DPM-OT: A New Diffusion Probabilistic Model Based on Optimal Transport

Zezeng Li[1,2], ShengHao Li[1], Zhanpeng Wang[3], Na Lei[1]*, Zhongxuan Luo[1], Xianfeng Gu[4]
[1]School of Software, Dalian University of Technology, China
[2]Beijing Key Laboratory of Light-field Imaging and Digital Geometry, Capital Normal University, China
[3]School of Mathematical Sciences, University of the Chinese Academy of Sciences, China
[4]Computer Science and Applied Mathematics, State University of New York at Stony Brook, USA

Abstract

*Sampling from diffusion probabilistic models (DPMs) can be viewed as a piecewise distribution transformation, which generally requires hundreds or thousands of steps of the inverse diffusion trajectory to get a high-quality image. Recent progress in designing fast samplers for DPMs achieves a trade-off between sampling speed and sample quality by knowledge distillation or adjusting the variance schedule or the denoising equation. However, it can't be optimal in both aspects and often suffer from mode mixture in short steps. To tackle this problem, we innovatively regard inverse diffusion as an optimal transport (OT) problem between latents at different stages and propose the **DPM-OT**, a unified learning framework for fast DPMs with a direct expressway represented by OT map, which can generate high-quality samples within around 10 function evaluations. By calculating the semi-discrete optimal transport map between the data latents and the white noise, we obtain an expressway from the prior distribution to the data distribution, while significantly alleviating the problem of mode mixture. In addition, we give the error bound of the proposed method, which theoretically guarantees the stability of the algorithm. Extensive experiments validate the effectiveness and advantages of **DPM-OT** in terms of speed and quality (FID and mode mixture), thus representing an efficient solution for generative modeling. Source codes are available at* https://github.com/cognaclee/DPM-OT

1. Introduction

Diffusion probabilistic models (DPMs) [40, 17, 43] are a class of new prevailing generative models which use a parameterized Markov chain to produce samples matching the data distribution after a finite time. Transitions of this chain include two processes: the diffusion process gradually adds noise to a data distribution and the sampling process gradually reverses each step of the noise corruption over a long trajectory of timesteps. DPMs are able to produce high-quality samples and even superior to the current SOTAs generative adversarial networks (GANs) [15] on many tasks, such as image generation [11, 33, 10], video generation [18], text-to-image generation [37], point cloud generation [32, 34], shape generation [49, 47] and speech synthesis [7, 8]. Despite their success, the sampling of DPMs often requires iterating over thousands of timesteps, which is two or three orders of magnitude slower [41, 3] than single-step generative models GANs and VAEs [22].

To accelerate the sampling process, the community has been focusing on fast DPMs. Existing works have successfully accelerated DPMs by knowledge distillation [38, 31], or adjusting the variance schedule [39, 35, 26, 46] or the denoising equation [41, 19, 44, 36, 28, 3, 30, 48]. However, as [23, 28] remarks, early fast samplers cannot maintain the quality of samples and even introduce new noise at a high speedup rate, which limits their practicability. Moreover, existing methods try to approximate a continuous diffusion process with a deep neural network, but ignore the discontinuity of the target data manifold at the class boundary, which leads to mode mixture in the generated images.

To resolve the above issues, we cast the denoising process as an OT problem and then compute the Brenier potential [4, 5] to represent the OT map which is discontinuous at singularity sets [13, 9, 1] and thus avoids mode mixture. Then we construct an optimal trajectory between different timestep latents, which combines multiple denoising processes into an OT map, thus greatly shortening the sampling trajectory. Building upon it, we propose **DPM-OT** which can generate high-quality images within around 10 steps of inverse diffusion. In summary, our main contributions are:

- By combining OT and diffusion model, we propose a unified learning framework **DPM-OT** for fast DPMs.
- **DPM-OT** computes the Brenier potential to represent the OT map between different timesteps latents which relieves mode mixture significantly.

*Corresponding author: *Na Lei (nalei@dlut.edu.cn)*

- We theoretically analyze the single-step error and give the upper bound of the error between the generated data distribution and the target data distribution.
- Extensive experiments demonstrate **DPM-OT** outperforms SOTAs in quality, especially for mode mixture.

2. Preliminaries

In this paper, we are committed to providing a plug-and-play fast DPM framework by incorporating the OT into DPM. So, in this section, we first review the generalized DPMs from [17, 41, 43]. Then we introduce the semi-discrete optimal transport (SDOT) used later in this paper.

2.1. Generalized Diffusion Probabilistic Model

Given a data distribution $\boldsymbol{x}_0 \sim q(\boldsymbol{x}_0)$, DPMs define a diffusion process $q(\boldsymbol{x}_t|\boldsymbol{x}_{t-1})$ which produces a diffusion trajectory $\{\boldsymbol{x}_t\}_{t=1}^T$ by adding gaussian noise and a sampling process $p(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)$ which reverse the diffusion process to reconstruct the original data. As song et al. remark in [43], generalized DPMs can be expressed as solutions of stochastic differential equations (SDEs) of the form:

$$d\boldsymbol{x} = b(\boldsymbol{x}, t)dt + \sigma(t)d\boldsymbol{w} \tag{1}$$

where $\boldsymbol{w}$ is the standard Winener process (a.k.a., Brownian motion), $b(., t) : \mathbb{R}^d \to \mathbb{R}^d$ is vector-valued funciton called the *drift* coefficient of $\boldsymbol{x}(t)$, and $\sigma(\cdot) : \mathbb{R} \to \mathbb{R}$ is a scalar function known as the *diffusion* coefficient of $\boldsymbol{x}(t)$. Eq. (1) is the limit of the following discrete form (Eq. 2) in $\Delta t \to 0$, which is also known as **forward SDE**.

$$\boldsymbol{x}_{t+\Delta t} = \boldsymbol{x}_t + b(\boldsymbol{x}, t)\Delta t + \sigma(t)\sqrt{\Delta t}\boldsymbol{z}, \;\; \boldsymbol{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}) \tag{2}$$

From a probability point of view, Eq. (2) is reformulated in the following conditional probability:

$$q(\boldsymbol{x}_{t+\Delta t}|\boldsymbol{x}_t) \sim \mathcal{N}\left(\boldsymbol{x}_t + b(\boldsymbol{x}, t)\Delta t, \sigma^2(t)\Delta t\mathbf{I}\right). \tag{3}$$

With a sufficiently long diffusion trajectory $\{\boldsymbol{x}_t\}_{t=0}^T$ and a well-behaved schedule of $\{(b(\cdot, t), \sigma(t))\}_{t=0}^T$, the last latent $\boldsymbol{x}_T$ is nearly a Gaussian distribution. Starting from $\boldsymbol{x}_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, the exact reverse diffusion distribution $q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)$ is indispensable for the sampling process which gradually reverses each step of the noise corruption latents $\boldsymbol{x}_{t-1}$ from $\boldsymbol{x}_t$. However, since $q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)$ depends on the entire data distribution, DPMs approximate it using a neural network parameterized by θ as follows:

$$p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t) \coloneqq \mathcal{N}(\boldsymbol{x}_{t-1}; \mu_\theta(\boldsymbol{x}_t, t), \Sigma_\theta(\boldsymbol{x}_t, t)) \tag{4}$$

Using Bayes rule, the posterior satisfies $p(\boldsymbol{x}_{t-\Delta t}|\boldsymbol{x}_t) \sim \mathcal{N}(\boldsymbol{x}_t - [b(\boldsymbol{x}_t, t) - \sigma^2(t)\nabla_{\boldsymbol{x}_t}\log q(\boldsymbol{x}_t)]\Delta t, \sigma^2(t)\Delta t\mathbf{I})$. In $\Delta t \to 0$, it converges to the following **inverse SDE**:

$$d\boldsymbol{x} = [b(\boldsymbol{x}, t) - \sigma^2(t)\nabla_{\boldsymbol{x}_t}\log q(\boldsymbol{x}_t)]dt + \sigma(t)d\boldsymbol{w}. \tag{5}$$

The estimation of $\nabla_{\boldsymbol{x}_t}\log q(\boldsymbol{x}_t)$ is achieved by $\mathbf{s}_\theta$. The optimization of θ can be achieved by minimizing the variational lower bound (Eq. 22) on negative log-likelihood.

$$\begin{aligned} L_{vlb} = &- \log p_\theta(\boldsymbol{x}_0|\boldsymbol{x}_1) + D_{KL}(q(\boldsymbol{x}_T|\boldsymbol{x}_0))||p(\boldsymbol{x}_T)) \\ &+ \sum_{t>1} D_{KL}(q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t, \boldsymbol{x}_0)||p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)) \end{aligned} \tag{6}$$

