This WACV paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

Elucidating the Solution Space of Extended Reverse-Time SDE for Diffusion Models

Qinpeng Cui[1], Xinyi Zhang[1], Qiqi Bao[2], Qingmin Liao[1]
[1]Tsinghua University, China
[2]Zhejiang University of Science and Technology, China
{cqp22,xinyi-zh22,liaoqm}@mails.tsinghua.edu.cn, nora919530829@163.com

Abstract

Sampling from Diffusion Models can alternatively be seen as solving differential equations, where there is a challenge in balancing speed and image visual quality. ODE-based samplers offer rapid sampling time but reach a performance limit, whereas SDE-based samplers achieve superior quality, albeit with longer iterations. In this work, we formulate the sampling process as an Extended Reverse-Time SDE (ER SDE), unifying prior explorations into ODEs and SDEs. Theoretically, leveraging the semi-linear structure of ER SDE solutions, we offer exact solutions and approximate solutions for VP SDE and VE SDE, respectively. Based on the approximate solution space of the ER SDE, referred to as one-step prediction errors, we yield mathematical insights elucidating the rapid sampling capability of ODE solvers and the high-quality sampling ability of SDE solvers. Additionally, we unveil that VP SDE solvers stand on par with their VE SDE counterparts. Based on these findings, leveraging the dual advantages of ODE solvers and SDE solvers, we devise efficient high-quality samplers, namely ER-SDE-Solvers. Experimental results demonstrate that ER-SDE-Solvers achieve state-of-the-art performance across all stochastic samplers while maintaining efficiency of deterministic samplers. Specifically, on the ImageNet 128×128 *dataset, ER-SDE-Solvers obtain 8.33 FID in only 20 function evaluations. Code is available at https://github.com/QinpengCui/ER-SDE-Solver*

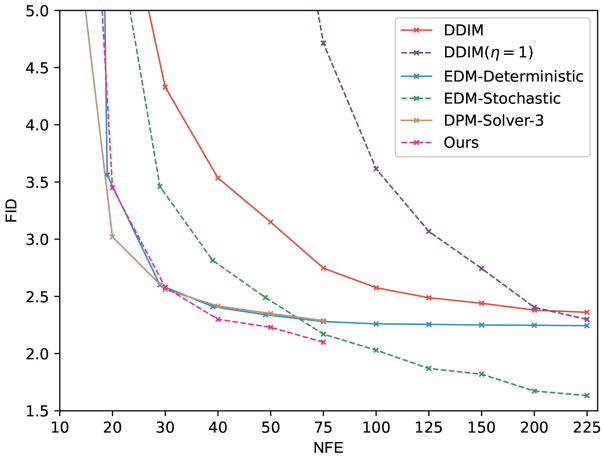


Figure 1. Sample quality (measured by FID↓) on ImageNet 64×64 versus number of function evaluations (NFE) for deterministic samplers (DDIM [29], EDM-Deterministic [13], DPM-Solver-3 [18]) and stochastic samplers (DDIM($\eta = 1$), EDM-Stochastic, Ours). Deterministic samplers excel in achieving rapid sampling but reach a mediocre quality with a large NFE, while stochastic samplers can further enhance image quality with an increase in NFE. Our efficient high-quality samplers demonstrate state-of-the-art performance among all stochastic samplers, simultaneously maintaining sampling efficiency comparable to deterministic samplers.

1. Introduction

Diffusion Models (DMs) demonstrate an aptitude for producing high-quality samples on many tasks, such as image synthesis [5, 10, 32], image super-resolution [6, 26], image restoration [3, 20], image editing [1, 21], image-to-image translation [33, 38], and similar domains. These models define a forward diffusion process by gradually incorporating Gaussian noise to the real data and use iterative backward processes to remove this noise addition. In comparison to alternative generative models like Generative Adversarial Networks (GANs) [8], DMs can lead to higher quality samples for image generation but this often comes with the trade-off of increased time required for sampling. Such inefficiency constrains the applicability of DMs in real-time scenarios.

Prior fast samplers for DMs can be categorized into training-based and training-free methods. While training-based methods can generate high quality samples at 2∼

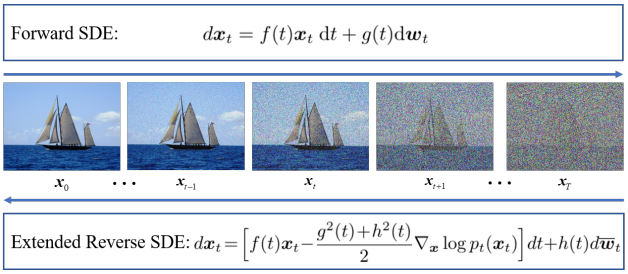


Figure 2. A unified framework for DMs: The forward process described by an SDE transforms real data into noise, while the reverse process characterized by an ER SDE generates real data from noise. Once the score function $\nabla_{\mathbf{x}} \log p_t(\mathbf{x}_t)$ is estimated by a neural network, solving the ER SDE enables the generation of high-quality samples.

4 sampling steps [22, 27, 30], the prerequisite for retraining renders them cost-intensive. Conversely, training-free methods directly utilize raw information without retraining, offering broad applicability and high flexibility. [32] have indicated that the image generation process is equivalent to solving ordinary differential equations (ODEs) or stochastic differential equations (SDEs) in reverse time. The essence of training-free methods lies in designing efficient solvers for ODEs [18, 36, 39] or SDEs [2, 37]. ODE-based samplers follow a deterministic sampling path, and SDE-based samplers incorporate randomness into the data state at each sampling step, leading to stochastic generation trajectories. Observations [13, 29, 34] (as shown in Fig.1) indicate that ODE-based samplers exhibit strong rapid sampling capabilities with fewer function evaluations. However, when aiming for even higher-quality images, increasing the number of function evaluations (NFE) leads to limited improvement in image quality. In contrast, SDE-based samplers show promise in producing data of superior quality with a large NFE, but at the cost of increased sampling time. We aim to design samplers that combine the advantages of both ODE-based samplers and SDE-based samplers in different NFE regimes, rendering it highly practical for real-world applications.

In this work, we present a unified framework for DMs, wherein Extended SDE formulation is proposed (see Fig.2). Within this framework, we define a solution space and design some highly efficient Extended Reverse-Time (ER)-SDE-Solvers to obtain high quality images. Specifically, we first model the sampling process as an ER SDE, which is an extension of [32] and [36]. Inspired by [18], we unveil the semi-linear structure inherent in the solutions. This structure consists of linear functions of data variables, nonlinear functions parameterized by neural networks and noise terms. Building on it, we deduce exact solutions for both Variance Exploding (VE) SDE and Variance Preserving (VP) SDE [32]. This is achieved by the analytical computation of the linear portions and noise terms, thereby circumventing associated prediction errors. Furthermore, we offer practical approximations for both VP SDE and VE SDE.

