

An Overview of Diffusion Models: Applications, Guided Generation, Statistical Rates and Optimization*

Minshuo Chen¹ Song Mei² Jianqing Fan¹ Mengdi Wang¹
¹Princeton University ²UC Berkeley

April 12, 2024

Abstract

Diffusion models, a powerful and universal generative AI technology, have achieved tremendous success in computer vision, audio, reinforcement learning, and computational biology. In these applications, diffusion models provide flexible high-dimensional data modeling, and act as a sampler for generating new samples under active guidance towards task-desired properties. Despite the significant empirical success, theory of diffusion models is very limited, potentially slowing down principled methodological innovations for further harnessing and improving diffusion models. In this paper, we review emerging applications of diffusion models, understanding their sample generation under various controls. Next, we overview the existing theories of diffusion models, covering their statistical properties and sampling capabilities. We adopt a progressive routine, beginning with unconditional diffusion models and connecting to conditional counterparts. Further, we review a new avenue in high-dimensional structured optimization through conditional diffusion models, where searching for solutions is reformulated as a conditional sampling problem and solved by diffusion models. Lastly, we discuss future directions about diffusion models. The purpose of this paper is to provide a well-rounded theoretical exposure for stimulating forward-looking theories and methods of diffusion models.

Contents

1	Introduction	2
2	Diffusion Model Preliminaries	4
	2.1 Forward and Backward Processes	4
	2.2 Conditional Diffusion Models	5
3	Emerging Applications of Diffusion Models	6
	3.1 Vision and Audio Generation	6
	3.2 Control and Reinforcement Learning	7
	3.3 Life-Science Applications	8
	3.4 Black-Box Optimization	9

*Emails: {minshuochen, jqfan, mengdiw}@princeton.edu, songmei@berkeley.edu

扩散模型理论

4	Theoretical Progress on Unconditional Diffusion Models	9
4.1	Learning Score Function	10
4.2	Score Approximation and Estimation	11
4.2.1	Score Approximation Guarantees	12
4.2.2	Sample Complexity of Score Estimation	13
4.2.3	Score Estimation in Graphical Models	13
4.3	Sampling and Distribution Estimation	14
4.3.1	Sampling Theory	14
4.3.2	Sample Complexity of Distribution Estimation	15
4.4	Alternative Formulation: Stochastic Localization	16
5	Theoretical Progress on Conditional Diffusion Models	16
5.1	Learning Conditional Score	17
5.2	Conditional Score and Distribution Estimation	18
5.3	Theoretical Insights on Strength of Guidance	19
6	Diffusion Model for Optimization	20
7	Future Directions	22
7.1	Connection to Stochastic Control	23
7.2	Adversarial Robustness and Distributionally Robust Optimization	23
7.3	Discrete Diffusion Models	24
8	Conclusion	25

条件扩散模型理论

1 Introduction

The field of artificial intelligence (AI) has been revolutionized by generative models, particularly large language models and diffusion models. Recognized as foundation models [1], they are trained on massive corpora of data and have opened up vibrant possibilities in machine learning research and applications. While large language models focus on generating coherent text based on context, diffusion models excel at modeling complex data distributions and generating diverse samples, both of which find widespread use across various domains.

Diffusion models, inspired by thermodynamics modeling [2], have emerged in recent years with ground-breaking performance, surpassing the previous state-of-the-art, such as Generative Adversarial Networks (GANs) [3, 4] and Variational AutoEncoders (VAEs) [5, 6]. Diffusion models are widely adopted in computer vision and audio generation tasks [7–22], and further utilized in text generation [23–25], sequential data modeling [26–29], reinforcement learning and control [30–35], as well as life-science [36–54]. For a more comprehensive exposition of applications, we refer readers to survey papers [22, 55–59].

The celebrated performance of diffusion models is indispensable to numerous methodological innovations that significantly expand the scope and boost the functionality of diffusion models, enabling high-fidelity generation, efficient sampling, and flexible control of the sample generation. For example, [60–63] extend diffusion models to discrete data generation, while the vanilla diffusion models target at continuous data. Meanwhile, there is an active line of research aiming to expedite the sample generation speed of diffusion models [64–74]. Last but not the least, a recent surge of research focuses on fine-tuning diffusion models towards generating samples of desired properties, such as generating images with peculiar aesthetic qualities [75–83]. These task-specific properties

are often encoded as guidance to the diffusion model, consisting of conditioning and control signals to steer the sample generation. Notably, guidance allows for the creation of diverse and relevant content across a wide range of applications, which underscores the versatility and adaptability of diffusion models. We term diffusion models with guidance as conditional diffusion models.

Despite the rapidly growing body of empirical advancements, theories of diffusion models fall far behind. Some recent theories view diffusion models as an unsupervised distribution learner and sampler, and thus establish their sampling convergence guarantees [84–89] and statistical distribution learning guarantees [90–92]. Such results offer invaluable theoretical insights into the efficiency and accuracy of diffusion models for modeling complex data, with a central focus on the unconditioned diffusion models in distribution estimation. This leaves a gap between theory and practice for conditional diffusion models. In specific, a theoretical foundation to support and motivate principled methodologies for guidance design and adapting diffusion models to task-specific needs is still lacking.

This paper serves as a contemporary exposure to diffusion models for stimulating sophisticated and forward-looking study on them. We mainly focus on the following fundamental theoretical questions of diffusion models:

- Can diffusion models learn data distributions accurately and efficiently? If so, what is the sample complexity, especially for structured data?
- Can conditional diffusion models generate distributions aligned to guidance? If so, how can we properly design the guidance and what is the sample complexity?

For a systematic study, we will first review how diffusion models work and their emerging applications. Then we provide an overview of the existing theoretical underpinnings pertinent to the questions above. Our ultimate goal is to demonstrate and harness the power of diffusion models, connecting to broad interdisciplinary areas of applied mathematics, statistics, computational biology, and operations research.

Paper Organization The rest of the paper is organized as follows. In Section 2, we present a continuous-time description of diffusion models using stochastic differential equations. The advantage of the continuous-time point of view lies in the clean and systematic formulation, and the seamless application of discretization schemes to replicate practical implementations.

In Section 3, we review emerging applications of diffusion models, especially in various controlled generation tasks, aiming to elucidate the conditional distributions that diffusion models attempt to capture. Then, in Section 3.4, we relate conditional generation to black-box optimization via gauging the quality of the generated samples under control by a reward function.

In Section 4, we delve into theoretical preliminaries and review theories of diffusion models. Specifically, in Section 4.1, we discuss how to learn the score function. Section 4.2 provides approximation theories for understanding proper neural network architectures for learning the score and statistical sample complexities of estimating the score function. Section 4.3 then discusses statistical sample complexities of distribution estimation using diffusion models and sampling theories by viewing diffusion models as samplers.

In Section 5, we focus on conditional diffusion models, continuing a similar study in Section 4. We introduce learning methods of conditional score functions in Section 5.1, connecting them to unconditional score by a so-called “guidance” term. This also motivates fine-tuning methods for conditional diffusion models. Section 5.2 then summarizes unconditional score approximation, estimation, and distribution learning theories. Section 5.3 revisits the guidance in conditional score functions and establishes theoretical insights for the impact of guidance.

In Section 6, we review theories and methodologies of data-driven black-box optimization using conditional diffusion models. We highlight that diffusion models generate high-fidelity solutions to the optimization objective function, preserving data latent structures, and the quality of the solutions aligns with the optimal off-policy bandit. This opens up new possibilities for optimization in high-dimensional complex and structured spaces through diffusion models.

Lastly, in Section 7, we discuss future directions and connections of diffusion models to broad research areas.

2 Diffusion Model Preliminaries

Roughly speaking, diffusion model consists of a forward process and a backward process. In the forward process, a clean sample from the data distribution is sequentially corrupted by Gaussian random noise, and in the infinite-time limit, the data distribution is transformed into pure noise. In the backward process, a denoising neural network is trained to sequentially remove the added noise distribution in data and restore new clean data distribution. The forward and backward processes are depicted in Figure 1.

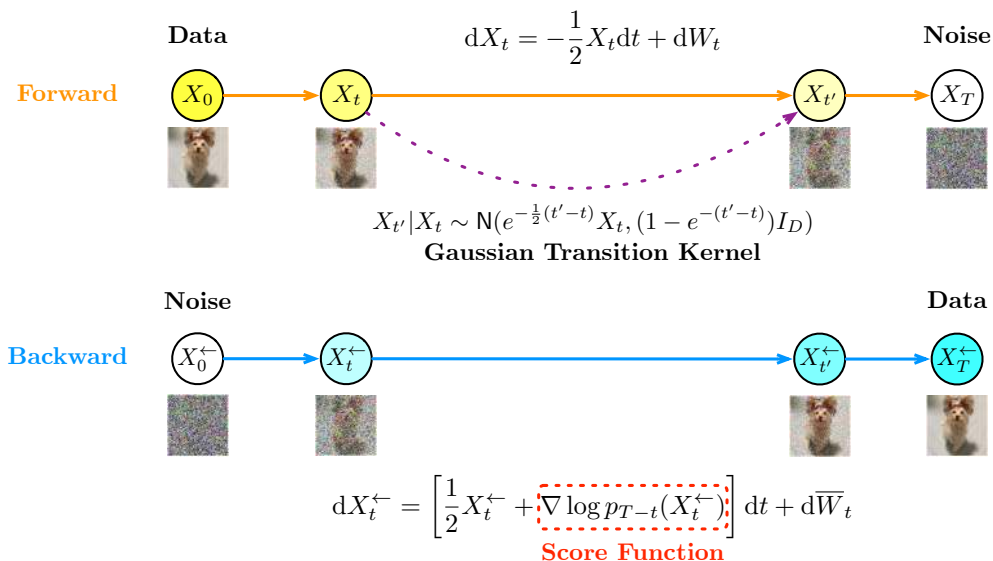


Figure 1: Demonstration of forward and backward processes in diffusion models. The forward process is a noise corruption process, while the backward process is used for new sample generation.

To fully decipher how diffusion models work, we describe the forward and backward processes in a continuous time limit and review how to implement the backward process. Next, we will introduce guidance to realize conditioning in controlled sample generation using conditional diffusion models.

2.1 Forward and Backward Processes

The forward process in diffusion models progressively adds noise to the original data. Here we consider the Ornstein-Uhlenbeck process, which is described by the following Stochastic Differential Equation (SDE),

保持差前向SDE: $f(x,t) = -\frac{1}{2}g(t)x_t, \quad \gamma(t) = \sqrt{g(t)}$

$$dX_t = -\frac{1}{2}g(t)X_t dt + \sqrt{g(t)}dW_t \quad \text{for } g(t) > 0, \quad (1)$$

随时间 X_t 会由高维分布至低维分布

$$\mathcal{N}(g(t)X_0, (1-g(t))I)$$

$$d(\alpha) = e^{-\int_0^t \frac{1}{2}g(s)ds}$$

where initial $X_0 \sim P_{\text{data}}$ follows the data distribution, $(W_t)_{t \geq 0}$ is a standard Wiener process, and $g(t)$ is a nondecreasing weighting function. We denote the marginal distribution of X_t at time t as P_t . After an infinitesimal time, the forward process (1) shrinks the magnitude of data and corrupts data by Gaussian noise. More precisely, given X_0 , the conditional distribution of $X_t|X_0$ is Gaussian $\mathcal{N}(\alpha(t)X_0, h(t)I_D)$, where $\alpha(t) = \exp(-\int_0^t \frac{1}{2}g(s)ds)$ and $h(t) = 1 - \alpha^2(t)$. Consequently, under mild conditions, (1) transforms initial distribution P_{data} to $P_\infty = \mathcal{N}(0, I_D)$. Therefore, (1) is known as the variance preserving forward SDE [10, 93].

The value of $g(t)$ controls the noise corruption speed in the forward process. In real-world usage, various choices on $g(t)$ are implemented. One example is to choose $g(t)$ so that the variance of the Gaussian noise in the forward process increases linearly with respect to time [9]. Later, several improvement techniques of $g(t)$ are proposed, such as a cosine-based variance schedule [94]. To simplify our presentation, we take $g(t) = 1$ for all t in the sequel.

The forward process (1) will terminate at a sufficiently large time $T > 0$, where the corrupted marginal distribution P_T is expected to be close to the standard Gaussian distribution. Then diffusion models generate fake data by reversing the time of (1), which leads to the following backward SDE,
 ← 表示与正向过程区分

$$dX_t^\leftarrow = \left[\frac{1}{2}X_t^\leftarrow + \nabla \log p_{T-t}(X_t^\leftarrow) \right] dt + d\bar{W}_t \quad \text{for } t \in [0, T], \quad (2)$$

where $\nabla \log p_t(\cdot)$ is the so-called “score function”, i.e., the gradient of the log probability density function of P_t , \bar{W}_t is another Wiener process independent of W_t , and we use the superscript \leftarrow for distinguishing with the forward process (1). Under mild conditions, when initialized at $X_0^\leftarrow \sim P_T$, the backward process $(X_t^\leftarrow)_{0 \leq t < T}$ has the same distribution as the forward process $(X_{T-t})_{0 \leq t < T}$ [95, 96].

Working with (2), however, leads to difficulties, as both the score function $\nabla \log p_t$ and the distribution P_T are unknown. Therefore, several surrogates are deployed in practice. Firstly, we replace the unknown distribution P_T by the standard Gaussian distribution $\mathcal{N}(0, I_D)$. Secondly, we denote $\hat{s}(x, t)$ as an estimator to the ground truth score function $\nabla \log p_t(x)$. The estimated score \hat{s} is often parameterized by a deep neural network and takes data and time as inputs. Substituting \hat{s} into the backward process, we obtain the following practical continuous-time backward SDE,

反向 SDE:
$$d\tilde{X}_t^\leftarrow = \left[\frac{1}{2}\tilde{X}_t^\leftarrow + \hat{s}(\tilde{X}_t^\leftarrow, T-t) \right] dt + d\bar{W}_t \quad \text{with } \tilde{X}_0^\leftarrow \sim \mathcal{N}(0, I_D). \quad (3)$$

Diffusion models then generate data by simulating a discretization of (3) with a proper step size. A common practice is to set the step size of order $\mathcal{O}(1/1000)$ so that the backward SDE (3) is discretized to hundreds of steps [9, 64, 94].

It is worth mentioning that simulating the backward process for thousands of steps to generate a sample is time-consuming. Accelerating the sampling speed of diffusion models is an active research direction [64–74]. Some notable methods include sampling with stride to reduce the backward steps [64, 68, 94], replacing the backward SDE (3) with an ODE or DDIM (Denoising Diffusion Implicit Models) [65, 70, 71], using a pre-trained VAE to extract low-dimensional data representations and then implementing diffusion processes – known as latent diffusion [66], training distillation and consistency models [67, 69, 97], as well as rectified flows [73]. These methods have found extensive adoption in highly fine-tuned diffusion models, such as Sora and Stable Diffusion [98, 99].

2.2 Conditional Diffusion Models

Conditional diffusion models generate samples analogous to the unconditioned one, while the major difference is the added conditional information. We denote the conditional information as y . Then

the goal of conditional diffusion models is to generate samples from the conditional data distribution $P(\cdot|y)$. The conditional forward process is again an Ornstein-Uhlenbeck process:

条件正向SDE:
$$dX_t^y = -\frac{1}{2}X_t^y dt + dW_t \quad \text{with } X_0^y \sim P_0(\cdot|y) \quad \text{and } t \in (0, T]. \quad (4)$$

Note that the initial distribution is now a conditional distribution $P_0(\cdot|y)$, which is different from the unconditioned forward process (1). The noise corruption is only performed on x , while y is kept fixed. We use the superscript y to emphasize the dependence of the process on y . Similarly, for sample generation, the backward process reverses the time in (4):

与一般SDE不同. 正向和反向均使 y 相等. 条件得分函数

$$dX_t^{y,\leftarrow} = \left[\frac{1}{2}X_t^{y,\leftarrow} + \nabla \log p_{T-t}(X_t^{y,\leftarrow}|y) \right] dt + d\bar{W}_t \quad \text{for } t \in [0, T]. \quad (5)$$

Here $\nabla \log p_{T-t}(X_t^{y,\leftarrow}|y)$ is the so-called “conditional score function”, which replaces the score function in (2). The initialization is identical to (2) as $X_0^{y,\leftarrow} \sim \mathcal{N}(0, I_D)$, independent of the guidance y . Despite the similarity in forward and backward processes, the major difference between conditional diffusion models and unconditioned ones lies in the estimation of the conditional score function $\nabla \log p_t(\cdot|y)$. In specific, the conditional score function can be related to the unconditioned one, which motivates a collection of practical learning and fine-tuning methods, e.g., classifier guidance and classifier-free guidance [9, 10, 100]. We defer an in-depth discussion to Section 5.1.

