Commun. Comput. Phys.
doi: 10.4208/cicp.OA-2024-0308

Vol. **40**, No. 1, pp. 153-175
July 2026

A Diffusion Model Based Iterative Convolution Thresholding Method for Structural Topological Optimization

Xiaonan Peng[1], Luyu Cen[2] and Xiaoping Wang[2,*]

[1] *Department of Mathematics, Hong Kong University of Science and Technology, Clear Water Bay, Kowloon, Hong Kong, China.*
[2] *School of Science and Engineering, The Chinese University of Hong Kong, Shenzhen, Guangdong 518172, China &*
Shenzhen International Center for Industrial and Applied Mathematics, Shenzhen Research Institute of Big Data, Guangdong 518172, China.

Received 16 December 2024; Accepted (in revised version) 30 March 2025

Abstract. In this study, we introduce the Diffusion Model with Iterative Convolution Thresholding Method (DICTM), a novel hybrid approach designed to address the minimum compliance problem in topology optimization. DICTM synergistically combines the robustness of diffusion models with the precision of threshold dynamics to tackle the complexities inherent in linear elasticity problems, while substantially enhancing computational efficiency. Our approach facilitates the generation of initial configurations via the diffusion model, which dramatically improves the efficiency of the subsequent threshold dynamics process, reducing the iteration count to about one-tenth of that required by traditional methods. This significant reduction in computational effort also enables more effective hyperparameter tuning without added cost. The integration of deep-generative models with a rigorous threshold dynamics framework positions DICTM as a powerful tool in topology optimization, producing designs not only with low compliance, but also in a computationally efficient way.

AMS subject classifications: 74B05, 90C26

Key words: Topology optimization, diffusion models, iterative convolution threshold dynamics, linear elasticity, deep generative models.

1 Introduction

Topology optimization is a powerful method to determine the material distribution within a given design domain to achieve the best possible structural performance under

*Corresponding author. *Email addresses:* wangxiaoping@cuhk.edu.cn (X. Wang), xpengak@connect.ust.hk (X. Peng), lcen@connect.ust.hk (L. Cen)

specified constraints. Traditional approaches, such as the Solid Isotropic Material with Penalization (SIMP) method [4], level set methods [1, 22], phase field methods [13, 23], threshold dynamics methods [6, 7], have been widely used to solve topology optimization problems. However, these methods often require careful tuning of parameters and can be computationally intensive, especially for complex structures.

With the advent of data-driven approaches, deep learning techniques have been integrated into topology optimization to accelerate the design process. Generative models have shown promise in achieving near-optimal compliance and shape similarity while substantially reducing computational demands [24]. Variational Autoencoders (VAE) [10, 25] and Generative Adversarial Networks (GAN) [17, 18] have been used to generate designs conditioned on boundary conditions and loads. Despite these advances, there remain challenges to establish reliable and generalizable mappings from boundary conditions to optimal structures. GANs, in particular, face issues with training stability and generalization due to their adversarial nature.

Diffusion models have recently emerged as a stable and effective alternative for generative modeling. First introduced by Sohl-Dickstein et al. [19] and later improved by Ho et al. [11] and Song et al. [21], diffusion models have outperformed GANs in generating high-quality images [9] and have found applications in computer vision [2, 14], natural language processing [3, 12], and medical imaging [5, 8].

However, deep learning methods often struggle to strictly adhere to the mathematical requirements inherent in optimization problems. Generated structures may lack necessary connectivity and smoothness, leading to suboptimal or infeasible solutions. Traditional numerical methods, while mathematically rigorous, can be computationally expensive and sensitive to initial guesses and parameter settings.

In [6], an iterative convolution thresholding method (ICTM) was introduced for topology optimization in linear elasticity. We propose a simple energy decaying algorithm to solve the two-phase minimum compliance problem. The material domain is implicitly represented by its characteristic function, and the problem is formulated as a minimization-minimization problem by the principle of minimum complementary energy. We show that the energy is decreasing in each iteration and a local minimum solution is quickly approached. Two effective continuation schemes are proposed to avoid being trapped in a bad local minimum.

In this study, we propose a novel approach that integrates diffusion models with the iterative convolution thresholding method (ICTM) for topology optimization in linear elasticity. Using the diffusion model to provide a high-quality initial guess, we significantly accelerate the convergence of the threshold dynamics method. This hybrid approach, termed the DICTM (Diffusion model with Iterative Convolution Thresholding Method), leverages the predictive capabilities of deep-generative models and the mathematical rigor of numerical methods.

Our contributions can be summarized as follows.

- We develop a hybrid topology optimization method that combines diffusion mod-

els with threshold dynamics, addressing the limitations of both deep learning and traditional numerical methods.

- We construct a comprehensive dataset with diverse boundary conditions and loading scenarios, enhancing the generalizability of the diffusion model.
- We demonstrate that DICTM achieves high-quality designs with reduced computational effort, significantly decreasing the number of iterations required compared to traditional methods.
- We validate the effectiveness of DICTM through numerical experiments, showing improved compliance and efficiency over existing methods.

The remainder of this paper is organized as follows. Section 2 provides background information on topology optimization, threshold dynamics, and diffusion models. Section 3 introduces our proposed DICTM method in detail. Section 4 describes the dataset used in our experiments. Section 5 presents numerical results and comparisons with traditional methods and deep learning approaches. Finally, Section 6 concludes the paper and discusses future research directions.

2 Background

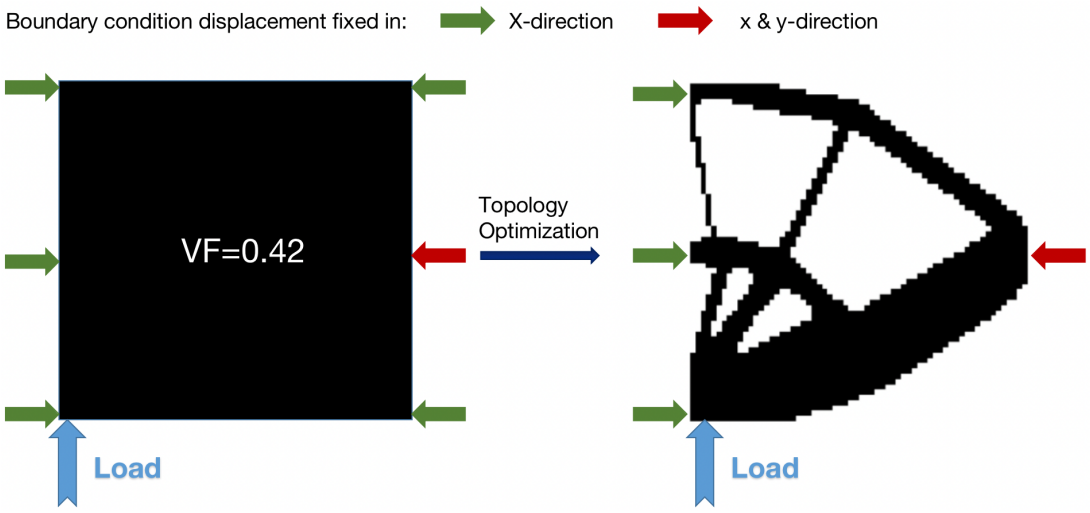

Figure 1: The left one is the domain and constraints, where VF is volume fraction of the material to the whole domain. The right one is the optimal structure generated by SIMP.

2.1 Topology optimization

Topology optimization is a widely used design methodology to find the optimal material distribution within a given design domain to achieve the best structural performance under specified constraints. The general structural topology optimization problem can be formulated as follows.

$$\min_{\rho} \mathcal{O}(\rho) \text{ subject to } \mathbf{g}(\rho) \leq \mathbf{0},\ \mathbf{h}(\rho) = \mathbf{0}, \tag{2.1}$$

where $\mathcal{O}(\rho)$ is the objective function that represents the performance measure to be optimized, ρ is the density function over the design domain, $\mathbf{g}(\rho)$ denotes the inequality constraints, and $\mathbf{h}(\rho)$ represents the equality constraints.

The minimum compliance problem is a fundamental problem in topology optimization, aiming to minimize the structural compliance (i.e., the inverse of stiffness)

$$l(\mathbf{u}) = \int_{\Gamma_T} \mathbf{u} \cdot \mathbf{t} ds, \tag{2.2}$$

where $\mathbf{u}$ is the displacement field and $\mathbf{t}$ is the traction force acting on Γ_T. In engineering applications, minimizing compliance often leads to an optimized structure with a better stiffness-to-weight ratio, resulting in improved performance and reduced material usage.