After the model is trained well, $\mathbf{s}_\theta$ is a function approximator intended to predict $\boldsymbol{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ from $\boldsymbol{x}_t$. To sample $\boldsymbol{x}_{t-1}$ from the posterior distribution defined in Eq. (4) is equivalent to inverse diffusion through Eq. (7).

$$\boldsymbol{x}_{t-\Delta t} = \boldsymbol{x}_t - [b(\boldsymbol{x}, t) - \sigma^2(t)\mathbf{s}_\theta(\boldsymbol{x}_t, t)]\Delta t + \sigma(t)\boldsymbol{z}. \tag{7}$$

2.2. Semi-discrete Optimal Transport

Suppose the source measure μ defined on a convex domain $\Omega \subset \mathbb{R}^d$, the target domain is a discrete set $\boldsymbol{Y} = \{\boldsymbol{y}_i\}_{i\in\mathcal{I}}, \boldsymbol{y}_i \in \mathbb{R}^d$. The target measure is a Dirac measure $\nu = \sum_{i\in\mathcal{I}} \nu_i \delta(\boldsymbol{y} - \boldsymbol{y}_i)$ and the source measure is equal to total mass as $\mu(\Omega) = \sum_{i\in\mathcal{I}} \nu_i$. Under a semi-discrete transport map $g : \Omega \to \boldsymbol{Y}$, a cell decomposition is induced $\Omega = \bigcup_{i\in\mathcal{I}} W_i$, such that every $\boldsymbol{x}$ in each cell W_i is mapped to the target $\boldsymbol{y}_i, g : \boldsymbol{x} \in W_i \mapsto \boldsymbol{y}_i$. The map g is measure preserving, denoted as $g_{\#}\mu = \nu$, if the μ-volume of each cell W_i equals to the ν-measure of the image $g(W_i) = \boldsymbol{y}_i, \mu(W_i) = \nu_i$. The cost function is given by $c : \Omega \times \boldsymbol{Y} \to \mathbb{R}$, where $c(\boldsymbol{x}, \boldsymbol{y})$ represents the cost for transporting a unit mass from $\boldsymbol{x}$ to $\boldsymbol{y}$. The total cost of transport map $g(x)$ is given by

$$\int_\Omega c(\boldsymbol{x}, g(\boldsymbol{x}))d\mu(\boldsymbol{x}) = \sum_{i\in\mathcal{I}} \int_{W_i} c(\boldsymbol{x}, \boldsymbol{y}_i)\, d\mu(\boldsymbol{x}). \tag{8}$$

The SDOT map g^* is a measure-preserving map that minimizes the total cost in Eq. (8),

$$g^* \coloneqq \arg\min_{g_{\#}\mu=\nu} \int_\Omega c(\boldsymbol{x}, g(\boldsymbol{x}))d\mu(\boldsymbol{x}). \tag{9}$$

Based on **Theorem 1.1** of **supplementary material**, when the cost function $c(\boldsymbol{x}, \boldsymbol{y}) = 1/2\|\boldsymbol{x} - \boldsymbol{y}\|^2$, we have $g^*(\boldsymbol{x}) = \nabla\boldsymbol{u}(\boldsymbol{x})$. This explains that the SDOT map is the gradient map of Brenier's potential $\boldsymbol{u}$. As [27, 1] remark, $\boldsymbol{u}$ is the upper envelope of a collection of hyperplanes $\pi_{\boldsymbol{h},i}(\boldsymbol{x}) = \boldsymbol{x}^T\boldsymbol{y}_i + h_i$ and can be parametrized uniquely up to an additive constant by a height vector $\boldsymbol{h} = (h_1, h_2, ..., h_{|\mathcal{I}|})^T$. In such a case, $\boldsymbol{u}_{\boldsymbol{h}}$ parameterized by $\boldsymbol{h}$ can be stated as follows,

$$\boldsymbol{u}_{\boldsymbol{h}}(\boldsymbol{x}) = \max_{i\in\mathcal{I}}\{\pi_{\boldsymbol{h},i}(\boldsymbol{x})\}, \boldsymbol{u}_{\boldsymbol{h}} : \Omega \to \mathbb{R}^n, \tag{10}$$

Given the target measure ν, there exists Brenier's potential $\boldsymbol{u}_{\boldsymbol{h}}$ in Eq. (10) whose projected volume of each support

plane is equal to the given target measure ν_i. To receive $\boldsymbol{u_h}$, we only need to optimal $\boldsymbol{h}$ by minimizing the following convex energy function:

$$E(\boldsymbol{h}) = \int_{\mathbf{0}}^{\boldsymbol{h}} \sum_{i\in\mathcal{I}} w_i(\eta)d\eta_i - \sum_{i\in\mathcal{I}} h_i\nu_i, \tag{11}$$

where $\omega_i(\eta)$ is the μ-volume of $W_i(\eta)$.

3. Diffusion Probabilistic Model Based on Optimal Transport

Leveraging the generative capabilities of DPMs and the distribution-aligned nature of OT, we propose a fast DPM whose framework is shown in Fig. 1. We give the definition of our **DPM-OT** sampler in Section 3.1 and error analysis in Section 3.3. Section 3.2 describes the proposed fast DPM from the perspective of the algorithm.

3.1. Optimal Trajectory and Sampler

Effective trajectory shortening method has been shown significant for sampling acceleration [46, 3, 2] and high-fidelity generation. To this end, Eric and Salimans et al. [31, 38] recursively distill the 2-step trajectory in the teacher network into a single-step using the knowledge distillation technique. Bao et al. [3] and Watson et al. [46] use dynamic programming to estimate optimal trajectory, which can quickly generate high-quality images. Bao et al. [2] propose to estimate the optimal covariance and its correction given imperfect means by learning these conditional expectations at each time step. Inspired by that prior wisdom, we propose a new SDOT based diffusion model **DPM-OT**, which builds a direct expressway represented by optimal trajectory. Thus, the single-step optimal trajectory replaces the multi-step trajectory in the vanilla DPM. The definition of optimal trajectory is given below.

Definition 3.1. (Optimal Trajectory). Given a M steps trajectory $\{\boldsymbol{x}_{t-i}\}_{i=0}^{M}$ at time t, $M \leq t$, $\boldsymbol{x}_t \in \boldsymbol{X}_t$, the optimal trajectory from $\boldsymbol{x}_t$ to $\boldsymbol{x}_{t-M}$ is a single-step trajectory that is obtained by minimizing the following transport cost:

$$g^* := \arg\min_{g_{\#}\mu=\nu} \int_{\boldsymbol{X}_t} c(\boldsymbol{x}_{t-M}, g(\boldsymbol{x}_t))d\mu(\boldsymbol{x}_t). \tag{12}$$

According to Brenier's theorem, the OT map $g^*(\cdot)$ is the gradient of the convex Brenier potential $\boldsymbol{u_h}$ which satisfies the Monge-Ampére equation. The existence and the uniqueness of the solution to the Monge-Ampére equation have been proved by the Fields medalist Figalli in Chapter 2 of [14], where he used Alexandrov's approach and claimed: i). The sequence of Dirac distributions $\{\nu_n\}$ weakly converges to data distribution ν; ii). For each Dirac measure ν_n, there exists an Alexandrov's solution $\boldsymbol{u}_n$ (which is exactly the discrete Brenier potential in this paper); iii). The weak solution $\boldsymbol{u}_n$ converges to the real solution $\boldsymbol{u_h}$ which is C^1 almost everywhere, except at the singular set. Thus, the OT map $g^*(\cdot)$ is continuous internally and discontinuous at the boundary singular set.

Definition 3.2. (DPM-OT Sampler). Given an initial latent code $\boldsymbol{x}_T \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I})$ at time T, the DPM-OT sampler needs to go through a $M+1$ steps trajectory $\boldsymbol{x}_T \cup \{\boldsymbol{x}_i\}_{i=M}^{0}$, where $M < T$. Let $g(\cdot)$ denote the OT map between $\boldsymbol{x}_T$ and $\boldsymbol{x}_M$, $f_t(\cdot,\cdot)$ denote the parameterized reverse diffusion process which can be any off-the-shelf DPM model. The DPM-OT sampler is defined as follows:

$$\begin{aligned} \boldsymbol{x}_M &= g(\boldsymbol{x}_T), \\ \boldsymbol{x}_{t-1} &= f_t(\boldsymbol{x}_t, \boldsymbol{z}), \quad t = M, ..., 1. \end{aligned} \tag{13}$$

The **DPM-OT Sampler** first transmits the white noise $\boldsymbol{x}_T$ to the manifold represented by the latent variable $\boldsymbol{x}_M$ through the OT map $g(\cdot)$, thus providing a near-perfect initial value for the subsequent inverse diffusion process. From **Definition** 3.1, $g(\cdot)$ is discontinuous at the singular set. Therefore, the manifold $\boldsymbol{x}_M$ can maintain the same attributes as the original data manifold, that is, discontinuity at the boundary singular point, thus avoiding mode mixture. Further, the subsequent M-step inverse diffusion gradually pushes latent variable $\boldsymbol{x}_M$ onto the target data manifold, and finally achieves high-quality data generation. The variable $\boldsymbol{z}$ in $f_t(\cdot,\cdot)$ is $\mathbf{0}$ when $t = 1$ or $\boldsymbol{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ when $t > 1$. M-step inverse diffusion process improves the generation ability of **DPM-OT Sampler**, which essentially completes the continuity of SDOT map.