With an estimated conditional score function $\hat{s}(x, y, t)$ replacing the ground truth conditional score $\nabla \log p_t(x|y)$, the conditional sample generation is to simulate the following backward process

条件反向SDE:
$$d\tilde{X}_t^{y,\leftarrow} = \left[\frac{1}{2}\tilde{X}_t^{y,\leftarrow} + \hat{s}(\tilde{X}_t^{y,\leftarrow}, y, T-t) \right] dt + d\bar{W}_t \quad \text{with } \tilde{X}_0^{y,\leftarrow} \sim \mathcal{N}(0, I_D).$$

In practical implementations, a proper discretization scheme is applied.

3 Emerging Applications of Diffusion Models

Through extensive developments [7, 9, 10, 94], modern diffusion models have achieved a startling success and are implanted in various applications (see, for example, the survey [55]). We highlight vast applications of diffusion models in the following, with a particular emphasis on conditional diffusion models for controlled sample generation.

3.1 Vision and Audio Generation

Diffusion models achieve state-of-the-art performance in image and audio generation tasks [7–22] and are one of the fundamental building blocks of image and audio synthesis systems, such as DALL-E [66], stable diffusion [101], and Diffwave [11].

Diffusion models’ performance is appraised of high-fidelity sample generation and allows versatile guidance to control the generation. The simplest example of generation under guidance is to generate images of certain categories, such as cats or dogs. Such categorical information is taken as a conditional signal and fed into conditional diffusion models. In more detail, we train conditional diffusion models using a labeled data set consisting of sample pairs (x_i, y_i) , where y_i is the label of an image x_i . The training is to estimate a conditional score function using the data set, modeling the correspondence between x and y . In this way, conditional diffusion models are learning the conditional distribution $P(x = \text{image} | y = \text{given label})$ and allow sampling from the distribution.

In text-to-image synthesis systems, the conditional information is an input text prompt, which can be a sentence consisting of objects or more abstract requirements, e.g., aesthetic quality. To generate images aligned with prompts, conditional diffusion models are trained with a massive annotated data set encompassing image and text summary pairs denoted as (x_i, y_i) . The text y_i will be transformed into a word embedding and taken as input to a conditional diffusion model. Similar to the generation of images in certain categories, conditional diffusion models for text-to-image synthesis learn the conditional distribution $P(x = \text{image} \mid y = \text{text prompt})$ and allow sampling from it. In more sophisticated synthesis systems, some fine-tuning steps are implemented to further enable abstract prompt conditioning and improve the quality of generated images. For example, [78] reformulates the discretized backward process (2) as a finite-horizon Markov Decision Process (MDP). The state space represents images, the conditional score function is viewed as a policy, and a reward function is defined to measure the alignment of an image to its desired text prompt. Therefore, to generate prompt-aligned images amounts to optimize reward via finding an optimal policy. [78] proposes a policy gradient-based method for fine-tuning pre-trained diffusion models. In Figure 2, we demonstrate a progressive improvement from left to right of fine-tuning a conditional diffusion model using the method in [78].

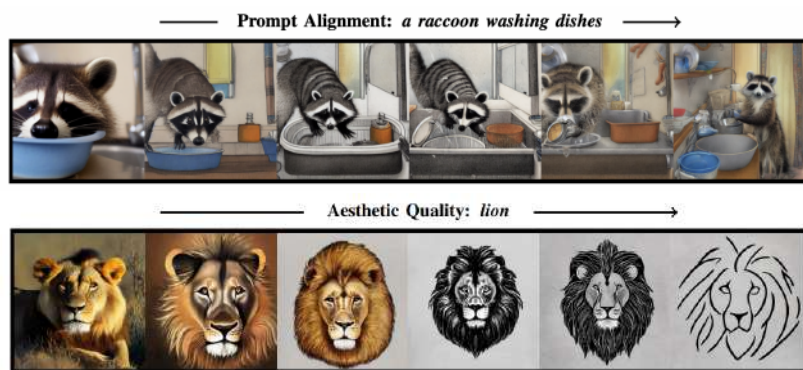


Figure 2: Conditional diffusion models generate images under various guidance [78]. The upper row demonstrates an alignment with text description consisting of multiple objects. The lower row demonstrates an abstract description of aesthetic quality.

Conditional diffusion models are also a powerful tool in image editing and restoration [102–109], as well as audio enhancement [110–113]; see also surveys [22, 56] and the references therein. To showcase the idea, we consider the image inpainting task as an example. The goal of inpainting is to predict missing pixels of an image. We denote the known region of an image as y and the original full image as x . Then inpainting boils down to sampling x from the conditional distribution $P(x = \text{full image} \mid y = \text{known region of the image})$. In all these applications, conditional diffusion models are shown to be highly expressive and effective in modeling the conditional distributions [10, 107].

3.2 Control and Reinforcement Learning

Apart from primary computer vision and audio tasks, diffusion models are actively deployed in Reinforcement Learning (RL) and control problems with appealing performance. For example, [30–33, 35] utilize conditional diffusion models to parameterize control/RL policies in highly complicated tasks, e.g., robot control and human behavior imitation. An extended review of the connection between diffusion models and RL can be found in [34]. In RL/control problems, a policy is a

conditional probabilistic distribution on the action space given the state of an underlying dynamical system. Accordingly, when using diffusion models to parameterize policies, the goal is to learn a distribution $P(a = \text{action} \mid y = \text{system states})$. [30, 32] focus on the imitation learning scenario, where the goal is to mimic the behaviors of an expert. The data set contains expert demonstrations denoted by (y_i, a_i) pairs. Here y_i is the state of the system and a_i is the expert’s chosen action. Analogous to text-to-image synthesis, we train a conditional score network using the data set to capture the dependency between states and actions. During inference, given a new system state, we use the learned conditional diffusion model to generate plausible actions. Diffusion-QL [114] further adds regularization to the training of the conditional diffusion model and tries to learn optimal actions based on a pre-collected data set.

Diffusion models also embody a new realm for algorithm design in control and RL problems by viewing sequential decision making as generative sequence modeling. In a typical task of reward-maximization planning in RL, the goal is to find an optimal policy that achieves large accumulative rewards. Conventional methods rely on iteratively solving for the Bellman optimality to obtain a corresponding policy. Generative sequence modeling, however, directly produces state-action trajectories of large rewards, avoiding explicitly solving for Bellman optimality. In other words, generative sequence modeling directly samples from the conditional distribution $P(\tau = \text{state-action trajectory} \mid \tau \text{ attains large reward})$. Early success was demonstrated with transformer generative models [115, 116]. Later, conditional diffusion models are deployed with state-of-the-art performance. Namely, Diffuser [117] generates state-action trajectories conditioned on high reward as guidance via conditional diffusion models. Decision Diffuser [118] presents conditional trajectory generation, taking reward, constraints, or skills as guidance and enhances Diffuser’s performance. For instance, given a pre-collected data set consisting of (τ_i, y_i) , where τ_i is the state-action trajectory and y_i is the accumulative reward of τ_i . We use a conditional diffusion model to model the conditional distribution $P(\tau|y)$, by estimating the conditional score function. After training, we specify a proper target reward value and deploy the conditional diffusion model to generate sample trajectories. A policy can then be extracted from the generated trajectories via an inverse dynamics model [119]. See the working flow of the decision diffuser in Figure 3. Adapt-Diffuser [120] further introduces a discriminator for fine-tuning the conditional diffusion models, allowing self-evolution and adaptation to out-of-distribution tasks.

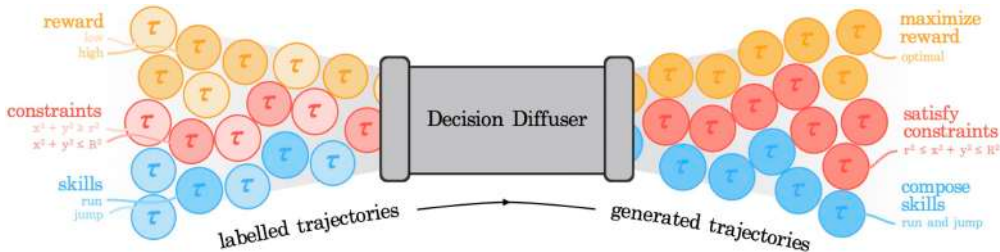


Figure 3: Decision diffuser in [118]. The model is trained on labeled trajectories and is capable of generating state-action trajectories conditioned on desired reward values, constraints, or skills.

3.3 Life-Science Applications

In life-science applications, conditional diffusion models are making ever profound impacts [36–54]. See also a survey [57] on applications of diffusion models in bioinformatics. These results cover diverse tasks including single-cell image analysis, protein design and generation, drug design, small

molecule generation, etc. The performance surpasses many of their predecessors using autoregressive, VAE, or GAN-type deep generative models [121–124].

To demonstrate the use of conditional diffusion models, we take protein design as an example. Protein design can be posed as a problem of finding a sequence w of certain length, where each coordinate of the sequence represents the structural information of the protein. A protein is only useful if it can be expressed in living cells. A widely adopted metric of usefulness is the likelihood of a protein sequence being a natural one [50]. In addition, the binding affinity and aggregation tendency are also vital properties of the protein structure. Combined with the usefulness metric, all these properties can be summarized by a vector-valued function $f(w)$. In this sense, conditional diffusion models actually generate protein sequences w following a conditional distribution $P(w \mid f(w) \in \mathcal{E})$, where \mathcal{E} is a set describing plausible protein structures. The training of conditional diffusion models for protein generation is analogous to text-to-image diffusion models, based on a training data set containing diverse protein structures with measured properties. In the inference stage, we can first sample one configuration from \mathcal{E} and conditioned on the configuration, we generate new proteins.

3.4 Black-Box Optimization

In control, RL, and life-science applications, various guidance may be summarized as an abstract reward function $V(\cdot)$. Then the goal is to generate new samples from a conditional distribution, aiming to optimize the reward. Consequently, conditional diffusion models act as an optimizer which generates optimal solutions.

We revisit the example of offline reward-maximization planning in RL. Recall that our data set comprises state-action trajectories τ_i and the associated accumulative rewards $y_i = V(\tau_i) + \epsilon_i$, where ϵ_i is an independent observation noise. Reward-maximization planning essentially seeks solutions to the black-box optimization problem $\operatorname{argmax}_{\tau} V(\tau)$. In this setting, we are prohibitive to interact with the target function V beyond the given data set [125]. Early existing works utilize GANs for optimal solution generation [126], yet suffer from training instability and mode collapse issues. Recently, [127] empirically presents superior performance of generating high-quality solutions using conditional diffusion models. The idea is to transform the black-box optimization problem into a conditional sampling problem. In specific, given a proper target value a , conditional diffusion models generate solutions from the conditional distribution $P(\tau \mid V(\tau) = a)$. The subtlety stems from how to properly choose the target value a to ensure the high-quality of the generated solutions. Roughly speaking, we are luring to choose a large a so that the generated solutions achieve large rewards. However, if we choose a too large compared to the given data set, significant extrapolation is required to generate corresponding solutions, leading to potential quality degradation. Consequently, a proper choice on a heavily depends on the coverage of the collected data set. [128] provides theoretical guidelines on how to choose a to ensure good generated solutions, which we will introduce in Section 6. Empirically, [127] proposes several methods to encourage large-reward solutions during the training of the conditional diffusion model, such as sample reweighting — assigning large weights to samples with large rewards.

4 Theoretical Progress on Unconditional Diffusion Models

This section reviews recent progress in the theoretical understanding of diffusion models. We recall from Section 2 that the score function is the key to implementing a diffusion model. From a theoretical perspective, the performance of diffusion models is intimately tied to whether or not the score function can be learned accurately. For a systematic treatment, we first introduce methods for learning the score and then dive into their theoretical insights. Specifically, we discuss how to

properly choose neural networks for learning the score function, based on the universal and adaptive approximation capability of neural networks. More importantly, we demonstrate structural properties in the score function induced by data distribution assumptions, e.g., low-dimensional support and graphical models. Then we provide statistical sample complexities for estimating the score using the chosen neural networks. We are particularly interested in understanding how score estimation circumvents the curse of dimensionality issues in high-dimensional settings. Lastly, we study statistical rates for estimating the data distribution.

4.1 Learning Score Function

We consider the goal of learning the score function $\nabla \log p_t(x_t)$ using neural networks. A naive objective function is the weighted quadratic loss:

$$\min_{s \in \mathcal{S}} \int_0^T w(t) \mathbb{E}_{x_t \sim P_t} \left[\|\nabla \log p_t(x_t) - s(x_t, t)\|_2^2 \right] dt, \quad (6)$$

(6) 的积分项, 又任以字=

where $w(t)$ is a weighting function and \mathcal{S} is a concept class (deep neural networks). However, such an objective function is not computable using samples, as the score function $\nabla \log p_t$ is unknown. As shown in the seminal works [129] and [130], rather than minimizing the integral (6), we can minimize an equivalent objective function,

$$\min_{s \in \mathcal{S}} \int_0^T w(t) \mathbb{E}_{x_0 \sim P_{\text{data}}} \left[\mathbb{E}_{x_t \sim \mathcal{N}(\alpha(t)x_0, h(t)I_D)} \left[\|\nabla_{x_t} \log \phi_t(x_t|x_0) - s(x_t, t)\|_2^2 \right] \right] dt. \quad (7)$$

降噪函数(7)的积分项

Here, $\phi_t(x_t|x_0)$ denotes the Gaussian transition kernel of the forward process, so that $\nabla \log \phi_t$ admits an analytical form

$$\nabla_{x_t} \log \phi_t(x_t|x_0) = -\frac{x_t - \alpha(t)x_0}{h(t)}.$$

d(t) = e^{-\int_0^t \frac{1}{2}g(s)ds}
h(t) = \int_0^t \sigma^2(s)ds

By this analytical expression, we could approximate the objective (7) using finite samples. Note that $\nabla_{x_t} \log \phi_t(x_t|x_0)$ is the noise added to x_0 at time t . Therefore, (7) is also known as the denoising score matching. As shown in [131, 132], denoising score matching can also be derived using a variational perspective, reproducing the evidence lower bound for regularized data negative likelihood minimization.