The compliance in (2.2) is implicitly determined by ρ, since $\mathbf{u}$ depends on the following linear elasticity equation

$$\begin{cases} -\nabla \cdot (\mathbf{E}(\rho)\varepsilon(\mathbf{u})) = \mathbf{0} & \text{in } \Omega, \\ \mathbf{u} = \mathbf{0} & \text{on } \Gamma_D, \\ (\mathbf{E}(\rho)\varepsilon(\mathbf{u})) \cdot \mathbf{n} = \mathbf{t} & \text{on } \Gamma_T, \\ (\mathbf{E}(\rho)\varepsilon(\mathbf{u})) \cdot \mathbf{n} = \mathbf{0} & \text{on } \partial\Omega \backslash (\Gamma_D \cup \Gamma_T), \end{cases} \tag{2.3}$$

where the 4-th order stiffness tensor field $\mathbf{E}$ depends on ρ, $\Gamma_D \subset \partial\Omega$ is the region of prescribed displacements, $\Gamma_T \subset \partial\Omega$ is the region of prescribed traction forces, $\mathbf{n}$ is the outward unit normal vector on the boundary, and $\varepsilon(\mathbf{u}) = \frac{1}{2}(\nabla \mathbf{u} + \nabla^{\mathrm{T}} \mathbf{u})$ is the strain tensor. The minimum compliance problem is

$$\min_{\rho} l(\mathbf{u}), \text{ subject to (2.3) and } \int_{\Omega} \rho \leq V. \tag{2.4}$$

In the two-phase optimization, any point in the design domain Ω either belongs to the material domain Ω_m or belongs to the void region $\Omega \backslash \Omega_m$. We assume that the void region has a very small stiffness tensor $E^{\min}\mathbf{E}^0$ where $E^{\min}$ is a positive number close to 0. Then we define

$$\mathbf{E}[x] = \begin{cases} \mathbf{E}^0, & x \in \Omega_m, \\ E^{\min}\mathbf{E}^0, & x \in \Omega \backslash \Omega_m, \end{cases} \tag{2.5}$$

where $\mathbf{E}^0$ is the stiffness tensor of the material, which is positive definite. Therefore, if we restrict ρ to be in $L^\infty(\Omega:\{0,1\})$ and represent Ω_m according to

$$\rho[x]=\begin{cases}1, & x\in\Omega_m,\\ 0, & x\in\Omega\backslash\Omega_m,\end{cases}$$

then

$$\mathbf{E}(\rho)=\Big(E^{\min}+(1-E^{\min})\rho\Big)\mathbf{E}^0.$$

2.2 The iterative convolution thresholding method

The iterative convolution thresholding method (ICTM) is a new method to solve minimum compliance problem proposed by Cen et al. [6]. This method is an efficient energy decaying algorithm to solve the two-phase minimum compliance problem. The material domain is implicitly represented by smoothed characteristic function, and the problem is formulated into a minimization-minimization problem by the principle of minimum complementary energy. We limit the solution space to the space only containing the Gaussian convolution with a characteristic function χ satisfying the volume constraint

$$K_\tau=\Big\{\rho:\rho=G_\tau*\chi,\ \int_\Omega\chi\le V,\ \chi\in L^\infty(\Omega:\{0,1\})\Big\},$$

where

$$G_\tau=\frac{1}{4\pi\tau}e^{-\frac{||\mathbf{x}||^2}{4\tau}}. \tag{2.6}$$

It has been shown as in [6] that

$$\min_{\rho\in K_\tau} l(\mathbf{u}) \text{ subject to (2.3)} \Leftrightarrow \min_{\rho\in K_\tau}\min_{\sigma\in S}\mathcal{J}(\sigma,\rho), \tag{2.7}$$

where

$$\mathcal{J}(\sigma,\rho):=\int_\Omega \mathbf{E}^{-1}(\rho)\sigma:\sigma,$$
$$S:=\{\sigma\in L^2(\Omega:\mathbb{R}^{2\times2}):\sigma_{ij}=\sigma_{ji},\ \nabla\cdot\sigma=0,\ \sigma\cdot\mathbf{n}=\mathbf{t}\text{ on }\Gamma_T,\ \sigma\cdot\mathbf{n}=0\text{ on }\partial\Omega\backslash(\Gamma_D\cup\Gamma_T)\},$$

and we define the following $\mathbf{E}(\rho)$ to make the energy $\mathcal{J}(\sigma,\rho)$ linear respect to ρ so that updates would be easy to obtain,

$$\mathbf{E}(\rho)=\Big(\frac{1}{E^{\min}}+\Big(1-\frac{1}{E^{\min}}\Big)\rho\Big)^{-1}\mathbf{E}^0.$$

The above min-min formulation allows the coordinate descent scheme which updates alternatively in the order of

$$\rho^0,\sigma^0,\rho^1,\cdots,\rho^k,\sigma^k,\rho^{k+1},\cdots.$$

At the k-th step, $k\geq 0$, given $\rho^k = G_\tau * \chi^k$, new iterates are computed according to

$$\sigma^k = \underset{\sigma\in S}{\operatorname{argmin}} \mathcal{J}(\sigma,\rho^k), \tag{2.8}$$

$$\rho^{k+1} = G_\tau * \chi^{k+1} \in \underset{\rho\in K_\tau}{\operatorname{argmin}} \mathcal{J}(\sigma^k,\rho). \tag{2.9}$$

The detailed algorithm is summarized in Algorithm 1. The energy $\mathcal{J}(\sigma,\rho)$ is proved to decay after each step.

Theorem 2.1. *The total energy $\mathcal{J}(\chi,\sigma)$ is nonincreasing in each iteration of σ and χ in Algorithm 1 for all $\tau>0$. That is,*

$$\mathcal{J}(\rho^{k+1},\sigma^{k+1}) \leq \mathcal{J}(\rho^k,\sigma^k). \tag{2.10}$$

Proof. It is a special case of Theorem 3.2 in [6], where the coefficient γ of the perimeter term is 0, since here we no longer consider the influence of the perimeter term. □

Algorithm 1: An iterative convolution thresholding method for the topology optimization problem (2.7).

Input: Set $k=0$, parameters $\tau>0$, material stiffness tensor $\mathbf{E}^0$, $E^{\min}\in(0,1)$, prescribed volume $V<|\Omega|$, initial guess χ^0.

Output: An approximate solution $\rho\in K_\tau$ to (2.7)

k=0

repeat

// update $\sigma^k = \underset{\sigma\in S}{\operatorname{argmin}} \mathcal{J}(\rho^k,\sigma)$ by the following.

$\mathbf{E}(\rho^k) \leftarrow \frac{1}{[1/E^{\min}+\rho^k(1-1/E^{\min})]}\mathbf{E}^0$;

Solve $\mathbf{u}^k$ from (2.3) with $\mathbf{E}(\rho^k)$;

$\sigma^k \leftarrow \mathbf{E}(\rho^k)\varepsilon(\mathbf{u}^k)$;

// update $\rho^{k+1}\in \underset{\rho\in K_\tau}{\operatorname{argmin}} \mathcal{J}(\rho,\sigma^k)$ by the following.

$\phi^k \leftarrow (1-1/E^{\min})G_\tau * (\mathbf{E}(\rho^k)^{-1}\sigma^k:\sigma^k)$;

$\delta \leftarrow \sup\{a : |\{x\in\Omega : \phi^k(x)<a\}|\leq V\}$;

Find A_δ satisfying $A_\delta\subset\{x\in\Omega:\phi^k(x)\leq\delta\}$ and $|A_\delta|=V$;

if $x\in A_\delta$ **then**

$\chi^{k+1}[x]\leftarrow 1$;

else

$\chi^{k+1}[x]\leftarrow 0$;

end

$\rho^{k+1}\leftarrow G_\tau * \chi^{k+1}$;

$k\leftarrow k+1$;

until *convergence*;

In the finite element discretization, we assume that the material distribution is constant per element. Let N be the number of elements, $\boldsymbol{\chi}=[\chi_{e_i}]_{i=1}^N\in\{0,1\}^N$ be the material indicating vector, $\boldsymbol{\rho}=\boldsymbol{G}\boldsymbol{\chi}=[\rho_{e_i}]_{i=1}^N\in[0,1]^N$ be the material physical density vector where $\boldsymbol{G}$ is a matrix approximating the Gaussian convolution operator. $\boldsymbol{G}$ is volume preserving, i.e. $\sum_{i=1}^N(\boldsymbol{G}\boldsymbol{\chi})_{e_i}=\sum_{i=1}^N\chi_{e_i}$. Compared to density methods like SIMP, the ICTM method naturally obtains a 0-1 topology from $\boldsymbol{\chi}$ that satisfies the volume constraint. The details of the numerical implementation are given in [6].

2.3 The diffusion model

The Denoising Diffusion Probabilistic Model (DDPM) was proposed by Ho et al. in [11]. Let $\boldsymbol{\rho}_0$ be our data sample, $\boldsymbol{\rho}_T$ be pure random noise. Suppose the generation involves T steps, the entire process can be viewed as:

$$\boldsymbol{\rho}=\boldsymbol{\rho}_0\to\boldsymbol{\rho}_1\to\boldsymbol{\rho}_2\to\cdots\to\boldsymbol{\rho}_{T-1}\to\boldsymbol{\rho}_T=\boldsymbol{z}. \tag{2.11}$$

The key idea of the diffusion model is to transform the one-step generation process (from $\boldsymbol{z}$ to $\boldsymbol{\rho}$) into a multistep generation process.

