LEARNING PARETO MANIFOLDS FOR DIFFUSION PROCESS

Anonymous authors
Paper under double-blind review

ABSTRACT

We propose Pareto-front Guided Diffusion (PGDiff), a novel diffusion framework that improves the learning of the reverse process by enforcing three categories of constraints, including initial/boundary constraints, governing constraints and inequality constraints. Unlike existing methods that applies constraints post-hoc or via penalty terms, PGDiff introduces a step-wise optimization architecture that inserts the intermediate state optimized by the Multiple Gradient Descent Algorithm between denoising steps. This enables the generated samples to move toward the Pareto front in the objective space, allowing progressive satisfaction of multiple complex constraints while preserving sample fidelity. Experiments on the path planning tasks and the PDEs solvers show PGDiff achieves a high constraint satisfaction under multi-constraints settings, outperforming several state-of-the-art baselines.

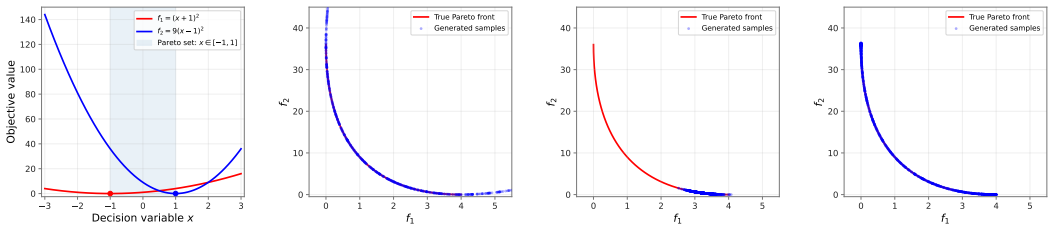
1 INTRODUCTION

Recently, diffusion models have emerged as a powerful class of generative methods, achieving state-of-the-art performance in image synthesis (Rombach et al., 2022; Saharia et al., 2022a), audio generation (Kong et al., 2020; Popov et al., 2021), and molecular/chemistry design (Huang et al., 2023; Anand & Achim, 2022). Particularly, diffusion models also show great promise in generating samples that satisfy multiple properties and constraints. However, real-world data from scientific and engineering domains is usually governed by several interacting physical objectives (Wang et al., 2023). Standard diffusion models lack explicit mechanisms to handle these physical objectives, resulting in the generation of many dominated samples. These samples may be reasonable in terms of data distribution but physically infeasible as shown in Figure 1b. Moreover, multiple physical objectives may compete with each other, causing improving one may harm another as shown in Figure 1c. How to enable generative models to address these issues remains a challenge.

A natural solution to these challenges might be to train models using data that already satisfy the constraints. However, the randomness in the diffusion sampling process does not guarantee that all the generated samples will be valid, even if all the training data used are valid. More critically, real-world applications may introduce conditions not present in the training data. Other methods enforce constraints through either post-generation or penalty-based training objectives. The former may reduce sample fidelity, while the latter may fail to handle competing constraint trade-offs and may produce results that can't perfectly consistent with real physic world. Overall, these methods lack explicit mechanism for coordinating competing constraints during generation, making it difficult to improve all objectives without sacrificing sample fidelity.

In this study, we focus on three categories of constraints which include initial/boundary constraints, governing constraints and inequality constraints. The first category specifies the state of a generated solution at particular spatial locations or time points. The second defines the relation among variables or describes how a state evolves. And the third requires the generated result to lie within an admissible region rather than strictly satisfying an equality. These categories cover a class of physical laws with explicit PDE forms, for which the governing equations and initial/boundary conditions can be represented as residuals similar to PINNs (Raissi et al., 2019). The proposed model, Pareto-front Guided Diffusion (PGDiff), represents a method different from conventional approaches by offering a integration of multiple interacting constraints within the generation process. The PGDiff model introduces a step-wise optimization architecture, directly embedding multi-objective optimization

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.



(a) Two Objectives (b) Standard Diffusion (c) Weighted Diffusion (d) PGDiff

Figure 1: A two-objective toy example for different models. (a) The plot of two competing objectives $f_1 = (x+1)^2$ and $f_2 = 9(x-1)^2$, whose conflicting region forms the Pareto set $x \in [-1, 1]$. (b) The Standard Diffusion generates many dominated samples away from the Pareto front. (c) The weighted Diffusion with fair weights ($\omega_1 = \omega_2 = 0.5$) generates samples with a particular compromise. (d) PGDiff progressively guides generated samples toward the Pareto front, enabling balanced satisfaction of competing objectives.

into the diffusion process. This approach progressively guides the physical objectives towards the Pareto front, while satisfying competing constraints simultaneously.

Our contributions are as follows:

- We propose a unified physics-constrained diffusion framework that systematically incorporates three categories of constraints, including initial/boundary constraints, governing constraints and inequality constraints. These categories can cover a class of practical physical laws. This formulation enables diffusion models to incorporate explicit physical knowledge in addition to simply matching the data distribution.
- We develop Pareto-front Guided Diffusion (PGDiff), a step-wise diffusion model that resolves competing physical objectives throughout the reverse process. PGDiff determines a Pareto direction at each denoising step through multi-objective optimization, allowing to improve progressive constraint satisfaction without sacrificing sample fidelity.

2 Related Work

2.1 Multi-Objective Optimization in Generation

Many generation tasks involve more than one objectives. Molecular design often requires consideration of several properties, such as potency, safety, and synthesizability (Fromer & Coley, 2023). Trajectory generation may need to balance multiple constraints such as collision-free, path length and smoothness (Ajeil et al., 2020). However, it is often difficult to optimize these potentially conflicting objectives simultaneously (Brockhoff & Zitzler, 2006), so how to balance them becomes a big challenge.