Score Blowup and Early-Stopping One challenge of optimizing (7) is the score blowup issue [64, 133]. To demonstrate the phenomenon, we consider a data distribution which lies in a linear subspace, where $x = Az$ for a representation matrix $A \in \mathbb{R}^{D \times d}$ and a latent variable $z \in \mathbb{R}^d$. Here D represents the ambient dimension of data and d is the intrinsic dimension, which is often much smaller than D . As shown in [91], the ground truth score $\nabla \log p_t(x)$ assumes the following orthogonal decomposition,

$$\nabla \log p_t(x) = A \nabla \log p_t^{\text{ld}}(A^\top x) + \underbrace{\frac{1}{1 - e^{-t}}(I - AA^\top)}_{(\mathcal{T})} x, \quad (8)$$

where p_t^{ld} is the marginal density function of applying the forward diffusion process (1) on the latent variable z . As can be seen, the term (\mathcal{T}) is orthogonal to the subspace spanned by matrix A . More importantly, as t approaches 0, the magnitude of (\mathcal{T}) grows to infinity as long as $x \neq 0$. The reason behind this is that (\mathcal{T}) enforces the orthogonal component to vanish so that the low-dimensional subspace structure is reproduced in generated samples. Such a blowup issue appears

早期停止

in all geometric data [133]. As a consequence, an early stopping time $t_0 > 0$ is introduced and the practical score estimation loss is written as

$$\min_{s \in \mathcal{S}} \int_{t_0}^T w(t) \mathbb{E}_{x_0 \sim P_{\text{data}}} \left[\mathbb{E}_{x_t \sim \mathcal{N}(\alpha(t)x_0, h(t)I_D)} \left[\left\| \nabla_{x_t} \log \phi_t(x_t | x_0) - s(x_t, t) \right\|_2^2 \right] \right] dt. \quad (9)$$

For practical implementation, we approximate (9) by its empirical version. Specifically, given n i.i.d. data points $x_i \sim P_{\text{data}}$ for $i = 1, \dots, n$, we sample x_t given $x_0 = x_i$ from the Gaussian distribution $\mathcal{N}(\alpha(t)x_i, h(t)I_D)$. We also sample time t from the interval $[t_0, T]$ to approximate the integration with respect to t .

Remark 1 (Network class \mathcal{S}). A common choice of the network class \mathcal{S} is the U-Net [134], as demonstrated in Figure 4. The network architecture utilizes convolution layers and shortcut connections. In the network, an input is first compressed into a low-dimensional representation and then gradually lifted back to the original dimension. This encoder-decoder type structure aims to extract intrinsic structures in data and leads to an efficient learning. More recently, using transformer-based score network has demonstrated outstanding performance [98, 135, 136], which excels in capturing spatial-temporal dependencies in data.

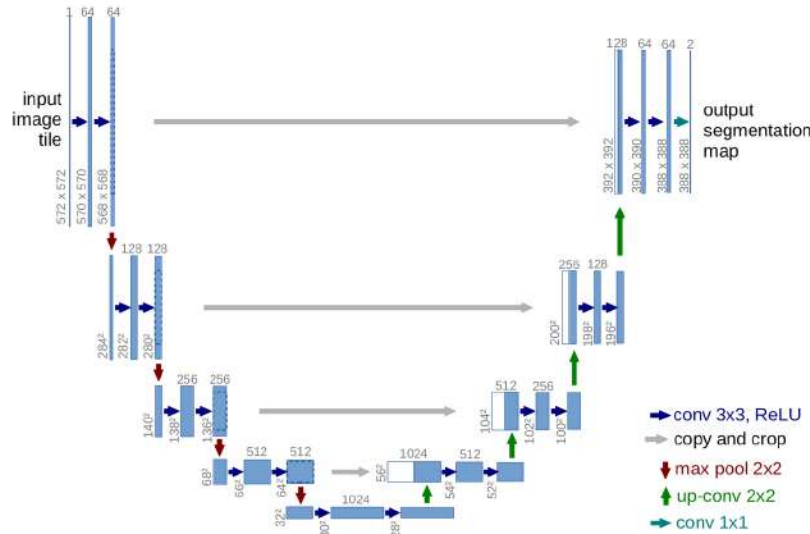


Figure 4: U-Net architecture in [134] for 32×32 resolution RGB images. When generating new samples using a discretized backward process, diffusion models utilize the U-Net at each discretization step for transforming samples. The image sample together with a time embedding is first compressed into a low-dimensional representation and then lifted back to the original dimension.

4.2 Score Approximation and Estimation

The choice of concept class \mathcal{S} is vital to the learning of the score function as in (9). There are two requirements on \mathcal{S} : 1) Class \mathcal{S} should be rich enough to well-approximate the ground truth score function, i.e., there exists a candidate in \mathcal{S} close to $\nabla \log p_t$; 2) Class \mathcal{S} should not be overly complicated to obscure the learning process with finite training samples. We present theoretical insights on both aspects, and address 1) from a function approximation perspective and 2) from a statistical learning perspective.

4.2.1 Score Approximation Guarantees

The question underscores score approximation is what the size and the architecture of score networks to ensure the existence of an ϵ -error approximation to the score function. Here $\epsilon > 0$ is the desired error level and often represents an L^2 distance measure. Such a question is reminiscent of the universal function approximation ability of neural networks [137–144]. However, we highlight some fundamental differences between score approximation and conventional function approximation. Firstly, the score function is defined on all of the high-dimensional Euclidean space, due to the added Gaussian noise, while conventional neural network approximation theory focuses on compact domains. Secondly, the score function depends on an additional time dimension, which complicates its approximation.

Concurrent works [90] and [91] tackle the challenges via very different approaches and develop score approximation theories for Euclidean data and low-dimensional linear subspace data. [90] rewrites the score function as $\nabla \log p_t = \frac{\nabla p_t}{p_t}$ and uses neural networks to approximate p_t and ∇p_t separately. To address the time dependency, [90] proposes a series of “diffused basis functions”. More formally, diffused basis functions are the convolutions of the Gaussian transition kernel in the forward process (1) with time-independent polynomials, such as Taylor polynomials and B-splines. The idea behind the diffused basis functions can be understood as tracking the evolution of p_t with respect to time t . Indeed, **once we can approximate the density of the clean data distribution P_{data} with time-independent polynomials, the corresponding diffused polynomials automatically approximate the density p_t for all t .**

On the other hand, [91] resorts to a local Taylor approximation of the score function using neural networks. In this case, the score function $\nabla \log p_t$ is viewed as a multi-dimensional input-output mapping of certain regularity. Built upon the existing universal approximation theories of neural networks, [91] devises a score approximation result. More interestingly, **[91] considers low-dimensional linear subspace data and shows that the ground truth score $\nabla \log p_t$ decomposes into two terms as in (8). In this regard, a simplified U-Net architecture (Figure 5) with linear encoder and decoder is constructed for efficient score approximation, indicating that the data subspace structures circumvent the dependence on data ambient dimension.**

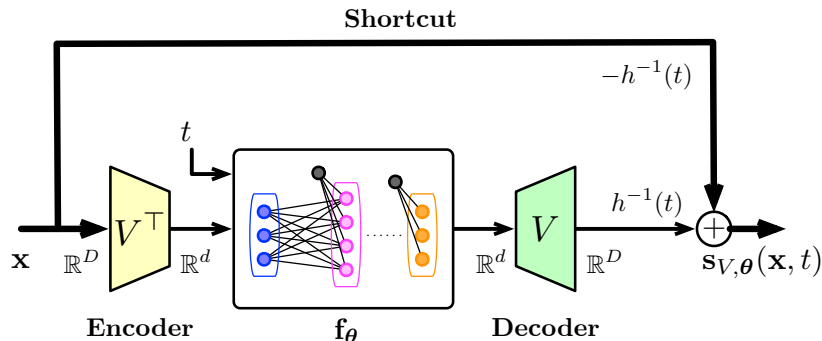


Figure 5: Simplified U-Net architecture in [91] for approximating score functions in the low-dimensional subspace data setting. Matrix V represents the linear encoder and decoder, which is to be jointly learned with parameter θ during the optimization of loss (9). Here f_θ is a network with input and output dimensions being the subspace dimension.

In deriving the approximation guarantees, [90, 91] leverage sophisticated input truncation to deal with the unbounded domain. The approximation error is in turn measured in the L^2 norm sense, instead of commonly used L^∞ norm. In order to achieve an ϵ approximation error, the

network size scales in the order $\tilde{\mathcal{O}}(\epsilon^{-\gamma})$, where γ is data dimension dependent. We emphasize that when there exists low-dimensional subspace structures in data, γ only depends on the subspace dimension.

4.2.2 Sample Complexity of Score Estimation

We shift to understand how many samples are needed to learn a score estimator by optimizing (9). The learned estimator should generalize in the sense that its deviation to the ground truth score is small. This requires more than a good score network class \mathcal{S} , but also the learnability within \mathcal{S} , which is characterized by some complexity measure of \mathcal{S} .

An early work [84] provides a sample complexity bound for score estimation. Yet the bound depends on some unknown Rademacher complexity of the score network class. Built upon the score approximation theory, [90] and [91] establish score estimation theories from the nonparametric statistics point of view. [90] assumes the clean data distribution is supported on a unit cube with a Besov continuous density. In order to obtain an ϵ -accurate score estimator in the L^2 norm, the sample size grows in the order $\tilde{\mathcal{O}}(\epsilon^{-\frac{D+2\beta}{\beta}})$, where D is the data dimension and β is the smoothness index of the density. As can be seen, the sample complexity indicates the curse of dimensionality, and [90] reduces the dependence on D when data has a known linear subspace structure. In the independent study, [91] focuses on linear subspace data without knowing the subspace in advance. Under assumptions of data having Gaussian tail and the score being Lipschitz, [91] establishes a $\tilde{\mathcal{O}}(\epsilon^{-(d+5)})$ sample complexity, where d is the subspace dimension. While free of the curse of dimensionality, [91] also proves that the unknown subspace can be automatically estimated via score estimation. Turning towards a kernel-based approach, [145, 146] establish optimal statistical score estimation rates using a regularized Gaussian kernel density estimation method. The obtained sample complexity is $\tilde{\mathcal{O}}(\epsilon^{-(d+4)})$ for Lipschitz continuous score functions.

Optimization Guarantees on Score Estimation On the algorithmic side, we are aware of [147] studying score estimation in Gaussian mixture models. They provide convergence analysis of using gradient descent to minimize the score estimation loss (9). The algorithmic behavior can be characterized in two phases, where in the large-noise phase, i.e., time t large in (9), gradient descent is analogous to power iteration. In the the small-noise phase, i.e., t small, gradient descent is akin to the EM algorithm. Besides, [148] studies the optimization guarantee of using two-layer neural networks for score estimation.

4.2.3 Score Estimation in Graphical Models

Besides considering data distributions in continuous spaces, such as Euclidean space and linear subspace, [92] studies score approximation and estimation in graphical models. Graphical models such as Markov random fields and restricted Boltzmann machines have been widely used for modeling image distributions in the literature [149, 150], yet they are fundamentally different from distributions on continuous variables. [92] proposes a novel approach for controlling the sample complexity of score estimation in high dimensions. In particular, the authors view the neural networks in diffusion models as a denoising algorithm, enabling an efficient score approximation.

Specifically, [92] assumes the data distribution follows an Ising model $P_{\text{data}}(\sigma) \propto \exp\{\langle \sigma, A\sigma \rangle\}$ for $\sigma \in \{\pm 1\}^D$. Under certain high-temperature conditions of the matrix $A \in \mathbb{R}^{D \times D}$, the score function $s(x_t, t)$ can be approximately computed by variational inference algorithms such as message passing [151–153]. This is an iterative algorithm of the form $m^{\ell+1} = \tanh((A - K)m^\ell + c_t x_t)$ with $m^0 = 0$ for certain matrix $K \in \mathbb{R}^{D \times D}$. Each step of the message-passing algorithm comprises

simple operations, including matrix-vector multiplication and pointwise nonlinearity, which could be efficiently approximated by one block of the residual network $u^{\ell+1} = u^\ell + W_1 \text{ReLU}(W_2 u^\ell)$. This renders an efficient approximation of Ising model score functions using a residual network with $\mathcal{O}(D^2L)$ parameters, where L is the number of neural network layers, allowing a moderate dependence on the problem size. Incorporating standard Rademacher complexity generalization error bound, [92] provides an estimation error bound without the exponential dependence on dimensionality.

4.3 Sampling and Distribution Estimation

Our ultimate goal of diffusion models is to learn the data distribution and provide easy access to generating new samples. This section first reviews sampling theories of diffusion models via the backward process (3), with a basic assumption on the accuracy of the estimated score function \hat{s} . Next, we move to an end-to-end analysis of diffusion models, by presenting sample complexity bounds for learning distributions.

4.3.1 Sampling Theory

Several recent sampling theories of diffusion models prove that the distribution generated by the backward process is close to the data distribution, as long as the score function is accurately estimated. The central contribution is a relationship between ϵ_{dis} and ϵ_{score} , where ϵ_{dis} is a discrepancy between the sampled data distribution and the ground truth distribution, and ϵ_{score} is the score estimation error. Specifically, [154, 155] establish upper bounds of ϵ_{dis} using ϵ_{score} for diffusion Schrödinger bridges. The error ϵ_{dis} is measured in the total variation distance and ϵ_{score} is measured in the L^∞ norm. More concrete bounds of ϵ_{dis} are provided in [84–87, 156]. These works specialize ϵ_{score} to be the L^2 error of the estimated score function, and ϵ_{dis} to be the total variation distance between the generated distribution and the data distribution. [85] requires the data distribution satisfying a log-Sobolev inequality. Concurrent works [86] and [87] relax the log-Sobolev assumption on the data distribution to only having bounded moments. The upper bound in [86] takes the form

$$\epsilon_{\text{dis}} = \tilde{\mathcal{O}} \left(\sqrt{T} \epsilon_{\text{score}} + \text{discretization-error} + \text{forward-error} \right).$$

Here T is the terminal time in the forward process. The **discretization-error** depends on the regularity of the data distribution and the step size in the discretized backward process. The **forward-error** quantifies the divergence between P_T and $P_\infty = \mathcal{N}(0, I_D)$, since the forward process is terminated as a finite time T . It is worth mentioning that [87] allows ϵ_{score} to be time-dependent and [156] improves the data dimension dependency. Recently, [88, 89, 157–159] largely enrich the study of sampling theory using diffusion models. Specifically, novel analyses based on Taylor expansions of the discretized backward process [159] or localization method [89, 156] are developed, which improve the upper bound on ϵ_{dis} . Further, [157] extends to DDIM sampling scheme and [88] considers the probabilistic ODE backward sampling.

Besides Euclidean data, [160] makes the first attempt to analyze diffusion models for learning low-dimensional manifold data. Assuming ϵ_{score} is small under the L^∞ norm (extension to the L^2 norm is also provided), [160] bound ϵ_{dis} of diffusion models in terms of the Wasserstein distance. The obtained bound has an exponential dependence on the diameter of the data manifold. Moreover, [161] considers using diffusion processes to sample from noisy observations of symmetric spiked models and [162] studies polynomial-time algorithms for sampling from Gibbs distributions based

on diffusion processes. The construction of diffusion processes in [161, 162] leverages the idea of stochastic localization [163–166].

Computational efficiency of sampling through diffusion models Sampling from certain high-dimensional distributions can be computationally challenging. For instance, [167] demonstrates the hardness of sampling from the low-temperature Sherrington-Kirkpatrick model using any stable algorithms. An intriguing line of inquiry would be to understand the computational complexity of sampling through diffusion models and its connection to the complexity of sampling via the Langevin dynamics.

Using heuristic physics methods, [168] investigated the relationship between the computational complexity of sampling through the Langevin dynamics and diffusion models in high-dimensional distributions widely studied in the statistical physics of disordered systems. They utilized the hardness of computing the score function as a proxy for the hardness of sampling with diffusion models. They generated phase diagrams of the computational complexity of sampling from these high-dimensional models, and identified parameter regions where diffusion models are unable to sample efficiently, while the Langevin dynamics can; conversely, they also identified regions where the Langevin dynamics are inefficient, yet diffusion models perform well.

4.3.2 Sample Complexity of Distribution Estimation

Distribution estimation theory of diffusion models is explored in [169] and [170] from an asymptotic statistics point of view. These results do not provide an explicit sample complexity bound. Given the aforementioned sampling theory and score estimation theory, we can develop an end-to-end analysis of diffusion models. The following theorem summarizes the existing sample complexity bounds of diffusion models in [90] and [91].

Theorem 1 (Sample Complexity of Distribution Estimation). Suppose the data distribution P_{data} is supported on a cube $[-1, 1]^D$ with a density function of smoothness index s . Under some conditions¹, diffusion models can learn a distribution \hat{P} satisfying

$$d_{\text{TV}}(\hat{P}, P_{\text{data}}) = \tilde{\mathcal{O}}\left(n^{-\frac{s}{2s+D}}\right), \quad (10)$$

where d_{TV} is the total variation distance.

