Schrödinger-Föllmer Sampler

Jian Huang, Yuling Jiao, Lican Kang, Xu Liao, Jin Liu, and Yanyan Liu

Abstract— Sampling from probability distributions is a critical problem in statistics and machine learning, particularly in Bayesian inference, where direct integration over the posterior distribution is often infeasible, making sampling from the posterior essential for inference. This paper introduces the Schrödinger-Föllmer sampler (SFS), a novel approach for sampling from potentially unnormalized distributions. The SFS leverages the Schrödinger-Föllmer diffusion process on the unit interval, incorporating a time-dependent drift term that evolves the distribution from a degenerate form at time zero to the target distribution at time one. Unlike existing Markov chain Monte Carlo methods that rely on ergodicity, SFS operates independently of ergodicity. Computationally, SFS is straightforward to implement using the Euler-Maruyama discretization. In our theoretical analysis, we derive non-asymptotic error bounds for the SFS sampling distribution in the Wasserstein distance, subject to reasonable conditions. Numerical experiments demonstrate that SFS generates higher-quality samples than several established methods.

Index Terms— Euler-Maruyama discretization, non-asymptotic error bound, Schrödinger bridge, unnormalized distribution, Wasserstein distance.

Received 7 February 2024; revised 11 November 2024; accepted 20 December 2024. Date of publication 25 December 2024; date of current version 22 January 2025. This work was supported in part by Guangdong Provincial Key Laboratory of Mathematical Foundations for Artificial Intelligence under Grant 2023B12120001 and in part by Shenzhen Key Laboratory of Cross-Modal Cognitive Computing under Grant ZDSYS20220626091302006. The work of Jian Huang was supported in part by the National Natural Science Foundation of China under Grant 72331005 and in part by The Hong Kong Polytechnic University. The work of Yuling Jiao was supported in part by the National Natural Science Foundation of China under Grant 12371283; and in part by the University Development Foundation of The Chinese University of Hong Kong, Shenzhen, under Grant UDFP1003033. The work of Yanyan Liu is with the School of Artificial Intelligence and Hubei Key Laboratory of Computational Intelligence and Bioinformatics. The authors: Lican Kang; Jin Liu.)

Jian Huang is with the Department of Data Science and AI and the Department of Applied Mathematics, The Hong Kong Polytechnic University, Hong Kong, SAR, China (e-mail: j.huang@polyu.edu.hk).

Yuling Jiao is with the School of Artificial Intelligence and Hubei Key Laboratory of Computational Science, Wuhan University, Wuhan 430072, China (e-mail: yulingjiaomath@whu.edu.cn).

Lican Kang is with the Institute for Mathematics and AI and the School of Mathematics and Statistics, Wuhan University, Wuhan 430072, China (e-mail: kanglican@whu.edu.cn).

Xu Liao is with the School of Data Science, The Chinese University of Hong Kong, Shenzhen 518172, China, and also with the Center of Quantitative Medicine, Duke-NUS Medical School, Singapore 169857 (e-mail: liaoxu@u.duke.nus.edu).

Jin Liu is with the School of Data Science, The Chinese University of Hong Kong, Shenzhen 518172, China (e-mail: liujinlab@cuhk.edu.cn).

Yanyan Liu is with the School of Mathematics and Statistics, Wuhan University, Wuhan 430072, China (e-mail: liuyy@whu.edu.cn).

Communicated by R. Venkataramanan, Associate Editor for Machine Learning and Statistics, Communications, Signal Processing and Source Coding.

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TIT.2022.3522494.

Digital Object Identifier 10.1109/TIT.2022.3522494

I. INTRODUCTION

Sampling from a probability distribution is a fundamental problem in statistics and machine learning. Efficiently sampling from an unnormalized posterior distribution, for instance, is essential to the success of Bayesian inference. Numerous sampling methods have been developed, particularly in the field of Markov Chain Monte Carlo (MCMC). Notable examples include the Metropolis-Hastings (MH) algorithm [1], [2], [3], the Gibbs sampler [4], [5], the Langevin algorithm [6], [7], [8], the bouncy particle sampler [9], [10], and the zig-zag sampler [11], among others. For additional background, see [12], [13], [14], and [15] and references therein. The Langevin sampler, based on the Euler-Maruyama discretization of Langevin diffusion, has garnered significant attention. Langevin diffusion is given by