We refer to the errors of the approximate solutions from predicting every data state in the reverse process as one-step prediction errors. Our analysis reveals that varying levels of one-step prediction errors emerge due to the incorporation of different noise scales. This phenomenon gives rise to the solution space inherent in the ER SDE, and also provides a clearer insight into the different performance of ODE-based and SDE-based samplers in different NFE regimes. We ascertain that the minimal one-step prediction errors correspond to ODE solvers within the solution space, which mathematically demonstrates that the rapid sampling performance of ODE solvers when NFE is limited. Additionally, due to the noise injected during the reverse process can gradually corrects the accumulated prediction errors [13], SDE solvers have the capability to generate higher-quality images as NFE increases. Moreover, given the consistency of the pre-trained models, we theoretically establish that the VP SDE solvers yield image quality equivalent to VE SDE solvers. To take advantages of both ODE and SDE solvers for efficient high-quality sampling, we devise some specialized ER-SDE-Solvers through selecting the noise scale functions carefully.

In summary, we have made several theoretical and practical contributions: 1) We formulate an ER SDE and provide an exact solution as well as approximations for VP SDE and VE SDE, respectively. 2) Through a rigorous analysis of one-step prediction errors in the approximate solutions, we provide a mathematical exposition of the rapid sampling capability of ODE solvers and the high-quality sampling ability of SDE solvers. Moreover, we theoretically demonstrate that VP SDE solvers achieve the same level of image quality compared with VE SDE solvers. 3) By harnessing the dual advantages of ODE and SDE solvers, we present specialized ER-SDE-Solvers for efficient high-quality sampling. Extensive experimentation reveals that ER-SDE-Solvers achieve state-of-the-art performance across all stochastic samplers while maintaining efficiency of deterministic samplers. Additionally, the utilization of classifier guidance further enhances the efficiency of ER-SDE-Solvers.

2. Diffusion Models

Diffusion models (DMs) represent a category of probabilistic generative models encompassing both forward and backward processes. During the forward process, DMs gradually incorporate noise at different scales, while noise is gradually eliminated to yield real samples in the backward process. In the context of continuous time, the forward and backward processes can be described by SDEs or ODEs. In this section, we primarily review the stochastic differential equations (SDEs) and ordinary differential equations (ODEs)

pertinent to DMs.

2.1. Forward Diffusion SDEs

The forward process can be expressed as a linear SDE [14]:

$$d\boldsymbol{x}_t = f(t)\boldsymbol{x}_t\,\mathrm{d}t + g(t)\mathrm{d}\boldsymbol{w}_t, \quad \boldsymbol{x}_0 \sim p_0\left(\boldsymbol{x}_0\right), \tag{1}$$

where $\boldsymbol{x}_0 \in \mathbb{R}^D$ is a D-dimensional random variable following an unknown probability distribution $p_0(\boldsymbol{x}_0)$. $\{\boldsymbol{x}_t\}_{t\in[0,T]}$ denotes each state in the forward process, and $\boldsymbol{w}_t$ stands for a standard Wiener process. When the coefficients $f(t)$ and $g(t)$ are piecewise continuous, a unique solution exists [24]. By judiciously selecting these coefficients, Eq.(1) can map the original data distribution to a priory known tractable distribution $p_T(x_T)$, such as the Gaussian distribution.

The selection of $f(t)$ and $g(t)$ in Eq.(1) is diverse. Based on the distinct noise employed in SMLD [31] and DDPM [10, 28], two distinct SDE formulations [32] are presented.

Variance Exploding (VE) SDE: The noise perturbations used in SMLD can be regarded as the discretization of the following SDE:

$$d\boldsymbol{x}_t = \sqrt{\frac{\mathrm{d}\sigma_t^2}{\mathrm{d}t}}\,\mathrm{d}\boldsymbol{w}_t, \tag{2}$$

where σ_t is the positive noise scale. As $t \to \infty$, the variance of this stochastic process also tends to infinity, thus earning the appellation of Variance Exploding (VE) SDE.

Variance Preserving (VP) SDE: The noise perturbations used in DDPM can be considered as the discretization of the following SDE:

$$d\boldsymbol{x}_t = \frac{d\log\alpha_t}{dt}\boldsymbol{x}_t\mathrm{d}t + \sqrt{\frac{d\sigma_t^2}{dt} - 2\frac{d\log\alpha_t}{dt}\sigma_t^2}\mathrm{d}\boldsymbol{w}_t, \tag{3}$$

where α_t is also the positive noise scale. Unlike the VE SDE, the variance of VP SDE remains bounded as $t \to \infty$. Therefore, it is referred to as Variance Preserving (VP) SDE.

2.2. Reverse Diffusion SDEs

The backward process can similarly be described by a reverse-time SDE [32]:

$$d\boldsymbol{x}_t = \left[f(t)\boldsymbol{x}_t - g^2(t)\nabla_{\boldsymbol{x}}\log p_t\left(\boldsymbol{x}_t\right)\right]\mathrm{d}t + g(t)\mathrm{d}\overline{\boldsymbol{w}}_t, \tag{4}$$

where $\overline{\boldsymbol{w}}_t$ is the standard Wiener process in the reverse time. $p_t(\boldsymbol{x}_t)$ represents the probability distribution of the state $\boldsymbol{x}_t$, and its logarithmic gradient $\nabla_{\boldsymbol{x}}\log p_t\left(\boldsymbol{x}_t\right)$ is referred to as the score function, which is often estimated by a neural network $\boldsymbol{s}_\theta(\boldsymbol{x}_t, t)$.

There are also studies [35, 36] that consider the following reverse-time SDE:

$$d\boldsymbol{x}_t = \left[f(t)\boldsymbol{x}_t - \frac{1+\lambda^2}{2}g^2(t)\nabla_{\boldsymbol{x}}\log p_t\left(\boldsymbol{x}_t\right)\right]\mathrm{d}t + \lambda g(t)\mathrm{d}\overline{\boldsymbol{w}}_t, \tag{5}$$

where the parameter $\lambda \geq 0$. Eq.(5) similarly shares the same marginal distribution as Eq.(1).

Once the score-based network $\boldsymbol{s}_\theta(\boldsymbol{x}_t, t)$ is trained, generating images only requires solving the reverse-time SDE in Eq.(4) or Eq.(5). The conventional ancestral sampling method [10] can be viewed as a first-order SDE solver [32], yet it needs thousands of function evaluations for high-quality images. Numerous efforts [2, 12] have aimed to enhance sampling speed by devising highly accurate SDE solvers, but they still require hundreds of function evaluations, presenting a gap compared to ODE solvers.

2.3. Reverse Diffusion ODEs

In the backward process, in addition to directly solving the reverse-time SDE in Eq.(4), a category of methods [18, 36, 39] focuses on solving the probability flow ODE corresponding to Eq.(4), expressed specifically as

$$d\boldsymbol{x}_t = \left[f(t)\boldsymbol{x}_t - \frac{1}{2}g^2(t)\nabla_{\boldsymbol{x}}\log p_t\left(\boldsymbol{x}_t\right)\right]\mathrm{d}t. \tag{6}$$

Eq.(6) shares the same marginal distribution at each time t with the SDE in Eq.(4), and the score function $\nabla_{\boldsymbol{x}}\log p_t\left(\boldsymbol{x}_t\right)$ can also be estimated by a neural network. Unlike SDEs introducing stochastic noise at each step, ODEs correspond to a deterministic sampling process. Despite several experiments [32] suggesting that ODE solvers outperform SDE solvers in terms of rapid sampling, SDE solvers can generate higher-quality images with an increase in NFE.