3.2. Sampling Algorithm

The overall framework of our **DPM-OT** is shown in Fig. 1, which includes an optimal trajectory from $\boldsymbol{x}_T$ to $\boldsymbol{x}_M$ and an M-step inverse diffusion process gradually pushing latent variable $\boldsymbol{x}_M$ onto the target data manifold. We summarize our framework in **Algorithm 1** and **Algorithm 2**. Specifically, to sample high-quality images, we need to compute the SDOT map $g : \boldsymbol{x}_T \to \boldsymbol{x}_M$ by **Algorithm 1**, and then use **Algorithm 2** to generate images.

Given the target dataset $\boldsymbol{Y} = \{\boldsymbol{y}_i\}_{i\in\mathcal{I}}$, our goal is to efficiently generate high-quality images distributed on the manifold represented by $\boldsymbol{Y}$. Intuitively, each point $\boldsymbol{y}_i$ in $\boldsymbol{Y}$ is distributed on the target manifold discretely, so the manifold induced by $\boldsymbol{Y}$ is often discontinuous, especially on the class boundary. Unfortunately, in the generation task, what the community cares about is the coverage and alignment of the entire target data manifold, while the properties of the manifold itself such as the discrete nature of the boundaries are often ignored. As a result, existing DPMs try to approximate a continuous diffusion process with a deep neural network, which leads to mode mixture in their output images.

To accelerate the sampling process and avoid the mode mixture, we first utilize a well-behaved schedule

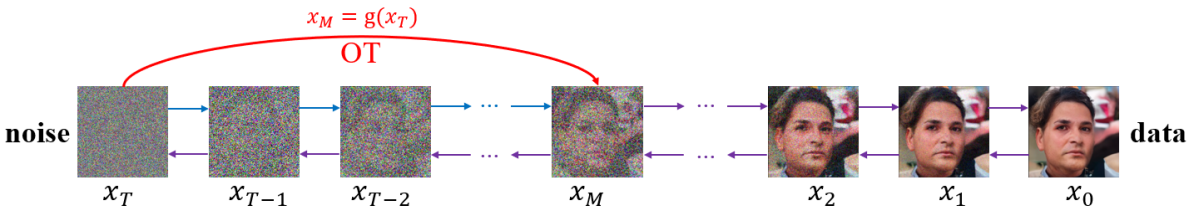


Figure 1. The framework of the proposed **DPM-OT**. The red curve indicates the **Optimal Trajectory**, which is induced by the SDOT map $g(\cdot)$ between $\boldsymbol{x}_T$ and $\boldsymbol{x}_M$. Correspondingly, the blue line indicates the first T-M steps inverse diffusion of the vanilla DPM.

Algorithm 1 SDOT Map

Require: Target dataset $\boldsymbol{Y} = \{\boldsymbol{y}_i\}_{i\in\mathcal{I}}$ with empirical distribution $\nu = \frac{1}{|\mathcal{I}|}\sum_{i\in\mathcal{I}} \nu_i \delta(\boldsymbol{y} - \boldsymbol{y}_i)$, number of Monte Carlo samples N, learning rate lr, threshold τ, positive integer s, reverse diffusion steps M and a well-behaved schedule of $\{(b_t, \sigma_t)\}_{t=0}^T$.
Ensure: OT map $g(\cdot)$.
Initialize $\boldsymbol{h} = (h_1, h_2, \cdots, h_{|\mathcal{I}|}) \leftarrow (0, 0, \cdots, 0)$
Diffuse $\boldsymbol{y}_i$ forwardly by M steps according to Eq. (2), obtain the latents set $\boldsymbol{X}_M = \{\boldsymbol{x}_M^i\}_{i\in\mathcal{I}}$
repeat
  Sample N white noise samples $\{\boldsymbol{x}_T \sim \mathcal{N}(0, I)\}_{j=1}^N$
  Calculate $\nabla \boldsymbol{h} = (\hat{w}_i(\boldsymbol{h}) - \nu_i)^T$.
  $\nabla \boldsymbol{h} = \nabla \boldsymbol{h} - mean(\nabla \boldsymbol{h})$.
  Update $\boldsymbol{h}$ by Adam algorithm with $\beta_1 = 0.9$,$\beta_2 = 0.5$.
  Calculate $E(\boldsymbol{h})$ by Eq. (11)
  if $E(\boldsymbol{h})$ has not decreased for s steps **then**
    $N \leftarrow 2 \times N$; $lr \leftarrow 0.8 \times lr$
  end if
until $E(\boldsymbol{h}) < \tau$
OT map $g(\cdot) \leftarrow \nabla(\max_i\{\langle \boldsymbol{x}_T, \boldsymbol{x}_M^i\rangle_F + h_i\})$.

$\{(b_t, \sigma_t)\}_{t=0}^T$ to diffuse the original data $\boldsymbol{y}$ into the latent variable $\boldsymbol{x}_M$ and then calculate the OT map between $\boldsymbol{x}_T$ and $\boldsymbol{x}_M$, which induces the **Optimal Trajectory** from $\boldsymbol{x}_T$ to $\boldsymbol{x}_M$. Considering that both $\boldsymbol{x}_T$ and $\boldsymbol{x}_M$ are matrices, we set the Brenier's potential $\boldsymbol{u_h} = \max_{i\in\mathcal{I}}\{\langle \boldsymbol{x}_T, \boldsymbol{x}_M^i\rangle_F + h_i\}$, where $\langle\cdot,\cdot\rangle_F$ denotes Frobenius inner product. In **Algorithm 1**, we use the Monte Carlo method to solve the SDOT map. For better convergence, we double the number of samples N and multiply the learning rate lr by 0.8 when the energy function $E(\boldsymbol{h})$ has not decreased for s steps.

After obtaining the SDOT map $g(\cdot)$, we sample with the help of an off-shelf pre-trained model and summarize the process in **Algorithm 2**. With the optimal trajectory, our **DPM-OT Sampler** can perform sampling quickly and with few mode mixture. Given different schedule $\{(b_t, \sigma_t)\}_{t=0}^T$, our framework **DPM-OT** will be instantiated into different fast DPM models. Specifically, if we use the Langevin dynamic in NCSNv2 [42] to instantiate the diffusion process, we can obtain the following sampling process:

$$\boldsymbol{x}_{t-1} = \boldsymbol{x}_t + \sigma^2(t)\mathbf{s}_\theta(\boldsymbol{x}_t, t) + \sigma(t)\boldsymbol{z},\ t = M, ..., 1. \quad (14)$$

Algorithm 2 DPM-OT Sampling

Require: Reverse diffusion steps M, OT map $g(\cdot)$, a well-trained $\mathbf{s}_\theta$ and a well-behaved schedule of $\{(b_t, \sigma_t)\}_{t=0}^T$.
Ensure: Generated image $\boldsymbol{x}_0$.
Sample $\boldsymbol{x}_T \sim \mathcal{N}(0, I)$
$\boldsymbol{x}_M = g(\boldsymbol{x}_T)$
for $t = M$ **to** 1 **do**
  $\boldsymbol{z} \sim \mathcal{N}(0, I)$ if $t > 1$, else $\boldsymbol{z} = 0$
  $\boldsymbol{x}_{t-1} = \boldsymbol{x}_t - [b(\boldsymbol{x}, t) - \sigma^2(t)\mathbf{s}_\theta(\boldsymbol{x}_t, t)] + \sigma(t)\boldsymbol{z}$
end for
return $\boldsymbol{x}_0$

3.3. Error Analysis

In this section, we analyze the error bound of **DPM-OT**. First, we prove that the single-step error is controllable. Then, we give the upper bound of the error between the generated data distribution and the target data distribution.