A common approach is to combine all objectives into a single function, also known as scalarization. This method is required to determine the relative importance between different objectives. Jain et al. (2023) samples preference weights from the Dirichlet distribution and uses them as conditions for the generative model. Rame et al. (2023) instead selects several weight vectors over the preference simplex and trains one policy for each vector. However, the generated results depend strongly on the choice and distribution of these weights. An unsuitable setting may lead to a bias towards one objective or fail to optimize some objectives effectively.

Pareto-based methods provide a more flexible way to handle this problem. Instead of selecting one single compromise in advance, they aim to explore the trade-offs among multiple objectives and seek Pareto optimal. Deb et al. (2002) proposed an evolutionary and population-based method which maintains a group of candidate solutions. Lin et al. (2022) proposed a preference-conditioned method which takes a preference vector as input and generates its corresponding Pareto solution. Désidéri (2012) proposed a gradient-based method which uses the gradients of the multiple objectives to find a Pareto descent direction that decreases all objectives. Sener & Koltun (2018) apply

this idea to multi-task learning and update the model parameters with the resulting direction. Yao et al. (2024) further used multiple property gradients to guide the process of the diffusion models during inference. These works provide an important basis for gradient-based Pareto generation.

2.2 Iterative Refinement in Diffusion Models

Traditional generative models often produce samples through a single forward pass which limits their ability to progressively refine the output according to complex constraints. Iterative refinement provides a natural way to address this limitation. Denoising diffusion models already introduced a multi-step generation process, but the standard formulation does not explicitly guide intermediate states toward constraint satisfaction (Ho et al., 2020). Several studies have therefore introduced different forms of refinement into diffusion models. Yuan et al. (2023) incorporated physics-based guidance into the denoising loop for human motion generation. For image super-resolution, Saharia et al. (2022b) proposed cascaded diffusion models that progressively increase resolution. Shmakov et al. (2023) developed latent variational diffusion models for inverse problems in high-energy physics, where intermediate representations are iteratively refined using domain information. Christopher et al. (2024) proposed a projected diffusion framework that combines diffusion sampling with constrained optimization through iterative projection.

Iterative refinement has also been explored in other domains. Liang et al. (2024) explored iterative refinement with diffusion models for video grounding, progressively aligning visual content with textual queries. Uehara et al. (2025) introduced reward-guided test-time refinement for biological sequence design without retraining. Despite these advances, most existing methods perform refinement through cascaded models or post-processing steps rather than an integral part of the generation process. The idea of interleaving optimization steps directly within the generative process remains explored, especially in the context of physical constraint satisfaction.

2.3 Physics-Constrained Learning

Physical knowledge can be introduced into data-driven models through different forms of constraints. Physics-Informed Neural Networks (PINNs) represent a foundational approach for integrating governing equations and initial/boundary conditions into machine learning models (Raissi et al., 2019). Yang & Perdikaris (2018) extended this concept to generative modeling, creating deep generative models that satisfy physical constraints while quantifying uncertainty. Bastek et al. (2024) integrated physical constraints into diffusion models for fluid flow simulation, while Shu et al. (2023) developed methods to reconstruct high-fidelity fields from low-quality data. Greydanus et al. (2019) introduced energy conservation into data-driven dynamics learning. Cranmer et al. (2020) embedded Lagrangian mechanics into neural networks for dynamics learning. They introduced the governing relations through structured models. Zhi et al. (2022) proposed an auto-differentiable model for collision avoidance which can be formulated as an inequality constraint and provided gradients for trajectory optimization in motion planning. These studies show that initial/boundary conditions, governing relations and inequality constraints can be introduced into learning systems.

3 Methodology

We propose Pareto-front Guided Diffusion Model (PGDiff), a novel diffusion framework that inserts a projected intermediate state into the each reverse denoising step, thereby forming a step-wise optimization architecture. Through these intermediate states, the denoising trajectory is progressively guided toward improved satisfaction of three categories of physical constraints we focus on while preserving sample fidelity. At each intermediate state, PGDiff leverages the Multiple Gradient Descent Algorithm (MGDA) to identify a Pareto direction, enabling trade-offs among competing physical objectives.

3.1 Problem Formulation

Let $x \in \mathbb{R}^d$ denote a decision variable. Conventional generation focuses on producing samples with high fidelity to the data manifold. And it aims to learn a parameterized distribution $p_\theta(x)$ that

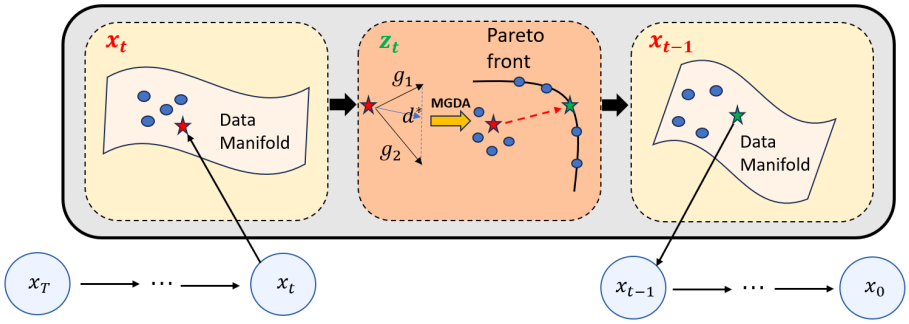


Figure 2: Reverse process of the Pareto-front Guided Diffusion Model (PGDiff). The model begins with pure noise state x_T and generates sample x_0 through iterative denoising steps. The key innovation of PGDiff is the introduction of intermediate projection states z_t, which embed constraints directly into the diffusion process through a specific optimization method. After each denoising step, the model employs Multiple Gradient Descent Algorithm (MGDA) to solve the multi-objective problem, ensuring that generated samples simultaneously satisfy multiple constraints while maintaining high sample quality.

approximates the underlying data distribution $q_{\text{data}}(x)$,

$$\min_{\theta} D\left[q_{\text{data}}(x) \,||\, p_\theta(x)\right] \tag{1}$$

where D(·,·) denotes the distribution divergence.