We define the forward (diffusion) process as:

$$\boldsymbol{\rho}_t=\alpha_t\boldsymbol{\rho}_{t-1}+\beta_t\boldsymbol{\epsilon}_t,\quad \boldsymbol{\epsilon}_t\sim\mathcal{N}(0,I), \tag{2.12}$$

where it holds that $\alpha_t^2+\beta_t^2=1$, and $\alpha_t,\beta_t>0$. For large α_t and small β_t, the diffusion process can be intuitively regarded as a demolition: the original 'building' $\boldsymbol{\rho}_{t-1}$ is partially preserved (scaled by α_t) while "bricks" (noise scaled by β_t) are added.

We can express the relation between $\boldsymbol{\rho}_t$ and $\boldsymbol{\rho}_0$ as:

$$\boldsymbol{\rho}_t=\bar{\alpha}_t\boldsymbol{\rho}_0+\bar{\beta}_t\bar{\boldsymbol{\epsilon}}_t,\quad \bar{\boldsymbol{\epsilon}}_t\sim\mathcal{N}(0,I), \tag{2.13}$$

where $\bar{\alpha}_t=\alpha_t\cdots\alpha_1$, $\bar{\beta}_t=\sqrt{1-(\alpha_t\cdots\alpha_1)^2}$ and $\bar{\boldsymbol{\epsilon}}_t$ is a standard random noise. Thus any $\boldsymbol{\rho}_t$ can be directly obtained from $\boldsymbol{\rho}_0$ once the parameters $\{\bar{\alpha}_t\}$ and $\{\bar{\beta}_t\}$ are specified. The forward process is carried out until $\bar{\alpha}_T$ approaches zero and $\boldsymbol{\rho}_T$ becomes nearly pure noise.

Next, we consider the reverse process (i.e., the construction or sampling process), which aims to recover $\boldsymbol{\rho}_{t-1}$ from $\boldsymbol{\rho}_t$. A simple approach is to define a model $\mu(\boldsymbol{\rho}_t)$ that approximates $\boldsymbol{\rho}_{t-1}$ using an L2 loss:

$$||\boldsymbol{\rho}_{t-1}-\mu(\boldsymbol{\rho}_t)||^2. \tag{2.14}$$

Using the forward process relation,

$$\boldsymbol{\rho}_t=\alpha_t\boldsymbol{\rho}_{t-1}+\beta_t\boldsymbol{\epsilon}_t\quad\Rightarrow\quad\boldsymbol{\rho}_{t-1}=\frac{1}{\alpha_t}\Big(\boldsymbol{\rho}_t-\beta_t\boldsymbol{\epsilon}_t\Big), \tag{2.15}$$

we are led to design the reverse model as

$$\hat{\boldsymbol{\rho}}_{t-1}=\mu(\boldsymbol{\rho}_t)=\frac{1}{\alpha_t}\Big(\boldsymbol{\rho}_t-\beta_t\boldsymbol{\epsilon}_\theta(\boldsymbol{\rho}_t,t)\Big), \tag{2.16}$$

where $\boldsymbol{\epsilon}_\theta(\boldsymbol{\rho}_t,t)$ represents the learnable (and only) parametric component. Although it appears that $\boldsymbol{\epsilon}_\theta(\boldsymbol{\rho}_t,t)$ is used to directly predict the noise $\boldsymbol{\epsilon}_t$, note that our aim is not to predict a particular sample of noise but to approximate its conditional expectation at step t. Consequently, the loss function is defined as

$$\mathcal{L}(\theta)=\mathbb{E}_{\boldsymbol{\rho}_0,\bar{\boldsymbol{\epsilon}}_t,\boldsymbol{\epsilon}_t,t}\Big[\|\boldsymbol{\epsilon}_t-\boldsymbol{\epsilon}_\theta(\bar{\alpha}_t\boldsymbol{\rho}_0+\bar{\beta}_t\bar{\boldsymbol{\epsilon}}_t,t)\|^2\Big]. \tag{2.17}$$

Derivation of the Loss Function:

For brevity, define

$$\boldsymbol{\rho}'_t=\bar{\alpha}_t\boldsymbol{\rho}_0+\bar{\beta}_t\bar{\boldsymbol{\epsilon}}_t, \tag{2.18}$$

so that the loss becomes

$$\mathcal{L}(\theta)=\mathbb{E}_{\boldsymbol{\rho}'_t,\boldsymbol{\epsilon}_t}\Big[\|\boldsymbol{\epsilon}_t-\boldsymbol{\epsilon}_\theta(\boldsymbol{\rho}'_t,t)\|^2\Big]. \tag{2.19}$$

For any candidate function $\boldsymbol{\epsilon}_\theta(\boldsymbol{\rho}'_t,t)$, we decompose the error as follows:

$$\|\boldsymbol{\epsilon}_t-\boldsymbol{\epsilon}_\theta(\boldsymbol{\rho}'_t,t)\|^2=\|\boldsymbol{\epsilon}_t-\mathbb{E}[\boldsymbol{\epsilon}_t|\boldsymbol{\rho}'_t]+\mathbb{E}[\boldsymbol{\epsilon}_t|\boldsymbol{\rho}'_t]-\boldsymbol{\epsilon}_\theta(\boldsymbol{\rho}'_t,t)\|^2. \tag{2.20}$$

Expanding the square, we obtain:

$$\begin{aligned}\|\boldsymbol{\epsilon}_t-\boldsymbol{\epsilon}_\theta(\boldsymbol{\rho}'_t,t)\|^2=&\|\boldsymbol{\epsilon}_t-\mathbb{E}[\boldsymbol{\epsilon}_t|\boldsymbol{\rho}'_t]\|^2+\|\mathbb{E}[\boldsymbol{\epsilon}_t|\boldsymbol{\rho}'_t]-\boldsymbol{\epsilon}_\theta(\boldsymbol{\rho}'_t,t)\|^2\\&+2\Big\langle\boldsymbol{\epsilon}_t-\mathbb{E}[\boldsymbol{\epsilon}_t|\boldsymbol{\rho}'_t],\mathbb{E}[\boldsymbol{\epsilon}_t|\boldsymbol{\rho}'_t]-\boldsymbol{\epsilon}_\theta(\boldsymbol{\rho}'_t,t)\Big\rangle.\end{aligned} \tag{2.21}$$

Taking the expectation with respect to $\boldsymbol{\epsilon}_t$ conditioned on $\boldsymbol{\rho}'_t$ and using the fact that

$$\mathbb{E}\Big[\boldsymbol{\epsilon}_t-\mathbb{E}[\boldsymbol{\epsilon}_t|\boldsymbol{\rho}'_t]\Big|\boldsymbol{\rho}'_t\Big]=\mathbf{0}, \tag{2.22}$$

the cross term vanishes. Therefore, we have:

$$\mathbb{E}\Big[\|\boldsymbol{\epsilon}_t-\boldsymbol{\epsilon}_\theta(\boldsymbol{\rho}'_t,t)\|^2\Big]=\mathbb{E}\Big[\|\boldsymbol{\epsilon}_t-\mathbb{E}[\boldsymbol{\epsilon}_t|\boldsymbol{\rho}'_t]\|^2\Big]+\mathbb{E}\Big[\|\mathbb{E}[\boldsymbol{\epsilon}_t|\boldsymbol{\rho}'_t]-\boldsymbol{\epsilon}_\theta(\boldsymbol{\rho}'_t,t)\|^2\Big]. \tag{2.23}$$

Since the term $\mathbb{E}\big[\|\boldsymbol{\epsilon}_t-\mathbb{E}[\boldsymbol{\epsilon}_t|\boldsymbol{\rho}'_t]\|^2\big]$ is independent of θ, the minimization of the loss is achieved when

$$\boldsymbol{\epsilon}_\theta(\boldsymbol{\rho}'_t,t)=\mathbb{E}[\boldsymbol{\epsilon}_t|\boldsymbol{\rho}'_t]. \tag{2.24}$$

This derivation rigorously confirms that our network does not simply predict a realization of $\boldsymbol{\epsilon}_t$, but approximates its conditional expectation, which is crucial for the stability and correctness of the reverse process.