Theorem 3.3. *Let $\tilde{\boldsymbol{x}}_t$ and $\boldsymbol{x}_t$ be the samples of step t obtained by **DPM-OT** and forward diffusion respectively, and $t \leqslant M$, $\boldsymbol{\zeta}_M$ be the error at step M induced by optimal trajectory, then there is a constant $C_t > 0$ satisfies*

$$\|\tilde{\boldsymbol{x}}_t - \boldsymbol{x}_t\| \leqslant C_t \|\boldsymbol{\zeta}_M\|. \quad (15)$$

Since the weak solution $\boldsymbol{u}_n$ converge to the real solution $\boldsymbol{u_h}$ [14, 1, 27] and the OT map $g(\cdot)$ is obtained by the Monte Carlo method, its error $\boldsymbol{\zeta}_M$ is $O(N^{-\frac{1}{2}})$ according to **Theorem 2.1** of [6]. So we can find a small enough error $\boldsymbol{\zeta}_M$ to make $\|\tilde{\boldsymbol{x}}_t - \boldsymbol{x}_t\| \leqslant \delta$ hold for any given error bound $\delta > 0$, which indicates the error of **DPM-OT** is controllable. Building upon **Theorem A.2**, we obtain **Theorem 3.4** and give their proofs in **supplementary material**.

Theorem 3.4. *Let L_{dpm_ot} be the error between the data distribution generated by **DPM-OT** and the target data distribution which is defined in Eq. (21), L_{vlb} is the variational lower bound on negative log-likelihood between data distribution generated by vanilla DPM and the target data distribution which is defined in Eq. (22). We have $L_{dpm\_ot} \leqslant$*

Table 1. Sample quality measured by FID ↓ on CIFAR-10, CelebA, and FFHQ, the number of function evaluations(NFE) is the number of times the neural network called.

Dataset	Models	NFE	FID ↓
CIFAR-10	DDIM [41]	100	4.60
	NCSNv2 [42]	1160	10.23
	DPM-Solver [30]	20	3.72
	EDM [20]	27	3.73
	UniPC [48]	10	3.87
	DPM-OT	5	3.78
		10	3.61
		20	3.33
		30	3.12
		50	**2.92**
CelebA 64 × 64	DDIM [41]	100	6.53
	NCSNv2 [42]	2500	10.23
	DPM-Solver [30]	20	3.13
	Analytic-DDIM [3]	10	15.62
	Analytic-DDIM [3]	50	6.13
	DPM-OT	5	3.30
		10	3.21
		20	3.12
		30	3.01
		50	**2.85**
FFHQ 256 × 256	DPM-Solver [30]	10	7.39
	UniPC [48]	10	6.99
	NCSNv2 [42]	6933	12.73
	DPM-OT	5	4.69
		10	4.46
		20	4.32
		30	4.26
		50	**4.11**

L_{vlb}. Hence, L_{vlb} is the upper bound of L_{dpm_ot}.

$$\begin{aligned} L_{dpm\_ot} &= L_0 + L_1 + ... + L_M + L_T \\ &= -\log \tilde{p}_\theta(\boldsymbol{x}_0|\boldsymbol{x}_1) + D_{KL}(q(\boldsymbol{x}_T|\boldsymbol{x}_0)), p(\boldsymbol{x}_T)) \\ &+ \sum_{t=1}^{M-1} D_{KL}(q(\boldsymbol{x}_t|\boldsymbol{x}_{t+1}, \boldsymbol{x}_0)||\tilde{p}_\theta(\boldsymbol{x}_t|\boldsymbol{x}_{t+1})) \\ &+ D_{KL}(q(\boldsymbol{x}_M|\boldsymbol{x}_T, \boldsymbol{x}_0)||\tilde{p}_\theta(\boldsymbol{x}_M|\boldsymbol{x}_T)), \end{aligned} \tag{16}$$

where $\tilde{p}_\theta(\boldsymbol{x}_M|\boldsymbol{x}_T) = p_\theta(\boldsymbol{x}_M - \boldsymbol{\zeta}_M|\boldsymbol{x}_T)$, $\tilde{p}_\theta(\boldsymbol{x}_t|\boldsymbol{x}_{t+1}) = p_\theta(\boldsymbol{x}_t - \boldsymbol{\zeta}_t|\boldsymbol{x}_{t+1})$ and $\boldsymbol{\zeta}_t = \tilde{\boldsymbol{x}}_t - \boldsymbol{x}_t$ for $t = 0$ to $t = M-1$. **Theorem 3.4** shows that our **DPM-OT** sampler can fit the target data distribution no worse than vanilla DPMs, which theoretically guarantees the robustness of the algorithm.

4. Experiments

Section 3 analyzes the benefits of our model from the theoretical perspective, and then we will further evaluate the performance of the model experimentally. The experimental results indicate that the **DPM-OT** has the following pros: **1)** it can be well embedded in pre-trained diffusion models to accelerate sampling; **2)** the generation efficiency of our model has been drastically enhanced, and high-quality images can be generated with only 5 function evaluations; **3)** mode mixture can be alleviated via the proposed method.

Table 2. Comparison of precision ↑ and recall ↑ .

Dataset	Models	Precision ↑	Recall ↑
CelebA 64 × 64	DDIM [41]	0.75	0.42
	NCSNv2 [42]	**0.85**	0.42
	DPM-Solver [30]	0.71	0.46
	DPM-OT	<u>0.79</u>	**0.78**

More specifically, we instantiate our framework **DPM-OT** with pre-trained models of NCSNv2 [42] on CIFAR10 [24], CelebA [29] and FFHQ [21] respectively. Compared to the NCSNv2, our method has a great improvement in sampling speed. NCSNv2 needs 1160, 2500, and 6933 function evaluations on the corresponding three datasets to get preferable images, while our model can get high-quality images with only 5 function evaluations. Furthermore, we employ FID score [16] and the improved precision and recall metric [25] to assess its performance against the SOTA models on the above three datasets. These SOTA models include DDIM [41], Analytic-DDIM [3], DPM-Solver [30], EDM [20], UniPC [48].

To measure the ability of generative models to generate images with obvious categories, that is, not to produce images that do not exist in the real world with multiple categories mixed together, we design a mode mixture indicator to judge whether the image is mixed. Exactly, for each generated image, we use a pre-trained classification model which is the SOTAs model in the community to evaluate its probability of belonging to each category. If the image has probabilities greater than a given threshold λ on more than two categories, it is identified with mode mixture. Given the classification model $cls(\boldsymbol{y})$=$(p_1^{\boldsymbol{y}}, p_2^{\boldsymbol{y}}, ..., p_C^{\boldsymbol{y}})$, number of categories C, the definition of **mode mixture indicator** is:

$$\mathbf{I}(\boldsymbol{y}) = \begin{cases} 1, & |\{p_j^{\boldsymbol{y}} | p_j^{\boldsymbol{y}} \geq \lambda\}| \geq 2 \\ 0, & others, \end{cases} \quad j = 1, \cdots, C, \tag{17}$$

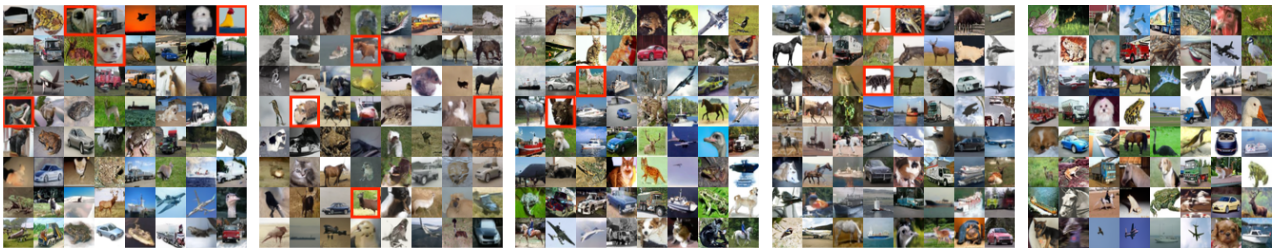
where $|\cdot|$ denotes the number of elements in the set, $p_j^{\boldsymbol{y}}$ represents the probability that the image $\boldsymbol{y}$ belongs to the j-th category. Then, We use mode mixture ratio (**MMR**) to measure the performance of the generative model in avoiding mode mixture, which is defined as follows:

$$\mathbf{MMR} = \frac{1}{K} \sum_{i=1}^{K} \mathbf{I}(\boldsymbol{y}_i), \quad i = 1, \cdots, K, \tag{18}$$

where K is the number of images. $\mathbf{I}(\boldsymbol{y}_i)$ indicates the prediction of whether there is a mode mixture on i-th image.

Table 3. Comparison of Mode mixture in **CIFAR-10** dataset. NIG: the number of images generated, threshold: the image has a probability greater than a given threshold on more than two categories, it is identified as mode mixture.