\[ dL_t = -\nabla V(L_t)dt + \sqrt{2d}dB_t, \]

where \(-\nabla V(\cdot)\) is the drift term, and \(\{B_t\}_{t \geq 0}\) represents a standard \(p\)-dimensional Brownian motion process. Under appropriate conditions, the Langevin diffusion process \(\{L_t\}_{t \geq 0}\) in (1) has an invariant distribution \(\mu(x) = \exp(-V(x))/C\) for \(x \in \mathbb{R}^p\), where \(C > 0\) denotes the normalization constant ensuring that \(\mu(x)\) integrates to one [16], [17]. The Langevin sampler’s desirable convergence properties, especially under the strongly convex potential assumption, have been established by various researchers [7], [18], [19], [20], [21], [22]. In addition, alternative conditions, such as the dissipativity condition for the drift term [23], [24], [25] and the local convexity condition for the potential function outside a ball [8], [26], [27], [28], can replace the strongly convex assumption to guarantee the log-Sobolev inequality for the target distribution. Despite significant advancements, sampling from distributions with multiple modes or high-dimensional distributions remains challenging [14]. For instance, even for the one-dimensional Gaussian mixture model \(0.5 \mathcal{N}(-1, \sigma^2) + 0.5 \mathcal{N}(1, \sigma^2)\), the optimally tuned Hamiltonian Monte Carlo and random walk Metropolis algorithms exhibit mixing times proportional to \(\exp(1/(2\sigma^2))\) [14], [29], which grows exponentially as \(\sigma\) approaches 0. The constant in the log-Sobolev inequality may also depend exponentially on dimensionality [23], [30], [31], [32], indicating that the Langevin sampler’s efficiency may be compromised by the curse of dimensionality when the ambient dimension \(p\) is high. Furthermore, tempered MCMC methods [33], [34], [35], [36] have been developed to improve mixing efficiency. These challenges in existing MCMC methods arise from the need for ergodicity in the constructed Markov chain, which ensures that each region of the target distribution’s sample space is visited with positive probability. In practice, however, exploring every...
region of a high-dimensional sample space is computationally prohibitive, rendering these MCMC methods inherently limited.

In this paper, we propose the Schrödinger-Föllmer sampler (SFS), a novel approach to sampling that does not require ergodicity. SFS is based on the Schrödinger-Föllmer diffusion:

$$dX_t = b(X_t, t)dt + dB_t, \quad X_0 = 0, \quad t \in [0, 1],$$

(2)

where the function $b: \mathbb{R}^p \times [0, 1] \to \mathbb{R}^p$ is a time-dependent drift term defined by the target distribution. The specific form of $b$ is provided in (6) below. According to [37] and [38], the process $\{X_t\}_{t \in [0, 1]}$ (2) was first formulated by Föllmer [39], [40], [41] in the context of the Schrödinger bridge problem [42].

The law of $\{X_t\}_{t \in [0, 1]}$ in (2) minimizes the relative entropy with respect to the Wiener measure among all processes with laws interpolating $\delta_0$ (the degenerate distribution at $X_0 = 0$) and the target distribution $\mu$ [37], [43], [44]. Consequently, the diffusion process (2) allows sampling from the target distribution $\mu$ by evolving the initial degenerate distribution at $t = 0$ to the target distribution $\mu$ at $t = 1$. For numerical implementation, we use the Euler-Maruyama method to discretize the diffusion process (2) as

$$Y_{tk+1} = Y_{tk} + \delta_k b(Y_{tk}, tk) + \sqrt{\delta_k} \epsilon_{k+1},$$

$$k = 0, 1, \ldots, K - 1, \quad K \geq 2,$$

(3)

where $K$ represents the number of grid points on $[0, 1]$ with $0 = t_0 < t_1 < \ldots < t_K = 1$, $\delta_k = t_{k+1} - t_k$ is the $k$-th step size, and $\{\epsilon_k\}_{k=1}^K$ are independent and identically distributed (i.i.d.) random vectors from $N(0, I_p)$. Based on (3), we can start from $Y_{t_0} = 0$ and iteratively update this initial value to obtain a realization of the random vector $Y_{tk}$, which is approximately distributed as the target distribution $\mu$ under suitable conditions.

