiments are conducted on CIFAR10, with the same setting as Sect. 5.1.

We set α as 0.1, 0.5, 1 and e as 0.01, 0.03, 0.05, respectively. We observe in Table 3 that PROUD is not sensitive to the choice of the hyper-parameters α and e.

We set γ as 0, 0.1, 0.2, 1. The results are summarized in Fig. 6a–d, showing that: (1) With an appropriate diversity coefficient, our PROUD can well cover the Pareto front. (2) Without the diversity regularization, PROUD can only obtain a small set of Pareto solutions. This demonstrates the necessity of the diversity loss, consistent with the finding in the former work (Liu et al., 2021a). (3) With a too large value of γ, the generated samples could fall outside the Pareto front. The effect of the diversity coefficient on DM+m-MGD (Fig. 6e–h) is similar.

To further investigate the effects of the diversity coefficient on the generation quality, we collect FID results in Table 4. With $\gamma = 0.2$, PROUD obtains both the best FID and HV, which is thus set as the hyper-parameter used in Sect. 5.1.

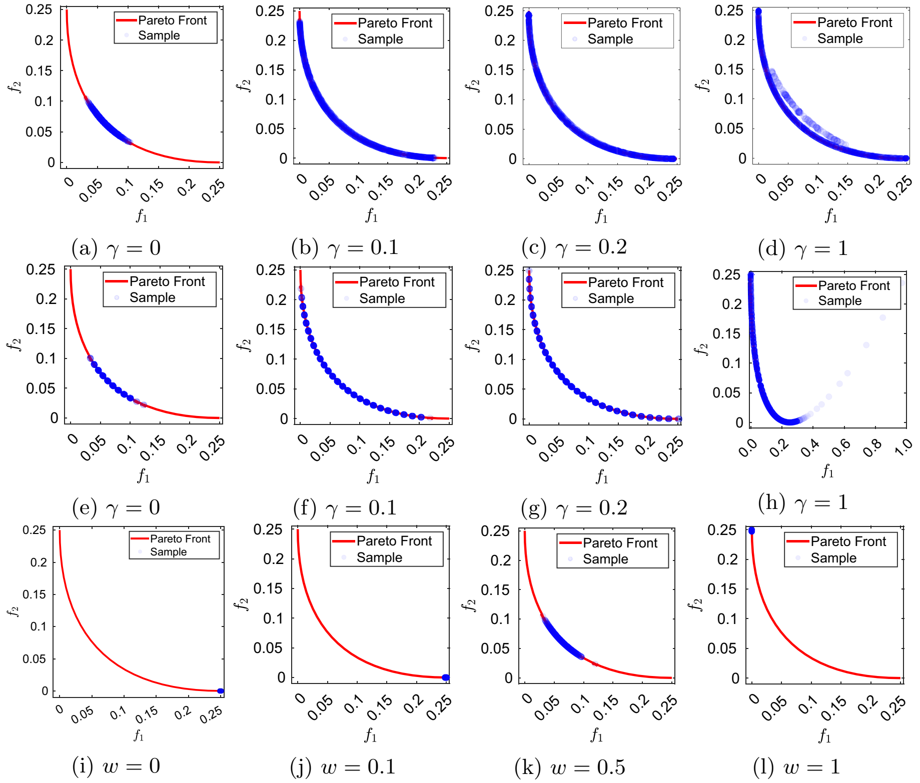
To demonstrate that the single-objective generation would fail to cover the Pareto front even with a uniform grid of weighting, we set the weight coefficient w for combining two objectives into a single objective in DM+single "$w \times f_1(x) + (1 - w) \times f_2(x)$" as 0 to 1 with a step 0.1. We put the results of 0, 0.1, 0.5, 1 in Fig. 6i–l and rest in "Appendix". With $w = 0, 0.1, 0.2, 0.3, 0.4$, the single objective is dominated by $f_2(x)$. Consequently, the generated samples achieve the smallest value for $f_2(x)$ but the largest one for $f_1(x)$; vice versa. With an equal weight, the generated samples are supposed to obtain the comprise value between two objectives, i.e., (0.0625, 0.0625). We notice that

[5] We only sample 5, 000 protein sequence since the computation cost of SASA values is very high.

Table 4 Sensitivity analysis on γ

Metric	$\gamma = 0$	$\gamma = 0.01$	$\gamma = 0.1$	$\gamma = 0.2$	$\gamma = 0.3$	$\gamma = 1$
FID	34.80	30.98	31.80	**31.48**	31.63	33.59
HV	0.0483	0.0498	0.0521	**0.0521**	0.0521	0.0521

α and e are set to 0.5 and 0.03, respectively. The best results are marked in bold



(a) $\gamma = 0$ (b) $\gamma = 0.1$ (c) $\gamma = 0.2$ (d) $\gamma = 1$

(e) $\gamma = 0$ (f) $\gamma = 0.1$ (g) $\gamma = 0.2$ (h) $\gamma = 1$

(i) $w = 0$ (j) $w = 0.1$ (k) $w = 0.5$ (l) $w = 1$

Fig. 6 Analysis on the effects of the diversity coefficient γ in Eq. (16) to our PROUD (1st row) and DM+m-MGD (2nd row). As DM+single (3rd row) degenerates to SOG and does not have the diversity regularization, we conduct sensitivity analysis on its weight coefficient for combining two objectives, i.e., $(1 - w)f_1 + wf_2$. The depth of color represents sample density, the deeper the higher

the generated samples cover a small range around this point. This diversity could result from the diffusion noise in diffusion models.