In summary, SDE-based methods can generate higher-quality samples, but exhibit slower convergence in high dimensions [15, 18]. Conversely, ODE-based methods demonstrate the opposite behavior. To strike a balance between high-quality and efficiency in the generation process, in Sec.3, we model the backward process as an extended SDE, and provide analytical solutions as well as approximations for both VP SDE and VE SDE. Furthermore, we devise some ER-SDE-Solvers in Sec.4, achieving efficient high-quality sampling.

3. Extended Reverse-Time SDE Solvers

There are three methods for recovering samples from noise in DMs. The first predicts the noise added in the forward process, achieved by a noise prediction model $\boldsymbol{\epsilon}_\theta\left(\boldsymbol{x}_t, t\right)$ [10]. The second utilizes a score prediction model $\boldsymbol{s}_\theta(\boldsymbol{x}_t, t)$ to match the score function $\nabla_{\mathbf{x}}\log p_t(\mathbf{x}_t)$ [11, 32]. The last directly restores the original data from the noisy samples, achieved by a data prediction model $\boldsymbol{x}_\theta\left(\boldsymbol{x}_t, t\right)$. These models can be mutually derived [14]. Previously, most SDE solvers relied on the score-based model [2, 12, 32]. Based on modeling the backward process as an extended SDE in Sec.3.1, we proceed to solve the VE SDE and VP SDE

for the data prediction model in Sec.3.2 and Sec.3.3, respectively. Compared with the other two types of models, data prediction model can be synergistically combined with thresholding methods [10, 25] to mitigate the adverse impact of large guiding scales, thereby finding broad application in guided image generation [19].

3.1. Extended Reverse-Time SDE

Besides Eq.(4) and Eq.(5), an infinite variety of diffusion processes can be employed. In this paper, we consider the following family of SDEs (referred to as Extended Reverse-Time SDE (ER SDE)):

$$d\boldsymbol{x}_t = \Big[f(t)\boldsymbol{x}_t - \frac{g^2(t)+h^2(t)}{2}\nabla_{\boldsymbol{x}}\log p_t(\boldsymbol{x}_t)\Big]dt + h(t)d\overline{\boldsymbol{w}}_t. \tag{7}$$

The score function $\nabla_{\boldsymbol{x}}\log p_t(\boldsymbol{x}_t)$ can be estimated using the pretrained neural network. Hence, generating samples only requires solving Eq.(7), guaranteed by Proposition 3.1.

Proposition 3.1 (The validity of the ER SDE, proof in Supp.1.1)**.** *When $\boldsymbol{s}_\theta(\boldsymbol{x}_t, t) = \nabla_{\boldsymbol{x}}\log p_t(\boldsymbol{x}_t)$ for all $\boldsymbol{x}_t$, $\overline{p}_T(\boldsymbol{x}_T) = p_T(\boldsymbol{x}_T)$, the marginal distribution $\overline{p}_t(\boldsymbol{x}_t)$ of Eq.(7) matches $p_t(\boldsymbol{x}_t)$ of the forward diffusion Eq.(1) for all $0 \le t \le T$.*

Eq.(7) extends the reverse-time SDE proposed in [13, 32, 34, 36]. Specifically, in [32], the noise scale $g(t)$ added at each time step t of the reverse process is the same as that of the corresponding moment in the forward process. [13, 34, 36] introduce a non-negative parameter to control the extent of noise added during the reverse process. However, the form of the noise scale is relevant to $g(t)$. In contrast, our ER SDE introduces a completely new noise scale $h(t)$ for the reverse process. This implies that the noise scale $h(t)$ added during the reverse process may not necessarily be correlated with the scale $g(t)$ of the forward process. Particularly, the ER SDE reduces to the reverse-time SDE in Eq.(4), Eq.(5) and the ODE depicted in Eq.(6) respectively when the specific values of $h(t)$ are chosen.

By expanding the reverse-time SDE, we not only unify ODEs and SDEs under a single framework, facilitating the comparative analysis of these two methods, but also lay the groundwork for designing more efficient samplers. Further details are discussed in Sec.4.

3.2. VE ER-SDE-Solvers

For the VE SDE, $f(t) = 0$ and $g(t) = \sqrt{\frac{\mathrm{d}\sigma_t^2}{\mathrm{d}t}}$ [14]. The relationship between the score prediction model $\boldsymbol{s}_\theta(\boldsymbol{x}_t, t)$ and the data prediction model $\boldsymbol{x}_\theta(\boldsymbol{x}_t, t)$ is $-[\boldsymbol{x}_t - \boldsymbol{x}_\theta(\boldsymbol{x}_t, t)]/\sigma_t^2 = \boldsymbol{s}_\theta(\boldsymbol{x}_t, t)$. By replacing the score function with the data prediction model, Eq.(7) can be expressed as

$$d\boldsymbol{x}_t = \frac{1}{2\sigma_t^2}\Big[\frac{d\sigma_t^2}{dt} + h^2(t)\Big][\boldsymbol{x}_t - \boldsymbol{x}_\theta(\boldsymbol{x}_t, t)]dt + h(t)d\overline{\boldsymbol{w}}_t. \tag{8}$$

Denote $d\boldsymbol{w}_\sigma := \sqrt{\frac{d\sigma_t}{dt}}d\overline{\boldsymbol{w}}_t$, $h^2(t) = \xi(t)\frac{d\sigma_t}{dt}$, we can rewrite Eq.(8) w.r.t σ as

$$d\boldsymbol{x}_\sigma = \Big[\frac{1}{\sigma} + \frac{\xi(\sigma)}{2\sigma^2}\Big][\boldsymbol{x}_\sigma - \boldsymbol{x}_\theta(\boldsymbol{x}_\sigma, \sigma)]d\sigma + \sqrt{\xi(\sigma)}d\boldsymbol{w}_\sigma. \tag{9}$$

We propose the exact solution for Eq.(9) using *variation-of-constants* formula [19].

Proposition 3.2 (Exact solution of the VE SDE, proof in Supp.1.2)**.** *Given an initial value $\boldsymbol{x}_s$ at time $s > 0$, the solution $\boldsymbol{x}_t$ at time $t \in [0, s]$ of VE SDE in Eq.(9) is:*

$$\begin{aligned}\boldsymbol{x}_t = &\underbrace{\frac{\phi(\sigma_t)}{\phi(\sigma_s)}\boldsymbol{x}_s}_{\text{(a) Linear term}} + \underbrace{\phi(\sigma_t)\int_{\sigma_t}^{\sigma_s}\frac{\phi^{(1)}(\sigma)}{\phi^2(\sigma)}\boldsymbol{x}_\theta(\boldsymbol{x}_\sigma, \sigma)d\sigma}_{\text{(b) Nonlinear term}} \\ &+ \underbrace{\sqrt{\sigma_t^2 - \sigma_s^2\Big[\frac{\phi(\sigma_t)}{\phi(\sigma_s)}\Big]^2}\boldsymbol{z}_s}_{\text{(c) Noise term}},\end{aligned} \tag{10}$$

where $\boldsymbol{z}_s \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$. $\phi(x)$ is derivable and $\int \frac{1}{\sigma} + \frac{\xi(\sigma)}{2\sigma^2}d\sigma = \ln\phi(\sigma)$.