But in physics-constrained generation, generated samples is expected not only to remain data fidelity, but also to satisfy a set of physical constraints. Rewarding the three categories of constraints that we focus on, the initial and boundary constraints can be formulated separately as $\mathcal{R}_{\text{ic}}(z) = \mathcal{I}(z) - g_0$ on $\partial\Gamma_0$ and $\mathcal{R}_{\text{bc}}(z) = \mathcal{B}(z) - g$ on $\partial\Omega$. Here, $\mathcal{I}$ and $\mathcal{B}$ represent the initial operator and boundary operator. And $\partial\Gamma_0$ and $\partial\Omega$ means the initial position/time and domain boundary respectively. The governing constraints, $\mathcal{F}(z) = 0$ can be formulated as $\mathcal{R}_{\text{gov}} = \mathcal{F}(z)$. For inequality constraints $\mathcal{G}_j(z) \leq 0$, we can define $\mathcal{R}_{\text{ineq}} = \max\left[0, \mathcal{G}_j(z)\right]$. Then we can define a feasible set that satisfies these constraints,

$$\mathcal{C} = \left\{x : \mathcal{R}_{\text{ic/bc}}(x) = 0,\ \mathcal{R}_{\text{gov}}(x) = 0,\ \mathcal{G}_j(x) \leq 0\right\} \tag{2}$$

We can formulate the physics-constrained problem as

$$\min_{\theta} D\left[q_{\text{data}}(x) \,||\, p_\theta(x)\right] \quad \text{s.t.} \quad \text{supp}(p_\theta(x)) \subseteq \mathcal{C} \tag{3}$$

where $\text{supp}(p_\theta(x))$ is the support set of $p_\theta(x)$, which is the set of all values that $p_\theta(x)$ can take.

3.2 Pareto Reformulation and Step-wise Intermediate State

It is difficult to directly solve Equation 3 since the feasible set $\mathcal{C}$ is implicitly defined by multiple constraints and generally highly non-convex, we cannot directly describe or characterize it. Instead of directly handling it, we transform the physical constraints into a set of objectives. We define $\mathbf{F}(x) = [f_1(x), \ldots, f_m(x)]$, and $f_i = \|\mathcal{R}_i\|_2^2$ where i denotes the initial/boundary constraints, governing constraints and inequality constraints. This transformation recasts the original constrained problem as a multi-objective optimization problem without requiring an explicit construction of the complex geometry of $\mathcal{C}$. The resulting constrained solution space can be characterized by the Pareto set, which consists of Pareto optimal solutions. A solution is Pareto optimal if no objective can be further improved without deteriorating at least one other objective. And we can define Pareto set as

$$\mathcal{P}^* = \{x : \nexists\, y \ \text{s.t}\ \mathbf{F}(y) \prec \mathbf{F}(x)\} \tag{4}$$

where $\prec$ denotes dominant relationship i.e., $\mathbf{F}(y) \prec \mathbf{F}(x)$ if $f_i(y) \leq f_i(x)$ for all i and $f_j(y) < f_j(x)$ for at least one objective j. $\mathbf{F}(\mathcal{P}^*)$ forms the Pareto front in the objective space. We can reformulate the physics-constrained problem as

$$\min_{\theta} D[q_{\text{data}}(x) \,||\, p_\theta(x)] \quad \text{s.t.} \quad \text{supp}(p_\theta(x)) \subseteq \mathcal{P}^* \tag{5}$$

To solve Equation 5, instead of generating samples through a direct Markov chain from noise to data as most common diffusion models do, our Pareto-front Guided Diffusion (PGDiff) addresses it by reformulating the diffusion reverse process to incorporate multi-objective projection at each denoising step. Specifically, our model proposes the step-wise architecture that transforms the standard diffusion process

$$x_0 \leftarrow x_1 \leftarrow \cdots \leftarrow x_T \tag{6}$$

into an augmented process with projected intermediate states z_t:

$$x_0 \leftarrow \cdots \leftarrow x_{t-1} \leftarrow z_t \leftarrow x_t \leftarrow \cdots \leftarrow x_T \tag{7}$$

By interleaving the standard denoising steps with intermediate projection state, PGDiff distributes multi-objective optimization across the entire reverse denoising process. The sequence of intermediate states progressively guides the generation process toward Pareto front, while the diffusion process preserves the sample fidelity.

3.3 Pareto Projection For Diffusion Models

The key issue is how to construct and solve the intermediate state z_t such that the samples can tend to the Pareto set $\mathcal{P}^*$ while satisfying multiple physical constraints that we focus on. A common approach is scalarization, that is, to combine these objectives into a single objective. However, this method is difficult to adjust the preferences among multiple objectives, and it may miss solutions on non-convex parts of the Pareto front.