Furthermore, considering the forward process relation:

$$\boldsymbol{\rho}_t = \alpha_t \boldsymbol{\rho}_{t-1} + \beta_t \boldsymbol{\epsilon}_t, \tag{2.25}$$

we can rearrange it to obtain:

$$\boldsymbol{\rho}_{t-1} = \frac{1}{\alpha_t}\Big(\boldsymbol{\rho}_t - \beta_t \boldsymbol{\epsilon}_t\Big). \tag{2.26}$$

Taking the conditional expectation on both sides with respect to $\boldsymbol{\rho}_t$, and substituting the optimal prediction $\mathbb{E}[\boldsymbol{\epsilon}_t|\boldsymbol{\rho}_t]$, we arrive at:

$$\mathbb{E}[\boldsymbol{\rho}_{t-1}\,|\,\boldsymbol{\rho}_t] = \frac{1}{\alpha_t}\Big(\boldsymbol{\rho}_t - \beta_t \mathbb{E}[\boldsymbol{\epsilon}_t|\boldsymbol{\rho}_t]\Big). \tag{2.27}$$

This conditional expectation forms the basis for the reverse update scheme, which is consistent with our model design.

To accelerate the generation process, we introduce DDIM [20] (Denoising Diffusion Implicit Models). Since DDPM typically requires 1000 steps—which is computationally intensive—DDIM allows us to generate high-quality samples in only a few steps.

During the training process, given the sequence $\{\bar{\alpha}_t\}_{t=1}^T$, the trained model is fixed. An interesting observation is that the training result of a T-step DDPM inherently contains the training results for any subsequence. More precisely, suppose we choose a subsequence $\tau=[\tau_1,\tau_2,\cdots,\tau_{dim(\tau)}]$ from $[1,2,\cdots,T]$. Then by setting $\{\bar{\alpha}_t\}_{t\in\tau}$ as parameters, one obtains a $dim(\tau)$-step DDPM. Consequently, if a T-step DDPM is well-trained, it implies that for any selected subsequence τ, the corresponding DDIM sampler based on $\{\bar{\alpha}_t\}_{t\in\tau}$ will generate high-quality samples in only $dim(\tau)$ steps.

The reverse process of DDIM is given by:

$$p(\boldsymbol{\rho}_{\tau_{i-1}}|\boldsymbol{\rho}_{\tau_i}) = \mathcal{N}\left(\frac{\bar{\alpha}_{\tau_{i-1}}}{\bar{\alpha}_{\tau_i}}\left(\boldsymbol{\rho}_{\tau_i} - \left(\bar{\beta}_{\tau_i} - \frac{\bar{\alpha}_{\tau_i}}{\bar{\alpha}_{\tau_{i-1}}}\sqrt{\bar{\beta}^2_{\tau_{i-1}} - \tilde{\sigma}^2_{\tau_i}}\right)\boldsymbol{\epsilon}_\theta\Big(\boldsymbol{\rho}_{\tau_i},\tau_i\Big)\right), \tilde{\sigma}^2_{\tau_i}\boldsymbol{I}\right), \tag{2.28}$$

where

$$\tilde{\sigma}_{\tau_i} = \frac{\bar{\beta}_{\tau_{i-1}}}{\bar{\beta}_{\tau_i}}\sqrt{1 - \frac{\bar{\alpha}^2_{\tau_i}}{\bar{\alpha}^2_{\tau_{i-1}}}}. \tag{2.29}$$

Conditional Diffusion Model:

In many applications, including our topology optimization problem, there exists additional physical condition information (e.g., boundary conditions, loads, volume fraction, etc.) denoted by y. To incorporate such information, we extend the denoising network to a conditional version $\boldsymbol{\epsilon}_\theta(\boldsymbol{\rho}_t, y, t)$. The associated conditional loss function becomes:

$$\mathcal{L}_{\text{cond}}(\theta) = \mathbb{E}_{\boldsymbol{\rho}_0,\bar{\boldsymbol{\epsilon}}_t,\boldsymbol{\epsilon}_t,t}\Big[\|\boldsymbol{\epsilon}_t - \boldsymbol{\epsilon}_\theta(\bar{\alpha}_t\boldsymbol{\rho}_0 + \bar{\beta}_t\bar{\boldsymbol{\epsilon}}_t, y, t)\|^2\Big]. \tag{2.30}$$

In our dataset, each sample $\boldsymbol{\rho}_0$ is naturally paired with the condition y. For training, we randomly sample a time step t and generate the noisy sample

$$\boldsymbol{\rho}_t = \bar{\alpha}_t \boldsymbol{\rho}_0 + \bar{\beta}_t \bar{\boldsymbol{\epsilon}}_t. \tag{2.31}$$

This process yields paired data $\{(\boldsymbol{\rho}_t, y)\}$ which are treated as i.i.d. during training to optimize the conditional loss. This extension enables our model to learn the conditional distribution of optimal structures given specific physical constraints.

In summary, our diffusion model employs a loss function whose minimizer is the conditional expectation $\mathbb{E}[\boldsymbol{\epsilon}_t|\boldsymbol{\rho}_t]$, ensuring that the predicted noise is well-behaved and consistent with the reverse process. Furthermore, the conditional extension seamlessly incorporates extra information y into the denoising process, enabling its application in topology optimization problems.

3 Diffusion model with iterative convolution thresholding method

The minimum compliance problem requires identifying the optimal structure that satisfies a set of constraints and physical conditions. We assume that the optimal structure, denoted by $\boldsymbol{\rho}$, adheres to a conditional distribution:

$$\boldsymbol{\rho} \sim p_{data}(\boldsymbol{\rho}|\boldsymbol{y}), \tag{3.1}$$

where $\boldsymbol{y}$ is the constraints and physical conditions corresponding to $\boldsymbol{\rho}$.

To generate a new sample of $\boldsymbol{\rho}$ from an approximate distribution, we introduce noise to an initial sample $\boldsymbol{\rho}_0$ from $\boldsymbol{p}_{data}$ and then employ a neural network to iteratively learn the noise introduced at each step. This process is analogous to the Denoising Diffusion Probabilistic Model (DDPM) framework, where noise is incrementally added over 1000 steps.

The forward diffusion process, depicted in Fig. 2, is formalized by the equation

$$q(\boldsymbol{\rho}_t|\boldsymbol{\rho}_0) = \mathcal{N}\left(\sqrt{\bar{\alpha}_t}\boldsymbol{\rho}_0, (1-\bar{\alpha}_t)I\right), \tag{3.2}$$

where the parameters α_t, $\bar{\alpha}_t$, and β_t are interrelated as follows:

$$\alpha_t = 1-\beta_t, \quad \bar{\alpha}_t = \prod_{i=1}^{t} \alpha_i, \quad \beta_t = 1 - \frac{\bar{\alpha}_t}{\bar{\alpha}_{t-1}}. \tag{3.3}$$

Hyperparameters can be determined through either linear or cosine interpolation:

- Linear interpolation is defined by setting $\beta^2_{\min} = 0.0001$ and $\beta^2_{\max} = 0.02$, leading to:

$$\beta_t^2 = \beta^2_{\min} + \frac{(\beta^2_{\max} - \beta^2_{\min}) \cdot t}{T}. \tag{3.4}$$

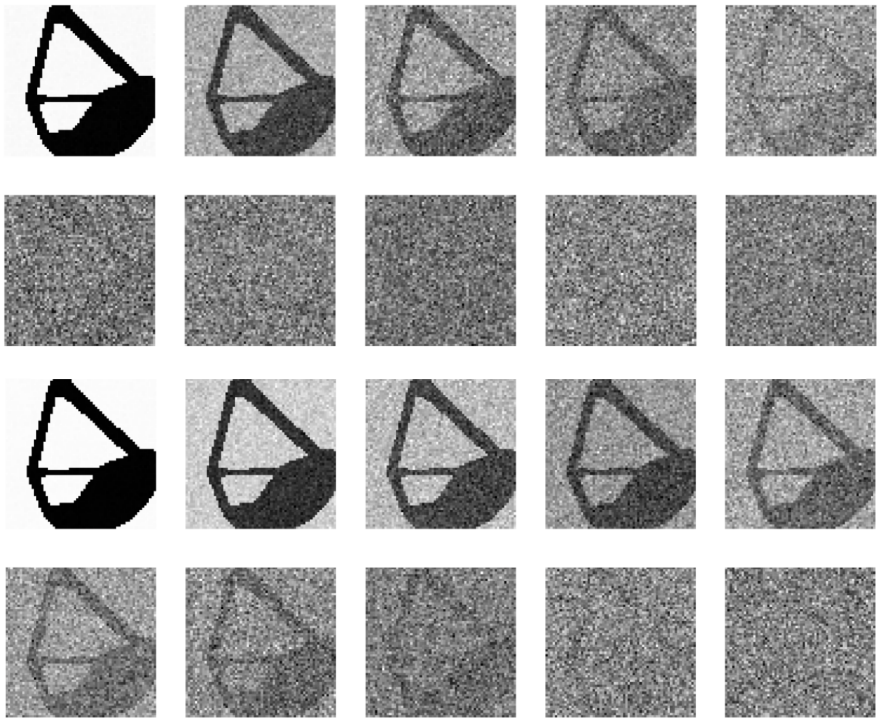

Figure 2: Diffusion process: images with different levels of noise. The top two rows utilize a linear noise scheduler, while the bottom two rows utilize a cosine noise scheduler. For each scheduler, there are 10 images representing $\{\rho_0,\rho_{111},\cdots,\rho_{999}\}$.