6 Conclusion

This paper studies the problem of optimizing deep generative models with multiple conflicting objectives. We highlight this problem setting by treating the optimization of samples with multiple properties and the process of sample generation as a unified task. By analyzing the connections and differences from multi-objective optimization, we introduce a constrained optimization formulation to solve the multi-objective generation problem, based on which we developed PROUD. Our experiments demonstrate the efficacy of PROUD in both image and protein sequence generation. While we explored the white-box multi-objectives in this work, it would be interesting to explore our PROUD in the black-box setting in the future. The multiple gradient descent technique used can be replaced by methods such as Bayesian multi-objective optimization (Stanton et al., 2022).

Appendix A: Complete sensitivity analysis for single-objective generation

We set the weight coefficient w for combining two objectives in DM+single "$w \times f_1(x) + (1 - w) \times f_2(x)$" as 0 to 1 with a step 0.1. The results is shown in Fig. 7:

- when $w < 0.5$, the resultant final objective is dominated by $f_2(x)$. Consequently, the leading objective is optimized to the best where all the generated samples have the smallest value for $f_2(x)$ but the largest one for $f_1(x)$.
- when $w > 0.5$, the resultant final objective is dominated by $f_1(x)$. Therefore, the generated samples achieve the smallest value for the first objective but the largest one for the second objective.
- when $w = 0.5 = \frac{1}{m}$, the generated samples are supposed to obtain the comprise value between $f_1(x)$ and $f_2(x)$, i.e., (0.0625, 0.0625). We notice that the generated samples cover a small range around this point. This diversity could result from the diffusion noise in diffusion models (Figs. 8, 9, 10).

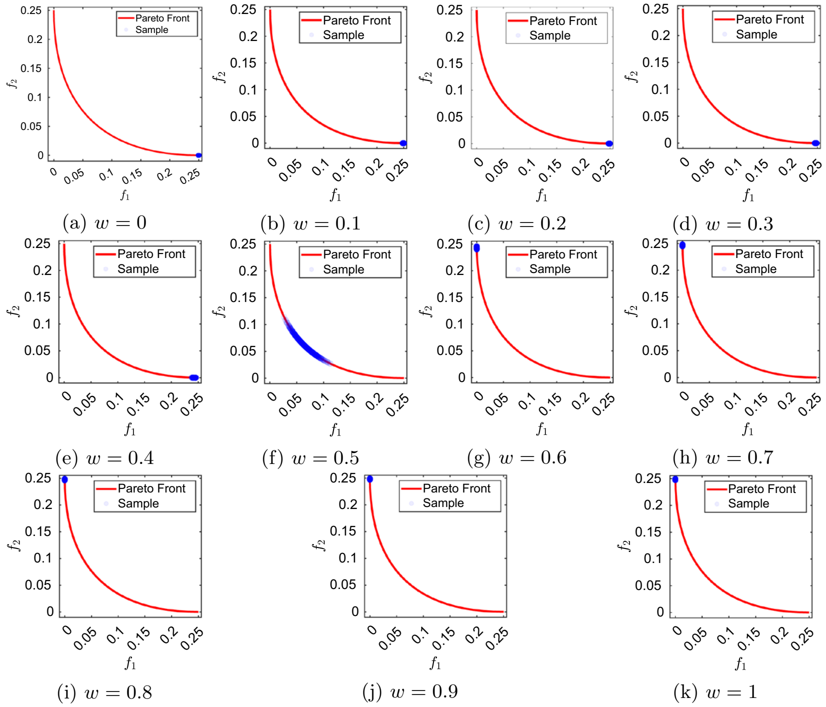
Appendix B: More experimental settings and analyses

Image Generation

According to Ishibuchi et al. (2013); Li et al. (2017)[6], we can obtain that: (1) the Pareto solutions of the two objective setting are the points on the line between 1_Ω and 0.5_Ω. Namely, the Pareto solutions are $\{x | x_\Omega = \kappa_\Omega, \kappa_\Omega \in [0.5_\Omega, 1_\Omega]\}$.[7] When taking images from CIFAR10 based on the Pareto set (Fig. 12), we follow Liu et al. (2021b) to sample images in a small neighborhood around κ_Ω, namely, $\|x_\Omega - \kappa_\Omega\|_2^2 \leq \epsilon$, where $\epsilon = 8 \times 10^{-4}$. (2) The Pareto solutions of the three objective setting are the points on the convex polygonal formed by three points $a_\Omega, b_\Omega, c_\Omega$. For easy understanding, we assume $\Omega = 3 \times 1 \times 1$, which is actually to constrain the middle point of CIFAR10 images to be certain colors.

[6] Our problem setting is slightly different as we take the distance square in order to obtain a non-linear shape of the Pareto front. We also refer reviewer to example-1 in Liu et al. (2021a) that defines a same two-objective problem but with 1-D decision variable for easy understanding.