NIG	models	NFE	threshold				
			$\lambda = 0.1$	$\lambda = 0.11$	$\lambda = 0.13$	$\lambda = 0.16$	$\lambda = 0.2$
49984	DDIM	100	3834(7.69%)	3626(7.25%)	3306(6.61%)	2859(5.71%)	2416(4.83%)
50000	DPM-solver	20	3824(7.65%)	3614(7.22%)	3295(6.59%)	2868(5.73%)	2397(4.79%)
50000	EDM	27	1579(3.16%)	1512(3.02%)	1379(2.75%)	1203(2.41%)	980(1.96%)
50000	NCSNv2	1160	8876(17.75%)	8217(16.43%)	7463(14.92%)	6527(13.05%)	5552(11.10%)
50000	**DPM-OT**	5	**699(1.40%)**	**743(1.48%)**	**678(1.35%)**	**585(1.17%)**	**426(0.85%)**
		10	**778(1.56%)**	**746(1.49%)**	**674(1.34%)**	**590(1.18%)**	**492(0.98%)**
		20	**829(1.66%)**	**835(1.67%)**	**746(1.49%)**	**645(1.29%)**	**527(1.05%)**
		30	**840(1.68%)**	**834(1.66%)**	**749(1.49%)**	**645(1.29%)**	**496(0.99%)**
		50	**984(1.97%)**	**951(1.90%)**	**872(1.74%)**	**760(1.52%)**	**622(1.24%)**



(a) DDIM (b) NCSNv2 (c) EDM (d) DPM-solver (e) DPM-OT

Figure 2. The visual comparison on CIFAR-10 dataset.

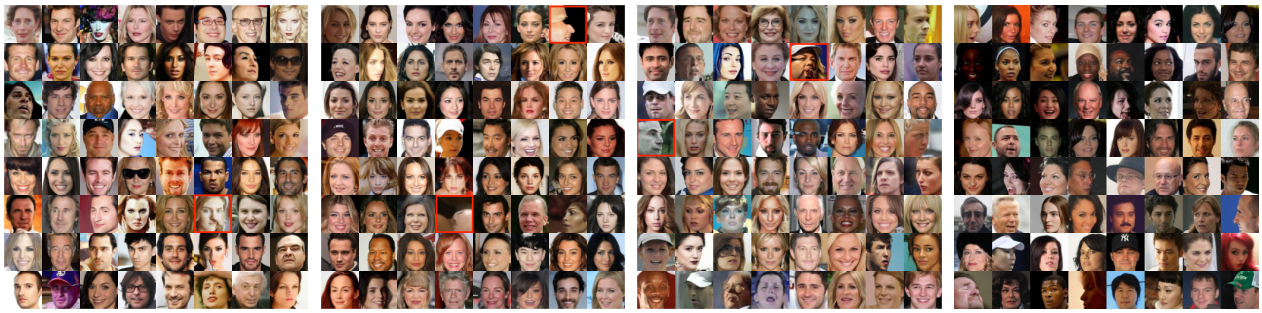
MMR is a simple but effective metric, which circumvents the dilemma of accuracy calculations but no labels. The lower its value, the fewer images with mode mixture.

4.1. Sample Quality and Efficiency

To illustrate the remarkable performance of the **DPM-OT**, we give its analysis of sampling quality and efficiency. As the universal evaluation metrics of image generation, the FID scores for our model on CIFAR-10, CelebA, and FFHQ datasets are reported in Tab. 1. Moreover, in order to evaluate the quality of the generated models in several dimensions, we give the results of precision and recall in Tab. 2 on CelebA. Precision is quantified by querying for each generated image whether the image is within the estimated manifold of real images. Symmetrically, recall is calculated by querying for each real image whether the image is within the estimated manifold of the generated image.

Combining the above metrics, the following result analysis is provided. Results on CIFAR-10, the FID of our model outperforms other fast DPM methods, this shows that our model is able to improve the image quality while accelerating sampling speed. And compared with DDIM, NCSNv2, EDM, and DPM-solver, the images generated by our method are much sharper, while other models have some images with mode mixture. Such as a bird head with a horse body, as marked in the red box of Fig. 2. Results on celebA, Tab. 1 and Fig. 3 provide quality evaluation and visualization comparisons in CelebA, respectively. Our model only needs 5 function evaluaitons to achieve excellent performance in image quality assessment and compared to the other models in Tab. 1, the **DPM-OT** is superior in terms of both speed and quality. Although DDIM, NCSNv2, and DPM-solver get highly realistic images, their results appear to some not exist face images in the real world caused by mode mixture, such as the red box marks in Fig. 3, while our method does not arise in this case. Results on FFHQ, in comparison to DPM-solver and UniPC, our method attains superior results after 5 times of neural network inference. Fig. 3 shows the **DPM-OT** generated high fidelity and almost no mixture images, yet the NCSNv2 appears some deformed face images in Fig. 3 with a marked red box. In addition to this, The results of precision and recall of our model on CelebA are shown in Tab. 2. Compared with NCSNv2, though the precision of our model is slightly lower, the recall is over a lot, which shows that our model has enhanced in both diversity and image quality.

In summary, the quantitative evaluation and visualized results demonstrate our model has superior performance. In particular, the sampling speed has been greatly enhanced, requiring only $5-10$ NFEs to generate high-quality images.



(a) DDIM (b) NCSNv2 (c) DPM-solver (d) DPM-OT

Figure 3. The visual comparison on CelebA dataset.

(a) NCSNv2

(b) DPM-OT

Figure 4. The visual comparison on FFHQ dataset.

4.2. Validity of Mode Mixture Indicator

To verify the validity of the mode mixture indicator, i.e., Eq. (17), we conduct empirical analysis and give the relevant examples and results which are reported in Fig. 5 and Fig. 6. We randomly select an image with mode mixture and an image with significant category characteristics to obtain classification results by the pre-training model ViT-B16 [12], judging whether the image is with mode mixture or not. Fig. 5 shows that the probability of a horse and deer in the mixed image is 0.3 and 0.4, respectively, this also aligns with our visual perception which is a mixture of a horse and a deer. While the other image can immediately be recognized as an automobile, and the output probability of the classification model is close to 1, this also indicates that the selected classification model is effective, and the indicator defined in Eq. (17) can effectively detect mode mixture.

Furthermore, we further randomly collected 100 clear images from CIFAR-10 and 50 images with mode mixture from generated results, the accuracy of the indicator's results are shown in Fig. 6. We found that the mode mixture indicator has a great ability to recognize clear images. Moreover, when the threshold λ is about 0.11, the image with mode mixture can be effectively identified. In summary of the results, it is reasonable to assume the proposed indicator is effective in the detection of mode mixture.

4.3. Mode Mixture Analysis

This part will demonstrate that the **DPM-OT** can mitigate mode mixture from experimental results. We devised a new metric to quantitatively assess model mixture, i.e. Eq. (18), which assumes that no less than two components of the probability vector are greater than the setup threshold. Furthermore, we utilize the best pre-trained classification model ViT-B16 to count the number of blended images. The results of **MMR** are reported in Tab. 3, with a nearly consistent number of images, we calculate the ratio of obfuscated images under the corresponding threshold. According to the results in Section 4.2, we know that when the value of γ is around 0.1, the prediction of the mode mixture indicator on clear images and mixed images will be more accurate, so we have selected five values in the interval $[0.1, 0.2]$ for comparative experiments. Judging from the results, the **DPM-OT** consistently maintains excellent

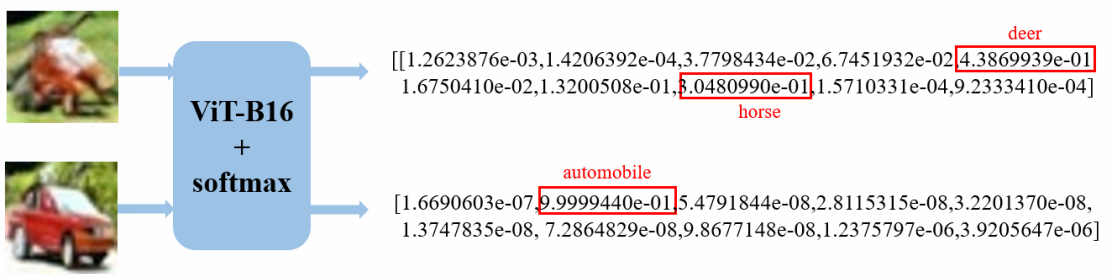


Figure 5. Examples of verifying mode mixture indicator. The image of the top row and the bottom row are judged as existence and nonexistence mode mixture by the indicator in Eq. (17).

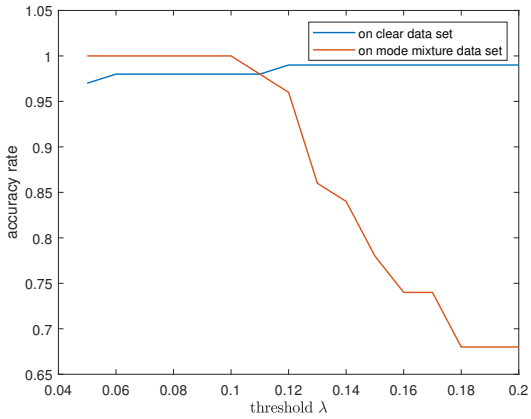


Figure 6. Randomly chosen mixed and unambiguous images to obtain the mode mixture indicator results under different thresholds.

performance under different thresholds, which indicates our model can effectively alleviate mode mixture.