- Cosine interpolation first computes $\bar{\alpha}$ as:

$$\bar{\alpha}_t=\frac{f(t)}{f(0)},\quad \text{where}\quad f(t)=\cos\left(\frac{t/T+s}{1+s}\cdot\frac{\pi}{2}\right). \tag{3.5}$$

To avoid singularities and ensure stability in the diffusion process, β_t is clipped to be no larger than 0.999. Additionally, to prevent excessively small values near $t=0$, a small offset $s=0.008$ is applied.

The difference between the linear and cosine schedulers lies in their application; the linear scheduler is preferred for high-resolution images, whereas the cosine scheduler is advantageous for images of resolutions 64×64 and 32×32. The rate of noise addition and its implications on image clarity are evident in Fig. 3, where the linear scheduler results in a nearly complete noise-dominant image by ρ_{555}, and the cosine scheduler maintains visible structural features up to ρ_{888}. These scheduler types are compared to assess their suitability for different resolutions, as detailed in [16]. We use linear scheduler in our model since the image is 128×128 in our dataset.

For the reverse (sampling) process, we utilize the Denoising Diffusion Implicit Mod-

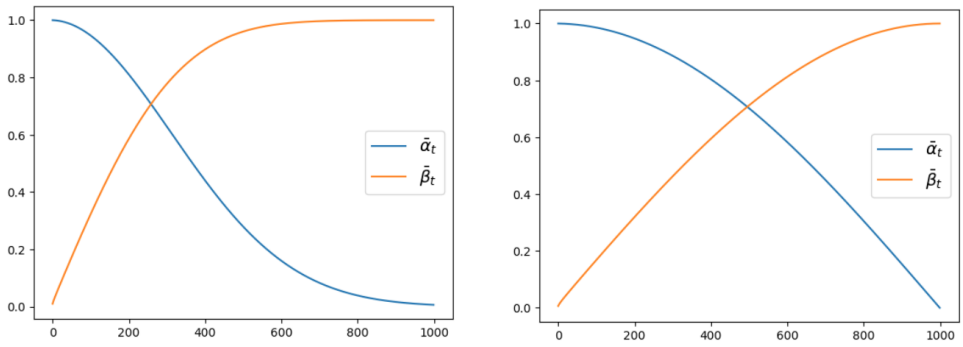

Figure 3: The left graph illustrates the linear noise scheduler, and the right graph illustrates the cosine noise scheduler.

els (DDIM) [20], employing the following formula:

$$\hat{\boldsymbol{\rho}}_{t-1} = \bar{\alpha}_{t-1} \underbrace{\left(\frac{\boldsymbol{\rho}_t - \bar{\beta}_t \epsilon_\theta^{(t)}(\boldsymbol{\rho}_t, \boldsymbol{y})}{\bar{\alpha}_t} \right)}_{\text{"predicted } \rho_0\text{"}} + \underbrace{\sqrt{\bar{\beta}_{t-1}^2 - \sigma_t^2} \cdot \epsilon_\theta^{(t)}(\boldsymbol{\rho}_t, \boldsymbol{y})}_{\text{"direction pointing to } \rho_t\text{"}} + \underbrace{\sigma_t \boldsymbol{\epsilon}_t}_{\text{random noise}} . \tag{3.6}$$

The equation $\hat{\boldsymbol{\rho}}_{t-1}$ differs from the simpler form $\boldsymbol{\rho}_{t-1} = \bar{\alpha}_{t-1}\boldsymbol{\rho}_0 + \bar{\beta}_{t-1}\bar{\epsilon}_{t-1}$ because it incorporates the predicted noise and directionality terms explicitly. The term σ_t is introduced to control the level of stochasticity in the reverse process. The presence of σ_t allows the model to incorporate random noise ϵ_t, which helps in generating diverse samples and avoiding deterministic paths that might not cover the data distribution adequately.

The introduction of σ_t ensures that the sampling process can explore different possible trajectories, making the generated samples more diverse. Specifically, σ_t is defined as:

$$\sigma_t = \eta \frac{\bar{\beta}_{t-1}\beta_t}{\bar{\beta}_t}, \tag{3.7}$$

where $\eta \in [0,1]$ is a hyperparameter that controls the variance, allowing for fine-tuning of the noise level during the sampling process.

Fig. 4 illustrates the overall workflow of our model. The input is a shape-multichannel tensor $8 \times 128 \times 128$, comprising the noisy image, the volume fraction matrix with each element set to the volume fraction (VF), the external load matrices in the x and y directions, the displacement boundary conditions, as well as the stress and strain fields computed using the Finite Element Method (FEM). The model's objective is to output the optimized structure that meets the specified criteria.

Despite its potential, our deep learning-based model, like others, faces limitations, notably in generating disconnected or non-smooth structures, which may manifest as isolated 'islands' or spikes. Additionally, the model's sampling speed is relatively slow, as generation time is proportional to the number of sampling steps.

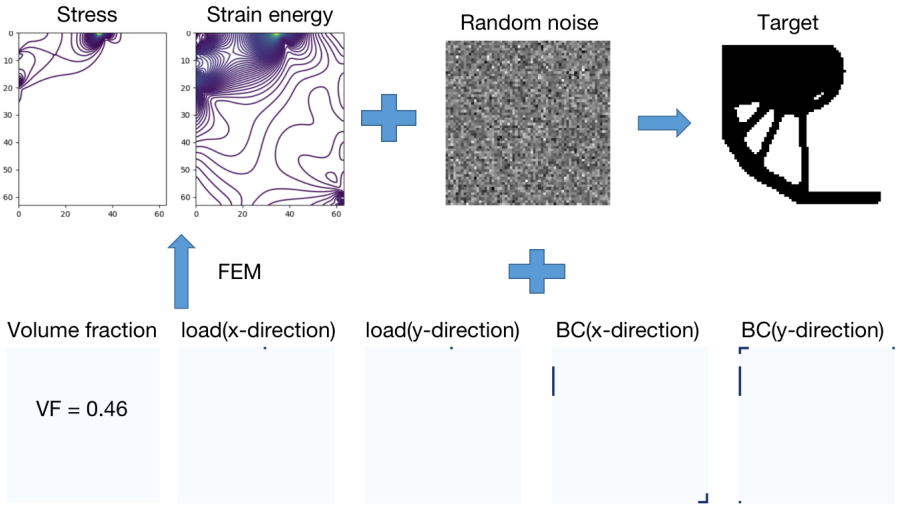

Figure 4: Illustration of the model's input consisting of various physical condition channels as condition and the random noise.

To overcome these challenges, we have devised a new method that combines the diffusion model and the threshold dynamics. By initializing the threshold dynamics with the output from a 5-step diffusion model, we can achieve layouts that are both smoother and better connected. This integrated approach, which we have termed Diffusion Model based Iterative Convolution thresholding Method (DICTM), promises to surmount the limitations of individual models while capitalizing on their strengths.

Conditional Denoising Network and Training Data Pairing:

In our application, the structural designs are conditioned on additional physical information, represented by $\boldsymbol{y}$. To integrate this conditioning in the reverse process, we extend our denoising network to the conditional form:

$$\epsilon_\theta^{(t)}(\boldsymbol{\rho}_t, \boldsymbol{y}), \tag{3.8}$$

which explicitly takes both the current noisy image $\boldsymbol{\rho}_t$ and the physical conditions $\boldsymbol{y}$ as inputs.

The corresponding conditional loss function is given by:

$$\mathcal{L}_{\text{cond}}(\theta) = \mathbb{E}_{\boldsymbol{\rho}_0, \bar{\boldsymbol{\epsilon}}_t, \boldsymbol{\epsilon}_t, t}\left[\|\boldsymbol{\epsilon}_t - \epsilon_\theta^{(t)}(\bar{\alpha}_t \boldsymbol{\rho}_0 + \bar{\beta}_t \bar{\boldsymbol{\epsilon}}_t, \boldsymbol{y})\|^2 \right]. \tag{3.9}$$

Here, each training sample $\boldsymbol{\rho}_0$ in the dataset is naturally paired with the corresponding condition $\boldsymbol{y}$ (which encompasses the volume fraction, boundary conditions, loads, FEM fields, etc.). For a given $\boldsymbol{\rho}_0$, we generate the noisy version $\boldsymbol{\rho}_t$ using:

$$\boldsymbol{\rho}_t = \bar{\alpha}_t \boldsymbol{\rho}_0 + \bar{\beta}_t \bar{\boldsymbol{\epsilon}}_t, \tag{3.10}$$

which guarantees that the paired data $(\boldsymbol{\rho}_t, \boldsymbol{y})$ are i.i.d. This pairing forms the basis for training the conditional denoising network by minimizing $\mathcal{L}_{\text{cond}}(\theta)$.