Inspired by Désidéri (2012), we adopt Multiple Gradient Descent Algorithm (MGDA) to overcome this challenge, where MGDA guides the initial point toward Pareto optimality by progressively solving for the update direction. Specifically, at the k-th projection step, we compute the gradient $g_i^k = \nabla_{z_t^k} f_i(z_t^k, t)$ of each physical objective with respect to the intermediate state. MGDA solves

$$d^k = \arg\max_{d \in \mathbb{R}^n} \left\{ \min_i (g_i^k)^\top d - \frac{1}{2}\|d\|^2 \right\} \tag{8}$$

Using Lagrange strong duality, the solution of Equation 8 can be framed into (Ye & Liu, 2022)

$$d^k = \sum_{i=1}^m \lambda_i g_i^k, \quad \text{where } \boldsymbol{\lambda} = \arg\min_{\boldsymbol{\lambda}} \left\| \sum_{i=1}^m \lambda_i g_i^k \right\|_2^2 \tag{9}$$

where $\sum_{i=1}^m \lambda_i = 1,\ \lambda_i \geq 0,\ \forall i$. The resulting direction d^k is the minimum-norm convex combination of the objective gradients, and repeatedly following the update direction progressively drives the current state toward a Pareto stationary solution.

The reformulation problem in Equation 5 requires the generated distribution to approximate the data distribution while driving generated samples toward the Pareto set defined by physical objectives. We address this problem through Langevin-based dynamics (Song & Ermon, 2019). Specifically, for **the distribution-matching objective** $\mathbf{D}[\mathbf{q}_{\text{data}}(\mathbf{x})||\mathbf{p}_\theta(\mathbf{x})]$, given the initial value $x_T \sim \mathcal{N}(0, I)$, the reverse process progressively evolves the sample by

$$x_{t-1} = x_t + \eta \nabla_x \log q_t(x_t) + \sqrt{2\eta}\,\xi, \quad t = T, T-1, \ldots, 1 \tag{10}$$

where $\xi \sim \mathcal{N}(0, I)$, $\nabla_x \log q_t(x_t)$ is the score function of the perturbed data distribution $q_t(x_t)$ at timestep t. When $\eta \to 0$ and $T \to \infty$, x_0 obtained from the procedure converges to a sample from $q(x)$ under some regularity conditions. And we can employ a neural network $\epsilon_\theta(x_t, t)$ to learn the corresponding denoising direction (Ho et al., 2020), thereby enabling the $p_\theta(x)$ to progressively approach the data distribution $q_{\text{data}}(x)$. For **the Pareto constraint** $\mathbf{supp}(\mathbf{p}_\theta(\mathbf{x})) \subseteq \mathcal{P}^*$, the multiple gradient direction obtained by MGDA provides a natural counterpart to the Langevin evolution. Following Liu et al. (2021), the MGDA direction can be incorporated in the Langevin dynamics by

$$z_t^{k+1} = z_t^k - \gamma d_t^k + \sqrt{2\gamma\tau}\,\xi, \quad k = 1, \ldots, K \tag{11}$$

where γ denotes the step size, τ controls the stochastic perturbation, and d_t^k is the multiple gradient direction in Equation 9. The drift term progressively drives the current state toward Pareto stationarity while the stochastic term preserves the exploratory behavior of Langevin dynamics.

Algorithm 1 Training of Pareto-front Guided Diffusion (PGDiff)

Require: Training dataset $\mathcal{D}$, diffusion model ϵ_θ, ProjectionNet z_ϕ, physical objectives $\{\mathcal{L}_{R_i}\}_{i=1}^m$, diffusion schedule $\{\beta_t\}_{t=1}^T$, Pareto updates K, step size γ

repeat
  Sample $x_0 \sim \mathcal{D}$, $t \sim \mathcal{U}\{1, \dots, T\}$, and $\epsilon \sim \mathcal{N}(0, I)$
  $x_t \leftarrow \sqrt{\bar{\alpha}_t} x_0 + \sqrt{1 - \bar{\alpha}_t}\epsilon$
  $z_t^{(0)} \leftarrow x_t$
  for $k = 1, \dots, K$ **do**
    $g_i^k \leftarrow \nabla_{z_t^k} f_i(z_t^k, t), \quad i = 1, \dots, m$
    $\boldsymbol{\lambda} \leftarrow \arg\min_{\boldsymbol{\lambda}} \left\| \sum_{i=1}^m \lambda_i g_i^k \right\|_2^2$
    $d_t^k \leftarrow \sum_{i=1}^m \lambda_i g_i^k$
    $z_t^{k+1} \leftarrow z_t^k - \gamma d_t^k$
  end for
  Update θ, ϕ using $\|\epsilon - \epsilon_\theta(x_t, t)\|_2^2 + \omega \left\| z_t^K - z_\phi(x_t, t) \right\|^2$
until converged

Algorithm 2 Sampling of Pareto-front Guided Diffusion (PGDiff)

Require: Trained diffusion model ϵ_θ, trained ProjectionNet z_ϕ, diffusion schedule $\{\beta_t\}_{t=1}^T$

Sample $x_T \sim \mathcal{N}(0, I)$
for $t = T, \dots, 1$ **do**
  $z_{\phi,t} \leftarrow z_\phi(x_t, t)$
  Sample $\xi \sim \mathcal{N}(0, I)$ if $t > 1$, otherwise $\xi \leftarrow 0$
  $x_{t-1} \leftarrow \frac{1}{\sqrt{\alpha_t}} \left(z_{\phi,t} - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}} \epsilon_\theta(z_{\phi,t}, t) \right) + \sigma_t \xi$
end for
return x_0

With the two Langevin dynamics above, we can couple the diffusion process with the multi-objective optimization through the process introduced in Equation 7, which can be explicitly connected to the reformulated Problem 3. Starting from the current diffusion state x_t, we first project it to z_t^0 and perform K MGDA-based Langevin updates to obtain z_t^K, which can be named as $x_t \to z_t$. This evolution corresponds to the Pareto constraint $\mathrm{supp}(p_\theta(x)) \subseteq \mathcal{P}^*$, since the multiple gradient drift progressively drives the state toward Pareto optimality under the physical objectives. The resulting z_t^K is introduced to the reverse diffusion transition with $x_{t-1} \sim p_\theta(x_{t-1} \mid z_t^K, t)$, which can be named as $z_t \to x_{t-1}$. And it continues the distribution-matching process associated with $\min_\theta D[q_{\mathrm{data}}(x) || p_\theta(x)]$. Repeating this process throughout the reverse chain progressively incorporates the multiple constraints while preserving the diffusion generation process, allowing the generated samples to maintain high fidelity while satisfying multiple physical requirements.