An important difference between SFS and existing MCMC methods is that ergodicity is not required for SFS to generate valid samples. This is due to the basic property of the Schrödinger-Föllmer diffusion (2) on the unit interval $[0, 1]$, which transports the initial distribution at $t = 0$ to the exact target distribution $\mu$ at $t = 1$. The sampling error of SFS arises solely from the Euler-Maruyama discretization and the drift term approximation in Schrödinger-Föllmer diffusion; these approximation errors can be made arbitrarily small under appropriate conditions.

Our main contributions are as follows:

(i) We propose a novel sampling method, SFS, that does not require ergodicity. SFS is based on the Euler-Maruyama discretization applied to the Schrödinger-Föllmer diffusion and is applicable to unnormalized distributions.

(ii) We establish non-asymptotic bounds on the Wasserstein distance between the distribution of samples generated by SFS and the target distribution $\mu$ under suitable conditions. When the drift term $b$ has a closed-form expression, as in the case where the target $\mu$ is a finite mixture of Gaussians, we demonstrate that

$$W_2(\text{Law}(Y_{tk}), \mu) = \mathcal{O}(\sqrt{p/K}),$$

holds under specific smoothness conditions for the drift term (see Theorem 1). When the drift term requires Monte Carlo approximation, we prove that

$$W_2(\text{Law}(\tilde{Y}_{tk}), \mu) = \mathcal{O}(\sqrt{p(1/\sqrt{K} + 1/\sqrt{m})})$$

holds under the assumption that the potential $U(x, t)$ in (5) below is strongly convex with respect to $x$, where $m$ is the number of Gaussian samples used in the Monte Carlo approximation of $b$ (see Theorem 2).

(iii) We perform numerical experiments to assess the effectiveness of SFS, demonstrating that it outperforms several existing MCMC methods in applications to Gaussian mixture models, demonstrating that it outperforms several existing MCMC methods in applications to Gaussian mixture models, demonstrating that it outperforms several existing MCMC methods in applications to Gaussian mixture models, demonstrating that it outperforms several existing MCMC methods in applications to Gaussian mixture models, demonstrating that it outperforms several existing MCMC methods in applications to Gaussian mixture models, demonstrating that it outperforms several existing MCMC methods in applications to Gaussian mixture models.

The rest of the paper is organized as follows. In Section II, we present the proposed SFS in detail. In Section III, we establish non-asymptotic bounds on the Wasserstein distance between the distribution of the samples generated by SFS and the target distribution. Section IV provides an overview of related work. In Section V, we conduct simulation studies to evaluate the performance of SFS. Concluding remarks are given in Section VI. Background information on the Schrödinger-Föllmer diffusion and proofs for all the theorems are provided in Appendix I.

We end this section by introducing some notations used throughout the paper. Let $\mathcal{B}(\mathbb{R}^p)$ denote the Borel set of $\mathbb{R}^p$, and let $\mathcal{P}(\mathbb{R}^p)$ represent the collection of probability measures on $(\mathbb{R}^p, \mathcal{B}(\mathbb{R}^p))$. The gradient of a smooth function $\varphi(x), x \in \mathbb{R}^p$ is denoted by $\nabla \varphi(x)$. Likewise, we denote the partial derivative of $\phi(x, t), (x, t) \in \mathbb{R}^p \times [0, 1]$ with respect to $x$ by $\nabla_x \phi(x, t)$. For symmetric matrices $A, B \in \mathbb{R}^{p \times p}$, $A \succeq B$ indicates that $A - B$ is a positive definite matrix. For $a, b \in \mathbb{R}$, we write $a \preceq b$ to indicate $a \leq cb$ for some constant $c > 0$. Let $\|\beta\|_\ell = (\sum_{i=1}^p |\beta_i|^\ell)^{1/\ell}$ be the $\ell$-norm of the vector $\beta = (\beta_1, \ldots, \beta_p)\top \in \mathbb{R}^p$, and let $\|X\|_{L_2} = (\mathbb{E}[|X|^2])^{1/2}$ denote the $L_2$ norm of a random vector $X$. The expectation of the random vector $X$ is denoted by $\mathbb{E}_X$.