In addition, the visualized results further illustrate the ability of our model to mitigate mode mixture in Fig. 2 $\sim$ Fig. 4. The visual results on the CIFAR-10 in Fig. 2, (a) $\sim$ (d) appear confusing images, such as a red box marked either a variety of animal images mixed together or blurred and unclear. And for face image results on the celebA 64×64 in Fig. 3, DDIM has a broken face formed by mixing faces of different sizes, NCSNv2 produces a completely deformed face, DPM-solver generates strange unnatural faces. This also demonstrates that the **DPM-OT** can better mitigate mode mixture on face images. The results of the high-resolution face image on FFHQ 256×256 are displayed in Fig. 4, Although NCSNv2 generates clean face images, there are still existed missing ears and overlapping facial features. The high-quality unconfused images are generated by our model. To sum up, from the **MMR** and visual results, the output results of other models have a serious mode mixture, and the **MMR** of our model maintains a very small prediction rate, this shows that our proposed method can well mitigate this problem and also improves the quality of images. We then briefly analyze the reason for mode mixture via optimal transport theory.

According to the regularity theory of Monge-Ampére, if the support set of the target distribution is non-convex, then there exists a singular set of points with zero measure and the transport mapping is interrupted at the singular points. While traditional deep neural networks (DNN) can only approximate continuous maps, this internal conflict leads to mode collapse/mixture. We are aware that the transformation from noise to generated images involves a transition from continuous to discrete distribution, which results in singular boundaries, and the images generated at the boundary points inevitably appear mode mixture. While existing DPMs attempt to use DNN to approximate the continuous diffusion process, which arise intrinsic conflict and cause mode mixture. Our method calculates the SDOT map from $\boldsymbol{x}_T$ to $\boldsymbol{x}_M$ to avoid using DNNs approximate continuous map, which can eliminate mode mixture at the first step of **DPM-OT** sampler. Therefore, it can effectively mitigate mode mixture of our model.

5. Conclusion

In this paper, we propose a novel fast sampling diffusion probability model in combination with optimal transport, i.e. **DPM-OT**, which can generate high-quality images while greatly speeding up the sampling. Our method built an optimal trajectory from the prior distribution to the target latents distribution by calculating the SDOT map between them. The optimal trajectory provides a near-perfect initial value for the subsequent diffusion process through a single-step transmission, which greatly shortens the sampling trajectory, thus improving the sampling efficiency. Moreover, the discontinuity of the SDOT map at the boundary singular point dramatically alleviates the problem of mode mixture in the generated image. Furthermore, the error bound of the proposed method is provided, which theoretically guarantees the stability of the algorithm. To detect mode mixture without labels, an effective indicator is proposed and verified. Extensive experiments validate the proposed **DPM-OT** can generate high-quality samples with almost no mode mixture within only $5-10$ function evaluations.

Limitations One limitation of our approach is that the noisy training data samples $\boldsymbol{x}_M$ need to be stored for use at sampling time. This means additional storage requirements, although we have designed batch-processing algorithms to reduce the demand for device storage. Another limitation is that we just only consider unconditional generation here. In future research, it would be interesting to incorporate the **DPM-OT** framework into conditional synthesis tasks.

References

[1] Dongsheng An, Yang Guo, Na Lei, Zhongxuan Luo, Shing-Tung Yau, and Xianfeng Gu. Ae-ot: a new generative model based on extended semi-discrete optimal transport. *ICLR 2020*, 2019. 1, 2, 4

[2] Fan Bao, Chongxuan Li, Jiacheng Sun, Jun Zhu, and Bo Zhang. Estimating the optimal covariance with imperfect mean in diffusion probabilistic models. *arXiv preprint arXiv:2206.07309*, 2022. 3

[3] Fan Bao, Chongxuan Li, Jun Zhu, and Bo Zhang. Analytic-DPM: An analytic estimate of the optimal reverse variance in diffusion probabilistic models. In *International Conference on Learning Representations*, 2022. 1, 3, 5

[4] Yann Brenier. Polar decomposition and increasing rearrangement of vector-fields. *COMPTES RENDUS DE L ACADEMIE DES SCIENCES SERIE I-MATHEMATIQUE*, 305(19):805–808, 1987. 1

[5] Yann Brenier. Polar factorization and monotone rearrangement of vector-valued functions. *Communications on pure and applied mathematics*, 44(4):375–417, 1991. 1

[6] Russel E Caflisch. Monte carlo and quasi-monte carlo methods. *Acta numerica*, 7:1–49, 1998. 4

[7] Nanxin Chen, Yu Zhang, Heiga Zen, Ron J. Weiss, Mohammad Norouzi, and William Chan. Wavegrad: Estimating gradients for waveform generation. In *International Conference on Learning Representations*, 2021. 1

[8] Nanxin Chen, Yu Zhang, Heiga Zen, Ron J. Weiss, Mohammad Norouzi, Najim Dehak, and William Chan. Wavegrad 2: Iterative refinement for text-to-speech synthesis. In *International Speech Communication Association*, pages 3765–3769, 2021. 1

[9] Shibing Chen and Alessio Figalli. Partial w2, p regularity for optimal transport maps. *Journal of Functional Analysis*, 272(11):4588–4605, 2017. 1

[10] Jooyoung Choi, Sungwon Kim, Yonghyun Jeong, Youngjune Gwon, and Sungroh Yoon. Ilvr: Conditioning method for denoising diffusion probabilistic models. in 2021 ieee. In *CVF International Conference on Computer Vision (ICCV)*, pages 14347–14356, 2021. 1

[11] Prafulla Dhariwal and Alexander Quinn Nichol. Diffusion models beat GANs on image synthesis. In *Advances in Neural Information Processing Systems*, volume 34, pages 8780–8794, 2021. 1

[12] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 7

[13] Alessio Figalli. Regularity properties of optimal maps between nonconvex domains in the plane. *Communications in Partial Differential Equations*, 35(3):465–479, 2010. 1

[14] Alessio Figalli. *The Monge–Ampère equation and its applications*. 2017. 3, 4

[15] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron C. Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in Neural Information Processing Systems*, volume 27, pages 2672–2680, 2014. 1

[16] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. GANs trained by a two time-scale update rule converge to a local Nash equilibrium. In Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30, pages 6626–6637, 2017. 5

[17] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems*, volume 33, pages 6840–6851, 2020. 1, 2

[18] Jonathan Ho, Tim Salimans, Alexey Gritsenko, William Chan, Mohammad Norouzi, and David J Fleet. Video diffusion models. *arXiv preprint arXiv:2204.03458*, 2022. 1

[19] Alexia Jolicoeur-Martineau, Ke Li, Rémi Piché-Taillefer, Tal Kachman, and Ioannis Mitliagkas. Gotta go fast when generating data with score-based models. *arXiv preprint arXiv:2105.14080*, 2021. 1

[20] Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models. *arXiv preprint arXiv:2206.00364*, 2022. 5

[21] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 5, 12

[22] Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. In *International Conference on Learning Representations*, 2014. 1

[23] Zhifeng Kong and Wei Ping. On fast sampling of diffusion probabilistic models. *arXiv preprint arXiv:2106.00132*, 2021. 1

[24] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009. 5, 12

[25] Tuomas Kynkäänniemi, Tero Karras, Samuli Laine, Jaakko Lehtinen, and Timo Aila. Improved precision and recall metric for assessing generative models. *Advances in Neural Information Processing Systems*, 32, 2019. 5

[26] Max WY Lam, Jun Wang, Rongjie Huang, Dan Su, and Dong Yu. Bilateral denoising diffusion models. *arXiv preprint arXiv:2108.11514*, 2021. 1

[27] Na Lei, Dongsheng An, Yang Guo, Kehua Su, Shixia Liu, Zhongxuan Luo, Shing-Tung Yau, and Xianfeng Gu. A geometric understanding of deep learning. *Engineering*, 6(3):361–374, 2020. 2, 4, 11

[28] Luping Liu, Yi Ren, Zhijie Lin, and Zhou Zhao. Pseudo numerical methods for diffusion models on manifolds. In *International Conference on Learning Representations*, 2022. 1

[29] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3730–3738, 2015. 5, 12

[30] Cheng Lu, Yuhao Zhou, Fan Bao, Jianfei Chen, Chongxuan Li, and Jun Zhu. Dpm-solver: A fast ode solver for diffusion probabilistic model sampling in around 10 steps. *arXiv preprint arXiv:2206.00927*, 2022. 1, 5