3.4 Training and Sampling

The K-step multi-objective dynamics provides the desired intermediate state z_t^K between two consecutive diffusion states. Rather than explicitly performing these iterative updates during generation, we introduce a ProjectionNet parameterized by ϕ to learn the resulting state z_t^K. Given the current diffusion state x_t, timestep t, ProjectionNet predicts $z_\phi(x_t, t)$ such that z_ϕ approximates the state obtained after the K-step MGDA process.

After establishing the step-wise diffusion architecture and having a learnable neural network, we can propose the complete training objective and process. During training, the overall training loss combines a standard diffusion objective with our learned projection state z_ϕ, and the denoising network and ProjectionNet are jointly optimized:

$$\mathcal{L} = \mathbb{E}_{x_0, t, \epsilon} \left[\|\epsilon - \epsilon_\theta(x_t, t)\|^2 + \omega \|z_t^K - z_\phi(x_t, t)\|^2 \right] \tag{12}$$

where the first term is the standard diffusion objective and the second term enforces the model to learn the intermediate state that remain closely Pareto optimality. The entire process of training algorithm is described in Algorithm 1.

During sampling, given the current state x_t, we can use the ProjectionNet which represents the explicit K-step MGDA optimization to predict the intermediate state $z_\phi(x_t, t)$. The denoising network evaluates the reverse diffusion transition conditioned on $z_\phi(x_t, t)$ yielding $x_{t-1} \sim p_\theta(x_{t-1} \mid z_\phi(x_t, t), t)$ where the transition is parameterized by $\epsilon_\theta(z_\phi(x_t, t), t)$. The entire process of sampling algorithm is descirbed in Algorithm 2. Consequently, the complete learned reverse process follows

$$x_T \to \dots x_t \to z_\phi, t \to x_{t-1} \cdots \to x_0 \tag{13}$$

Our integrated approach leverages the complementary strengths of explicit step-wise projection mechanism and diffusion process. The loss function encourages the model to learn the relationship between standard diffusion states and their multi-objective optimal counterparts. This creates a generation process that maintains both sample quality and plausibility, enabling the production of valid samples even when multiple competing objectives must be satisfied simultaneously.

4 EXPERIMENT

To demonstrate the universality and capability of our technique in addressing complex problems, we conduct experiments on two multi-constraint tasks. We apply PGDiff in a complicated path planning scenario inside a non-convex environment. And we second evaluate its effectiveness in the challenging task of solving partial differential equations, comparing it with several baseline models in multi-constraint contexts.

4.1 PATH PLANNING

To assess PGDiff under competing constraints, we apply it to path planning in significantly non-convex environments. This task evaluates the model's ability to balance collision avoidance and path length. A simulated environment has been developed, featuring intricate static non-convex obstacle arrangements. These configurations lead to the creation of multiple interconnected open areas and narrow corridors. The environment consists of a series of test mazes, each characterized by distinct starting and ending points. This task involves the diffusion model predicting a sequence of points (p0, p1, …, pN), where each consecutive pair of points denotes a line segment. The trajectory's initial and final points are pseudo-randomly assigned for each problem instance, whereas the topography remains consistent across various instances. The objectives to be achieved include: (1) Collision avoidance, no collision with obstacles, which represents the third category of constraints we mentioned earlier (2) Selecting the shortest possible path to achieve the target, which matches the second category of the constraints.

Additionally, obstacles, represented by the gray elements in Figure 3, are introduced during the inference process. These obstacles are absent during training; consequently, certain training data is out of the distribution. This subsequently evaluates the generalization capability of these methods. The performance evaluation relies on two sets of maps adapted from Carvalho et al. (2023).

Conventional approaches entail sampling numerous batches of trajectories and subsequently identifying a viable solution, if available. In this experiment, the MPD motion planning diffusion model served as the baseline conditional model. The MPD model utilizes the method established by Carvalho et al. (2023). Furthermore, we incorporated the Projected Diffusion Model (PDM), a representative constrained diffusion approach that enforces feasibility through projection during the generative process, (Christopher et al., 2024) into the comparison. And we also select the Denoising Diffusion Probabilistic Models (DDPM) as the baseline model (Ho et al., 2020). Path quality is evaluated using two metrics: Path Length, and Collision Violation (C_V). The Collision Violation quantifies the severity of any encroachment into the obstacles:

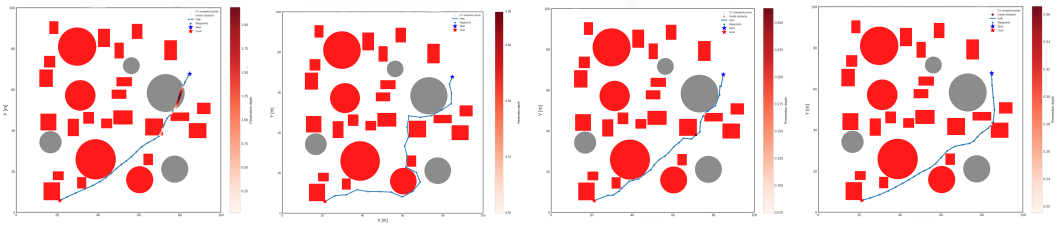
$$\mathcal{C}_V(P) = \sum_{i=0}^{N} \max(0, -\min_{o \in O} s_o(p_i))^2, \qquad s_o(p) = \begin{cases} d(p, \partial o), & p \notin o, \\ -d(p, \partial o), & p \in o \end{cases}$$

where O represents the set of all obstacles, ∂o represents the boundary of obstacles , $d(p_i, \partial o)$ denotes the distance from a path point p_i to an obstacle boundary ∂o. A score of zero indicates a perfectly collision-free path.

Figure 3 illustrates that PGDiff is capable of planning feasible paths with sampling, while also achieving notable enhancements in path length. This finding indicates that the proposed architecture

Table 1: Comparison of Path Planning Models

Metric	DDPM	PDM	PGDiff	MPD
Path Length	**92.67**	126.03	<u>96.90</u>	100.25
C_V	66.22	**0.00**	<u>0.09</u>	0.18



(a) DDPM (b) PDM (c) PGDiff (d) MPD

Figure 3: Comparison of path planning results. The red color on the path represents collision with obstacles, and the deeper the color, the more intense the collision. (a) DDPM produces the shortest trajectory but exhibits severe obstacle penetration due to the lack of explicit physical constraints. (b) PDM eliminates collision through explicit projection, but results in a substantially longer trajectory. (c) PGDiff generates a smooth and short trajectory while achieving the smallest constraint violation. (d) MPD also reduces obstacle collisions, but produces a longer trajectory with sightly higher violation than PGDiff.

demonstrates strong performance in non-convex scenarios, underscoring the wide applicability and effectiveness of PGDiff in complex contexts.

4.2 Solving Partial Differential Equations

Experimental Settings. In this experiment, we evaluate our method on three different PDE systems: the Burgers equation, the Poisson equation, and the Helmholtz equation. These systems are governed by multiple physical requirements, including the governing PDEs, boundary conditions, and initial conditions. We express these requirements as residual objectives, following the formulation introduced in the previous section. Therefore, these tasks provide a natural setting for evaluating our multi-objective constrained generation framework. For a fair comparison, we use the publicly available datasets and pretrained diffusion backbones provided by Huang et al. (2024). We follow the experimental setting of Yuan et al. (2025), where the goal is to generate the entire physical fields.

We compare PGDiff with three constrained generation methods. (a) EDM is the base diffusion model provided by Huang et al. (2024) for generating PDE fields. (b) DiffusionPDE guides a pretrained diffusion model during the reverse process using gradients of the PDE residual, without retraining the backbone. (Huang et al., 2024) (c) CoCoGen (Jacobsen et al., 2025) is a score-based diffusion framework that improves physical consistency at inference time by iterative minimizing PDE residuals without retraining the model. To match our multi-objective setting, we extend these models to include the multiple physical objectives. The scale of each objective is determined from its statics on a separate calibration set, which avoids one objective dominating the others only because of its numerical magnitude.

For each equation, we report three evaluation metrics. PDE Residual is the mean squared residual of the governing equation. BC Residual measures the violation of the corresponding boundary condition. Prior Loss measures the deviation between the generated fields and data prior learned by the diffusion model. Lower values indicate better performance for all metrics. We evaluate each method using 1000 generated samples.

Main Result. The results, presented in Table 2, demonstrate that PGDiff almost outperforms all baselines across the multi-objectives constrained generation for solving PDEs. PGDiff achieves the best overall performance on the Burgers equation and Poisson equation. It obtains the lowest values on all metrics. On the Helmholtz equation, PGDiff also provides the best PDE residual. This

Table 2: Results on PDE-constrained generation.

Equation	Method	PDE Residual	BC Residual
Burgers	EDM	1.77E-01	2.96E-04
	DiffusionPDE	2.67E-02	1.64E-04
	CoCoGen	2.43E-02	2.74E-04
	PGDiff	**5.32E-03**	**3.31E-05**
Poisson	EDM	4.19	4.71E-09
	DiffusionPDE	3.88	4.48E-09
	CoCoGen	3.82	4.40E-09
	PGDiff	**2.66**	**1.44E-09**
Helmholtz	EDM	2.03	4.83E-09
	DiffusionPDE	1.74	3.03E-09
	CoCoGen	1.18	**1.41E-09**
	PGDiff	**0.76**	1.96E-09

Table 3: Multi-objective analysis of PGDiff across multiple equations.

Equation	Conflict Rate (%)	Accepted Rate (%)	PI Rate (%)
Burgers	38.5	90.1	93.0
Poisson	18.2	99.0	93.0
Helmholtz	35.5	96.3	94.5

performance comes from the proposed step-wise optimization architecture. With this architecture, PGDiff improves multiple competing objectives without strongly sacrificing generation quality.

Multi-objective analysis of PGDiff. We further analyze the internal behavior of PGDiff in Table 3. We evaluate three different indicators. Conflict Rate is the fraction of optimization step in which at least one pair of objective gradients has a negative inner product. A conflict means that directly reducing one objective may increase another. Accepted Rate is the fraction of MGDA updates that pass the objective descent and feasibility checks. A high value means that the Pareto direction usually gives a valid and stable optimization under the current step. PI (Pareto improvement) Rate is computed from paired EDM and PGDiff samples. A sample is counted as improved when all active objectives are no worse than those of the EDM samples, and at least one objective is strictly better.