Moreover, suppose the data distribution is supported on a d -dimensional subspace, i.e., data $x = Az$ with an unknown matrix $A \in \mathbb{R}^{D \times d}$ of orthonormal columns. We recall that d is the intrinsic dimension and much smaller than D . Specializing the smoothness index $s = 1$ and under some conditions, diffusion models can estimate the subspace and learn a distribution \hat{P}_{sub} in the subspace satisfying²

$$d_{\text{TV}}(\tilde{P}_{\text{data}}, \hat{P}_{\text{sub}}) = \tilde{\mathcal{O}}\left(n^{-\frac{1}{d+5}}\right), \quad (11)$$

where \tilde{P}_{data} is a slightly perturbed data distribution.

From (10), we conclude that if the density function has a higher smoothness s , the distribution estimation is more efficient. Moreover, (10) matches the minimax optimal rate of distribution estimation in Euclidean spaces, indicating diffusion models are powerful and efficient distribution

¹These are technical conditions on the data distribution approaching the boundary of the hypercube. A precise statement can be found in [90].

²The precise statement requires a set of notations. Interested readers may refer to [91, Theorem 3].

estimators. The result in (11) further unveils the adaptivity of diffusion models, since the convergence rate is only dependent on the subspace dimension d , which can be much smaller than D . This result provides valuable insights of why diffusion models yield startling practical performance, since real-world high-dimensional data often has rich low-dimensional geometric structures and diffusion models are efficient in capturing these structures for an efficient learning.

4.4 Alternative Formulation: Stochastic Localization

Stochastic localization is a measure-valued stochastic process employed to study isoperimetric inequalities [163, 171, 172]. As a mathematical technique, stochastic localization has been successfully utilized in proving versions of the Kannan-Lovász-Simonovits (KLS) conjecture [173, 174]. The process was later generalized in [161, 167, 175] as a sampling algorithm with provable sampling error bounds. The connections between stochastic localization and the DDPM (Denoising Diffusion Probabilistic Model) diffusion models are demonstrated in [164].

We introduce the simplest stochastic localization process, following the presentation in [164]. Given the measure P_{data} , the stochastic localization process is a stochastic differential equation defined as:

$$dY_t = m_t(Y_t)dt + dW_t \quad \text{for } t \in [0, \infty), \quad Y_0 = 0, \quad (12)$$

where $m_t(y) = \mathbb{E}_{(x,g) \sim P_{\text{data}} \otimes \mathbf{N}(0, I_D)}[x \mid tx + \sqrt{t}g = y]$ is the posterior expectation of Y_t upon observing $y = tx + \sqrt{t}g$. Standard theory implies that the marginal distribution of Y_t satisfies $Y_t \stackrel{d}{=} tx + \sqrt{t}g$, where $(x, g) \sim P_{\text{data}} \otimes \mathbf{N}(0, I_D)$. Consequently, $\lim_{t \rightarrow \infty} Y_t/t$ converges to a random variable following the distribution P_{data} . In generative modeling tasks, one could fit the posterior expectation $m_t(y)$ using neural networks and training samples, and discretize the SDE as in Eq. (12), similar to DDPM diffusion models.

In sampling tasks for the distribution P_{data} being spin-glasses models and posterior of spiked matrix models, [161, 167, 175] show that the posterior expectation m_t can be approximately computed using variational inference algorithms in the high-temperature regime, enabling efficient sampling from these distributions.

A firm connection between stochastic localization to DDPM diffusion models is shown in [164]: The stochastic localization process $\{Y_t\}_{t \geq 0}$ as in Eq. (12) is equivalent to the backward SDE of the diffusion model (2) up to time and scale reparametrization. [164] further generalizes the stochastic localization scheme to general stochastic processes.

5 Theoretical Progress on Conditional Diffusion Models

Although conditional diffusion models share many characteristics with their unconditional counterpart, their unique reliance on guidance requires new understanding and insights. As a result, theoretical results on conditional diffusion models are highly limited. In this section, we mimic the study of unconditional diffusion models yet put an extra emphasis on distinct uses and methods of conditional diffusion models. We first introduce the training of conditional diffusion models, which is to estimate the conditional score function. Interestingly, the conditional score function can be related to the unconditional score function, motivating a fine-tuning perspective for training conditional diffusion models. Next, we present conditional score estimation and distribution estimation guarantees. The last section is devoted to theoretical insights on the influence of guidance in Gaussian mixture models, where we corroborate common observations and reveal curious new discoveries.

5.1 Learning Conditional Score

For conditional sample generation via (5), the conditional score function $\nabla \log p_t(x|y)$ needs to be estimated. We slightly abuse the notation to denote s as a conditional score network and \mathcal{S} as the corresponding network class. By introducing an early-stopping time t_0 , a conceptual quadratic loss for conditional score estimation is defined as

$$\operatorname{argmin}_{s \in \mathcal{S}} \int_{t_0}^T w(t) \mathbb{E}_{(x_t, y)} \left[\|\nabla \log p_t(x_t|y) - s(x_t, y, t)\|_2^2 \right] dt, \quad (13)$$

where $w(t)$ is a time dependent reweighting function. Inspired by [129] and [130], [176, Proposition 3.1] asserts the equivalence of (13) to the following implementable loss function,

$$\operatorname{argmin}_{s \in \mathcal{S}} \int_{t_0}^T \mathbb{E}_{(x_0, y)} \left[\mathbb{E}_{x_t \sim \mathcal{N}(\alpha(t)x_0, h(t)I_D)} \left[\|\nabla_{x_t} \log \phi_t(x_t|x_0) - s(x_t, y, t)\|_2^2 \right] \right] dt, \quad (14)$$

which shares the same spirit as (3).

Classifier and Classifier-Free Guidance Practical implementations of conditional score estimation, such as classifier and classifier-free guidance methods, build upon (14) for reduced computational cost or better performance [100, 177]. We begin with the classifier guidance method [177], which is arguably the first method to allow conditional generation in diffusion models similar to GANs or flow models [178, 179]. Specifically, when conditional information y is discrete, e.g., image categories, the conditional score $\nabla \log p_t(x_t|y)$ is rewritten via Bayes' rule as

$$\nabla \log p_t(x_t|y) = \nabla \log p_t(x_t) + \nabla \log c_t(y|x_t),$$

where c_t is the likelihood function of an external classifier. In other words, classifier guidance combines the unconditional score function with the gradient of an external classifier. The external classifier is trained using the diffused data points in the forward process. As a result, the performance of classifier guidance methods is sometimes limited, since it is difficult to train the external classifier with highly corrupted data.

Later, classifier-free guidance proposes to remove the external classifier, circumventing the limitation caused by classifier training. The idea of classifier-free guidance is to introduce a mask signal to randomly ignore y and unifies the learning of conditional and unconditional scores. Specifically, let $\tau \in \{\emptyset, \text{id}\}$ be a mask signal, where \emptyset means to ignore the conditional information y and id to keep y . Corresponding to the two circumstances, we have

$$\begin{aligned} \tau = \emptyset : & \int_{t_0}^T \mathbb{E}_{(x_0, y)} \left[\mathbb{E}_{x_t \sim \mathcal{N}(\alpha(t)x_0, h(t)I_D)} \left[\|s(x_t, \emptyset, t) - \nabla_{x_t} \log \phi_t(x_t|x_0)\|_2^2 \right] \right] dt \\ \tau = \text{id} : & \int_{t_0}^T \mathbb{E}_{(x_0, y)} \left[\mathbb{E}_{x_t \sim \mathcal{N}(\alpha(t)x_0, h(t)I_D)} \left[\|s(x_t, y, t) - \nabla_{x_t} \log \phi_t(x_t|x_0)\|_2^2 \right] \right] dt. \end{aligned} \quad (15)$$

Note that (15) coincides with (14), and recall that t_0 is an early-stopping time. Combining the two cases, classifier-free guidance method minimizes the following loss function:

$$\hat{s} \in \operatorname{argmin}_{s \in \mathcal{S}} \int_{t_0}^T \mathbb{E}_{(x_0, y)} \left[\mathbb{E}_{\tau \sim P_\tau, x_t \sim \mathcal{N}(\alpha(t)x_0, h(t)I_D)} \left[\|s(x_t, \tau y, t) - \nabla_{x_t} \log \phi_t(x_t|x_0)\|_2^2 \right] \right] dt. \quad (16)$$

Here τ is randomly chosen among \emptyset and id following distribution P_τ . The simplistic choice on P_τ is a uniform distribution on $\{\emptyset, \text{id}\}$, while it is preferred to bias towards $\tau = \text{id}$ in some applications [100].

Once the estimator \widehat{s} is learned from (16), we compute

$$\widetilde{s}(x, y, t) = (1 + \eta) \cdot \widehat{s}(x, y, t) - \eta \cdot \widehat{s}(x, \emptyset, t) \quad (17)$$

with some $\eta > 0$ for substitution into the backward process. From a theoretical point of view, choosing $\eta > 0$ is counter-intuitive, as the resulting \widetilde{s} does not correspond to the conditional score function $\nabla \log p_t(x|y)$. However, a properly chosen η leads to improved performance on benchmarks in practice. More interestingly, increasing η reduces the diversity of the generated samples but promotes distinguishability of them [100]. The coefficient η can also be chosen dependent on time t . Unfortunately, a principled guidance on how to choose η is still missing, yet some theoretical insights on the impact of η are developed [180]; see also Section 5.3.

Adapting Unconditional Score via Guidance In real use cases, the desired criteria or objectives of conditional sample generation may shift over time, which necessitates quick adaptation of conditional diffusion models. Although classifier-free guidance method has been adopted for training conditional diffusion models from scratch, it is not tailored for adapting or fine-tuning diffusion models owing to the computational overhead. Consequently, this opens up new possibilities of theories and methods for fine-tuning diffusion models without compromising the pre-training performance.

Recently, [75, 78, 79, 181, 182] propose efficient fine-tuning methods when the quality of generated samples is measured by a scalar-valued reward function. To guide a pre-trained model for generating high-reward samples, [75] assumes the differentiability of the reward function and directly fine-tunes parameters in the diffusion model by back-propagation. [78, 79] formulate the sample generation process of diffusion models as a finite-horizon Markov decision process. The score function is equivalent to a policy and allows for fine-tuning using reinforcement learning techniques, such as policy gradient methods.

A more interesting and principled fine-tuning method draws motivation from the classifier guidance. We revisit the Bayes’ rule for conditional score function,

$$\nabla \log p_t(x_t|y) = \underbrace{\nabla \log p_t(x_t)}_{\text{pre-trained score}} + \underbrace{\nabla \log c_t(y|x_t)}_{\text{guidance}},$$

where the classifier c_t acts as guidance to adapting the pre-trained score. Despite classifier guidance requires a discrete label y (yet can be multi-dimensional), the decomposition in the last display has a profound impact on guidance-based fine-tuning. Indeed, [19, 117, 183] extend guidance to arbitrary conditioning by incorporating gradients of a proper scalar-valued function. For demonstration, [19] defines the so-call “universal guidance” in the form of $\nabla_{x_t} \ell(y, f(\widehat{x}_0))$, where f is a function measuring the quality of samples, \widehat{x}_0 is the anticipated generated sample of the pre-trained diffusion model given current point x_t in the backward process, and ℓ is a loss function. Note that \widehat{x}_0 correlates with x_t and the gradient is nontrivial. As a special example, when y is the discrete label, f is the classification likelihood, and ℓ is the cross-entropy loss, universal guidance reproduces classifier guidance.

5.2 Conditional Score and Distribution Estimation

Theory of conditional score estimation and conditional distribution estimation is very limited. To the best of our knowledge, [176] provides an initial study of using (14) for conditional score estimation and distribution estimation. A systematic analysis of classifier-free guidance method is presented in [184], with results highlighted by approximation theories of conditional score functions

and sample complexities of conditional score estimation and distribution learning. In addition, [184] shows the utility of the developed statistical theory in elucidating the performance of conditional diffusion models for diverse applications, including model-based transition kernel estimation in reinforcement learning, solving inverse problems [103, 185–187], and reward conditioned sample generation.

The core contribution of [184] is the conditional score approximation theory, which is motivated by the idea of diffused basis approximation in [90]. In more detail, [184] substantially broadens the framework to unbounded data domains and conditional distributions. The authors rewrite the conditional score function as $\nabla \log p_t(x|y) = \frac{\nabla p_t(x|y)}{p_t(x|y)}$ and approximate $\nabla p_t(x|y)$ and $p_t(x|y)$ separately. On a technical side, the unbounded data domain and the conditioning on y lead to new challenges. More importantly, however, [184] lifts the technical conditions on data distributions in [90] and obtains optimal statistical rates with a mild bounded Hölder norm assumption. We remark that [184] takes the condition y as independent input variables, leaving an open direction to identify intrinsic smoothness with respect to y in the conditional distribution so as to improve the dimension dependency.

5.3 Theoretical Insights on Strength of Guidance

We conclude the discussion on conditional diffusion models by a recent work on the influence of the strength of guidance [180]. We are referring back to (17) and studying the influence of η on the sample generation. The same strength parameter can be introduced into classifier guidance as

$$\tilde{s}(x, y, t) = \nabla \log p_t(x_t) + \eta \nabla \log c_t(y|x_t).$$

Hence, we will not distinguish different guidance methods, and term η as the strength of guidance.

A common observation of the consequence yielded by η is best illustrated in Figure 6 on a three-component Gaussian mixture model. Here label y indicates the Gaussian components and x is a two-dimensional variable. When generating new samples, we fix a choice on y to obtain within component samples. We observe that with an increased guidance strength η , the generated conditional distribution shifts its probability mass farther away from other components, and most of the mass becomes concentrated in smaller regions.

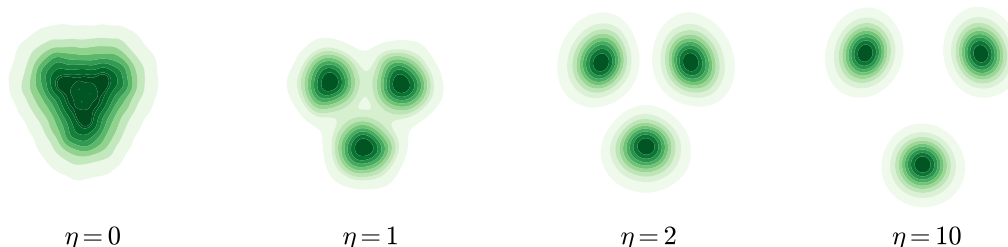


Figure 6: The effect of guidance strength η on a three-component GMM in \mathbb{R}^2 [100, 180]. Each component has weight $1/3$ and identity covariance, and the component centers are $(\sqrt{3}/2, 1/2)$, $(-\sqrt{3}/2, 1/2)$ and $(0, -1)$. The leftmost panel displays the unguided density. We increase the guidance strength from left to right. When generating samples, we use the ground truth score function.

The results in [180] theoretically characterize the influence of strength on diffusion models in the context of Gaussian mixture models. Under mild conditions, [180] proves that incorporating

不合适的指导强度 (η过大时) ① 增大与类间分离
② 降低分布多样性, 生成条件分布的微小扰动降低

③ 随着 η 增加, 采样数据会越来越稀, 高斯混合模型的中心量会对称分裂, 损害原始数据模态。

strong guidance not only boosts classification confidence but also diminishes distribution diversity, leading to a reduction in the differential entropy of the generated conditional distribution. These theories align closely with empirical observations.

On the other hand, [180] identifies a possible negative impact of large η under discretized backward sampling in Gaussian mixture models, as depicted in Figure 7. There exists a phase shift as strength η increases. Under large η , the center component of the original Gaussian mixture model splits into two symmetric clusters, harming the modality of the original data. The emergence of this negative effect is tied to the locations of the components and the discretization step size in the backward sampling process. Until this point, we are not aware of principled methods for tuning the strength η in different tasks, which might be encouraged by the obtained theoretical insights.