II. SCHRODINGER-FÖLLMER SAMPLER

In this section, we introduce the Schrödinger-Föllmer diffusion, with the detailed background deferred to Appendix A. We proceed to derive the closed-form expression for the drift term in the case where the target distribution is a Gaussian mixture, and we present the proposed SFS method based on the Euler-Maruyama discretization of the Schrödinger-Föllmer diffusion.

A. Schrödinger-Föllmer Diffusion Process

Let $\mu \in \mathcal{P}(\mathbb{R}^p)$ denote the target distribution of interest. Suppose $\mu$ is absolutely continuous with respect to the $p$-dimensional standard Gaussian distribution $N(0, I_p)$. Let $f$ denote the Radon-Nikodym derivative of $\mu$ with respect to $N(0, I_p)$, or the ratio of the density of $\mu$ over the density of $N(0, I_p)$, i.e.,

$$f(x) = \frac{d\mu}{dN(0, I_p)}(x), \quad x \in \mathbb{R}^p.$$
Let $Q_t$ be the heat semigroup defined by

$$ Q_t \tilde{f}(x) = E_{Z \sim N(0, I_p)}[\tilde{f}(x + \sqrt{t}Z)], \quad t \in [0, 1], $$

where $\tilde{f}$ is a measurable function. The Schrödinger-Föllmer diffusion process $\{X_t\}_{t \in [0, 1]}$ is defined as [39, 40, 41]

$$ dX_t = -\nabla_x U(X_t, t) dt + dB_t, \quad X_0 = 0, \quad t \in [0, 1], \quad (4) $$

where $U$ is the potential given by

$$ U(x, t) = -\log Q_{1-t}f(x). \quad (5) $$

This process $\{X_t\}_{t \in [0, 1]}$, as defined by (4), solves (27) with $\nu = \delta_0$, $\mu = \mu$, and $V_t(x) = -\nabla_x U(x, t)$ [38, 44, 45]. For simplicity, we denote the drift term of the SDE (4) by

$$ b(x, t) = -\nabla_x U(x, t) = \frac{E_Z[\nabla f(x + \sqrt{1-t}Z)]}{E_Z[f(x + \sqrt{1-t}Z)]}, \quad (6) $$

where $x \in \mathbb{R}^p$, $t \in [0, 1]$, $Z \sim N(0, I_p)$.

To ensure that the SDE (4) admits a unique strong solution, we assume that the drift term $b$ satisfies both a linear growth condition and a Lipschitz continuity condition [46, 47]:

$$ \|b(x, t)\|^2_2 \leq C_0(1 + \|x\|^2_2), \quad x \in \mathbb{R}^p, t \in [0, 1], \quad (C1) $$

and

$$ \|b(x, t) - b(y, t)\|_2 \leq C_1\|x - y\|_2, \quad x, y \in \mathbb{R}^p, t \in [0, 1], \quad (C2) $$

where $C_0$ and $C_1$ are finite positive constants.

**Proposition 1:** Assume (C1) and (C2) hold, then the Schrödinger-Föllmer SDE (4) has a unique strong solution $\{X_t\}_{t \in [0, 1]}$ with $X_0 \sim \delta_0$ and $X_1 \sim \mu$.

**Remark 1:**

(i) This SDE property well-established for the Schrödinger-Föllmer process [38, 44, 45, 48]. Refer also to the review [37] for further discussions and historical context on the Schrödinger problem.

(ii) The drift term $b(x, t)$ is scale-invariant with respect to $f$, meaning that $b(x, t) = \nabla \log Q_{1-t}f(x), \forall C > 0$. Consequently, the Schrödinger-Föllmer diffusion can be applied to sample from an unnormalized distribution $\mu$, without needing the normalizing constant of $\mu$.

(iii) If $f$ and $\nabla f$ are Lipschitz continuous and $f$ has a lower bound strictly greater than zero, it follows that Assumptions (C1) and (C2) are satisfied. When the target distribution $\mu$ is a Gaussian mixture distribution (10), both $f$ and $\nabla f$ are Lipschitz continuous if $I_p > \Sigma_i$, $i = 1, \ldots, \kappa$.