Table 3 shows that gradient conflicts occur in all three equations, confirming that the physical objectives are not always aligned. The conflicts are especially clear on Burgers and Helmholtz equations, where around one third of the optimization step contain competing gradients. Despite these conflicts, PGDiff accepts more than 90% of the MGDA updates for all equations, indicating that MGDA can usually find a stable direction that satisfies the direction and feasibility conditions. Following this direction enables the generated results to balance the physical objectives and move toward the Pareto front. Pareto Improvement Rate is also recorded in Table 3, meaning that PGDiff simultaneously improves all active objectives for more than 90% of the samples. Therefore, the improvements in the Table 2 are consistent across most samples rather than being driven by only a few successful cases.

5 Conclusions

In this work, we propose Pareto-front Guided Diffusion (PGDiff) to solve the challenge of generating physically robust samples that simultaneously satisfy three categories that we focus on. The PGDiff features a step-wise optimization architecture and adopts MGDA during the reverse diffusion process. Empirical studies demonstrate that the PGDiff outperforms state-of-the-art models in physical consistency and sample quality. We hope that future work can build on this foundation to further study how to integrate more types of physical constraints more harmoniously into the generation process, resulting in outcomes that better conform to physical laws.

REFERENCES

Fatin H Ajeil, Ibraheem Kasim Ibraheem, Mouayad A Sahib, and Amjad J Humaidi. Multi-objective path planning of an autonomous mobile robot using hybrid pso-mfb optimization algorithm. *Applied Soft Computing*, 89:106076, 2020.

Namrata Anand and Tudor Achim. Protein structure and sequence generation with equivariant denoising diffusion probabilistic models. *arXiv preprint arXiv:2205.15019*, 2022.

Jan-Hendrik Bastek, WaiChing Sun, and Dennis M Kochmann. Physics-informed diffusion models. *arXiv preprint arXiv:2403.14404*, 2024.

Dimo Brockhoff and Eckart Zitzler. On objective conflicts and objective reduction in multiple criteria optimization. *TIK Report*, 243, 2006.

Joao Carvalho, An T Le, Mark Baierl, Dorothea Koert, and Jan Peters. Motion planning diffusion: Learning and planning of robot motions with diffusion models. In *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 1916–1923. IEEE, 2023.

Jacob K Christopher, Stephen Baek, and Nando Fioretto. Constrained synthesis with projected diffusion models. *Advances in Neural Information Processing Systems*, 37:89307–89333, 2024.

Miles Cranmer, Sam Greydanus, Stephan Hoyer, Peter Battaglia, David Spergel, and Shirley Ho. Lagrangian neural networks. *arXiv preprint arXiv:2003.04630*, 2020.

Kalyanmoy Deb, Amrit Pratap, Sameer Agarwal, and TAMT Meyarivan. A fast and elitist multiobjective genetic algorithm: Nsga-ii. *IEEE transactions on evolutionary computation*, 6(2): 182–197, 2002.

Jean-Antoine Désidéri. Multiple-gradient descent algorithm (mgda) for multiobjective optimization. *Comptes Rendus. Mathématique*, 350(5-6):313–318, 2012.

Jenna C Fromer and Connor W Coley. Computer-aided multi-objective optimization in small molecule discovery. *Patterns*, 4(2), 2023.

Samuel Greydanus, Misko Dzamba, and Jason Yosinski. Hamiltonian neural networks. *Advances in neural information processing systems*, 32, 2019.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

Jiahe Huang, Guandao Yang, Zichen Wang, and Jeong Joon Park. Diffusionpde: Generative pde-solving under partial observation. *Advances in Neural Information Processing Systems*, 37: 130291–130323, 2024.

Lei Huang, Hengtong Zhang, Tingyang Xu, and Ka-Chun Wong. Mdm: Molecular diffusion model for 3d molecule generation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pp. 5105–5112, 2023.

Christian Jacobsen, Yilin Zhuang, and Karthik Duraisamy. Cocogen: Physically consistent and conditioned score-based generative models for forward and inverse problems. *SIAM Journal on Scientific Computing*, 47(2):C399–C425, 2025.

Moksh Jain, Sharath Chandra Raparthy, Alex Hernández-Garcıa, Jarrid Rector-Brooks, Yoshua Bengio, Santiago Miret, and Emmanuel Bengio. Multi-objective gflownets. In *International conference on machine learning*, pp. 14631–14653. PMLR, 2023.

Zhifeng Kong, Wei Ping, Jiaji Huang, Kexin Zhao, and Bryan Catanzaro. Diffwave: A versatile diffusion model for audio synthesis. *arXiv preprint arXiv:2009.09761*, 2020.

Xiao Liang, Tao Shi, Yaoyuan Liang, Te Tao, and Shao-Luo Huang. Exploring iterative refinement with diffusion models for video grounding. In *2024 IEEE International Conference on Multimedia and Expo (ICME)*, pp. 1–6. IEEE, 2024.

Xi Lin, Zhiyuan Yang, and Qingfu Zhang. Pareto set learning for neural multi-objective combinatorial optimization. *arXiv preprint arXiv:2203.15386*, 2022.

Xingchao Liu, Xin Tong, and Qiang Liu. Profiling pareto front with multi-objective stein variational gradient descent. *Advances in neural information processing systems*, 34:14721–14733, 2021.

Vadim Popov, Ivan Vovk, Vladimir Gogoryan, Tasnima Sadekova, and Mikhail Kudinov. Grad-tts: A diffusion probabilistic model for text-to-speech. In *International conference on machine learning*, pp. 8599–8608. PMLR, 2021.