Notably, the nonlinear term in Eq(10) involves the integration of a non-analytical neural network $\boldsymbol{x}_\theta(\boldsymbol{x}_\sigma, \sigma)$, which can be challenging to compute. For practical applicability, Proposition 3.3 furnishes high-stage solvers (followed by [7]) for Eq.(10).

Proposition 3.3 (High-stage approximations of the VE SDE, proof in Supp.1.3)**.** *Given an initial value $\boldsymbol{x}_T$ and $M+1$ time steps $\{t_i\}_{i=0}^M$ decreasing from $t_0 = T$ to $t_M = 0$. Starting with $\tilde{\boldsymbol{x}}_{t_0} = \boldsymbol{x}_T$, the sequence $\{\tilde{\boldsymbol{x}}_{t_i}\}_{i=1}^M$ is computed iteratively as follows:*

$$\begin{aligned}\tilde{\boldsymbol{x}}_{t_i} = &\frac{\phi(\sigma_{t_i})}{\phi(\sigma_{t_{i-1}})}\tilde{\boldsymbol{x}}_{t_{i-1}} + \Big[1 - \frac{\phi(\sigma_{t_i})}{\phi(\sigma_{t_{i-1}})}\Big]\boldsymbol{x}_\theta(\tilde{\boldsymbol{x}}_{\sigma_{t_{i-1}}}, \sigma_{t_{i-1}}) \\ &+ \sum_{n=1}^{k-1}\boldsymbol{x}_\theta^{(n)}(\tilde{\boldsymbol{x}}_{\sigma_{t_{i-1}}}, \sigma_{t_{i-1}})\Big[\frac{(\sigma_{t_i} - \sigma_{t_{i-1}})^n}{n!} + \phi(\sigma_{t_i}) \\ &\int_{\sigma_{t_i}}^{\sigma_{t_{i-1}}}\frac{(\sigma - \sigma_{t_{i-1}})^{n-1}}{(n-1)!\phi(\sigma)}d\sigma\Big] + \sqrt{\sigma_{t_i}^2 - \sigma_{t_{i-1}}^2\Big[\frac{\phi(\sigma_{t_i})}{\phi(\sigma_{t_{i-1}})}\Big]^2}\boldsymbol{z}_{t_{i-1}},\end{aligned} \tag{11}$$

where $k \ge 1$. $\boldsymbol{x}_\theta^{(n)}(\boldsymbol{x}_\sigma, \sigma) := \frac{d^n\boldsymbol{x}_\theta(\boldsymbol{x}_\sigma, \sigma)}{d\sigma^n}$ is the n-th order total derivative of $\boldsymbol{x}_\theta(\boldsymbol{x}_\sigma, \sigma)$ w.r.t σ.

$\int_{\sigma_{t_i}}^{\sigma_{t_{i-1}}}\frac{(\sigma - \sigma_{t_{i-1}})^{n-1}}{(n-1)!\phi(\sigma)}d\sigma$ in Eq.(11) lacks an analytical expression, and we resort to N-point numerical integration for estimation. The detailed algorithms refer to Supp.2.

3.3. VP ER-SDE-Solvers

For the VP SDE, $f(t) = \frac{d\log\alpha_t}{dt}$, $g(t) = \sqrt{\frac{d\sigma_t^2}{dt} - 2\frac{d\log\alpha_t}{dt}\sigma_t^2}$ [14]. The relationship between the

score prediction model $\boldsymbol{s}_\theta(\boldsymbol{x}_t, t)$ and the data prediction model $\boldsymbol{x}_\theta(\boldsymbol{x}_t, t)$ is $-[\boldsymbol{x}_t - \alpha_t \boldsymbol{x}_\theta(\boldsymbol{x}_t, t)]/\sigma_t^2 = \boldsymbol{s}_\theta(\boldsymbol{x}_t, t)$. By replacing the score function with the data prediction model, Eq.(7) can be written as:

$$d\boldsymbol{x}_t = \Big\{\Big[\frac{1}{\sigma_t}\frac{d\sigma_t}{dt} + \frac{h^2(t)}{2\sigma_t^2}\Big]\boldsymbol{x}_t - \Big[\frac{1}{\sigma_t}\frac{d\sigma_t}{dt} - \frac{1}{\alpha_t}\frac{d\alpha_t}{dt} + \frac{h^2(t)}{2\sigma_t^2}\Big] \alpha_t \boldsymbol{x}_\theta(\boldsymbol{x}_t, t)\Big\}dt + h(t)d\overline{\boldsymbol{w}}_t. \tag{12}$$

Let $h(t) = \eta(t)\alpha_t$, $\boldsymbol{y}_t = \frac{\boldsymbol{x}_t}{\alpha_t}$ and $\lambda_t = \frac{\sigma_t}{\alpha_t}$. Denote $d\boldsymbol{w}_\lambda := \sqrt{\frac{d\lambda_t}{dt}}d\overline{\boldsymbol{w}}_t$, $\eta^2(t) = \xi(t)\frac{d\lambda_t}{dt}$,and rewrite Eq.(12) w.r.t λ as

$$d\boldsymbol{y}_\lambda = \Big[\frac{1}{\lambda} + \frac{\xi(\lambda)}{2\lambda^2}\Big][\boldsymbol{y}_\lambda - \boldsymbol{x}_\theta(\boldsymbol{x}_\lambda, \lambda)]d\lambda + \sqrt{\xi(\lambda)}d\boldsymbol{w}_\lambda. \tag{13}$$

Following [19], we propose the exact solution for Eq.(13) using the *variation-of-constants* formula.

Proposition 3.4 (Exact solution of the VP SDE, proof in Supp.1.4)**.** *Given an initial value $\boldsymbol{x}_s$ at time $s > 0$, the solution $\boldsymbol{x}_t$ at time $t \in [0, s]$ of VP SDE in Eq.(13) is:*

$$\boldsymbol{x}_t = \underbrace{\frac{\alpha_t}{\alpha_s}\frac{\phi(\lambda_t)}{\phi(\lambda_s)}\boldsymbol{x}_s}_{\text{(a) Linear term}} + \underbrace{\alpha_t\phi(\lambda_t)\int_{\lambda_t}^{\lambda_s}\frac{\phi^{(1)}(\lambda)}{\phi^2(\lambda)}\boldsymbol{x}_\theta(\boldsymbol{x}_\lambda, \lambda)d\lambda}_{\text{(b) Nonlinear term}} + \underbrace{\alpha_t\sqrt{\lambda_t^2 - \lambda_s^2\Big[\frac{\phi(\lambda_t)}{\phi(\lambda_s)}\Big]^2}\boldsymbol{z}_s}_{\text{(c) Noise term}}, \tag{14}$$

where $\boldsymbol{z}_s \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$. $\phi(x)$ is derivable and $\int \frac{1}{\lambda} + \frac{\xi(\lambda)}{2\sigma^2}d\lambda = \ln\phi(\lambda)$.