By explicitly incorporating $\boldsymbol{y}$ into the network and loss function, the model learns the conditional distribution of optimized layouts given the physical constraints. This results in improved performance when generating structural designs that inherently satisfy the requisite physical conditions.

In summary, our DICTM framework first uses a 5-step conditional DDIM-based diffusion model to quickly generate an initial guess that captures key structural features under the given conditions. In a subsequent stage, iterative convolution thresholding refines this initial output, ensuring that the final design adheres strictly to the physical and topological constraints.

3.1 Network structure

In our model, we employ a U-Net architecture adapted for two-dimensional data, as implemented in the `UNet2DModel` from the `diffusers` library. The network configuration is as follows:

- **Layers per block**: 2 ResNet layers are used per U-Net block.
- **Block output channels**: The numbers of output channels for each block are $(64, 128, 256, 256)$, increasing with the depth of the network.
- **Downsampling blocks**: The downsampling path consists of:
  - 1 `DownBlock2D`
  - 3 `AttnDownBlock2D` blocks

  These blocks include attention mechanisms to capture long-range dependencies.
- **Upsampling blocks**: The upsampling path mirrors the downsampling path, consisting of:
  - 3 `AttnUpBlock2D` blocks
  - 1 `UpBlock2D`

This U-Net architecture allows the model to effectively capture both local and global features, which is crucial for high-fidelity image reconstruction in diffusion models. The inclusion of attention mechanisms enhances the network's ability to model complex relationships within the data.

4 Dataset

Our dataset consists of 32,000 examples, each with eight distinct channels, and each channel is represented as a 128×128 matrix. These channels encompass volume fraction, stress field, strain energy field, loads in both x and y directions, displacement boundary conditions, and the optimized structure derived via the SIMP method with a uniform material distribution that satisfies the volume constraint as the initial guess. We used the idea of physical fields proposed in [17]. Unlike previous datasets constrained by a limited set of fixed conditions, our dataset exhibits the following characteristics:

1. The volume fraction channel is a uniform matrix, with each element indicating the volume fraction, which varies in a range of $[0.2,0.6]$ in increments of 0.02.

2. The stress field is characterized by the von Mises stress, computed as:

$$\sigma_{vm}=\sqrt{\sigma_{11}^2-\sigma_{11}\sigma_{22}+\sigma_{22}^2+3\sigma_{12}^2}.$$

3. The strain energy density is calculated according to:

$$W=\frac{1}{2}(\sigma_{11}\epsilon_{11}+\sigma_{22}\epsilon_{22}+2\sigma_{12}\epsilon_{12}).$$

4. Load channels for the x and y directions are different from the conventional datasets, which only include point force on the boundary, in two aspects. First, we incorporate not only loads applied on the boundary but also those applied within the domain. The internal load is a point force. Second, the load on the boundary can be either point force or traction force along a segment of the boundary.

5. The displacement boundary conditions for the x and y directions are randomly generated, offering a wide spectrum of potential scenarios and improving the diversity of the data set. Our displacement boundary conditions randomly fall into three categories, cantilever beam ($\mathbf{u}=0$ on Γ_D), MBB beam ($u_x=0$ on Γ_{D_x}, $u_y=0$ on Γ_{D_y}), combination of MBB beam and cantilever beam ($\mathbf{u}=0$ on Γ_D, $u_x=0$ on Γ_{D_x}, $u_y=0$ on Γ_{D_y}). Γ_D, Γ_{D_x} and Γ_{D_y} are randomly selected from the pool of boundary points while satisfying certain requirements. Any point on the boundary is one of the following three cases, free, fixed, able to slide along the tangent direction but fixed along the normal direction. We call the last case the sliding points. In addition, it requires that with these boundary conditions (2.3) have a unique solution. Since $\mathbf{E}$ is already positive definite, in the case of a cantilever beam, a sufficient condition for the linear elasticity equation to have a unique solution is $|\Gamma_D|>0$. Therefore, we need at least two consecutive points in Γ_D. In the case of a MBB beam, we found that the linear system usually has a unique solution when the number of sliding points along the x direction and along the y direction are both larger than 1.

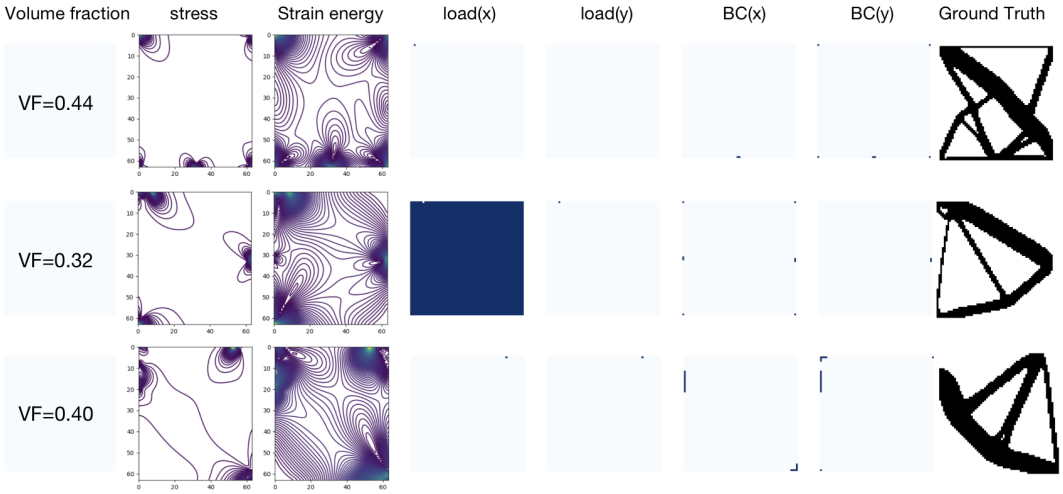

Figure 5: Dataset illustration: we choose three data samples from our dataset, each row is a sample that comprises eight channels representing different physical properties, each as a 128×128 matrix.

6. Load magnitudes are normalized to 1, with application angles generated from the uniform distribution in $[0,2\pi]$, which moves beyond the standard practice of selecting from a predefined set of angles.

7. The material distribution channel reflects the optimized structure as determined by the SIMP method.

The randomization of load positions, both boundary and internal, and the wide variety of displacement boundary conditions make our dataset different from previous works [15,17], which were limited to a few fixed configurations. This approach enhances the dataset's versatility and applicability to a broader range of structural optimization problems.

The dataset is divided into 30,000 training samples and 2,000 testing samples, providing a robust basis for model training. Fig. 5 illustrates the complexity and diversity of the dataset, which is crucial for developing models that are both accurate and generalizable across various scenarios.

5 Numerical results

In this section, we present the numerical results of applying the Diffusion model with Iterative Convolution Thresholding Method (DICTM) to the minimum compliance problem in topology optimization. We compare our method with traditional numerical methods such as the SIMP method and the ICTM, as well as contemporary deep learning approaches like TopologyGAN [17]. The numerical results demonstrate the effectiveness of DICTM in achieving high-quality designs with reduced computational cost.

For the ICTM method, we used the implementation described in [6], and the SIMP method was implemented following the approach in [4]. The dataset used consists of 30,000 training samples and 2,000 testing samples, as described in Section 4. Each sample includes eight channels representing different physical properties, each as a 128×128 matrix. The diffusion model was trained on this dataset to learn the conditional distribution of optimized structures given various boundary conditions and loads.

5.1 Comparison of deep generative models: Diffusion model vs. TopologyGAN

We first compare two deep generative models: our diffusion model and TopologyGAN [17], a GAN-based approach for topology optimization. Fig. 6 shows sample results on the training dataset, and Fig. 7 presents results on the testing dataset.

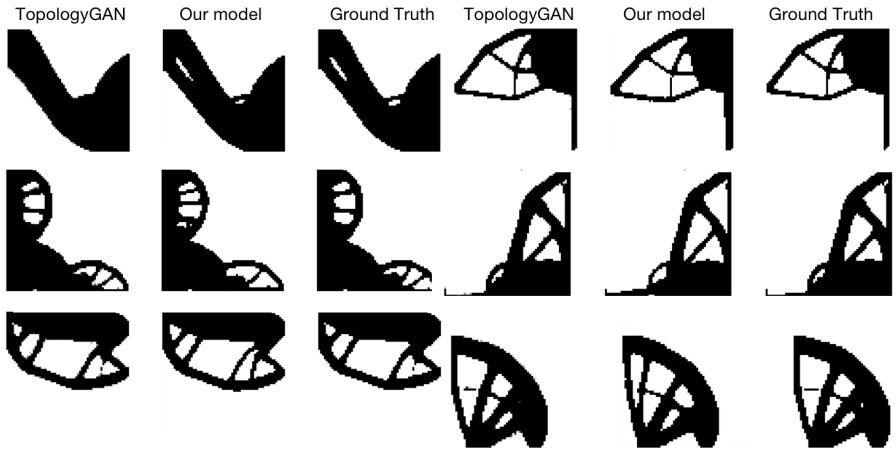

Figure 6: Training set results for TopologyGAN and Diffusion Model. Six examples are randomly selected from the training set.