Maziar Raissi, Paris Perdikaris, and George E Karniadakis. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational physics*, 378:686–707, 2019.

Alexandre Rame, Guillaume Couairon, Corentin Dancette, Jean-Baptiste Gaya, Mustafa Shukor, Laure Soulier, and Matthieu Cord. Rewarded soups: towards pareto-optimal alignment by interpolating weights fine-tuned on diverse rewards. *Advances in Neural Information Processing Systems*, 36:71095–71134, 2023.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 10684–10695, 2022.

Chitwan Saharia, William Chan, Huiwen Chang, Chris Lee, Jonathan Ho, Tim Salimans, David Fleet, and Mohammad Norouzi. Palette: Image-to-image diffusion models. In *ACM SIGGRAPH 2022 conference proceedings*, pp. 1–10, 2022a.

Chitwan Saharia, Jonathan Ho, William Chan, Tim Salimans, David J Fleet, and Mohammad Norouzi. Image super-resolution via iterative refinement. *IEEE transactions on pattern analysis and machine intelligence*, 45(4):4713–4726, 2022b.

Ozan Sener and Vladlen Koltun. Multi-task learning as multi-objective optimization. *Advances in neural information processing systems*, 31, 2018.

Alexander Shmakov, Kevin Greif, Michael Fenton, Aishik Ghosh, Pierre Baldi, and Daniel Whiteson. End-to-end latent variational diffusion models for inverse problems in high energy physics. *Advances in neural information processing systems*, 36:65102–65127, 2023.

Dule Shu, Zijie Li, and Amir Barati Farimani. A physics-informed diffusion model for high-fidelity flow field reconstruction. *Journal of Computational Physics*, 478:111972, 2023.

Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. *Advances in neural information processing systems*, 32, 2019.

Masatoshi Uehara, Xingyu Su, Yulai Zhao, Xiner Li, Aviv Regev, Shuiwang Ji, Sergey Levine, and Tommaso Biancalani. Reward-guided iterative refinement in diffusion models at test-time with applications to protein and dna design. *arXiv preprint arXiv:2502.14944*, 2025.

Tsun-Hsuan Johnson Wang, Juntian Zheng, Pingchuan Ma, Yilun Du, Byungchul Kim, Andrew Spielberg, Josh Tenenbaum, Chuang Gan, and Daniela Rus. Diffusebot: Breeding soft robots with physics-augmented generative diffusion models. *Advances in Neural Information Processing Systems*, 36:44398–44423, 2023.

Yibo Yang and Paris Perdikaris. Physics-informed deep generative models. *arXiv preprint arXiv:1812.03511*, 2018.

Yinghua Yao, Yuangang Pan, Jing Li, Ivor Tsang, and Xin Yao. Proud: Pareto-guided diffusion model for multi-objective generation. *Machine Learning*, 113(9):6511–6538, 2024.

Mao Ye and Qiang Liu. Pareto navigation gradient descent: a first-order algorithm for optimization in pareto set. In *Uncertainty in artificial intelligence*, pp. 2246–2255. PMLR, 2022.

Mingze Yuan, Pengfei Jin, Na Li, and Quanzheng Li. Pirf: Physics-informed reward fine-tuning for diffusion models. *arXiv preprint arXiv:2509.20570*, 2025.

Ye Yuan, Jiaming Song, Umar Iqbal, Arash Vahdat, and Jan Kautz. Physdiff: Physics-guided human motion diffusion model. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 16010–16021, 2023.

Yuheng Zhi, Nikhil Das, and Michael Yip. Diffco: Autodifferentiable proxy collision detection with multiclass labels for safety-aware trajectory optimization. *IEEE Transactions on Robotics*, 38(5): 2668–2685, 2022.

A Additional Samples distribution in Solving Partial Differential Equations

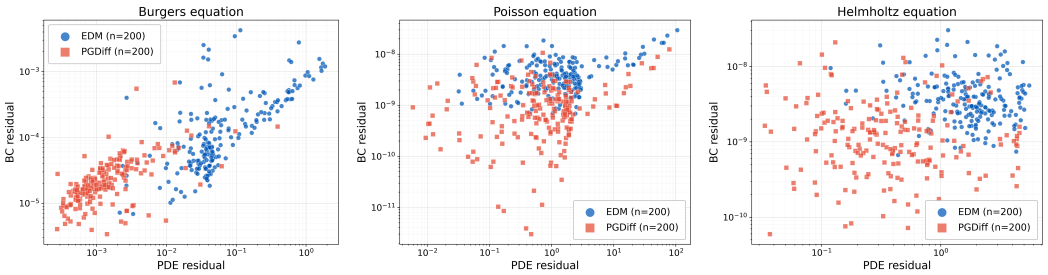


Figure 4: The distribution plots corresponding to EDM and PGDiff after each has sampled 200 points for three equations.

Figure 4 illustrates the joint distributions of PDE residual and BC residual for 200 samples generated by EDM and PGDiff on three representative equations. The lower-left region corresponds to samples with smaller residuals for both objectives and therefore indicates better simultaneous satisfaction of the governing equation and boundary conditions. Compared with EDM, the distribution of PGDiff samples consistently shifts toward this desirable region across the Burgers, Helmholtz, and Poisson equations. Rather than reducing a single residual while sacrificing the other, PGDiff tends to improve the two physical objectives jointly, resulting in fewer samples with large residuals and a higher concentration of samples in regions with better overall physical consistency.

B AI Use Statement

In this work, we used generative AI tools only for polishing language and searching related works. We have not used generative AI tools for research idea design, model architecture design, experimental implementation, result analysis, or conclusion derivation. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.