[31] Eric Luhman and Troy Luhman. Knowledge distillation in iterative generative models for improved sampling speed. *arXiv preprint arXiv:2101.02388*, 2021. 1, 3

[32] Shitong Luo and Wei Hu. Diffusion probabilistic models for 3d point cloud generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2837–2845, 2021. 1

[33] Chenlin Meng, Yang Song, Jiaming Song, Jiajun Wu, Jun-Yan Zhu, and Stefano Ermon. SDEdit: Image synthesis and editing with stochastic differential equations. In *International Conference on Learning Representations*, 2022. 1

[34] Alex Nichol, Heewoo Jun, Prafulla Dhariwal, Pamela Mishkin, and Mark Chen. Point-e: A system for generating 3d point clouds from complex prompts. *arXiv preprint arXiv:2212.08751*, 2022. 1

[35] Alexander Quinn Nichol and Prafulla Dhariwal. Improved denoising diffusion probabilistic models. In *International Conference on Machine Learning*, pages 8162–8171. PMLR, 2021. 1

[36] Vadim Popov, Ivan Vovk, Vladimir Gogoryan, Tasnima Sadekova, Mikhail Kudinov, and Jiansheng Wei. Diffusion-based voice conversion with fast maximum likelihood sampling scheme. In *International Conference on Learning Representations*, 2022. 1

[37] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with CLIP latents. *arXiv preprint arXiv:2204.06125*, 2022. 1

[38] Tim Salimans and Jonathan Ho. Progressive distillation for fast sampling of diffusion models. In *International Conference on Learning Representations*, 2022. 1, 3

[39] Robin San-Roman, Eliya Nachmani, and Lior Wolf. Noise estimation for generative diffusion models. *arXiv preprint arXiv:2104.02600*, 2021. 1

[40] 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. 1

[41] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *International Conference on Learning Representations*, 2021. 1, 2, 5

[42] Yang Song and Stefano Ermon. Improved techniques for training score-based generative models. *Advances in neural information processing systems*, 33:12438–12448, 2020. 4, 5

[43] Yang Song, Jascha Sohl-Dickstein, Diederik P. Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *International Conference on Learning Representations*, 2021. 1, 2

[44] Hideyuki Tachibana, Mocho Go, Muneyoshi Inahara, Yotaro Katayama, and Yotaro Watanabe. Quasi-Taylor sampling scheme for denoising diffusion probabilistic models using ideal derivatives. *arXiv preprint arXiv:2112.13339*, 2021. 1

[45] Cédric Villani. *Optimal transport: old and new*, volume 338. Springer Science & Business Media, 2008. 11

[46] Daniel Watson, William Chan, Jonathan Ho, and Mohammad Norouzi. Learning fast samplers for diffusion models by differentiating through sample quality. In *International Conference on Learning Representations*, 2022. 1, 3

[47] Xiaohui Zeng, Arash Vahdat, Francis Williams, Zan Gojcic, Or Litany, Sanja Fidler, and Karsten Kreis. Lion: Latent point diffusion models for 3d shape generation. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022. 1

[48] Wenliang Zhao, Lujia Bai, Yongming Rao, Jie Zhou, and Jiwen Lu. Unipc: A unified predictor-corrector framework for fast sampling of diffusion models. *arXiv preprint arXiv:2302.04867*, 2023. 1, 5

[49] Linqi Zhou, Yilun Du, and Jiajun Wu. 3d shape generation and completion through point-voxel diffusion. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5826–5835, 2021. 1

In this document, we provide proof of theorems, additional implementation details, and qualitative results. Limitations and future work are also discussed.

A. Proof of Theorems

Theorem A.1. *[27, 45] Given μ and ν on a compact convex domain $\Omega \subset \mathbb{R}^d$, there exists an OT plan for the cost $c(\boldsymbol{x}, \boldsymbol{y}) = \boldsymbol{g}(\boldsymbol{x} - \boldsymbol{y})$, with $\boldsymbol{g}$ strictly convex. It is unique and of the form (id, $T_{\#}$) μ (id: identity map), provided that μ is absolutely continuous with respect to Lebesgue measure and $\partial\Omega$ is negligible. Moreover, there exists a Kantorovich's potential φ, and OT map T can be represented as follows:*

$$T(\boldsymbol{x}) = \boldsymbol{x} - (\nabla \boldsymbol{g})^{-1}[\nabla \varphi(\boldsymbol{x})].$$

Theorem A.2. *Let $\tilde{\boldsymbol{x}}_t$ and $\boldsymbol{x}_t$ be the samples of step t obtained by **DPM-OT** and forward diffusion respectively, and $t \leq M$, $\boldsymbol{\zeta}_M$ be the error at step M induced by optimal trajectory, then there is a constant $C_t > 0$ satisfies the following inequality.*

$$\|\tilde{\boldsymbol{x}}_t - \boldsymbol{x}_t\| \leq C_t \|\boldsymbol{\zeta}_M\| \tag{19}$$

Proof. Since the reverse diffusion function sequence $\{f_t\}$ is continuous, there is continuous function $c_t(\cdot)$ from $\mathbb{R}^d$ to $\mathbb{R}$ that makes the following formula hold

$$\begin{aligned} \tilde{\boldsymbol{x}}_t &= f_t \circ \cdots \circ f_{M-1}(\boldsymbol{x}_M + \boldsymbol{\zeta}_M) \\ &= f_t \circ \cdots \circ f_{M-1}(\boldsymbol{x}_M) + c_t(\boldsymbol{\zeta}_M)\boldsymbol{\zeta}_M \\ &= \boldsymbol{x}_t + c_t(\boldsymbol{\zeta}_M)\boldsymbol{\zeta}_M \end{aligned} \tag{20}$$

So we can get

$$\|\tilde{\boldsymbol{x}}_t - \boldsymbol{x}_t\| = \|c_t(\boldsymbol{\zeta}_M)\boldsymbol{\zeta}_M\| \leq |c_t(\boldsymbol{\zeta}_M)| \, \|\boldsymbol{\zeta}_M\| \leq C_t \|\boldsymbol{\zeta}_M\|$$

□

Theorem A.3. *Let L_{dpm_ot} be the error between the data distribution generated by **DPM-OT** and the target data distribution which is defined in Eq. 21, L_{vlb} is the variational lower bound on negative log-likelihood between data distribution generated by vanilla DPM and the target data distribution which is defined in Eq. 22. We have $L_{dpm\_ot} \leqslant L_{vlb}$, i.e., L_{vlb} is the upper bound of L_{dpm_ot}.*

$$\begin{aligned} L_{dpm\_ot} =& L_0 + L_1 + ... + L_M + L_T \\ =& -\log \tilde{p}_\theta(\boldsymbol{x}_0|\boldsymbol{x}_1) + D_{KL}(q(\boldsymbol{x}_T|\boldsymbol{x}_0)), p(\boldsymbol{x}_T)) \\ &+ \sum_{t=1}^{M-1} D_{KL}(q(\boldsymbol{x}_t|\boldsymbol{x}_{t+1}, \boldsymbol{x}_0)||\tilde{p}_\theta(\boldsymbol{x}_t|\boldsymbol{x}_{t+1})) \\ &+ D_{KL}(q(\boldsymbol{x}_M|\boldsymbol{x}_T, \boldsymbol{x}_0)||\tilde{p}_\theta(\boldsymbol{x}_M|\boldsymbol{x}_T)), \end{aligned} \tag{21}$$

$$\begin{aligned} L_{vlb} =& -\log p_\theta(\boldsymbol{x}_0|\boldsymbol{x}_1) + D_{KL}(q(\boldsymbol{x}_T|\boldsymbol{x}_0))||p(\boldsymbol{x}_T)) \\ &+ \sum_{t>1} D_{KL}(q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t, \boldsymbol{x}_0)||p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)) \end{aligned} \tag{22}$$

Proof. It may be assumed that$\boldsymbol{X}_t = \boldsymbol{x}_t - \mu_{t+1}(\boldsymbol{x}_{t+1})$,

$$\begin{aligned} p_\theta(\boldsymbol{x}_t|\boldsymbol{x}_{t+1}) &= \mathcal{N}(\boldsymbol{x}_t|\mu_{t+1}(\boldsymbol{x}_{t+1}), \sigma_{t+1}^2 I) \\ &= \frac{1}{(2\pi)^{d/2}\sigma_{t+1}^d} \exp\left(-\frac{1}{2\sigma_{t+1}^2}\boldsymbol{X}_t^T \boldsymbol{X}_t\right) \end{aligned} \tag{23}$$

$$\begin{aligned} p_\theta(\tilde{\boldsymbol{x}}_t|\boldsymbol{x}_{t+1}) &= \mathcal{N}(\boldsymbol{x}_t + \boldsymbol{\zeta}_t|\boldsymbol{\mu}_{t+1}(\boldsymbol{x}_{t+1}), \sigma_{t+1}^2 I) \\ = \frac{1}{(2\pi)^{d/2}\sigma_{t+1}^d} \exp&\left(-\frac{1}{2\sigma_{t+1}^2}\left(\boldsymbol{X}_t - \boldsymbol{\zeta}_t\right)^T(\boldsymbol{X}_t - \boldsymbol{\zeta}_t)\right) \end{aligned} \tag{24}$$