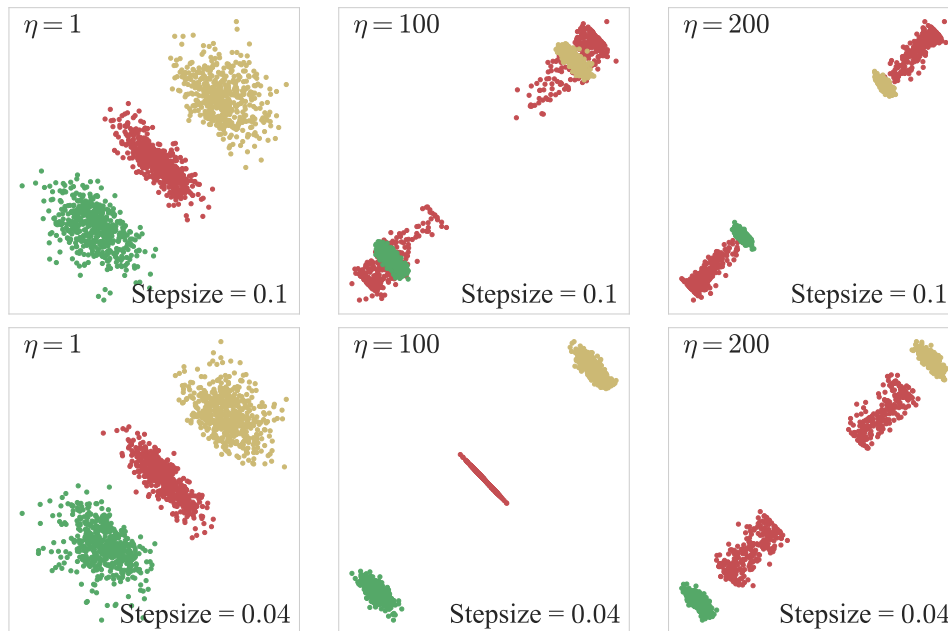


Figure 7: Illustration of a possible negative effect of large guidance strength in [180]. In this plot, the component means of the Gaussian mixture model are aligned on the same line. We increase the guidance strength η from left to right. The upper row uses a relatively large discretization step size in the backward process. With large η , the center component splits into two clusters at an earlier stage. The bottom row uses a much smaller discretization step size; the center component then splits only with a much larger η .

6 Diffusion Model for Optimization

This section introduces a novel avenue for optimization in high-dimensional complex and structured spaces through diffusion models. We focus on data-driven black-box optimization, where the goal is to generate new solutions that optimize an unknown objective function. Black-box optimization, also known as model-based optimization in machine learning, encapsulates various application domains such as reinforcement learning, computational biology and business management [54, 57, 118, 188–192].

Solving data-driven black-box optimization differentiates from conventional optimization, as interactions with the objective function beyond a pre-collected data set are prohibitive, diminishing

the possibility of sequentially searching for optimal solutions. Instead, people aim to extract pertinent information from the pre-collected data set and directly recommend solutions. To complicate matters, the solution space is often high-dimensional with rich latent structures. For example, in drug discovery, molecule structures need to satisfy global and local regularity to be expressive in living bodies. This poses a critical requirement for solving data-driven black-box optimization: we need to capture data latent structures to avoid suggesting unrealistic solutions that deviate severely from the original data domain.

To address the challenges, [176] formulates data-driven black-box optimization as sampling from a conditional distribution, as demonstrated in Figure 8. The objective function value is the conditioning in the conditional distribution, meanwhile the distribution implicitly captures data latent structures.

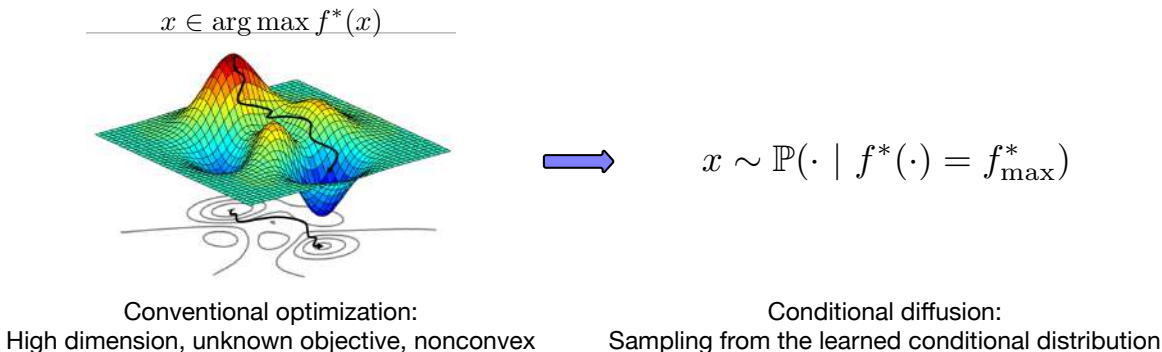


Figure 8: Reformulation of data-driven black-box optimization as conditional sampling in [176]. The conditional distribution is learned from a pre-collected data set.

The pre-collected data set in [176] consists of two parts: 1) a massive unlabeled part $\mathcal{D}_{\text{unlabel}}$ and 2) a smaller labeled part $\mathcal{D}_{\text{label}}$. By terming the objective function as a reward function, [176] considers two types of label feedback in $\mathcal{D}_{\text{label}}$:

- **(Real-valued reward)** The data set $\mathcal{D}_{\text{label}}$ consists of data and reward pairs, where the reward is a real-valued noise-perturbed version of the underlying ground truth reward;
- **(Human preference)** The data set $\mathcal{D}_{\text{label}}$ consists of triples taking two comparable data points and a binary preference label. The preference label indicates that the corresponding data point is likely to have an edge in the underlying reward over the other one.

Moreover, the data point $x \in \mathbb{R}^D$ is assumed to concentrate on a linear subspace, i.e., $x = Az$ for some unknown matrix $A \in \mathbb{R}^{D \times d}$, with $z \in \mathbb{R}^d$ being the latent variable. Therefore, newly generated samples should be kept close to the subspace to maintain high fidelity.

A semi-supervised learning algorithm is proposed in Figure 9. There are two training procedures: one in the first step for estimating the reward function and the other one in the third step for training the conditional diffusion model. In the fourth step, the target reward is set at a scalar value a , so that the generated samples follow the conditional distribution $\widehat{P}_a = \widehat{P}(\cdot \mid \widehat{\text{reward}} = a)$, where \widehat{P} and $\widehat{\text{reward}}$ emphasize that the distribution and the reward are estimated, rather than the ground truth. One may be curious about the quality of the generated samples. In particular, these two properties of the generated samples are of particular interest: 1) the reward levels of new samples and 2) their level of fidelity – how much do new samples deviate from the latent subspace.

The results in [176] provide a positive statistical answer. For the reward levels of new samples, [176] defines

$$\text{SubOpt}(a) = a - \mathbb{E}_{x \sim \hat{P}_a} [V(x)]$$

to measure the gap between the sample average reward and the target reward. In the language of bandit learning, **SubOpt** can be interpreted as a form of off-policy sub-optimality. The following theorem proves an upper bound on **SubOpt**.

Theorem 2 (Conditional Diffusion for Black-Box Optimization). Consider data $x = Az$ for some unknown matrix $A \in \mathbb{R}^{D \times d}$ with orthonormal columns. Suppose the reward function V decomposes into

$$V(x) = \underbrace{g(AA^\top x)}_{\geq 0, \text{ on-support reward}} + \underbrace{h((I - AA^\top)x)}_{\leq 0, \text{ off-support penalty}} .$$

Running the algorithm in Figure 9 generates high-fidelity samples and gives rise to

$$\text{SubOpt}(a) \leq \underbrace{\mathbb{E}_{P_a} [|g - \hat{g}|]}_{\text{reward estimation error}} + \underbrace{|\mathbb{E}_{P_a} [g] - \mathbb{E}_{\hat{P}_a} [g]|}_{\text{on-support diffusion error}} + \underbrace{|\mathbb{E}_{\hat{P}_a} [h]|}_{\text{off-support penalty}} ,$$

where \hat{g} is an estimated reward function and $P_a = P(\cdot \mid \widehat{\text{reward}} = a)$.

We note that the reward function V consists of two components: 1) The on-support reward g is nonnegative and measures the quality of samples by projecting it onto the subspace spanned by matrix A ; 2) The off-support penalty, however, is nonpositive and discourages the generated samples extrapolating in the space outside the subspace spanned by matrix A .

Theorem 2 says that the reward estimation error depends on the sample size in $\mathcal{D}_{\text{label}}$, which is often the dominating term. The on-support diffusion error and off-support penalty depend on the sample size in $\mathcal{D}_{\text{unlabel}}$ and rely on a statistical analysis of conditional diffusion models for distribution estimation. There is also a subtlety to explicitly quantify the three error terms, namely, the distribution shift, which is the mismatch between the training data distribution and the target data distribution. Diffusion models are learned to generate similar samples to the training distribution, however, optimizing the reward function drives the model to deviate from the training. In other words, the model needs to both “interpolate” and “extrapolate”. A higher value of a provides stronger guidance to the diffusion model, while the increasing distribution shift may hurt the generated samples’ quality.

Through detailed analysis, [176] instantiates Theorem 2 to parametric and nonparametric settings. For example, with a linear reward function g , the reward estimation error aligns with the optimal off-policy bandit sub-optimality [193, 194], where the distribution shift is explicitly computed and the dimension dependence is d instead of large ambient dimension D . In the human preference setting, [176] considers the Bradley-Terry-Luce choice model [195, 196] and derives a similar concrete sub-optimality bound.

7 Future Directions

We discuss several future directions of diffusion models, exploring their connections to stochastic control and distributional robustness; we also introduce discrete diffusion models.

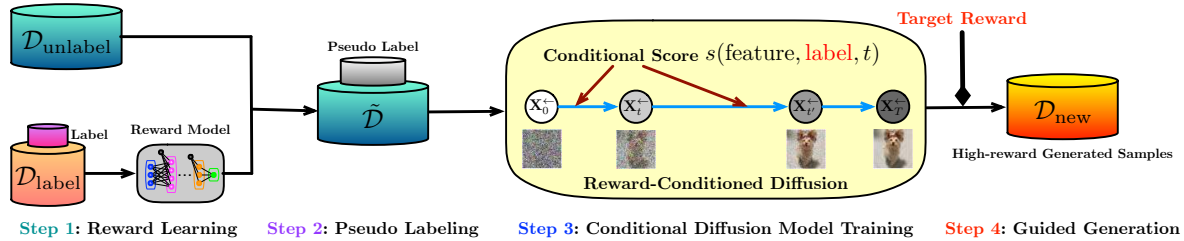


Figure 9: The learning algorithm proposed in [176] consists of four steps. In the first step, a reward model is learned from the labeled data $\mathcal{D}_{\text{label}}$. In the second step, the learned reward model is deployed as a pseudo-labeler to label $\mathcal{D}_{\text{unlabel}}$. In the third step, a conditional diffusion model is trained using the pseudo-labeled data. Lastly, in the fourth step, new samples are generated from the conditional distribution P_a by specifying a target reward value a .

7.1 Connection to Stochastic Control

In either unconditioned diffusion models or conditional diffusion models, generating samples using backward processes (2) or (5) can be viewed as a stochastic control problem [197, 198]. The goal of stochastic control is to design the evolution of the controlled variable, so that certain cost is minimized. In diffusion models, the score function constitutes the control and steers the quality of the generated samples. In the simplest form of unconditioned diffusion models, we define the cost to be the distribution divergence between the generated distribution and the data distribution, such as the total variation distance and the Wasserstein distance. Then the score estimation essentially amounts to finding the optimal control for minimizing such costs.

When using conditional diffusion models for black-box optimization, the cost is the negative of a reward function and the conditional score function is the control. The theory in [176] chooses a proper target reward to design the control for optimizing the cost. Leveraging this control perspective, a series of empirical results attempt to fine-tune diffusion models by designing the control based on various cost forms [75–83, 199]. For instance, [75, 78, 79] consider differentiable real-valued reward, while [77, 80] focus on the cost being human preferences. In terms of methodology, [78] uses the policy gradient method in reinforcement learning for fine-tuning the control (conditional score function). [75] resorts to the classifier guidance formula by directly augmenting the unconditioned score function by gradients of the cost.

In this regard, principled methodologies and accompanying theories [181–183] can be motivated from the stochastic control perspective, improving and analyzing diffusion models under various task objectives.

7.2 Adversarial Robustness and Distributionally Robust Optimization

Diffusion models exhibit the natural denoising property in the backward processes, which are leveraged for adversarial purification and promoting robustness [200–203]. To illustrate, in robust classification, a two-step classification procedure is proposed: A trained conditional diffusion model is first deployed to generate new samples given the input adversarial examples for multiple times, hoping to purify the added noise in the input sample. Then the generated samples are fed into a trained classifier to produce a predicted label. Due to the randomness in the diffusion models, multiple transformed samples of the same input adversarial example can be obtained. Therefore, a majority vote among the predicted labels is assigned as the label of the adversarial example. This method is motivated by a justification on the promotion of robustness using diffusion models and

empirically shown to be effective [203]. Yet an end-to-end analysis is still missing.

We also expect a close connection between diffusion models to Distributionally Robust Optimization (DRO) [204–207]. As shown in Theorem 1, diffusion models generate samples in the close vicinity of a target distribution, which can be viewed as providing a certain coverage of the distributional uncertainty set in DRO. In this sense, diffusion models can potentially simulate the worst-case scenario in the uncertainty set. We suspect the emergence of innovative methods and theories in the corresponding intersection area, where motivating attempts have been made in [208].

7.3 Discrete Diffusion Models

Discrete diffusion models, analogous to the previous continuous counterparts, are designed to keep the finite data support during the forward and backward processes [60, 61, 209–214]. Instead of using continuous Gaussian noise to corrupt clean data, discrete diffusion resorts to continuous-time Markov processes for transforming clean data. The discrete nature has appealing alignment to real data characterized by a massive but finite support, e.g., natural language represented by word tokens and molecular structures. As reported in [213], discrete diffusion achieves competitive or better performance in language tasks with comparable sized models. [215] demonstrates the possibility of using discrete diffusion for solving combinatorial problems.

We describe a discrete distribution by a probability vector p_{data} belonging to the simplex. Analogous to Gaussian noise corruption for continuous diffusion, we utilize a continuous-time Markov process driven by a time-dependent transition matrix Q_t , i.e.,

$$\frac{dp_t}{dt} = Q_t p_t \quad \text{with} \quad p_0 = p_{\text{data}}. \quad (18)$$

The process above is known as the forward discrete diffusion process. Several design choices of Q_t are summarized in [61], including discretized Gaussian, uniform, and absorbing transitions.

The discrete forward process (18) also asserts a time reversal:

$$\frac{dp_t^{\leftarrow}}{dt} = \bar{Q}_t p_t^{\leftarrow} \quad \text{with} \quad [\bar{Q}_t]_{ij} = \begin{cases} \frac{[p_{T-t}]_i}{[p_{T-t}]_j} [Q_{T-t}]_{ji} & \text{if } i \neq j \\ -\sum_{s \neq i} [Q_{T-t}]_{is} \frac{[p_{T-t}]_s}{[p_{T-t}]_i} & \text{if } i = j \end{cases}. \quad (19)$$

Here \bar{Q}_t is the backward transition matrix and $[\cdot]_i$ (or $[\cdot]_{ij}$) denotes the i -th (or (i, j) -th) entry. We observe from the backward process (19) that to generate new samples, we only need to estimate the ratios $\frac{[p_t]_i}{[p_t]_j}$ for $t \in [0, T]$. We can view this probability ratio as an analogy to the score function in the continuous distribution. Some caveats arise that estimating the ratios suffers from the massive support size of the data distribution and the magnitude of ratios can vary significantly. It is also likely that a large fraction of the ratios are zero or approximately zero, inducing sparse structures. There are different empirical methods for estimating the ratios [61, 209, 210, 213, 214], such as using a quadratic loss or an entropy loss.