[7] We use $[0.5_\Omega, 1_\Omega]$ to denote image patches in normalized RGB color values between [0.5, 0.5, 0.5] (grey) and [1, 1, 1] (white).

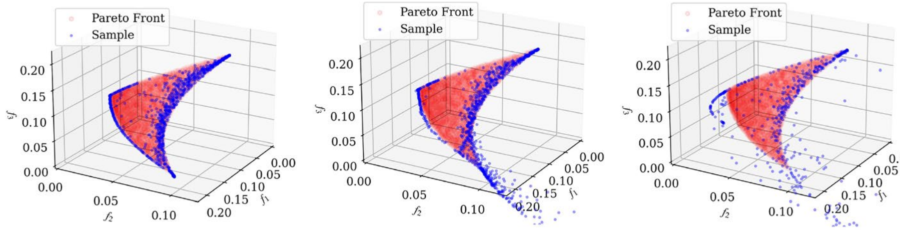


(a) $w = 0$ (b) $w = 0.1$ (c) $w = 0.2$ (d) $w = 0.3$

(e) $w = 0.4$ (f) $w = 0.5$ (g) $w = 0.6$ (h) $w = 0.7$

(i) $w = 0.8$ (j) $w = 0.9$ (k) $w = 1$

Fig. 7 Sensitivity analysis on the weight coefficient for combining two objectives, i.e., $(1-w)f_1 + wf_2$ in DM+single. The depth of color represents sample density, the deeper the higher



(a) $\gamma = 0.8$ (b) $\gamma = 1$ (c) $\gamma = 1.5$

Fig. 8 Different diversity coefficient γ for DM+m-MGD on CIFAR10 optimized with three objectives. 1000 generated samples are randomly selected for visualization

We visualize the Pareto front of these two settings in Fig. 11. Specifically, for the two objective setting, the Pareto optimal points lie on the line between [1, 1, 1] and [0.5, 0.5, 0.5] (Fig. 11a), which physically denote RGB values (normalized, RGB values [0, 255] divided by 255). Then, we calculate the objectives values $[f_1(x), f_2(x)]$ for these points accordingly, shown in Fig. 11b. Figure 11c, d are plotted for the three objective setting in a similar way. According to their Pareto fronts, we select [0.25, 0.25] and [0.2, 0.1, 0.2] as reference points to calculate the hypervolume (HV) for the two objective setting and the three objective setting in Table 2, respectively.

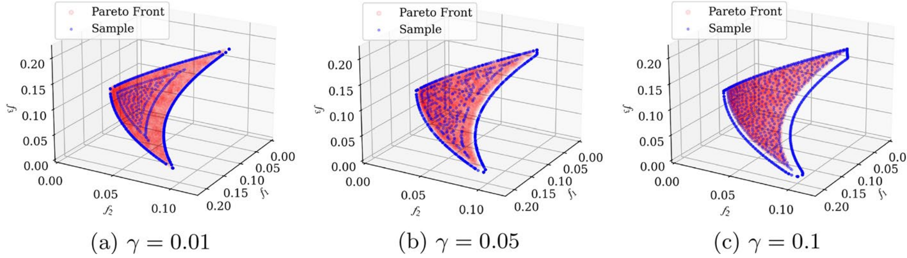
We sample CIFAR10 image using the constraint with different patch sizes to demonstrate its effect in Fig. 13. With a smaller size of the region Ω, more CIFAR10 images will meet the constraint.

Protein Sequence Generation

Our experiments in Section 5.2 adopted the same dataset and objectives as that in Section 5.2 of Gruver et al. (2023). Note that we did not include their other experiments, because the experiment in their Section 5.1 is not a generation task equipped with property optimization and the dataset for the experiment in Section 5.3 and 5.4 has not been released due to private data. We select $[1 \times 10^4, 0]$ as a reference point to calculate the HV for this task.

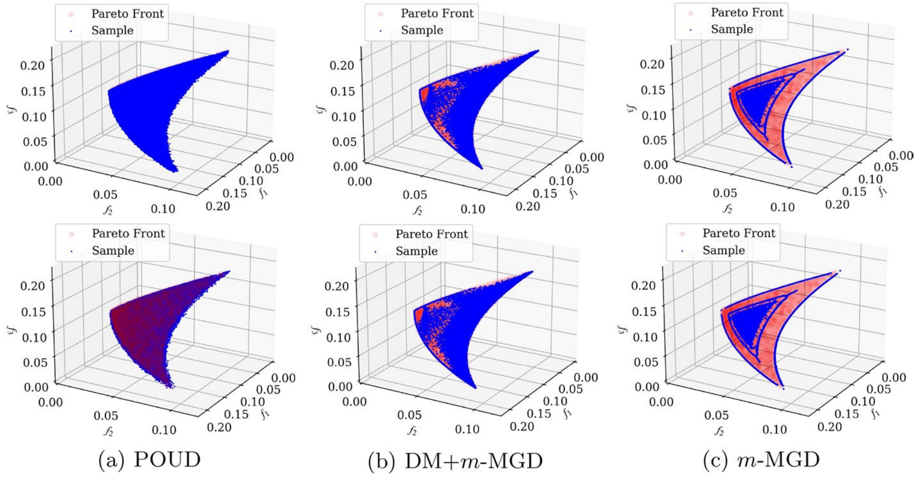
Justification of Our Experiment Designs

Our experiment designs can appropriately justify the motivation of the MOG problem. Both CIFAR10 and protein datasets are real-world datasets whose data lie on low-dimensional manifolds in high-dimensional space (Krizhevsky and Hinton, 2009; Gruver et al., 2023), thus applicable to our MOG problem setting. Meanwhile, the objectives considered for CIFAR10 are indeed benchmark multi-objective optimization problems with clear evaluations (Ishibuchi et al., 2013); the objectives considered for the protein design task