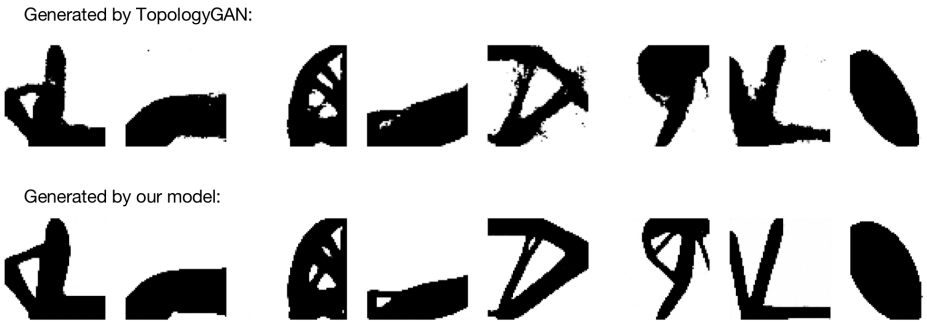

Figure 7: Testing set results for TopologyGAN and Diffusion Model. Eight examples are randomly selected from the testing set.

As observed in Fig. 6, TopologyGAN performs well on the training data, capturing primary structural features. However, its performance significantly degrades on the testing data (Fig. 7), indicating overfitting and poor generalization. In contrast, our diffusion model maintains consistent performance across both training and testing datasets, demonstrating strong generalization capabilities. This can be attributed to the diffusion model's ability to effectively learn the underlying data distribution.

After training, our diffusion model can be used with the DDIM sampler at an arbitrary number of steps. In principle, using the same number of steps as during training (e.g., 1000 steps) would yield very refined outputs; however, the computational cost increases with the number of steps. Since our goal here is to obtain an initial guess for ICTM, we opt for a balanced 5-step DDIM generation. Although the generated output on the test dataset appears visually good, our experiments show that some designs still do not fully satisfy the physical constraints. That is why, after generating the initial guess, we further refine it using ICTM iterations to produce a final layout that strictly conforms to both physical and mathematical requirements. In contrast, GAN-generated results are so degraded on the testing dataset that they are not even suitable as an initial guess—often causing ICTM to converge to a bad design.

Both TopologyGAN and our 5-step diffusion model have similar sampling speeds, producing 100 images in 0.49s and 0.36s on an Nvidia A-100 (40G) GPU, respectively. However, our diffusion model uses only 27 million parameters, significantly fewer than TopologyGAN's 130 million. This suggests that the diffusion model offers a more streamlined and efficient architecture without sacrificing performance.

5.2 Comparison with classical numerical methods

We further compare DICTM with classical numerical methods, specifically the SIMP method and ICTM without the diffusion model initialization. Since the results from DM and SIMP contain gray area, for fair comparison, we first obtained a 0-1 design and calculated its compliance, which we call "sharp compliance". We let the density be 1 if it exceeds a threshold and let it be 0 if it is below the same threshold. The threshold was determined so that the final 0-1 design would satisfy the volume constraint. Fig. 8 illustrates the results for a representative test case.

As shown in Fig. 8, we consider the design domain $\Omega = [0,1]^2$. A traction force $f = [-0.6484, 0.7613]$ is applied on the line segment $\{(x,1) : x \in [0.55,1]\}$ of the upper boundary. The boundary conditions for the displacement $\mathbf{u} = (u_x, u_y)$ are $u_x = 0$ on $\Gamma_x := \{(0,y) : y \in [0,0.38]\}$ and $u_y = 0$ on $\Gamma_y := \{(x,0) : x \in [0.0078,0.016]\}$. The second row shows the results generated from the 5-step diffusion model, the ICTM (with 67 iterations) and the SIMP respectively. The result from the diffusion model exhibits non-physical and disconnected structures. The result from ICTM has significantly higher compliance and is obviously trapped in a local minimum. The third row shows the results of the DICTM using different values of τ (the standard deviation of the convolution kernel). Using the result generated from the diffusion model as the initial distribution,

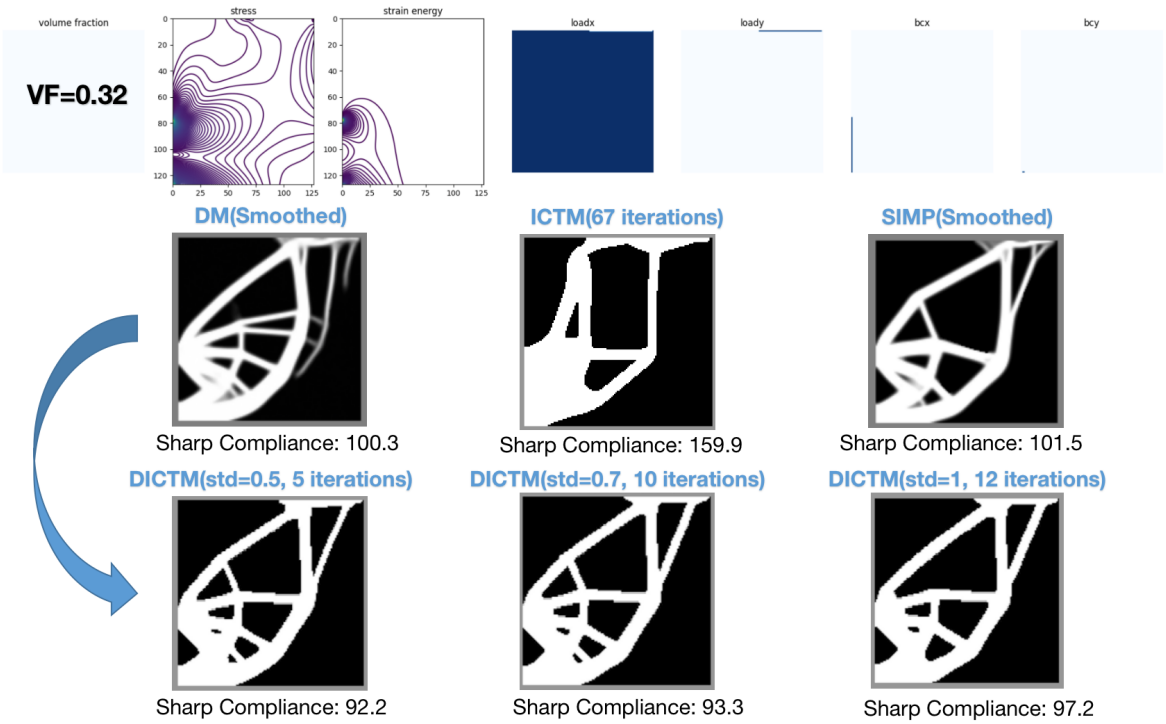

Figure 8: Comparison of DICTM with classical numerical methods. **First Row**: Boundary conditions and physical fields, including volume fraction, loads, and displacement boundary conditions. **Second Row**: Results generated by the 5-step diffusion model (DM), ICTM with a uniform initial guess, and the SIMP method. **Third Row**: Results from DICTM using different values of the convolution kernel standard deviation τ. DICTM results exhibit smoother and more connected structures while achieving lower compliance values.

the ICTM quickly converges to an optimized structure. In particular, it takes 5 iterations for ICTM to converge when $\tau = 0.5$ with the smallest compliance at 92.2. The designs generated by DICTM are also smoother and more connected compared to those from the diffusion model alone or the ICTM. The SIMP method produces structures with comparable compliance but requires more iterations.

Traditional numerical methods often require careful tuning of hyperparameters, such as the standard deviation τ of the Gaussian kernel in Eq. (2.6). Selecting appropriate hyperparameters is crucial for achieving optimal results but can significantly increase computational cost due to the need for multiple trial-and-error iterations.

In our approach, the diffusion model provides a high-quality initial guess, which substantially accelerates the convergence of the threshold dynamics method. Specifically, we observed that the number of iterations required for convergence is reduced to approximately one-tenth of what is needed when starting from a uniform initial guess. This acceleration means that even if we experiment with multiple τ values to find the optimal one, the total computational cost remains comparable to that of the traditional method with a fixed hyperparameter.