From a theoretical stand of point, discrete diffusion poses interesting open questions: How to efficiently estimate the ratios using finite samples, with potential sparse structures and ill-spread ranges of ratios. More importantly, how to smartly design principled transition kernels relevant to data distributions remains unclear. Nonetheless, assuming access to estimated ratios, [216] proves the first sampling theory of discrete diffusion models.

8 Conclusion

In this paper, we have surveyed how diffusion models generate samples, their wide applications, and existing theoretical underpinnings of them. We have adopted a continuous-time description of the forward and backward processes in diffusion models and discussed their training procedure, especially when there exists guidance to steer the sample generation. We have started with an exposure to theories of unconditional diffusion models, covering its score approximation, statistical estimation, and sampling theories. Built upon the insights from the unconditional diffusion models, we have then turned towards conditional diffusion models, with a focus on their unique design properties and theories. Next, we have made a connection between generative diffusion models to black-box optimization, paving a new avenue for high-dimensional optimization problems. Lastly, we have discussed several trending future directions.

References

- [1] R. Bommasani, D. A. Hudson, E. Adeli, R. Altman, S. Arora, S. von Arx, M. S. Bernstein, J. Bohg, A. Bosselut, E. Brunskill *et al.*, “On the opportunities and risks of foundation models,” *arXiv preprint arXiv:2108.07258*, 2021.
- [2] J. Sohl-Dickstein, E. Weiss, N. Maheswaranathan, and S. Ganguli, “Deep unsupervised learning using nonequilibrium thermodynamics,” in *Proceedings of the International conference on machine learning*. PMLR, 2015, pp. 2256–2265.
- [3] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, “Generative adversarial networks,” *Communications of the ACM*, vol. 63, no. 11, pp. 139–144, 2020.
- [4] A. Creswell, T. White, V. Dumoulin, K. Arulkumaran, B. Sengupta, and A. A. Bharath, “Generative adversarial networks: An overview,” *IEEE signal processing magazine*, vol. 35, no. 1, pp. 53–65, 2018.
- [5] D. P. Kingma and M. Welling, “Auto-encoding variational bayes,” *arXiv preprint arXiv:1312.6114*, 2013.
- [6] D. P. Kingma, M. Welling *et al.*, “An introduction to variational autoencoders,” *Foundations and Trends® in Machine Learning*, vol. 12, no. 4, pp. 307–392, 2019.
- [7] Y. Song and S. Ermon, “Generative modeling by estimating gradients of the data distribution,” *Advances in Neural Information Processing Systems*, vol. 32, 2019.
- [8] S. Dathathri, A. Madotto, J. Lan, J. Hung, E. Frank, P. Molino, J. Yosinski, and R. Liu, “Plug and play language models: A simple approach to controlled text generation,” *arXiv preprint arXiv:1912.02164*, 2019.
- [9] J. Ho, A. Jain, and P. Abbeel, “Denoising diffusion probabilistic models,” *Advances in Neural Information Processing Systems*, vol. 33, pp. 6840–6851, 2020.
- [10] Y. Song, J. Sohl-Dickstein, D. P. Kingma, A. Kumar, S. Ermon, and B. Poole, “Score-based generative modeling through stochastic differential equations,” *arXiv preprint arXiv:2011.13456*, 2020.
- [11] Z. Kong, W. Ping, J. Huang, K. Zhao, and B. Catanzaro, “Diffwave: A versatile diffusion model for audio synthesis,” *arXiv preprint arXiv:2009.09761*, 2020.
- [12] N. Chen, Y. Zhang, H. Zen, R. J. Weiss, M. Norouzi, and W. Chan, “Wavegrad: Estimating gradients for waveform generation,” *arXiv preprint arXiv:2009.00713*, 2020.
- [13] G. Mittal, J. Engel, C. Hawthorne, and I. Simon, “Symbolic music generation with diffusion models,” *arXiv preprint arXiv:2103.16091*, 2021.
- [14] R. Huang, Z. Zhao, H. Liu, J. Liu, C. Cui, and Y. Ren, “Prodiff: Progressive fast diffusion model for high-quality text-to-speech,” in *Proceedings of the 30th ACM International Conference on Multimedia*, 2022, pp. 2595–2605.
- [15] M. Jeong, H. Kim, S. J. Cheon, B. J. Choi, and N. S. Kim, “Diff-tts: A denoising diffusion model for text-to-speech,” *arXiv preprint arXiv:2104.01409*, 2021.

- [16] A. Ulhaq, N. Akhtar, and G. Pogrebna, “Efficient diffusion models for vision: A survey,” *arXiv preprint arXiv:2210.09292*, 2022.
- [17] O. Avrahami, D. Lischinski, and O. Fried, “Blended diffusion for text-driven editing of natural images,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 18 208–18 218.
- [18] G. Kim, T. Kwon, and J. C. Ye, “Diffusionclip: Text-guided diffusion models for robust image manipulation,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 2426–2435.
- [19] A. Bansal, H.-M. Chu, A. Schwarzschild, S. Sengupta, M. Goldblum, J. Geiping, and T. Goldstein, “Universal guidance for diffusion models,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 843–852.
- [20] C. Saharia, W. Chan, S. Saxena, L. Li, J. Whang, E. L. Denton, K. Ghasemipour, R. Gontijo Lopes, B. Karagol Ayan, T. Salimans *et al.*, “Photorealistic text-to-image diffusion models with deep language understanding,” *Advances in Neural Information Processing Systems*, vol. 35, pp. 36 479–36 494, 2022.
- [21] R. Po, W. Yifan, V. Golyanik, K. Aberman, J. T. Barron, A. H. Bermano, E. R. Chan, T. Dekel, A. Holynski, A. Kanazawa *et al.*, “State of the art on diffusion models for visual computing,” *arXiv preprint arXiv:2310.07204*, 2023.
- [22] C. Zhang, C. Zhang, S. Zheng, M. Zhang, M. Qamar, S.-H. Bae, and I. S. Kweon, “A survey on audio diffusion models: Text to speech synthesis and enhancement in generative ai,” *arXiv preprint arXiv:2303.13336*, vol. 2, 2023.
- [23] X. Li, J. Thickstun, I. Gulrajani, P. S. Liang, and T. B. Hashimoto, “Diffusion-lm improves controllable text generation,” *Advances in Neural Information Processing Systems*, vol. 35, pp. 4328–4343, 2022.
- [24] P. Yu, S. Xie, X. Ma, B. Jia, B. Pang, R. Gao, Y. Zhu, S.-C. Zhu, and Y. N. Wu, “Latent diffusion energy-based model for interpretable text modeling,” *arXiv preprint arXiv:2206.05895*, 2022.
- [25] J. Lovelace, V. Kishore, C. Wan, E. Shekhtman, and K. Weinberger, “Latent diffusion for language generation,” *arXiv preprint arXiv:2212.09462*, 2022.
- [26] J. M. L. Alcaraz and N. Strodthoff, “Diffusion-based time series imputation and forecasting with structured state space models,” *arXiv preprint arXiv:2208.09399*, 2022.
- [27] Y. Tashiro, J. Song, Y. Song, and S. Ermon, “CSDI: Conditional score-based diffusion models for probabilistic time series imputation,” *Advances in Neural Information Processing Systems*, vol. 34, pp. 24 804–24 816, 2021.
- [28] G. Tevet, S. Raab, B. Gordon, Y. Shafir, D. Cohen-Or, and A. H. Bermano, “Human motion diffusion model,” *arXiv preprint arXiv:2209.14916*, 2022.
- [29] M. Tian, B. Chen, A. Guo, S. Jiang, and A. R. Zhang, “Fast and reliable generation of ehr time series via diffusion models,” *arXiv preprint arXiv:2310.15290*, 2023.

- [30] T. Pearce, T. Rashid, A. Kanervisto, D. Bignell, M. Sun, R. Georgescu, S. V. Macua, S. Z. Tan, I. Momennejad, K. Hofmann, and S. Devlin, “Imitating human behaviour with diffusion models,” *arXiv preprint arXiv:2301.10677*, 2023.
- [31] C. Chi, S. Feng, Y. Du, Z. Xu, E. Cousineau, B. Burchfiel, and S. Song, “Diffusion Policy: Visuomotor policy learning via action diffusion,” *arXiv preprint arXiv:2303.04137*, 2023.
- [32] P. Hansen-Estruch, I. Kostrikov, M. Janner, J. G. Kuba, and S. Levine, “IDQL: Implicit Q-learning as an actor-critic method with diffusion policies,” *arXiv preprint arXiv:2304.10573*, 2023.
- [33] M. Reuss, M. Li, X. Jia, and R. Lioutikov, “Goal-conditioned imitation learning using score-based diffusion policies,” *arXiv preprint arXiv:2304.02532*, 2023.
- [34] Z. Zhu, H. Zhao, H. He, Y. Zhong, S. Zhang, Y. Yu, and W. Zhang, “Diffusion models for reinforcement learning: A survey,” *arXiv preprint arXiv:2311.01223*, 2023.
- [35] Z. Ding and C. Jin, “Consistency models as a rich and efficient policy class for reinforcement learning,” *arXiv preprint arXiv:2309.16984*, 2023.
- [36] C. Cao, Z.-X. Cui, S. Liu, D. Liang, and Y. Zhu, “High-frequency space diffusion models for accelerated mri,” *arXiv preprint arXiv:2208.05481*, 2022.
- [37] H. Chung, E. S. Lee, and J. C. Ye, “MR image denoising and super-resolution using regularized reverse diffusion,” *IEEE Transactions on Medical Imaging*, vol. 42, no. 4, pp. 922–934, 2022.
- [38] H. Chung and J. C. Ye, “Score-based diffusion models for accelerated MRI,” *Medical Image Analysis*, vol. 80, p. 102479, 2022.
- [39] A. Güngör, S. U. Dar, Ş. Öztürk, Y. Korkmaz, H. A. Bedel, G. Elmas, M. Ozbey, and T. Çukur, “Adaptive diffusion priors for accelerated MRI reconstruction,” *Medical Image Analysis*, p. 102872, 2023.
- [40] B. Jing, G. Corso, J. Chang, R. Barzilay, and T. Jaakkola, “Torsional diffusion for molecular conformer generation,” *Advances in Neural Information Processing Systems*, vol. 35, pp. 24 240–24 253, 2022.
- [41] N. Anand and T. Achim, “Protein structure and sequence generation with equivariant denoising diffusion probabilistic models,” *arXiv preprint arXiv:2205.15019*, 2022.
- [42] J. S. Lee, J. Kim, and P. M. Kim, “Proteinsgm: Score-based generative modeling for de novo protein design,” *bioRxiv*, pp. 2022–07, 2022.
- [43] S. Luo, Y. Su, X. Peng, S. Wang, J. Peng, and J. Ma, “Antigen-specific antibody design and optimization with diffusion-based generative models for protein structures,” *Advances in Neural Information Processing Systems*, vol. 35, pp. 9754–9767, 2022.
- [44] S. Mei, F. Fan, and A. Maier, “Metal inpainting in CBCT projections using score-based generative model,” *arXiv preprint arXiv:2209.09733*, 2022.
- [45] D. J. Waibel, E. Röoell, B. Rieck, R. Giryes, and C. Marr, “A diffusion model predicts 3d shapes from 2d microscopy images,” *arXiv preprint arXiv:2208.14125*, 2022.

- [46] J. Ingraham, M. Baranov, Z. Costello, V. Frappier, A. Ismail, S. Tie, W. Wang, V. Xue, F. Obermeyer, A. Beam *et al.*, “Illuminating protein space with a programmable generative model,” *BioRxiv*, pp. 2022–12, 2022.
- [47] Y. Huang, X. Peng, J. Ma, and M. Zhang, “3DLinker: an E (3) equivariant variational autoencoder for molecular linker design,” *arXiv preprint arXiv:2205.07309*, 2022.
- [48] A. Schneuing, Y. Du, C. Harris, A. Jamasb, I. Igashov, W. Du, T. Blundell, P. Lió, C. Gomes, M. Welling, M. Bronstein, and B. Correia, “Structure-based drug design with equivariant diffusion models,” *arXiv preprint arXiv:2210.13695*, 2022.
- [49] L. Wu, C. Gong, X. Liu, M. Ye, and Q. Liu, “Diffusion-based molecule generation with informative prior bridges,” *Advances in Neural Information Processing Systems*, vol. 35, pp. 36 533–36 545, 2022.
- [50] N. Gruver, S. Stanton, N. C. Frey, T. G. Rudner, I. Hotzel, J. Lafrance-Vanasse, A. Rajpal, K. Cho, and A. G. Wilson, “Protein design with guided discrete diffusion,” *arXiv preprint arXiv:2305.20009*, 2023.
- [51] T. Weiss, E. Mayo Yanes, S. Chakraborty, L. Cosmo, A. M. Bronstein, and R. Gershoni-Poranne, “Guided diffusion for inverse molecular design,” *Nature Computational Science*, pp. 1–10, 2023.
- [52] M. Xu, L. Yu, Y. Song, C. Shi, S. Ermon, and J. Tang, “Geodiff: A geometric diffusion model for molecular conformation generation,” *arXiv preprint arXiv:2203.02923*, 2022.
- [53] Y. Song, L. Shen, L. Xing, and S. Ermon, “Solving inverse problems in medical imaging with score-based generative models,” *arXiv preprint arXiv:2111.08005*, 2021.
- [54] J. L. Watson, D. Juergens, N. R. Bennett, B. L. Trippe, J. Yim, H. E. Eisenach, W. Ahern, A. J. Borst, R. J. Ragotte, and L. F. Milles, “De novo design of protein structure and function with rfdiffusion,” *Nature*, vol. 620, no. 7976, pp. 1089–1100, 2023.
- [55] L. Yang, Z. Zhang, Y. Song, S. Hong, R. Xu, Y. Zhao, Y. Shao, W. Zhang, B. Cui, and M.-H. Yang, “Diffusion models: A comprehensive survey of methods and applications,” *arXiv preprint arXiv:2209.00796*, 2022.
- [56] X. Li, Y. Ren, X. Jin, C. Lan, X. Wang, W. Zeng, X. Wang, and Z. Chen, “Diffusion models for image restoration and enhancement—a comprehensive survey,” *arXiv preprint arXiv:2308.09388*, 2023.
- [57] Z. Guo, J. Liu, Y. Wang, M. Chen, D. Wang, D. Xu, and J. Cheng, “Diffusion models in bioinformatics: A new wave of deep learning revolution in action,” *arXiv preprint arXiv:2302.10907*, 2023.
- [58] H. Cao, C. Tan, Z. Gao, Y. Xu, G. Chen, P.-A. Heng, and S. Z. Li, “A survey on generative diffusion model,” *arXiv preprint arXiv:2209.02646*, 2022.
- [59] F.-A. Croitoru, V. Hondru, R. T. Ionescu, and M. Shah, “Diffusion models in vision: A survey,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023.
- [60] E. Hoogeboom, D. Nielsen, P. Jaini, P. Forré, and M. Welling, “Argmax flows and multinomial diffusion: Learning categorical distributions,” *Advances in Neural Information Processing Systems*, vol. 34, pp. 12 454–12 465, 2021.