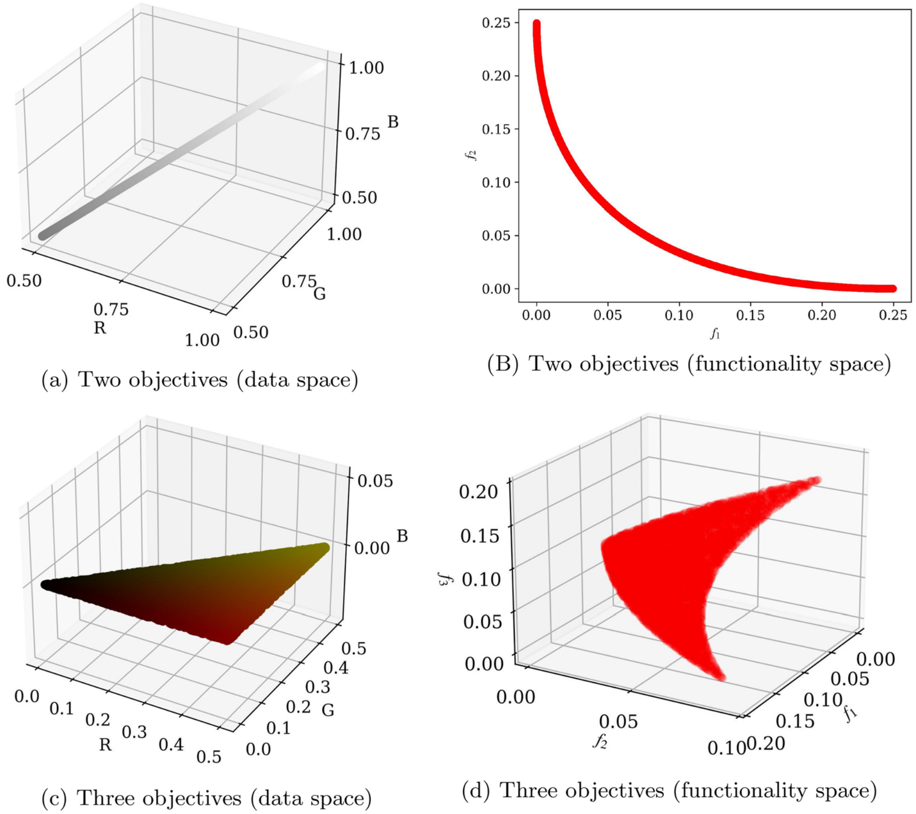
(a) $\gamma = 0.01$ (b) $\gamma = 0.05$ (c) $\gamma = 0.1$

Fig. 9 Different diversity coefficient γ for m-MGD on CIFAR10 optimized with three objectives. 1000 generated samples are randomly selected for visualization



(a) POUD (b) DM+m-MGD (c) m-MGD

Fig. 10 Approximation of Pareto front of various methods on CIFAR10 optimized with three objectives. The first row presents 50,000 generated samples while the second row presents non-dominated points out of 50,000 sample points, verifying the HV results obtained in Table 2



(a) Two objectives (data space)

(B) Two objectives (functionality space)

(c) Three objectives (data space)

(d) Three objectives (functionality space)

Fig. 11 Pareto front of two and three objectives in data space and functionality space optimized for CIFAR10 image generation

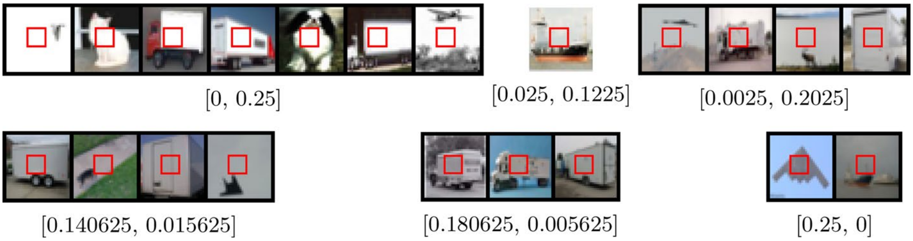
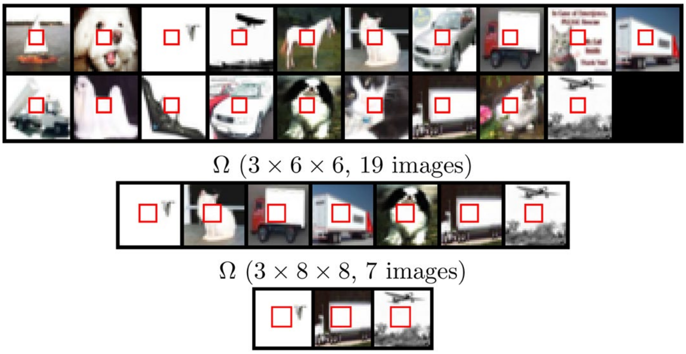


Fig. 12 Full resolution CIFAR10 images ($3 \times 32 \times 32$) in Fig. 1b of the manuscript. The red box denotes the region Ω ($3 \times 8 \times 8$) in the two objectives in Sect. 5.1

represent real-world scenarios (Gruver et al., 2023). Lastly, Fig. 2 and Table 2 demonstrate the necessity of considering generation quality, as the generation quality of all baseline methods suffers to some extent when optimizing multiple properties.