The solution of the VP SDE also involves integrating a non-analytical and nonlinear neural network. Proposition 3.5 furnishes high-stage solvers (followed by [7]) for Eq.(14).

Proposition 3.5 (High-stage approximations of the VP SDE, proof in Supp.1.5)**.** *Given an initial value $\boldsymbol{x}_T$ and $M+1$ time steps $\{t_i\}_{i=0}^M$ decreasing from $t_0 = T$ to $t_M = 0$. Starting with $\tilde{\boldsymbol{x}}_{t_0} = \boldsymbol{x}_T$, the sequence $\{\tilde{\boldsymbol{x}}_{t_i}\}_{i=1}^M$ is computed iteratively as follows:*

$$\tilde{\boldsymbol{x}}_{t_i} = \frac{\alpha_{t_i}}{\alpha_{t_{i-1}}}\frac{\phi(\lambda_{t_i})}{\phi(\lambda_{t_{i-1}})}\tilde{\boldsymbol{x}}_{t_{i-1}} + \alpha_{t_i}\Big[1 - \frac{\phi(\lambda_{t_i})}{\phi(\lambda_{t_{i-1}})}\Big]\boldsymbol{x}_\theta(\tilde{\boldsymbol{x}}_{\lambda_{t_{i-1}}}, \lambda_{t_{i-1}}) + \alpha_{t_i}\sum_{n=1}^{k-1}\boldsymbol{x}_\theta^{(n)}(\tilde{\boldsymbol{x}}_{\lambda_{t_{i-1}}}, \lambda_{t_{i-1}})\Big[\frac{(\lambda_{t_i} - \lambda_{t_{i-1}})^n}{n!} + \phi(\lambda_{t_i})\int_{\lambda_{t_i}}^{\lambda_{t_{i-1}}}\frac{(\lambda - \lambda_{t_{i-1}})^{n-1}}{(n-1)!\phi(\lambda)}d\lambda\Big] + \alpha_{t_i}\sqrt{\lambda_{t_i}^2 - \lambda_{t_{i-1}}^2\Big[\frac{\phi(\lambda_{t_i})}{\phi(\lambda_{t_{i-1}})}\Big]^2}\boldsymbol{z}_{t_{i-1}}, \tag{15}$$

where $k \geq 1$. $\boldsymbol{x}_\theta^{(n)}(\boldsymbol{x}_\lambda, \lambda) := \frac{d^n\boldsymbol{x}_\theta(\boldsymbol{x}_\lambda, \lambda)}{d\lambda^n}$ is the n-th order total derivative of $\boldsymbol{x}_\theta(\boldsymbol{x}_\lambda, \lambda)$ w.r.t λ.

Similarly, we employ N-point numerical integration to estimate $\int_{\lambda_{t_i}}^{\lambda_{t_{i-1}}}\frac{(\lambda - \lambda_{t_{i-1}})^{n-1}}{(n-1)!\phi(\lambda)}d\lambda$ in Eq.(15). The detailed algorithms are proposed in Supp.2.

4. Elucidating the Solution Space of ER SDE

This section primarily focuses on the solution space of ER SDE. Specifically, in Sec.4.1, we provide a mathematical explanation for experimental observations made in previous research. Furthermore, we introduce various specialized Extended Reverse-Time SDE Solvers (ER-SDE-Solvers) in Sec.4.2, which achieve efficient high-quality sampling.

4.1. Insights about the Solution Space of ER SDE

Sec.3 demonstrates that the exact solution of ER SDE comprises three components: a linear function of the data variables, a non-linear function parameterized by neural networks and a noise term. The linear and noise terms can be precisely computed, while errors arising from predicting the data state in the reverse process are present in the non-linear term. Due to the decreasing error as the stage increases (see Table1), the first-order error predominantly influences the overall error. Therefore, we exemplify the case with order $k = 1$ for error analysis. Specifically, the first-order approximation for VE SDE is given by

$$\tilde{\boldsymbol{x}}_{t_i} = \frac{\phi(\sigma_{t_i})}{\phi(\sigma_{t_{i-1}})}\tilde{\boldsymbol{x}}_{t_{i-1}} + \Big[1 - \frac{\phi(\sigma_{t_i})}{\phi(\sigma_{t_{i-1}})}\Big]\boldsymbol{x}_\theta(\tilde{\boldsymbol{x}}_{\sigma_{t_{i-1}}}, \sigma_{t_{i-1}}) + \sqrt{\sigma_{t_i}^2 - \sigma_{t_{i-1}}^2\Big[\frac{\phi(\sigma_{t_i})}{\phi(\sigma_{t_{i-1}})}\Big]^2}\boldsymbol{z}_{t_{i-1}}, \tag{16}$$

and the first-order approximation for VP SDE is

$$\frac{\tilde{\boldsymbol{x}}_{t_i}}{\alpha_{t_i}} = \frac{\phi(\lambda_{t_i})}{\phi(\lambda_{t_{i-1}})}\frac{\tilde{\boldsymbol{x}}_{t_{i-1}}}{\alpha_{t_{i-1}}} + \Big[1 - \frac{\phi(\lambda_{t_i})}{\phi(\lambda_{t_{i-1}})}\Big]\boldsymbol{x}_\theta(\tilde{\boldsymbol{x}}_{\lambda_{t_{i-1}}}, \lambda_{t_{i-1}}) + \sqrt{\lambda_{t_i}^2 - \lambda_{t_{i-1}}^2\Big[\frac{\phi(\lambda_{t_i})}{\phi(\lambda_{t_{i-1}})}\Big]^2}\boldsymbol{z}_{t_{i-1}}. \tag{17}$$

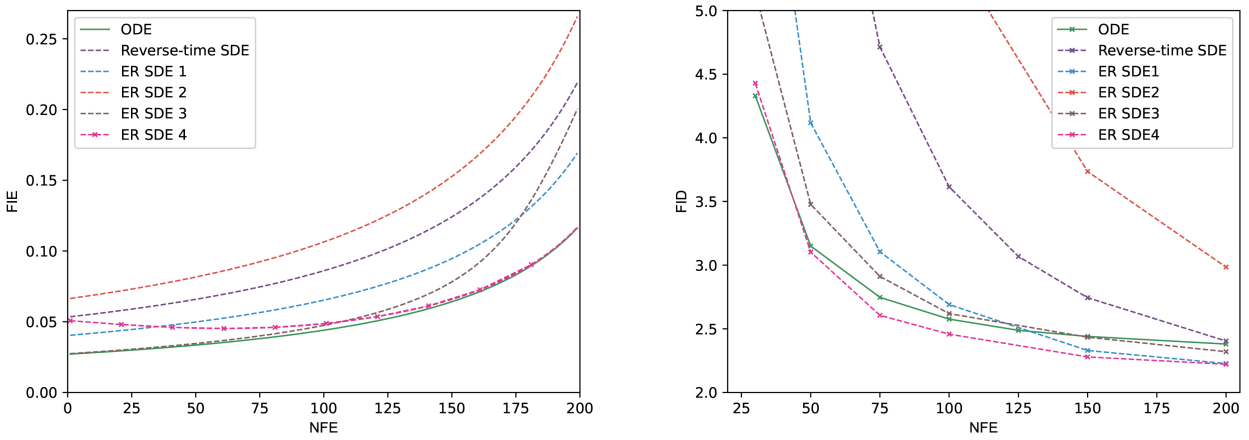
To elaborate more clearly, we refer to the errors arising from predicting every data state in the reverse process as one-step prediction errors. For the VE SDE, one-step prediction errors can be expressed as (derivation in Supp.1.6)