- [61] J. Austin, D. D. Johnson, J. Ho, D. Tarlow, and R. Van Den Berg, “Structured denoising diffusion models in discrete state-spaces,” *Advances in Neural Information Processing Systems*, vol. 34, pp. 17 981–17 993, 2021.
- [62] S. Gu, D. Chen, J. Bao, F. Wen, B. Zhang, D. Chen, L. Yuan, and B. Guo, “Vector quantized diffusion model for text-to-image synthesis,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 10 696–10 706.
- [63] Y. Ouyang, L. Xie, C. Li, and G. Cheng, “Missdiff: Training diffusion models on tabular data with missing values,” *arXiv preprint arXiv:2307.00467*, 2023.
- [64] Y. Song and S. Ermon, “Improved techniques for training score-based generative models,” *Advances in neural information processing systems*, vol. 33, pp. 12 438–12 448, 2020.
- [65] J. Song, C. Meng, and S. Ermon, “Denoising diffusion implicit models,” *arXiv preprint arXiv:2010.02502*, 2020.
- [66] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, “High-resolution image synthesis with latent diffusion models,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 10 684–10 695.
- [67] Y. Song, P. Dhariwal, M. Chen, and I. Sutskever, “Consistency models,” *arXiv preprint arXiv:2303.01469*, 2023.
- [68] C. Lu, Y. Zhou, F. Bao, J. Chen, C. Li, and J. Zhu, “Dpm-solver: A fast ode solver for diffusion probabilistic model sampling in around 10 steps,” *Advances in Neural Information Processing Systems*, vol. 35, pp. 5775–5787, 2022.
- [69] T. Salimans and J. Ho, “Progressive distillation for fast sampling of diffusion models,” *arXiv preprint arXiv:2202.00512*, 2022.
- [70] T. Karras, M. Aittala, T. Aila, and S. Laine, “Elucidating the design space of diffusion-based generative models,” *Advances in Neural Information Processing Systems*, vol. 35, pp. 26 565–26 577, 2022.
- [71] Q. Zhang, M. Tao, and Y. Chen, “gddim: Generalized denoising diffusion implicit models,” *arXiv preprint arXiv:2206.05564*, 2022.
- [72] F. Bao, C. Li, J. Zhu, and B. Zhang, “Analytic-dpm: an analytic estimate of the optimal reverse variance in diffusion probabilistic models,” *arXiv preprint arXiv:2201.06503*, 2022.
- [73] X. Liu, X. Zhang, J. Ma, J. Peng, and Q. Liu, “Instaflo: One step is enough for high-quality diffusion-based text-to-image generation,” *arXiv preprint arXiv:2309.06380*, 2023.
- [74] Q. Zhang and Y. Chen, “Fast sampling of diffusion models with exponential integrator,” *arXiv preprint arXiv:2204.13902*, 2022.
- [75] K. Clark, P. Vicol, K. Swersky, and D. J. Fleet, “Directly fine-tuning diffusion models on differentiable rewards,” *arXiv preprint arXiv:2309.17400*, 2023.
- [76] K. Lee, H. Liu, M. Ryu, O. Watkins, Y. Du, C. Boutilier, P. Abbeel, M. Ghavamzadeh, and S. S. Gu, “Aligning text-to-image models using human feedback,” *arXiv preprint arXiv:2302.12192*, 2023.

- [77] X. Wu, K. Sun, F. Zhu, R. Zhao, and H. Li, “Better aligning text-to-image models with human preference,” *arXiv preprint arXiv:2303.14420*, 2023.
- [78] K. Black, M. Janner, Y. Du, I. Kostrikov, and S. Levine, “Training diffusion models with reinforcement learning,” *arXiv preprint arXiv:2305.13301*, 2023.
- [79] Y. Fan, O. Watkins, Y. Du, H. Liu, M. Ryu, C. Boutilier, P. Abbeel, M. Ghavamzadeh, K. Lee, and K. Lee, “DPOK: Reinforcement learning for fine-tuning text-to-image diffusion models,” *arXiv preprint arXiv:2305.16381*, 2023.
- [80] J. Xu, X. Liu, Y. Wu, Y. Tong, Q. Li, M. Ding, J. Tang, and Y. Dong, “Imagereward: Learning and evaluating human preferences for text-to-image generation,” *arXiv preprint arXiv:2304.05977*, 2023.
- [81] Y. Hao, Z. Chi, L. Dong, and F. Wei, “Optimizing prompts for text-to-image generation,” *arXiv preprint arXiv:2212.09611*, 2022.
- [82] D. Watson, W. Chan, J. Ho, and M. Norouzi, “Learning fast samplers for diffusion models by differentiating through sample quality,” in *International Conference on Learning Representations*, 2021.
- [83] B. Wallace, A. Gokul, S. Ermon, and N. Naik, “End-to-end diffusion latent optimization improves classifier guidance,” *arXiv preprint arXiv:2303.13703*, 2023.
- [84] A. Block, Y. Mroueh, and A. Rakhlin, “Generative modeling with denoising auto-encoders and langevin sampling,” *arXiv preprint arXiv:2002.00107*, 2020.
- [85] H. Lee, J. Lu, and Y. Tan, “Convergence for score-based generative modeling with polynomial complexity,” *arXiv preprint arXiv:2206.06227*, 2022.
- [86] S. Chen, S. Chewi, J. Li, Y. Li, A. Salim, and A. R. Zhang, “Sampling is as easy as learning the score: theory for diffusion models with minimal data assumptions,” *arXiv preprint arXiv:2209.11215*, 2022.
- [87] H. Lee, J. Lu, and Y. Tan, “Convergence of score-based generative modeling for general data distributions,” *arXiv preprint arXiv:2209.12381*, 2022.
- [88] S. Chen, S. Chewi, H. Lee, Y. Li, J. Lu, and A. Salim, “The probability flow ode is provably fast,” *arXiv preprint arXiv:2305.11798*, 2023.
- [89] J. Benton, V. De Bortoli, A. Doucet, and G. Deligiannidis, “Linear convergence bounds for diffusion models via stochastic localization,” *arXiv preprint arXiv:2308.03686*, 2023.
- [90] K. Oko, S. Akiyama, and T. Suzuki, “Diffusion models are minimax optimal distribution estimators,” *arXiv preprint arXiv:2303.01861*, 2023.
- [91] M. Chen, K. Huang, T. Zhao, and M. Wang, “Score approximation, estimation and distribution recovery of diffusion models on low-dimensional data,” *arXiv preprint arXiv:2302.07194*, 2023.
- [92] S. Mei and Y. Wu, “Deep networks as denoising algorithms: Sample-efficient learning of diffusion models in high-dimensional graphical models,” *arXiv preprint arXiv:2309.11420*, 2023.

- [93] W. Tang and H. Zhao, “Score-based diffusion models via stochastic differential equations—a technical tutorial,” *arXiv preprint arXiv:2402.07487*, 2024.
- [94] A. Q. Nichol and P. Dhariwal, “Improved denoising diffusion probabilistic models,” in *Proceedings of the International Conference on Machine Learning*. PMLR, 2021, pp. 8162–8171.
- [95] B. D. Anderson, “Reverse-time diffusion equation models,” *Stochastic Processes and their Applications*, vol. 12, no. 3, pp. 313–326, 1982.
- [96] U. G. Haussmann and E. Pardoux, “Time reversal of diffusions,” *The Annals of Probability*, pp. 1188–1205, 1986.
- [97] S. Luo, Y. Tan, S. Patil, D. Gu, P. von Platen, A. Passos, L. Huang, J. Li, and H. Zhao, “Lcm-lora: A universal stable-diffusion acceleration module,” *arXiv preprint arXiv:2311.05556*, 2023.
- [98] Y. Liu, K. Zhang, Y. Li, Z. Yan, C. Gao, R. Chen, Z. Yuan, Y. Huang, H. Sun, and J. Gao, “Sora: A review on background, technology, limitations, and opportunities of large vision models,” *arXiv preprint arXiv:2402.17177*, 2024.
- [99] P. Esser, S. Kulal, A. Blattmann, R. Entezari, J. Müller, H. Saini, Y. Levi, D. Lorenz, A. Sauer, and F. Boesel, “Scaling rectified flow transformers for high-resolution image synthesis,” *arXiv preprint arXiv:2403.03206*, 2024.
- [100] J. Ho and T. Salimans, “Classifier-free diffusion guidance,” *arXiv preprint arXiv:2207.12598*, 2022.
- [101] A. Ramesh, P. Dhariwal, A. Nichol, C. Chu, and M. Chen, “Hierarchical text-conditional image generation with clip latents,” *arXiv preprint arXiv:2204.06125*, 2022.
- [102] Y. Wang, J. Yu, and J. Zhang, “Zero-shot image restoration using denoising diffusion null-space model,” *arXiv preprint arXiv:2212.00490*, 2022.
- [103] B. Kawar, M. Elad, S. Ermon, and J. Song, “Denoising diffusion restoration models,” *Advances in Neural Information Processing Systems*, vol. 35, pp. 23 593–23 606, 2022.
- [104] J. Choi, S. Kim, Y. Jeong, Y. Gwon, and S. Yoon, “ILVR: Conditioning method for denoising diffusion probabilistic models,” *arXiv preprint arXiv:2108.02938*, 2021.
- [105] C. Saharia, J. Ho, W. Chan, T. Salimans, D. J. Fleet, and M. Norouzi, “Image super-resolution via iterative refinement,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 4, pp. 4713–4726, 2022.
- [106] C. Saharia, W. Chan, H. Chang, C. Lee, J. Ho, T. Salimans, D. Fleet, and M. Norouzi, “Palette: Image-to-image diffusion models,” in *ACM SIGGRAPH 2022 Conference Proceedings*, 2022, pp. 1–10.
- [107] A. Lugmayr, M. Danelljan, A. Romero, F. Yu, R. Timofte, and L. Van Gool, “Repaint: Inpainting using denoising diffusion probabilistic models,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 11 461–11 471.
- [108] H. Li, Y. Yang, M. Chang, S. Chen, H. Feng, Z. Xu, Q. Li, and Y. Chen, “Srdiff: Single image super-resolution with diffusion probabilistic models,” *Neurocomputing*, vol. 479, pp. 47–59, 2022.

- [109] J. Whang, M. Delbraccio, H. Talebi, C. Saharia, A. G. Dimakis, and P. Milanfar, “Deblurring via stochastic refinement,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 16 293–16 303.
- [110] Y.-J. Lu, Y. Tsao, and S. Watanabe, “A study on speech enhancement based on diffusion probabilistic model,” in *2021 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*. IEEE, 2021, pp. 659–666.
- [111] S. Welker, J. Richter, and T. Gerkmann, “Speech enhancement with score-based generative models in the complex stft domain,” *arXiv preprint arXiv:2203.17004*, 2022.
- [112] J. Richter, S. Welker, J.-M. Lemercier, B. Lay, and T. Gerkmann, “Speech enhancement and dereverberation with diffusion-based generative models,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 2023.
- [113] C.-Y. Yu, S.-L. Yeh, G. Fazekas, and H. Tang, “Conditioning and sampling in variational diffusion models for speech super-resolution,” in *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2023, pp. 1–5.
- [114] Z. Wang, J. J. Hunt, and M. Zhou, “Diffusion policies as an expressive policy class for offline reinforcement learning,” *arXiv preprint arXiv:2208.06193*, 2022.
- [115] M. Janner, Q. Li, and S. Levine, “Offline reinforcement learning as one big sequence modeling problem,” *Advances in Neural Information Processing Systems*, 2021.
- [116] L. Chen, K. Lu, A. Rajeswaran, K. Lee, A. Grover, M. Laskin, P. Abbeel, A. Srinivas, and I. Mordatch, “Decision transformer: Reinforcement learning via sequence modeling,” *Advances in neural information processing systems*, vol. 34, pp. 15 084–15 097, 2021.
- [117] M. Janner, Y. Du, J. B. Tenenbaum, and S. Levine, “Planning with diffusion for flexible behavior synthesis,” *arXiv preprint arXiv:2205.09991*, 2022.
- [118] A. Ajay, Y. Du, A. Gupta, J. Tenenbaum, T. Jaakkola, and P. Agrawal, “Is conditional generative modeling all you need for decision-making?” *arXiv preprint arXiv:2211.15657*, 2022.
- [119] P. Agrawal, A. V. Nair, P. Abbeel, J. Malik, and S. Levine, “Learning to poke by poking: Experiential learning of intuitive physics,” *Advances in neural information processing systems*, vol. 29, 2016.
- [120] Z. Liang, Y. Mu, M. Ding, F. Ni, M. Tomizuka, and P. Luo, “AdaptDiffuser: Diffusion models as adaptive self-evolving planners,” *arXiv preprint arXiv:2302.01877*, 2023.
- [121] E. D. Zhong, T. Bepler, J. H. Davis, and B. Berger, “Reconstructing continuous distributions of 3d protein structure from cryo-em images,” *arXiv preprint arXiv:1909.05215*, 2019.
- [122] E. D. Zhong, T. Bepler, B. Berger, and J. H. Davis, “CryoDRGN: reconstruction of heterogeneous cryo-EM structures using neural networks,” *Nature Methods*, vol. 18, no. 2, pp. 176–185, 2021.
- [123] J.-E. Shin, A. J. Riesselman, A. W. Kollasch, C. McMahon, E. Simon, C. Sander, A. Manglik, A. C. Kruse, and D. S. Marks, “Protein design and variant prediction using autoregressive generative models,” *Nature Communications*, vol. 12, no. 1, p. 2403, 2021.

- [124] A. Strokach and P. M. Kim, “Deep generative modeling for protein design,” *Current Opinion in Structural Biology*, vol. 72, pp. 226–236, 2022.
- [125] B. Trabucco, X. Geng, A. Kumar, and S. Levine, “Design-bench: Benchmarks for data-driven offline model-based optimization,” in *Proceedings of the International Conference on Machine Learning*. PMLR, 2022, pp. 21 658–21 676.
- [126] A. Kumar and S. Levine, “Model inversion networks for model-based optimization,” *Advances in Neural Information Processing Systems*, vol. 33, pp. 5126–5137, 2020.
- [127] S. Krishnamoorthy, S. M. Mashkaria, and A. Grover, “Diffusion models for black-box optimization,” *arXiv preprint arXiv:2306.07180*, 2023.
- [128] H. Yuan, K. Huang, C. Ni, M. Chen, and M. Wang, “Reward-directed conditional diffusion: Provable distribution estimation and reward improvement,” *arXiv preprint arXiv:2307.07055*, 2023.
- [129] A. Hyvärinen and P. Dayan, “Estimation of non-normalized statistical models by score matching.” *Journal of Machine Learning Research*, vol. 6, no. 4, 2005.
- [130] P. Vincent, “A connection between score matching and denoising autoencoders,” *Neural computation*, vol. 23, no. 7, pp. 1661–1674, 2011.
- [131] C.-W. Huang, J. H. Lim, and A. C. Courville, “A variational perspective on diffusion-based generative models and score matching,” *Advances in Neural Information Processing Systems*, vol. 34, pp. 22 863–22 876, 2021.
- [132] C. Luo, “Understanding diffusion models: A unified perspective,” *arXiv preprint arXiv:2208.11970*, 2022.
- [133] A. Vahdat, K. Kreis, and J. Kautz, “Score-based generative modeling in latent space,” *Advances in Neural Information Processing Systems*, vol. 34, pp. 11 287–11 302, 2021.
- [134] O. Ronneberger, P. Fischer, and T. Brox, “U-net: Convolutional networks for biomedical image segmentation,” in *Medical Image Computing and Computer-Assisted Intervention—MICCAI 2015: 18th International Conference, Munich, Germany, October 5–9, 2015, Proceedings, Part III 18*. Springer, 2015, pp. 234–241.
- [135] W. Peebles and S. Xie, “Scalable diffusion models with transformers,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 4195–4205.
- [136] A. Gupta, L. Yu, K. Sohn, X. Gu, M. Hahn, L. Fei-Fei, I. Essa, L. Jiang, and J. Lezama, “Photorealistic video generation with diffusion models,” *arXiv preprint arXiv:2312.06662*, 2023.
- [137] G. Cybenko, “Approximation by superpositions of a sigmoidal function,” *Mathematics of Control, Signals and Systems*, vol. 2, no. 4, pp. 303–314, 1989.
- [138] K. Hornik, M. Stinchcombe, and H. White, “Universal approximation of an unknown mapping and its derivatives using multilayer feedforward networks,” *Neural Networks*, vol. 3, no. 5, pp. 551–560, 1990.
- [139] A. R. Barron, “Approximation and estimation bounds for artificial neural networks,” *Machine Learning*, vol. 14, pp. 115–133, 1994.