Significant Test

We apply the Friedman test under the null hypothesis positing that all methods perform similarly, alongside the Nemenyi post-hoc test for pairwise comparisons among the



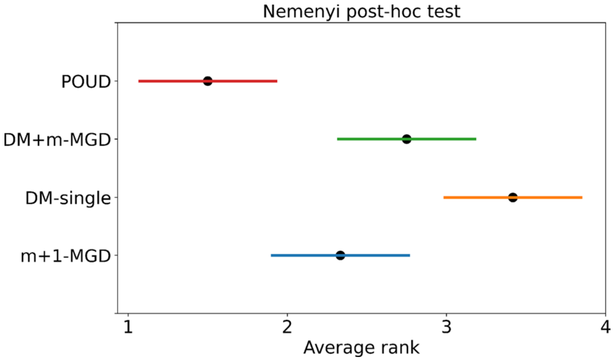
Ω (3 × 6 × 6, 19 images)

Ω (3 × 8 × 8, 7 images)

Ω (3 × 10 × 10, 3 images)

Fig. 13 Sampling CIFAR-10 images with regions of different patch sizes

Fig. 14 Nemenyi post-hoc test over four methods



four methods (Demšar, 2006). The number of factors was set to four, given the failure of m-MGD to produce qualified samples, leading to its exclusion. The dataset comprised 30 instances, with each of the four methods independently evaluated five times across three datasets, employing two evaluation criteria. The Friedman test shows that $\tau_F = 18.24$, greater than the critical value $F_{3,87} = 2.709$ when $\alpha = 0.05$. Therefore, the null hypothesis is rejected, which signifies a statistically significant difference among the four methods at the significance level of 0.05. Subsequent analysis via the Nemenyi post-hoc test in Fig. 14 unequivocally demonstrates that our PROUD exhibits marked superiority over the three baseline methods.

Appendix C: Discussions

The constrained MOO problem defines its decision space S on a constrained space expressed using specified linear, nonlinear, or box constraints (Afshari et al., 2019; Désidéri, 2018) in $\mathbb{R}^d$. Consequently, it is different from our MOG problems, whose manifold is delineated by a given dataset $\mathcal{X}$. Nevertheless, MOG problems could be understood as a type of constrained MOO problem in a broader context (Table 5).

Table 5 Comparison of the MOG problem with the relevant MOO problems

	Objectives	Decision/data space	Generation quality
MOO	$F(x)$	$x \in \mathbb{R}^d$	×
Constrained MOO	$F(x)$	$x \in S, S \subset \mathbb{R}^d$ defined by (non)linear or box constraints	×
MOG	$F(x)$	$x \in \mathcal{X}, \mathcal{X} \subset \mathbb{R}^d$	✓

The generation quality in MOG is usually modeled based on a given dataset $X \subset \mathcal{X}$, where $\mathcal{X}$ denotes a low-dimensional manifold embedded in a high dimensional space $\mathbb{R}^d$. $F(x) = [f_1(x), f_2(x), \ldots, f_m(x)]$

Author Contributions Idea: YY; Methodology and Experiment: YY, YP; Writing—comments/edits: all.

Funding This work was supported by the National Research Foundation, Singapore and DSO National Laboratories under the AI Singapore Programme (AISG Award No: AISG2-GC-2023-010-T), the A*STAR GAP project (Grant No. I23D1AG079), the A*STAR Career Development Fund (Grant No. C222812019), the A*STAR Pitchfest for ECR 232D800027, the A*STAR Centre for Frontier AI Research, the Program for Guangdong Introducing Innovative and Entrepreneurial Teams (Grant No. 2017ZT07X386), NSFC (Grant No. 62250710682), and the Program for Guangdong Provincial Key Laboratory (Grant No. 2020B121201001).

Availability of data and materials All datasets used in this work are available online and clearly cited.

Code availability The code of this work is available at https://github.com/EvaFlower/Pareto-guided-diffusion-model.

Declarations

Conflict of interest The authors have no financial or non-financial interests to disclose that are relevant to the content of this article.

Ethics approval Not applicable.

Consent to participate. Not applicable.

Consent to publish Not applicable.