$$|\boldsymbol{x}_t - \tilde{\boldsymbol{x}}_t| = \Big[1 - \frac{\phi(\sigma_t)}{\phi(\sigma_s)}\Big]|\boldsymbol{x}_0(\boldsymbol{x}_{\sigma_s}, \sigma_s) - \boldsymbol{x}_\theta(\boldsymbol{x}_{\sigma_s}, \sigma_s)| + \check{\mathcal{R}}_1, \tag{18}$$

and for VP SDE is (derivation in Supp.1.7)

$$|\boldsymbol{x}_t - \tilde{\boldsymbol{x}}_t| = \alpha_t\Big[1 - \frac{\phi(\lambda_t)}{\phi(\lambda_s)}\Big]|\boldsymbol{x}_0(\boldsymbol{x}_{\lambda_s}, \lambda_s) - \boldsymbol{x}_\theta(\boldsymbol{x}_{\lambda_s}, \lambda_s)| + \check{\mathcal{R}}_1. \tag{19}$$

We observe that one-step prediction errors of both VE SDE and VP SDE are influenced by the First-order Itô-Taylor



(a) FIE coefficients

(b) FID scores(↓) on ImageNet 64×64

Figure 3. FIE coefficients (a) and FID scores (b) versus NFE for distinct noise scale functions. 1st-order solver is used here with the pretrained EDM. In the solution space of ER SDE, ODE solver shows minimal one-step prediction errors. ER SDE 4 demonstrates elevated error in the initial 100 NFE and gradually converges to the ODE's error profile. Thus, ER SDE 4 exhibit comparable efficiency to ODE solver but can further generate high-quality images. Image quality deteriorates for ill-suited noise scale functions (like ER SDE 2).

Expansion (FIE) coefficient $1-\frac{\phi(x_t)}{\phi(x_s)}$, which is only determined by the noise scale function $\phi(x)$ introduced in the reverse process. As $\phi(x)$ is arbitrary, different noise scale functions correspond to different solutions, collectively forming the solution space of ER SDE (here we borrow the concept of solution space from linear algebra [17]).

ODE Solvers exhibit rapid sampling capability while SDE Solvers demonstrate high-quality sampling ability: Taking the first order approximation of VE SDE as an example, an intuitive strategy for reducing one-step prediction errors is to decrease the FIE coefficient. Due to $\frac{\phi(\sigma_t)}{\phi(\sigma_s)} \leq \frac{\sigma_t}{\sigma_s}$ (see Supp.1.9), the minimum value for the FIE coefficient is $1-\frac{\sigma_t}{\sigma_s}$. Interestingly, when the FIE coefficient reaches its minimum value, the ER SDE precisely reduces to ODE (in this case, $\phi(\sigma)=\sigma$). This implies that ODE solvers possess the minimal one-step prediction error, theoretically explaining the strong rapid sampling capabilities observed in Fig.1. Further analysis in Supp.1.8 reveals that ER SDE reduces to the reverse-time SDE when $\phi(\sigma)=\sigma^2$. In theory, smaller one-step prediction errors lead to higher image generation quality in the small NFE regime. However, why do SDE solvers produce higher-quality images compared with ODE solvers when increasing NFE further? This is because the larger FIE coefficient of SDE solvers corresponds to more noise during the reverse process, which gradually corrects the accumulated prediction errors [13]. As shown in Fig.3(a), ODE Solvers have the minimal FIE coefficient, thus demonstrating rapid sampling capability in Fig.3(b) in the small NFE regime. However, once the noise gradually corrects the accumulated prediction errors, SDE Solvers exhibits greater potential for more efficient high-quality sampling.

VP SDE Solvers achieve parity with VE SDE Solvers: The only difference between Eq.(16) and Eq.(17) lies in the latter being scaled by $1/\alpha_t$, but their relative errors remain the same. In other words, the performance of the VP SDE and the VE SDE solver is equivalent under the same NFE and pretrained model. Directly comparing them by experiments has been challenging in prior research due to the absence of a generative model simultaneously supporting both types of SDEs. This has led to divergent conclusions, with some studies [32] finding that VE SDE provides better sample quality than VP SDE, while others [12] reaching the opposite conclusion. Fortunately, EDM [13] allows us for a fair comparison between VP SDE and VE SDE, as elaborated in Supp.3.3.

4.2. Customized Efficient High-Quality ER-SDE-Solvers

In order to combine the rapid sampling performance of ODE solvers with the high-quality sampling capability of SDE solvers, we devise specialized ER-SDE-Solvers by carefully selecting noise scale functions.

To further demonstrate how the noise scale function $\phi(x)$ directly impacts the efficiency of the sampling process, we initially provide three different forms of $\phi(x)$:

$$\text{ER SDE 1:}\phi(x)=x^{1.5}, \quad \text{ER SDE 2:}\phi(x)=x^{2.5}$$

$$\text{ER SDE 3:}\phi(x)=x^{0.9}\log_{10}(1+100x)\,.$$

Fig.3 illustrates that unfavorable choices of $\phi(x)$ (such as ER SDE 2) lead to significant one-step prediction errors and

inefficient sampling. With inappropriate $\phi(x)$ and limited NFE, the role of noise correction is not prominent. Thus, it is crucial to carefully select the noise scale function $\phi(x)$ to achieve high-quality sampling with fewer NFE.

In order to achieve rapid sampling, we make the FIE coefficient as close as possible to the ODE case, which has the minimum one-step prediction errors. To further enhance the quality of generated images, we introduce a moderate amount of noise during the sampling process, i.e., allowing for a controlled amplification of one-step prediction errors when NFE is relatively small. Consequently, we propose a customized ER SDE solver where

$$\text{ER SDE 4:} \quad \phi(x) = x(e^{x^{0.3}} + 10). \tag{20}$$

Although ER SDE 4 exhibits more significant one-step prediction errors in the early stages ($\sim$100 NFE) in Fig.3(a), its later-stage errors closely approach the minimum error (i.e., the error of the ODE). As a result, ER SDE 4 has the potential to generate high-quality images while maintaining efficiency of ODE solvers (see Fig.3(b)).