By definitions of $p_\theta(\boldsymbol{x}_t|\boldsymbol{x}_{t+1})$ and $p_\theta(\tilde{\boldsymbol{x}}_t|\boldsymbol{x}_{t+1})$, we know the following equation

$$\begin{aligned}
&\left|\log \frac{p_\theta(\boldsymbol{x}_t|\boldsymbol{x}_{t+1})}{p_\theta(\tilde{\boldsymbol{x}}_t|\boldsymbol{x}_{t+1})}\right| \\
=&\frac{1}{2\sigma_{t+1}^2}\left|2\boldsymbol{x}_t\boldsymbol{\zeta}_t + \boldsymbol{\zeta}_t^T\boldsymbol{\zeta}_t - 2\boldsymbol{\mu}_{t+1}(\boldsymbol{x}_{t+1})^T\boldsymbol{\zeta}_t\right| \\
\leq&\frac{1}{2\sigma_{t+1}^2}\left(2||\boldsymbol{x}_t|| \cdot ||\boldsymbol{\zeta}_t|| + ||\boldsymbol{\zeta}_t||^2 + 2||\boldsymbol{\mu}_{t+1}(\boldsymbol{x}_{t+1})||\cdot \parallel \boldsymbol{\zeta}_t \parallel\right) \\
=&\frac{1}{2\sigma_{t+1}^2}\left(2||\boldsymbol{x}_t|| + ||\boldsymbol{\zeta}_t|| + 2||\boldsymbol{\mu}_{t+1}(\boldsymbol{x}_{t+1})||\right)||\boldsymbol{\zeta}_t||
\end{aligned} \tag{25}$$

Suppose $\boldsymbol{x}$ is bounded with $[a, b]^d$, then there is constant $A_t > 0$ makes

$$2||\boldsymbol{x}_t|| + ||\boldsymbol{\zeta}_t|| + 2||\boldsymbol{\mu}_{t+1}(\boldsymbol{x}_{t+1})|| \leq A_t \tag{26}$$

Applying inequality 26 to equation 25, we get

$$\begin{aligned}
|\tilde{D}_{KL} - D_{KL}| =& \left|\int_{X_t} q(\boldsymbol{x}_t|\boldsymbol{x}_{t+1}) \log \frac{p_\theta(\boldsymbol{x}_t|\boldsymbol{x}_{t+1})}{p_\theta(\tilde{\boldsymbol{x}}_t|\boldsymbol{x}_{t+1})} d\boldsymbol{x}_t\right| \\
\leq& \frac{A_t}{2\sigma_{t+1}^2} \cdot ||\boldsymbol{\zeta}_t||
\end{aligned} \tag{27}$$

where $\tilde{D}_{KL} = D_{KL}(q(\boldsymbol{x}_t|\boldsymbol{x}_{t+1})||p_\theta(\tilde{\boldsymbol{x}}_t|\boldsymbol{x}_{t+1}))$, $D_{KL} = D_{KL}(q(\boldsymbol{x}_t|\boldsymbol{x}_{t+1})||p_\theta(\boldsymbol{x}_t|\boldsymbol{x}_{t+1}))$. Therefore there is

$$\begin{aligned}
&|L_{dmot} - L_{vlb}^{0:M}| \\
\leq& \sum_{t=0}^{M} |\tilde{D}_{KL} - D_{KL}| \\
\leq& (M+1) \max_t \frac{A_t}{2\sigma_{t+1}^2} |C_t| \cdot ||\boldsymbol{\zeta}_M||
\end{aligned} \tag{28}$$

That is

$$L_{dmot} \leq L_{vlb}^{0:M} + (M+1) \max_t \frac{A_t}{2\sigma_{t+1}^2} |C_t| \cdot ||\boldsymbol{\zeta}_M|| \tag{29}$$

Because of $\boldsymbol{\zeta}_M = O(N^{-\frac{1}{2}})$, we can make $\boldsymbol{\zeta}_M$ arbitrarily small by increasing the number of OT samples N. For a given **DPM**, there exists $\boldsymbol{\zeta}_M$ such that the following formula is true.

$$(M+1) \max_t \frac{A_t}{2\sigma_{t+1}^2} |C_t| \cdot ||\boldsymbol{\zeta}_M|| \leq L_{vlb}^{M+1:T} \tag{30}$$

So we have

$$L_{dmot} \leq L_{vlb} \tag{31}$$

□

B. Implementation Details

Additional details about hyperparameter settings of **Algorithm 1** and **Algorithm 2** are elucidated in this section. In the experiments, we instantiate the DPM model $\mathbf{s}_\theta$ of **DPM-OT** with pre-trained models of NCSNv2 [**?**]. In addition, for a fair comparison, we also adopted the same sampling schedule $\{(b_t, \sigma_t)\}_{t=0}^T$ as NCSNv2.

In **Algorithm 1**, we use the Monte Carlo method to solve the SDOT map. We set the number of Monte Carlo samples $N = 10 \times |\mathcal{I}|$, where $|\cdot|$ denotes the number of elements in the set. For the learning rate lr, we set the lr on the datasets CIFAR10 [24], CelebA [29] and FFHQ [21] to $lr = 0.1$, $lr = 20$, and $lr = 50$, respectively. For better convergence, we double the number of samples N and multiply the learning rate lr by 0.8 when the energy function $E(\boldsymbol{h})$ has not decreased

for $s = 50$ steps. Moreover, we set threshold $\tau = 8 \times 10^{-4}$. When the energy function $E(\boldsymbol{h}) < \tau$ or the total number of iteration steps is greater than 10000, the optimization of $\boldsymbol{h}$ will be stopped.

In **Algorithm 2**, reverse diffusion steps M is a variable that satisfies $0 < M < T$. We have carried out five experiments on the datasets CIFAR10, CelebA, and FFHQ, where values of M are 5, 10, 20, 30 and 50 respectively. We find that with the increase of M, the image FID score will decrease, but this decline will tend to be flat with the increase of M, which is reflected in Table 1 of the paper.

C. Additional Qualitative Results

Figure 7. The visualization of our model on FFHQ (10 steps).

In this section, we show three more qualitative results of the proposed **DPM-OT** on FFHQ, Cifar10, and CelebA respectively. Among them, Fig. 7 and Fig. 8 show the results obtained by our model with 10 steps of inverse diffusion on the FFHQ 256×256 and Cifar10, respectively, and Fig. 9 shows the results obtained with only 5 steps of inverse diffusion on the CelebA dataset. As these results show, our model can obtain high-quality images after 5-10 reverse diffusion.

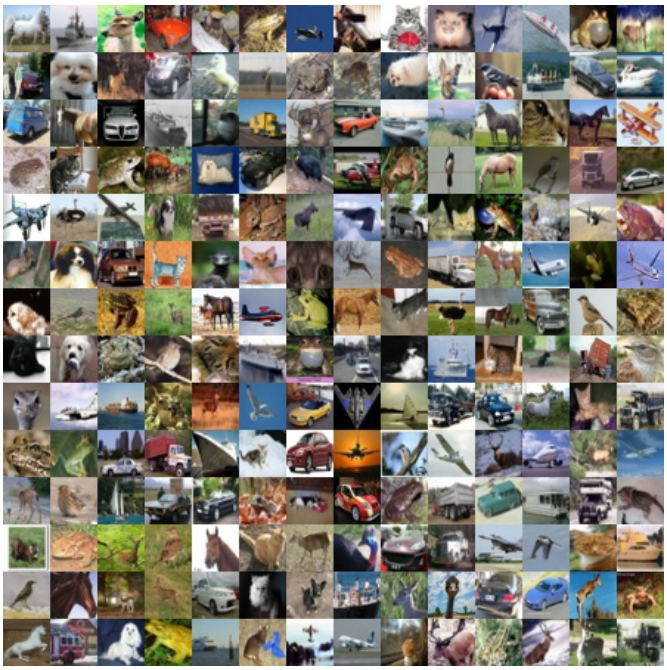


Figure 8. The visualization of our model on Cifar10 (10 steps).

Figure 9. The visualization of our model on CelebA (5 steps).