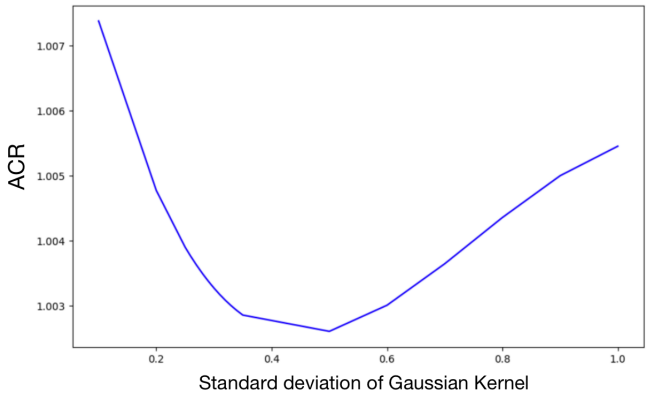

Figure 9: Average Compliance Ratio (ACR) versus τ for DICTM. The ACR remains close to 1 across a range of τ values, indicating that DICTM achieves comparable compliance to the SIMP method while reducing computational cost.

Table 1: ACR comparison across different models.

ICTM	DM (5-step)	DICTM with $\tau=0.5$	DICTM with optimal τ
1.0463	1.1142	1.0026	1.0016

Fig. 9 illustrates the Average Compliance Ratio (ACR) versus different τ values. The ACR is defined as:

$$\mathrm{ACR}=\mathbb{E}\left[\frac{C_{\mathrm{DICTM}}(\tau)}{C_{\mathrm{SIMP}}}\right],$$

where $C_{\mathrm{DICTM}}(\tau)$ is the compliance obtained using our method with a specific τ, and C_{SIMP} is the compliance from the SIMP method. The results demonstrate that our method consistently achieves compliance values close to or better than those obtained by SIMP across a range of τ values.

By integrating the deep-generative model to provide an effective initial guess, DICTM not only reduces the dependence on hyperparameter tuning but also combines the strengths of deep learning and traditional numerical algorithms. This synergy leads to efficient computations without compromising the quality of the optimized structures.

Although DICTM demonstrates significant improvements in efficiency and performance, there are limitations to consider. The current implementation is tested on two-dimensional problems with specific boundary conditions and load cases. Extending the method to three-dimensional problems or more complex structures may require additional adjustments to the model architecture and training process.

Furthermore, reliance on a high-quality data set means that generalization capabilities are influenced by the diversity of the training data. Future work could focus on expanding the dataset to include a wider range of scenarios and exploring data augmentation techniques to enhance robustness.

The experimental results demonstrate that DICTM effectively combines the strengths of deep generative models and traditional numerical methods. By providing a good initial guess through the diffusion model, the threshold dynamics method converges more rapidly and yields high-quality designs that meet or exceed the performance of those generated by traditional methods. The reduction in computation time and iterations highlights the practical advantages of our approach for topology optimization tasks.

6 Conclusion

In this study, we introduced the diffusion model with iterative convolution thresholding method (DICTM), a novel hybrid approach for solving the minimum compliance problem in topology optimization. By combining the strengths of diffusion models and threshold dynamics, DICTM effectively captures the complexities of the linear elasticity problem while significantly improving computational efficiency.

Our numerical results demonstrate that DICTM produces high-quality topological designs with strong generalization in various test scenarios. The integration of the diffusion model provides a superior initial guess for the threshold dynamics method, reducing the number of required iterations to approximately one-tenth of the original. This acceleration not only saves computational time, but also allows for efficient hyperparameter tuning without increasing the overall cost.

Using the predictive capabilities of deep-generative models and the rigorous mathematical framework of threshold dynamics, DICTM offers a promising direction for topology optimization. The synergy between deep learning and traditional numerical methods enables us to achieve designs that are both physically sound and computationally efficient.

Future work will focus on extending the DICTM approach to more complex optimization problems, exploring higher-dimensional design spaces, and further improving the integration between deep learning models and numerical algorithms. We believe that this methodology has the potential to advance the field of topology optimization and contribute to practical engineering applications.

Acknowledgments

X.-P. Wang acknowledges support from the National Natural Science Foundation of China (NSFC) (No. 12271461), the key project of NSFC (No. 12131010), Shenzhen Science and Technology Innovation Program (Grant: C10120230046), the Hetao Shenzhen-Hong Kong Science and Technology Innovation Cooperation Zone Project (No.HZQSWS-KCCYB-2024016) and the University Development Fund from The Chinese University of Hong Kong, Shenzhen (UDF01002028).

References

[1] Grégoire Allaire, François Jouve, and Anca-Maria Toader. Structural optimization using sensitivity analysis and a level-set method. *Journal of Computational Physics*, 194(1):363–393, 2004.

[2] Tomer Amit, Tal Shaharbany, Eliya Nachmani, and Lior Wolf. SegDiff: Image segmentation with diffusion probabilistic models. *arXiv preprint arXiv:2112.00390*, 2021.

[3] Jacob Austin, Daniel D Johnson, Jonathan Ho, Daniel Tarlow, and Rianne Van Den Berg. Structured denoising diffusion models in discrete state-spaces. *Advances in Neural Information Processing Systems*, 34:17981–17993, 2021.

[4] Martin P Bendsøe and Ole Sigmund. Material interpolation schemes in topology optimization. *Archive of Applied Mechanics*, 69:635–654, 1999.

[5] Chentao Cao, Zhuo-Xu Cui, Yue Wang, Shaonan Liu, Taijin Chen, Hairong Zheng, Dong Liang, and Yanjie Zhu. High-frequency space diffusion model for accelerated MRI. *IEEE Transactions on Medical Imaging*, 43.5:1853–1865, 2024.

[6] Luyu Cen, Wei Hu, Dong Wang, and Xiaoping Wang. An iterative thresholding method for the minimum compliance problem. *Communications in Computational Physics*, 33(4):1189–1216, 2023.

[7] Huangxin Chen, Haitao Leng, Dong Wang, and Xiao-Ping Wang. An efficient threshold dynamics method for topology optimization for fluids. *CSIAM Transactions on Applied Mathematics*, 3:26–56, 2022.

[8] Hyungjin Chung, Eun Sun Lee, and Jong Chul Ye. MR image denoising and super-resolution using regularized reverse diffusion. *IEEE Transactions on Medical Imaging*, 42(4):922–934, 2022.

[9] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat GANs on image synthesis. *Advances in Neural Information Processing Systems*, 34:8780–8794, 2021.

[10] Tinghao Guo, Danny J Lohan, Ruijin Cang, Max Yi Ren, and James T Allison. An indirect design representation for topology optimization using variational autoencoder and style transfer. In *2018 AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, page 0804, 2018.

[11] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020.

[12] Emiel Hoogeboom, Didrik Nielsen, Priyank Jaini, Patrick Forré, and Max Welling. Argmax flows and multinomial diffusion: Learning categorical distributions. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, volume 34, pages 12454–12465. Curran Associates, Inc., 2021.

[13] Bangti Jin, Jing Li, Yifeng Xu, and Shengfeng Zhu. An adaptive phase-field method for structural topology optimization. *Journal of Computational Physics*, 506:112932, 2024.

[14] Bahjat Kawar, Gregory Vaksman, and Michael Elad. Stochastic image denoising by sampling from the posterior distribution. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1866–1875, 2021.

[15] François Mazé and Faez Ahmed. Diffusion models beat GANs on topology optimization. *arXiv*, 2022.

[16] Alexander Quinn Nichol and Prafulla Dhariwal. Improved denoising diffusion probabilistic models. In *International Conference on Machine Learning*, pages 8162–8171. PMLR, 2021.

[17] Zhenguo Nie, Tong Lin, Haoliang Jiang, and Levent Burak Kara. TopologyGAN: Topology optimization using generative adversarial networks based on physical fields over the initial

domain. *Journal of Mechanical Design*, 143(3), 2021.

[18] Conner Sharpe and Carolyn Conner Seepersad. Topology design with conditional generative adversarial networks. In *International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, volume 59186, page V02AT03A062. American Society of Mechanical Engineers, 2019.

[19] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International Conference on Machine Learning*, pages 2256–2265. PMLR, 2015.

[20] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020.

[21] Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. *Advances in Neural Information Processing Systems*, 32, 2019.

[22] Michael Yu Wang, Xiaoming Wang, and Dongming Guo. A level set method for structural topology optimization. *Computer Methods in Applied Mechanics and Engineering*, 192(1-2):227–246, 2003.

[23] Qian Yu and Yibao Li. A second-order unconditionally energy stable scheme for phase-field based multimaterial topology optimization. *Computer Methods in Applied Mechanics and Engineering*, 405:115876, 2023.

[24] Yonggyun Yu, Taeil Hur, Jaeho Jung, and In Gwun Jang. Deep learning for determining a near-optimal topological design without any iteration. *Structural and Multidisciplinary Optimization*, 59(3):787–799, 2019.

[25] Yujie Zhang and Wenjing Ye. Deep learning–based inverse method for layout design. *Structural and Multidisciplinary Optimization*, 60:527–536, 2019.