References

Afshari, H., Hare, W., & Tesfamariam, S. (2019). Constrained multi-objective optimization algorithms: Review and comparison with application in reinforced concrete structures. *Applied Soft Computing, 83*, 105631. https://doi.org/10.1016/J.ASOC.2019.105631

Andrieu, C., De Freitas, N., Doucet, A., et al. (2003). An introduction to MCMC for machine learning. *Machine Learning, 50*, 5–43. https://doi.org/10.1023/A:1020281327116

Arjovsky, M., Chintala, S., & Bottou, L. (2017). Wasserstein generative adversarial networks. In *International conference on machine learning* (pp. 214–223). https://proceedings.mlr.press/v70/arjovsky17a.html

Borghi, G., Herty, M., & Pareschi, L. (2023). An adaptive consensus based method for multi-objective optimization with uniform pareto front approximation. *Applied Mathematics and Optimization, 88*(2), 58. https://doi.org/10.1007/s00245-023-10036-y

Cheng, R., Li, M., Tian, Y., et al. (2017). A benchmark test suite for evolutionary many-objective optimization. *Complex and Intelligent Systems, 3*, 67–81. https://doi.org/10.1007/s40747-017-0039-7

Chinchuluun, A., & Pardalos, P. M. (2007). A survey of recent developments in multiobjective optimization. *Annals of Operations Research, 154*(1), 29–50. https://doi.org/10.1007/S10479-007-0186-0

Cock, P. J., Antao, T., Chang, J. T., et al. (2009). Biopython: Freely available python tools for computational molecular biology and bioinformatics. *Bioinformatics, 25*(11), 1422–1423. https://doi.org/10.1093/bioinformatics/btp163

Dathathri, S., Madotto, A., & Lan, J. et al (2020). Plug and play language models: A simple approach to controlled text generation. In *International conference on learning representations*. https://openreview.net/forum?id=H1edEyBKDS

Deb, K. (2001). *Multi-objective optimization using evolutionary algorithms* (Vol. 16). Wiley.

Demšar, J. (2006). Statistical comparisons of classifiers over multiple data sets. *The Journal of Machine learning research, 7*, 1–30.

Deng, Y., Yang, J., Chen, D., et al (2020). Disentangled and controllable face image generation via 3D imitative-contrastive learning. In *IEEE/CVF conference on computer vision and pattern recognition* (pp. 5154–5163). https://doi.org/10.1109/CVPR42600.2020.00520

Désidéri, J. A. (2012). Multiple-gradient descent algorithm (MGDA) for multiobjective optimization. *Comptes Rendus Mathematique, 350*(5–6), 313–318. https://doi.org/10.1016/j.crma.2012.03.014

Désidéri, J. A. (2018). Quasi-Riemannian multiple gradient descent algorithm for constrained multiobjective differential optimization. Ph.D. thesis, Inria Sophia-Antipolis; Project-Team Acumes. https://inria.hal.science/hal-01740075

Dhariwal, P., & Nichol, A. (2021). Diffusion models beat GANs on image synthesis. In *Advances in neural information processing systems* (pp. 8780–8794). https://proceedings.neurips.cc/paper_files/paper/2021/file/49ad23d1ec9fa4bd8d77d02681df5cfa-Paper.pdf

Fefferman, C., Mitter, S., & Narayanan, H. (2016). Testing the manifold hypothesis. *Journal of the American Mathematical Society, 29*(4), 983–1049. https://doi.org/10.1090/jams/852

Ferruz, N., Schmidt, S., & Höcker, B. (2022). Protgpt2 is a deep unsupervised language model for protein design. *Nature Communications, 13*(1), 4348. https://doi.org/10.1038/s41467-022-32007-7

Gong, C., Liu, X., & Liu, Q. (2021). Bi-objective trade-off with dynamic barrier gradient descent. In *Advances in neural information processing systems* (pp. 29630–29642). https://proceedings.neurips.cc/paper_files/paper/2021/file/f7b027d45fd7484f6d0833823b98907e-Paper.pdf

Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., et al. (2014). Generative adversarial nets. In *Advances in neural information processing systems* (pp. 2672–2680). https://proceedings.neurips.cc/paper_files/paper/2014/file/5ca3e9b122f61f8f06494c97b1afccf3-Paper.pdf

Gruver, N., Stanton, S., Frey, N. C., et al. (2023). Protein design with guided discrete diffusion. In *Advances in neural information processing systems* (pp. 12489–12517). https://proceedings.neurips.cc/paper_files/paper/2023/file/29591f355702c3f4436991335784b503-Paper-Conference.pdf

Guo, X., Du, Y., & Zhao, L. (2020). Property controllable variational autoencoder via invertible mutual dependence. In *International conference on learning representations*. https://openreview.net/forum?id=tYxG_OMs9WE

Heusel, M., Ramsauer, H., Unterthiner, T., et al. (2017). GANs trained by a two time-scale update rule converge to a local Nash equilibrium. In *Advances in neural information processing systems* (pp. 6629–6640). https://proceedings.neurips.cc/paper_files/paper/2017/file/8a1d694707eb0fefe65871369074926d-Paper.pdf

Ho, J., Jain, A., & Abbeel, P. (2020). Denoising diffusion probabilistic models. In *Advances in neural information processing systems* (pp. 6840–6851). https://proceedings.neurips.cc/paper/2020/file/4c5bcfec8584af0d967f1ab10179ca4b-Paper.pdf

Ishibuchi, H., Tsukamoto, N., & Nojima, Y. (2008). Evolutionary many-objective optimization: A short review. In *IEEE congress on evolutionary computation* (pp. 2419–2426). https://doi.org/10.1109/CEC.2008.4631121