- [140] I. Gühring, G. Kutyniok, and P. Petersen, “Error bounds for approximations with deep relu neural networks in w, s, p norms,” *Analysis and Applications*, vol. 18, no. 05, pp. 803–859, 2020.
- [141] D. Yarotsky, “Optimal approximation of continuous functions by very deep relu networks,” in *Conference on learning theory*. PMLR, 2018, pp. 639–649.
- [142] J. Lu, Z. Shen, H. Yang, and S. Zhang, “Deep network approximation for smooth functions,” *SIAM Journal on Mathematical Analysis*, vol. 53, no. 5, pp. 5465–5506, 2021.
- [143] A. J. Schmidt-Hieber, “Nonparametric regression using deep neural networks with relu activation function,” *Annals of Statistics*, vol. 48, no. 4, pp. 1875–1897, 2020.
- [144] M. Chen, H. Jiang, W. Liao, and T. Zhao, “Efficient approximation of deep relu networks for functions on low dimensional manifolds,” *Advances in Neural Information Processing Systems*, vol. 32, 2019.
- [145] K. Y. Yang and A. Wibisono, “Convergence in kl and rényi divergence of the unadjusted langevin algorithm using estimated score,” in *NeurIPS 2022 Workshop on Score-Based Methods*, 2022.
- [146] A. Wibisono, Y. Wu, and K. Y. Yang, “Optimal score estimation via empirical bayes smoothing,” *arXiv preprint arXiv:2402.07747*, 2024.
- [147] K. Shah, S. Chen, and A. Klivans, “Learning mixtures of gaussians using the ddpm objective,” *arXiv preprint arXiv:2307.01178*, 2023.
- [148] Y. Han, M. Razaviyayn, and R. Xu, “Neural network-based score estimation in diffusion models: Optimization and generalization,” *arXiv preprint arXiv:2401.15604*, 2024.
- [149] S. Geman and C. Graffigne, “Markov random field image models and their applications to computer vision,” in *Proceedings of the international congress of mathematicians*, vol. 1. Berkeley, CA, 1986, p. 2.
- [150] M. Ranzato, A. Krizhevsky, and G. Hinton, “Factored 3-way restricted boltzmann machines for modeling natural images,” in *Proceedings of the thirteenth international conference on artificial intelligence and statistics*. JMLR Workshop and Conference Proceedings, 2010, pp. 621–628.
- [151] M. Mezard and A. Montanari, *Information, physics, and computation*. Oxford University Press, 2009.
- [152] R. Eldan, “Gaussian-width gradient complexity, reverse log-sobolev inequalities and nonlinear large deviations,” *Geometric and Functional Analysis*, vol. 28, no. 6, pp. 1548–1596, 2018.
- [153] M. Celentano, Z. Fan, and S. Mei, “Local convexity of the tap free energy and amp convergence for z 2-synchronization,” *The Annals of Statistics*, vol. 51, no. 2, pp. 519–546, 2023.
- [154] V. De Bortoli, J. Thornton, J. Heng, and A. Doucet, “Diffusion schrödinger bridge with applications to score-based generative modeling,” *Advances in Neural Information Processing Systems*, vol. 34, pp. 17 695–17 709, 2021.
- [155] M. S. Albergo, N. M. Boffi, and E. Vanden-Eijnden, “Stochastic interpolants: A unifying framework for flows and diffusions,” *arXiv preprint arXiv:2303.08797*, 2023.

- [156] J. Benton, V. De Bortoli, A. Doucet, and G. Deligiannidis, “Nearly d -linear convergence bounds for diffusion models via stochastic localization,” in *Proceedings of the International Conference on Learning Representations*, 2024.
- [157] S. Chen, G. Daras, and A. Dimakis, “Restoration-degradation beyond linear diffusions: A non-asymptotic analysis for ddim-type samplers,” in *International Conference on Machine Learning*. PMLR, 2023, pp. 4462–4484.
- [158] G. Li, Y. Huang, T. Efimov, Y. Wei, Y. Chi, and Y. Chen, “Accelerating convergence of score-based diffusion models, provably,” *arXiv preprint arXiv:2403.03852*, 2024.
- [159] G. Li, Y. Wei, Y. Chen, and Y. Chi, “Towards faster non-asymptotic convergence for diffusion-based generative models,” *arXiv preprint arXiv:2306.09251*, 2023.
- [160] V. De Bortoli, “Convergence of denoising diffusion models under the manifold hypothesis,” *arXiv preprint arXiv:2208.05314*, 2022.
- [161] A. Montanari and Y. Wu, “Posterior sampling from the spiked models via diffusion processes,” *arXiv preprint arXiv:2304.11449*, 2023.
- [162] A. El Alaoui, A. Montanari, and M. Sellke, “Sampling from mean-field gibbs measures via diffusion processes,” *arXiv preprint arXiv:2310.08912*, 2023.
- [163] R. Eldan, “Thin shell implies spectral gap up to polylog via a stochastic localization scheme,” *Geometric and Functional Analysis*, vol. 23, no. 2, pp. 532–569, 2013.
- [164] A. Montanari, “Sampling, diffusions, and stochastic localization,” *arXiv preprint arXiv:2305.10690*, 2023.
- [165] Y. Chen and R. Eldan, “Localization schemes: A framework for proving mixing bounds for markov chains,” in *2022 IEEE 63rd Annual Symposium on Foundations of Computer Science (FOCS)*. IEEE, 2022, pp. 110–122.
- [166] A. El Alaoui and A. Montanari, “An information-theoretic view of stochastic localization,” *IEEE Transactions on Information Theory*, vol. 68, no. 11, pp. 7423–7426, 2022.
- [167] A. El Alaoui, A. Montanari, and M. Sellke, “Sampling from the sherrington-kirkpatrick gibbs measure via algorithmic stochastic localization,” in *2022 IEEE 63rd Annual Symposium on Foundations of Computer Science (FOCS)*. IEEE, 2022, pp. 323–334.
- [168] D. Ghio, Y. Dandi, F. Krzakala, and L. Zdeborova, “Sampling with flows, diffusion and autoregressive neural networks: A spin-glass perspective,” *arXiv preprint arXiv:2308.14085*, 2023.
- [169] Y. Song, S. Garg, J. Shi, and S. Ermon, “Sliced score matching: A scalable approach to density and score estimation,” in *Uncertainty in Artificial Intelligence*. PMLR, 2020, pp. 574–584.
- [170] X. Liu, L. Wu, M. Ye, and Q. Liu, “Let us build bridges: Understanding and extending diffusion generative models,” *arXiv preprint arXiv:2208.14699*, 2022.
- [171] R. Eldan, “Taming correlations through entropy-efficient measure decompositions with applications to mean-field approximation,” *Probability Theory and Related Fields*, vol. 176, no. 3, pp. 737–755, 2020.

- [172] —, “Analysis of high-dimensional distributions using pathwise methods,” in *Proc. Int. Cong. Math.*, vol. 6, 2022, pp. 4246–4270.
- [173] Y. T. Lee and S. S. Vempala, “Eldan’s stochastic localization and the kls conjecture: Isoperimetry, concentration and mixing,” *arXiv preprint arXiv:1612.01507*, 2016.
- [174] Y. Chen, “An almost constant lower bound of the isoperimetric coefficient in the kls conjecture,” *Geometric and Functional Analysis*, vol. 31, pp. 34–61, 2021.
- [175] A. E. Alaoui, A. Montanari, and M. Sellke, “Sampling from mean-field gibbs measures via diffusion processes,” *arXiv preprint arXiv:2310.08912*, 2023.
- [176] Z. Li, H. Yuan, K. Huang, C. Ni, Y. Ye, M. Chen, and M. Wang, “Diffusion model for data-driven black-box optimization,” *arXiv preprint arXiv:2403.13219*, 2024.
- [177] P. Dhariwal and A. Nichol, “Diffusion models beat gans on image synthesis,” *Advances in Neural Information Processing Systems*, vol. 34, pp. 8780–8794, 2021.
- [178] A. Brock, J. Donahue, and K. Simonyan, “Large scale gan training for high fidelity natural image synthesis,” *arXiv preprint arXiv:1809.11096*, 2018.
- [179] D. P. Kingma and P. Dhariwal, “Glow: Generative flow with invertible 1x1 convolutions,” *Advances in Neural Information Processing Systems*, vol. 31, 2018.
- [180] Y. Wu, M. Chen, Z. Li, M. Wang, and Y. Wei, “Theoretical insights for diffusion guidance: A case study for gaussian mixture models,” *arXiv preprint arXiv:2403.01639*, 2024.
- [181] M. Uehara, Y. Zhao, K. Black, E. Hajiramezanali, G. Scalia, N. L. Diamant, A. M. Tseng, S. Levine, and T. Biancalani, “Feedback efficient online fine-tuning of diffusion models,” *arXiv preprint arXiv:2402.16359*, 2024.
- [182] W. Tang, “Fine-tuning of diffusion models via stochastic control: entropy regularization and beyond,” *arXiv preprint arXiv:2403.06279*, 2024.
- [183] M. Uehara, Y. Zhao, K. Black, E. Hajiramezanali, G. Scalia, N. L. Diamant, A. M. Tseng, T. Biancalani, and S. Levine, “Fine-tuning of continuous-time diffusion models as entropy-regularized control,” *arXiv preprint arXiv:2402.15194*, 2024.
- [184] H. Fu, Z. Yang, M. Wang, and M. Chen, “Unveil conditional diffusion models with classifier-free guidance: A sharp statistical theory,” *arXiv preprint arXiv:2403.11968*, 2024.
- [185] H. Chung, J. Kim, M. T. Mccann, M. L. Klasky, and J. C. Ye, “Diffusion posterior sampling for general noisy inverse problems,” *arXiv preprint arXiv:2209.14687*, 2022.
- [186] L. Rout, N. Raoof, G. Daras, C. Caramanis, A. Dimakis, and S. Shakkottai, “Solving linear inverse problems provably via posterior sampling with latent diffusion models,” *Advances in Neural Information Processing Systems*, vol. 36, 2024.
- [187] H. Chung, B. Sim, D. Ryu, and J. C. Ye, “Improving diffusion models for inverse problems using manifold constraints,” *Advances in Neural Information Processing Systems*, vol. 35, pp. 25 683–25 696, 2022.

- [188] Y. Pan and D. Wu, “A novel recommendation model for online-to-offline service based on the customer network and service location,” *Journal of Management Information Systems*, vol. 37, no. 2, pp. 563–593, 2020.
- [189] Y. Pan, D. Wu, C. Luo, and A. Dolgui, “User activity measurement in rating-based online-to-offline (o2o) service recommendation,” *Information Sciences*, vol. 479, pp. 180–196, 2019.
- [190] D. G. Luenberger, Y. Ye *et al.*, *Linear and nonlinear programming*. Springer, 1984, vol. 2.
- [191] J. Fu and S. Levine, “Offline model-based optimization via normalized maximum likelihood estimation,” *arXiv preprint arXiv:2102.07970*, 2021.
- [192] J. Bu, D. Simchi-Levi, and L. Wang, “Offline pricing and demand learning with censored data,” *Management Science*, vol. 69, no. 2, pp. 885–903, 2023.
- [193] T. Nguyen-Tang, S. Gupta, A. T. Nguyen, and S. Venkatesh, “Offline neural contextual bandits: Pessimism, optimization and generalization,” *arXiv preprint arXiv:2111.13807*, 2021.
- [194] Y. Jin, Z. Yang, and Z. Wang, “Is pessimism provably efficient for offline RL?” in *International Conference on Machine Learning*. PMLR, 2021, pp. 5084–5096.
- [195] R. A. Bradley and M. E. Terry, “Rank analysis of incomplete block designs: I. the method of paired comparisons,” *Biometrika*, vol. 39, no. 3/4, pp. 324–345, 1952.
- [196] B. Zhu, M. Jordan, and J. Jiao, “Principled reinforcement learning with human feedback from pairwise or k-wise comparisons,” in *Proceedings of the International Conference on Machine Learning*. PMLR, 2023, pp. 43 037–43 067.
- [197] K. J. Åström, *Introduction to stochastic control theory*. Courier Corporation, 2012.
- [198] W. H. Fleming and R. W. Rishel, *Deterministic and stochastic optimal control*. Springer Science & Business Media, 2012, vol. 1.
- [199] A. Lou and S. Ermon, “Reflected diffusion models,” *arXiv preprint arXiv:2304.04740*, 2023.
- [200] W. Nie, B. Guo, Y. Huang, C. Xiao, A. Vahdat, and A. Anandkumar, “Diffusion models for adversarial purification,” *arXiv preprint arXiv:2205.07460*, 2022.
- [201] Q. Wu, H. Ye, and Y. Gu, “Guided diffusion model for adversarial purification from random noise,” *arXiv preprint arXiv:2206.10875*, 2022.
- [202] N. Carlini, F. Tramer, K. D. Dvijotham, L. Rice, M. Sun, and J. Z. Kolter, “(certified!!) adversarial robustness for free!” *arXiv preprint arXiv:2206.10550*, 2022.
- [203] C. Xiao, Z. Chen, K. Jin, J. Wang, W. Nie, M. Liu, A. Anandkumar, B. Li, and D. Song, “Densepure: Understanding diffusion models for adversarial robustness,” in *The Eleventh International Conference on Learning Representations*, 2022.
- [204] E. Delage and Y. Ye, “Distributionally robust optimization under moment uncertainty with application to data-driven problems,” *Operations research*, vol. 58, no. 3, pp. 595–612, 2010.
- [205] D. Kuhn, P. M. Esfahani, V. A. Nguyen, and S. Shafieezadeh-Abadeh, “Wasserstein distributionally robust optimization: Theory and applications in machine learning,” in *Operations research & management science in the age of analytics*. Informs, 2019, pp. 130–166.

- [206] J. Goh and M. Sim, “Distributionally robust optimization and its tractable approximations,” *Operations research*, vol. 58, no. 4-part-1, pp. 902–917, 2010.
- [207] H. Rahimian and S. Mehrotra, “Distributionally robust optimization: A review,” *arXiv preprint arXiv:1908.05659*, 2019.
- [208] C. Xu, J. Lee, X. Cheng, and Y. Xie, “Flow-based distributionally robust optimization,” *IEEE Journal on Selected Areas in Information Theory*, 2024.
- [209] C. Meng, K. Choi, J. Song, and S. Ermon, “Concrete score matching: Generalized score matching for discrete data,” *Advances in Neural Information Processing Systems*, vol. 35, pp. 34 532–34 545, 2022.
- [210] A. Campbell, J. Benton, V. De Bortoli, T. Rainforth, G. Deligiannidis, and A. Doucet, “A continuous time framework for discrete denoising models,” *Advances in Neural Information Processing Systems*, vol. 35, pp. 28 266–28 279, 2022.
- [211] J. Benton, Y. Shi, V. De Bortoli, G. Deligiannidis, and A. Doucet, “From denoising diffusions to denoising markov models,” *arXiv preprint arXiv:2211.03595*, 2022.
- [212] J. E. Santos, Z. R. Fox, N. Lubbers, and Y. T. Lin, “Blackout diffusion: generative diffusion models in discrete-state spaces,” in *Proceedings of the International Conference on Machine Learning*. PMLR, 2023, pp. 9034–9059.
- [213] A. Lou, C. Meng, and S. Ermon, “Discrete diffusion language modeling by estimating the ratios of the data distribution,” *arXiv preprint arXiv:2310.16834*, 2023.
- [214] H. Sun, L. Yu, B. Dai, D. Schuurmans, and H. Dai, “Score-based continuous-time discrete diffusion models,” *arXiv preprint arXiv:2211.16750*, 2022.
- [215] Y. Li, J. Guo, R. Wang, and J. Yan, “From distribution learning in training to gradient search in testing for combinatorial optimization,” *Advances in Neural Information Processing Systems*, vol. 36, 2024.
- [216] H. Chen and L. Ying, “Convergence analysis of discrete diffusion model: Exact implementation through uniformization,” *arXiv preprint arXiv:2402.08095*, 2024.