We select ER SDE 4 as the noise scale function by default in the subsequent experiments. This strategy not only facilitates rapid sampling but also contributes to preserving the stochastic noise introduced by the reverse process, thereby enhancing higher-quality in the generated images. In fact, there are countless possible choices for $\phi(x)$, and we have only provided a few examples here. Researchers should select the suitable one based on specific application scenarios, as detailed in Supp.1.10.

Table 1. Sample quality measured by FID↓ on ImageNet 64×64 for different stages of VE(P) ER-SDE-Solvers with EDM, varying the NFE. VE(P)-x denotes the x-th stage VE(P) ER-SDE-Solver.

Method\NFE	10	20	30	50
VE-2	11.81	3.67	2.67	2.31
VP-2	11.94	3.73	2.67	2.27
VE-3	11.46	3.45	2.58	2.24
VP-3	11.32	3.48	2.58	2.28

Table 2. Sample quality measured by FID↓ on ImageNet 64×64 with the pretrained model EDM, varying the NFE. The upper right $-$ indicates a reduction of NFE by one, and $+$ signifies an increase in NFE by one.

Sampling method\NFE	35	40	45	50
Stochastic Sampling				
DDIM($\eta = 1$)	11.4	9.49	8.37	7.35
SDE-DPM-Solver++(2M)	3.02	2.73	2.50	2.29
EDM-Stochastic	2.97	2.82^-	2.57	2.49^-
SEEDS(ETD-SERK)	53.7^+	46.9^-	34.0	25.9^+
Ours(ER-SDE-Solver-3)	**2.46**	**2.30**	**2.25**	**2.22**
Deterministic Sampling				
DDIM	3.85	3.54	3.31	3.15
EDM-Deterministic	**2.46**	2.41^-	2.37	2.34^-
DPM-Solver-3	2.48	2.42	2.38	2.35
DPM-Solver++(2M)	2.47	2.42	2.39	2.35
SEEDS(ETD-ERK)	**2.46**	2.39	2.37^-	2.34

5. Experiments

In this section, we demonstrate that ER-SDE-Solvers can significantly accelerate the sampling process of pretrained DMs. We vary the number of function evaluations (NFE), i.e., the invocation number of the data prediction model, and compare the sample quality between ER-SDE-Solvers of different stages and other training-free samplers. For each experiment, we draw 50K samples and employ the widely-used FID score [9] and sFID [23] to evaluate sample quality, where a lower FID/sFID typically signifies better sample quality. Detailed implementation and experimental settings refer to Supp.3.

5.1. Different Stages of VE and VP ER-SDE-Solvers

To ensure a fair comparison between VP ER-SDE-Solvers and VE ER-SDE-Solvers, we opt for EDM [13] as the pretrained model, as detailed in Supp.3.3. It can be observed from Table 1 that the image generation quality produced by both of them is similar, consistent with the findings in Sec.4.1. Additionally, the high-stage ER-SDE-Solver-3 converges faster than ER-SDE-Solver-2, particularly in the few-step regime under 20 NFE. This is because higher stages result in more minor discretization errors, which constitute one of the components of one-step prediction errors (see Supp.1.6 and Supp.1.7). We also arrive at the same conclusions on CIFAR-10 dataset [16] as can be found in supplementary material Table.1.

5.2. Comparisons with Other Training-Free Methods

We compare ER-SDE-Solvers with other training-free sampling methods, including stochastic samplers such as SDE-DPM-Solver++ [19] and SEEDS [7], as well as deterministic samplers like DDIM [29], DPM-Solver [18] and DPM-Solver++ [19]. Table 3 presents experimental results on the ImageNet 128×128 dataset [4] using the same pretrained Guided-diffusion model [5]. It is evident that ER-SDE-Solvers emerge as the most efficient stochastic samplers, achieving a remarkable $2 \sim 8\times$ speedup compared with previously state-of-the-art stochastic sampling methods. Specifically, ER-SDE-Solvers achieve high-quality

Table 3. Sample quality measured by FID↓ and sFID↓ on class-conditional ImageNet 128×128 with the pretrained model Guided-diffusion (without classifier guidance, linear noise schedule), varying the NFE.

Sampling method\NFE	FID↓					sFID↓				
	20	**25**	**30**	**40**	**50**	**20**	**25**	**30**	**40**	**50**
Stochastic Sampling										
DDIM($\eta = 1$)	21.23	17.31	14.95	12.05	10.42	27.77	22.66	19.62	15.67	13.30
SDE-DPM-Solver++(2M)	9.61	8.75	8.38	7.85	7.83	7.89	7.03	6.62	5.90	5.57
Ours(ER-SDE-Solver-3)	**8.33**	**7.87**	**7.84**	**7.78**	**7.72**	5.90	**5.39**	**5.13**	**5.00**	**4.88**
Deterministic Sampling										
DDIM	11.49	10.59	9.79	8.99	8.65	7.88	6.87	6.24	5.51	5.27
DPM-Solver-3	9.55	9.45	9.39	9.13	9.02	**5.75**	5.52	5.30	5.16	5.06
DPM-Solver++(2M)	10.24	9.99	9.93	9.76	9.61	6.52	6.21	6.01	5.70	5.55

Table 4. Sample quality measured by FID↓ on class-conditional ImageNet 256×256 with the pretrained Guided-diffusion (classifier guidance scale = 2.0, linear noise schedule), varying the NFE.

Sampling method\NFE	10	20	30	50
Stochastic Sampling				
DDIM($\eta = 1$)	17.97	10.23	8.19	6.85
SDE-DPM-Solver++(2M)	9.21	6.01	5.47	5.19
Ours(ER-SDE-Solver-3)	**6.24**	**4.76**	**4.62**	**4.57**
Deterministic Sampling				
DDIM	8.63	5.60	5.00	4.59
DPM-Solver-3	6.45	5.03	4.94	4.92
DPM-Solver++(2M)	7.19	5.54	5.32	5.16

sampling with FID = 7.84 requiring only 30 NFE, while SDE-DPM-Solver++ needs 50 NFE. Additionally, compared to deterministic samplers, ER-SDE-Solvers can generate higher-quality images when NFE is fixed. As illustrated in Fig.1, increasing NFE further enhances the image quality produced by ER-SDE-Solvers, whereas the improvement in image quality for deterministic samplers is limited. In summary, ER-SDE-Solvers achieve state-of-the-art performance among all stochastic samplers by enhancing image generation quality while maintaining efficiency. We also provide comparisons on ImageNet 64×64 using EDM [13] as the pretrained model in Table 2, yielding consistent conclusions.

Particularly, we combine ER-SDE-Solvers with *classifier guidance* to generate high-resolution images. Table 4 provides comparative results on ImageNet 256×256 [4] using Guided-diffusion [5] as the pretrained model. We surprisingly find that ER-SDE-Solvers with *classifier guidance* exhibit high image generation quality even with very low NFE. This may be attributed to the customized noise injected into the sampling process, which mitigates the inaccuracies in data estimation introduced by classifier gradient guidance. Further investigation is left for future work.