Ishibuchi, H., Yamane, M., Akedo, N., et al. (2013). Many-objective and many-variable test problems for visual examination of multiobjective search. In *IEEE congress on evolutionary computation* (pp. 1491–1498). https://doi.org/10.1109/CEC.2013.6557739

Jain, M., Raparthy, S. C., & Hernández-Garcıa, A., et al. (2023). Multi-objective gflownets. In *International conference on machine learning* (pp. 14631–14653). https://proceedings.mlr.press/v202/jain23a.html

Jin, W., Barzilay, R., & Jaakkola, T. (2020). Multi-objective molecule generation using interpretable substructures. In *International conference on machine learning* (pp. 4849–4859). http://proceedings.mlr.press/v119/jin20b.html

Kingma, D. P., & Welling, M. (2014). Auto-encoding variational Bayes. In *International conference on learning representations*. https://openreview.net/forum?id=33X9fd2-9FyZd

Klys, J., Snell, J., & Zemel, R. (2018). Learning latent subspaces in variational autoencoders. In *Advances in neural information processing systems* (pp. 6445–6455). https://proceedings.neurips.cc/paper_files/paper/2018/file/73e5080f0f3804cb9cf470a8ce895dac-Paper.pdf

Krizhevsky, A., & Hinton, G., et al. (2009). Learning multiple layers of features from tiny images. https://www.cs.utoronto.ca/~kriz/learning-features-2009-TR.pdf

Li, M., Grosan, C., Yang, S., et al. (2017). Multiline distance minimization: A visualized many-objective test problem suite. *IEEE Transactions on Evolutionary Computation, 22*(1), 61–78. https://doi.org/10.1109/TEVC.2017.2655451

Li, S., Liu, M., & Walder, C. (2022). Editvae: Unsupervised parts-aware controllable 3d point cloud shape generation. In *AAAI conference on artificial intelligence* (pp. 1386–1394). https://doi.org/10.1609/AAAI.V36I2.20027

Liao, Y., Schwarz, K., Mescheder, L., et al. (2020). Towards unsupervised learning of generative models for 3D controllable image synthesis. In *IEEE/CVF conference on computer vision and pattern recognition* (pp. 5871–5880). https://doi.org/10.1109/CVPR42600.2020.00591

Liu, X., Tong, X., & Liu, Q. (2021a). Profiling pareto front with multi-objective stein variational gradient descent. In *Advances in neural information processing systems* (pp. 14721–14733). https://proceedings.neurips.cc/paper/2021/file/7bb16972da003e87724f048d76b7e0e1-Paper.pdf

Liu, X., Tong, X., & Liu, Q. (2021b). Sampling with trusthworthy constraints: A variational gradient framework. In *Advances in neural information processing systems* (pp. 23557–23568). https://papers.nips.cc/paper/2021/file/c61aed648da48aa3893fb3eaadd88a7f-Paper.pdf

McInnes, L., Healy, J., & Melville, J. (2018). Umap: Uniform manifold approximation and projection for dimension reduction. arXiv:1802.03426

Olsen, T. H., Boyles, F., & Deane, C. M. (2022). Observed antibody space: A diverse database of cleaned, annotated, and translated unpaired and paired antibody sequences. *Protein Science, 31*(1), 141–146. https://doi.org/10.1002/pro.4205

Papamakarios, G., Nalisnick, E., Rezende, D. J., et al. (2021). Normalizing flows for probabilistic modeling and inference. *Journal of Machine Learning Research, 22*(57), 1–64.

Roweis, S. T., & Saul, L. K. (2000). Nonlinear dimensionality reduction by locally linear embedding. *Science, 290*(5500), 2323–2326. https://doi.org/10.1126/science.290.5500.2323

Ruffolo, J. A., Chu, L. S., Mahajan, S. P., et al. (2023). Fast, accurate antibody structure prediction from deep learning on massive set of natural antibodies. *Nature Communications, 14*(1), 2389. https://doi.org/10.5281/zenodo.7709609

Sanchez-Lengeling, B., & Aspuru-Guzik, A. (2018). Inverse molecular design using machine learning: Generative models for matter engineering. *Science, 361*(6400), 360–365. https://doi.org/10.1126/science.aat2663

Sener, O., & Koltun, V. (2018). Multi-task learning as multi-objective optimization. In *Advances in neural information processing systems* (pp. 525–536). https://proceedings.neurips.cc/paper/2018/file/432aca3a1e345e339f35a30c8f65edce-Paper.pdf

Shen, M. W., Bengio, E., & Hajiramezanali, E., et al. (2023). Towards understanding and improving gflownet training. In *International conference on machine learning* (pp. 30956–30975). https://proceedings.mlr.press/v202/shen23a.html

Sohl-Dickstein, J., Weiss, E., Maheswaranathan, N., et al. (2015). Deep unsupervised learning using nonequilibrium thermodynamics. In *International conference on machine learning* (pp. 2256–2265). http://proceedings.mlr.press/v37/sohl-dickstein15.html

Song, Y., & Ermon, S. (2019). Generative modeling by estimating gradients of the data distribution. In *Advances in neural information processing systems* (pp. 11918–11930). https://proceedings.neurips.cc/paper_files/paper/2019/file/3001ef257407d5a371a96dcd947c7d93-Paper.pdf