6. Conclusion

We address the challenges of sampling speed and image visual quality in DMs. Initially, we formulate the sampling process as an ER SDE, which unifies ODEs and SDEs in previous studies. Leveraging the semi-linear structure of ER SDE solutions, we provide exact solutions and high-stage approximations for both VP SDE and VE SDE. Building upon it, we introduce one-step prediction errors and establish two crucial findings from mathematical standpoints: the superior performance of ODE solvers for rapid sampling and the high-quality sampling ability of SDE solvers, and the comparable performance of VP SDE solvers with VE SDE solvers. Finally, leveraging the advantages of both ODE solvers and SDE solvers, we introduce state-of-the-art efficient high-quality samplers, known as ER-SDE-Solvers.

Impact Statement

In line with other advanced deep generative models like GANs, DMs can be harnessed to produce deceptive or misleading content, particularly in manipulated images. The efficient solvers we propose herein offer the capability to expedite the sampling process of DMs, thereby enabling faster image generation and manipulation, potentially leading to the creation of convincing but fabricated visuals. As with any technology, this acceleration could accentuate the potential ethical concerns associated with DMs, particularly their susceptibility to misuse or malicious applications. For instance, more frequent image generation might elevate the likelihood of unauthorized exposure of personal information, facilitate content forgery and dissemination of false information, and potentially infringe upon intellectual property rights.

References

[1] Omri Avrahami, Dani Lischinski, and Ohad Fried. Blended diffusion for text-driven editing of natural images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18208–18218, 2022. 1

[2] 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. 2, 3

[3] Hyungjin Chung, Byeongsu Sim, and Jong Chul Ye. Come-closer-diffuse-faster: Accelerating conditional diffusion models for inverse problems through stochastic contraction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12413–12422, 2022. 1

[4] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 7, 8

[5] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34:8780–8794, 2021. 1, 7, 8

[6] Sicheng Gao, Xuhui Liu, Bohan Zeng, Sheng Xu, Yanjing Li, Xiaoyan Luo, Jianzhuang Liu, Xiantong Zhen, and Baochang Zhang. Implicit diffusion models for continuous super-resolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10021–10030, 2023. 1

[7] Martin Gonzalez, Nelson Fernandez, Thuy Vinh Dinh Tran, Elies Gherbi, Hatem Hajri, and Nader Masmoudi. SEEDS: Exponential SDE solvers for fast high-quality sampling from diffusion models. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. 4, 5, 7

[8] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Advances in neural information processing systems*, 27, 2014. 1

[9] 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. *Advances in neural information processing systems*, 30, 2017. 7

[10] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020. 1, 3, 4

[11] Aapo Hyvärinen and Peter Dayan. Estimation of non-normalized statistical models by score matching. *Journal of Machine Learning Research*, 6(4), 2005. 3

[12] 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. 3, 6

[13] Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models. *Advances in Neural Information Processing Systems*, 35:26565–26577, 2022. 1, 2, 4, 6, 7, 8

[14] Diederik Kingma, Tim Salimans, Ben Poole, and Jonathan Ho. Variational diffusion models. *Advances in neural information processing systems*, 34:21696–21707, 2021. 3, 4

[15] Peter E Kloeden and Eckhard Platen. *Stochastic differential equations*. Springer, 1992. 3

[16] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009. 7

[17] Steven J Leon, Lisette G De Pillis, and Lisette G De Pillis. *Linear algebra with applications*. Pearson Prentice Hall Upper Saddle River, NJ, 2006. 6

[18] 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. *Advances in Neural Information Processing Systems*, 35:5775–5787, 2022. 1, 2, 3, 7

[19] Cheng Lu, Yuhao Zhou, Fan Bao, Jianfei Chen, Chongxuan Li, and Jun Zhu. Dpm-solver++: Fast solver for guided sampling of diffusion probabilistic models. *arXiv preprint arXiv:2211.01095*, 2022. 4, 5, 7

[20] Ziwei Luo, Fredrik K Gustafsson, Zheng Zhao, Jens Sjölund, and Thomas B Schön. Image restoration with mean-reverting stochastic differential equations. In *Proceedings of the 40th International Conference on Machine Learning*, pages 23045–23066. PMLR, 2023. 1

[21] Chenlin Meng, Yutong He, Yang Song, Jiaming Song, Jiajun Wu, Jun-Yan Zhu, and Stefano Ermon. Sdedit: Guided image synthesis and editing with stochastic differential equations. In *International Conference on Learning Representations*, 2021. 1

[22] Chenlin Meng, Robin Rombach, Ruiqi Gao, Diederik Kingma, Stefano Ermon, Jonathan Ho, and Tim Salimans. On distillation of guided diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14297–14306, 2023. 2

[23] Charlie Nash, Jacob Menick, Sander Dieleman, and Peter W Battaglia. Generating images with sparse representations. *arXiv preprint arXiv:2103.03841*, 2021. 7

[24] Bernt Oksendal. *Stochastic differential equations: an introduction with applications*. Springer Science & Business Media, 2013. 3

[25] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in Neural Information Processing Systems*, 35:36479–36494, 2022. 4

[26] Chitwan Saharia, Jonathan Ho, William Chan, Tim Salimans, David J Fleet, and Mohammad Norouzi. Image super-resolution via iterative refinement. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(4):4713–4726, 2022. 1

[27] Tim Salimans and Jonathan Ho. Progressive distillation for fast sampling of diffusion models. In *International Conference on Learning Representations*, 2021. 2

[28] 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. 3

[29] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *International Conference on Learning Representations*, 2021. 1, 2, 7

[30] Yang Song, Prafulla Dhariwal, Mark Chen, and Ilya Sutskever. Consistency models. *arXiv preprint arXiv:2303.01469*, 2023. 2

[31] Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. *Advances in neural information processing systems*, 32, 2019. 3

[32] 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, 3, 4, 6

[33] Xuan Su, Jiaming Song, Chenlin Meng, and Stefano Ermon. Dual diffusion implicit bridges for image-to-image translation. In *The Eleventh International Conference on Learning Representations*, 2022. 1

[34] Shuchen Xue, Mingyang Yi, Weijian Luo, Shifeng Zhang, Jiacheng Sun, Zhenguo Li, and Zhi-Ming Ma. Sa-solver: Stochastic adams solver for fast sampling of diffusion models. *arXiv preprint arXiv:2309.05019*, 2023. 2, 4

[35] Qinsheng Zhang and Yongxin Chen. Diffusion normalizing flow. *Advances in Neural Information Processing Systems*, 34:16280–16291, 2021. 3

[36] Qinsheng Zhang and Yongxin Chen. Fast sampling of diffusion models with exponential integrator. In *International Conference on Learning Representations*, 2023. 2, 3, 4

[37] Qinsheng Zhang, Molei Tao, and Yongxin Chen. gddim: Generalized denoising diffusion implicit models. In *International Conference on Learning Representations*, 2023. 2

[38] Min Zhao, Fan Bao, Chongxuan Li, and Jun Zhu. Egsde: Unpaired image-to-image translation via energy-guided stochastic differential equations. *Advances in Neural Information Processing Systems*, 35:3609–3623, 2022. 1

[39] 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. 2, 3