Song, Y., & Ermon, S. (2020). Improved techniques for training score-based generative models. In *Advances in neural information processing systems* (pp. 12438–12448). https://papers.neurips.cc/paper_files/paper/2020/file/92c3b916311a5517d9290576e3ea37ad-Paper.pdf

Song, Y., & Kingma, D. P. (2021). How to train your energy-based models. arXiv:2101.03288

Song, Y., Durkan, C., Murray, I., et al. (2021a). Maximum likelihood training of score-based diffusion models. In *Advances in neural information processing systems* (pp. 1415–1428). https://papers.nips.cc/paper/2021/file/0a9fdbb17feb6ccb7ec405cfb85222c4-Paper.pdf

Song, Y., Sohl-Dickstein, J., Kingma, D.P., et al (2021b). Score-based generative modeling through stochastic differential equations. In *International conference on learning representations*. https://openreview.net/forum?id=PxTIG12RRHS

Stanton, S., Maddox, W., Gruver, N., et al. (2022). Accelerating Bayesian optimization for biological sequence design with denoising autoencoders. In *International conference on machine learning* (pp. 20459–20478). https://proceedings.mlr.press/v162/stanton22a.html

Tagasovska, N., Frey, N. C., Loukas, A., et al. (2022). A pareto-optimal compositional energy-based model for sampling and optimization of protein sequences. In *NeurIPS 2022 workshop AI for science: progress and promises*. https://openreview.net/forum?id=U2rNXaTTXPQ

Tanabe, R., & Ishibuchi, H. (2020). An easy-to-use real-world multi-objective optimization problem suite. *Applied Soft Computing, 89*, 106078. https://doi.org/10.1016/J.ASOC.2020.106078

Van Veldhuizen, D. A., Lamont, G. B., et al (1998). Evolutionary computation and convergence to a pareto front. In *Late breaking papers at the genetic programming 1998 conference* (pp. 221–228). https://citeseerx.ist.psu.edu/document?repid=rep1 &type=pdf &doi=f329eb18a4549daa83fae28043d19b83fe8356fa

Wang, S., Guo, X., Lin, X., et al. (2022). Multi-objective deep data generation with correlated property control. In *Advances in neural information processing systems* (pp. 28889–28901). https://proceedings.neurips.cc/paper_files/paper/2022/file/b9c2e8a0bbed5fcfaf62856a3a719ada-Paper-Conference.pdf

Wang, S., Du, Y., Guo, X., et al. (2024). Controllable data generation by deep learning: A review. *ACM Computing Surveys*. https://doi.org/10.1145/3648609

Wang, Z., Zhao, L., & Xing, W. (2023). Stylediffusion: Controllable disentangled style transfer via diffusion models. In *IEEE/CVF international conference on computer vision* (pp. 7677–7689). https://doi.org/10.1109/ICCV51070.2023.00706

Watson, J. L., Juergens, D., Bennett, N. R., et al. (2023). De novo design of protein structure and function with rfdiffusion. *Nature, 620*(7976), 1089–1100. https://doi.org/10.1038/s41586-023-06415-8

Welling, M., & Teh, Y. W. (2011). Bayesian learning via stochastic gradient Langevin dynamics. In *International conference on machine learning* (pp. 681–688). https://icml.cc/2011/papers/398_icmlpaper.pdf

Yang, L., Zhang, Z., Song, Y., et al. (2023). Diffusion models: A comprehensive survey of methods and applications. *ACM Computing Surveys, 56*(4), 1–39. https://doi.org/10.1145/3626235

Yao, K., Gao, P., Yang, X., et al. (2022). Outpainting by queries. In *European conference on computer vision* (pp. 153–169). https://doi.org/10.1007/978-3-031-20050-2_10

Ye, M., & Liu, Q. (2022). Pareto navigation gradient descent: A first-order algorithm for optimization in pareto set. In *Uncertainty in artificial intelligence* (pp. 2246–2255). https://proceedings.mlr.press/v180/ye22a.html

Zhang, S., Qian, Z., Huang, K., et al. (2023). Robust generative adversarial network. *Machine Learning, 112*, 5135–5161. https://doi.org/10.1007/s10994-023-06367-0

Zitzler, E., & Thiele, L. (1999). Multiobjective evolutionary algorithms: A comparative case study and the strength pareto approach. *IEEE Transactions on Evolutionary Computation, 3*(4), 257–271. https://doi.org/10.1109/4235.797969

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Authors and Affiliations

Yinghua Yao[1,2] · Yuangang Pan[1,2] · Jing Li[1,2] · Ivor Tsang[1,2] · Xin Yao[3,4]

✉ Yuangang Pan
yuangang.pan@gmail.com

Yinghua Yao
eva.yh.yao@gmail.com

Jing Li
j.lee9383@gmail.com

Ivor Tsang
ivor.tsang@gmail.com

Xin Yao
xinyao@ln.edu.hk

[1] Centre for Frontier AI Research, Agency for Science, Technology and Research (A*STAR), Singapore 138632, Singapore

[2] Institute of High Performance Computing, Agency for Science, Technology and Research (A*STAR), Singapore 138632, Singapore

[3] School of Data Science, Lingnan University, Hong Kong, China

[4] School of Computer Science, University of Birmingham, Birmingham, UK