Suppose the target distribution has a density function with respect to the Lebesgue measure $\mathcal{L}$ on $(\mathbb{R}^p, B(\mathbb{R}^p))$. Let $\mu$ also represent the density function, with a normalizing constant $C > 0$. Without loss of generality, we express this as

$$ \mu(x) = \frac{1}{C} \exp(-V(x)), \quad x \in \mathbb{R}^p, \quad (7) $$

where $V$ has a known form, but $C$ may be unknown. The Radon-Nikodym derivative of $\mu$ with respect to $N(0, I_p)$ can then be written as $f(x) = C^{-1}(2\pi)^{p/2}g(x)$, where

$$ g(x) = \exp\left(-V(x) + \frac{1}{2}\|x\|^2\right), \quad x \in \mathbb{R}^p. \quad (8) $$

If the potential $V$ in (7) has the form

$$ V(x) = a_1x^\top A x + a_2\eta^\top x + a_3, \quad (9) $$

where $a_1, a_2, a_3 \in \mathbb{R}$ are constants, $A \in \mathbb{R}^{p \times p}$ is a positive definite matrix, and $\eta \in \mathbb{R}^p$ is a vector, a closed-form expression of $b$ can be computed. Several common distributions are included as special cases: $\mu$ simplifies to a uniform distribution if $a_1 = a_2 = 0$ and its support is a bounded subset of $\mathbb{R}^p$; it is an exponential distribution if $a_1 = 0$; and it becomes a normal distribution for $a_1 > 0$. Also, when the target distribution $\mu$ is a finite mixture of distributions with potential function given by (9), the drift terms can be calculated explicitly. Using widely used Gaussian mixture models as an illustrative example, we derive the explicit form expression of the corresponding drift terms in the following subsection.

### B. Gaussian Mixture Distributions

Assume that the target distribution $\mu$ is a Gaussian mixture, i.e.,

$$ \mu = \sum_{i=1}^\kappa \theta_i N(\alpha_i, \Sigma_i) \sum_{i=1}^\kappa \theta_i = 1 \text{ and } 0 \leq \theta_i \leq 1, i = 1, \ldots, \kappa, \quad (10) $$

where $\kappa$ is the number of mixture components, $N(\alpha_i, \Sigma_i)$ is the $i$th Gaussian component with mean $\alpha_i \in \mathbb{R}^p$ and covariance matrix $\Sigma_i \in \mathbb{R}^{p \times p}$. Clearly, the target distribution $\mu$ in (10) is absolutely continuous with respect to the $p$-dimensional standard Gaussian distribution $N(0, I_p)$. The density ratio can be expressed as $f = \sum_{i=1}^\kappa \theta_i f_i$, where $f_i = \frac{\pi_i N(\alpha_i, \Sigma_i)}{\pi N(0, I_p)}$ represents the density ratio of $N(\alpha_i, \Sigma_i)$ relative to $N(0, I_p)$. The drift term of the Schrödinger-Föllmer SDE (4) is

$$ b(x, t) = \sum_{i=1}^\kappa \theta_i E_x \nabla f_i(x + \sqrt{1-t}Z) \sum_{i=1}^\kappa \theta_i E_Z f_i(x + \sqrt{1-t}Z), \quad Z \sim N(0, I_p). \quad (11) $$

To obtain the expression of the drift term $b(x, t)$ in (11), we only need to derive the expressions of $E_Z \nabla f_i(x + \sqrt{1-t}Z)$ and $E_Z f_i(x + \sqrt{1-t}Z)$, $i = 1, \ldots, \kappa$. Let $\Sigma_i(t) = (1-t)\Sigma_i^{-1}$, some tedious calculations show that

$$ E_Z \nabla f_i(x + \sqrt{1-t}Z) = \Sigma_i^{-1} \alpha_i + (I_p - \Sigma_i^{-1})[dI_p + \Sigma_i(t)]^{-1} \Sigma_i(t) \alpha_i + x \quad (12) $$

and

$$ E_Z f_i(x + \sqrt{1-t}Z) = \frac{g_i(x, t)}{t\Sigma_i + (1-t)I_p}^{1/2}, \quad (13) $$

where

$$ g_i(x, t) = \exp\left(\frac{1}{2} - \frac{1}{2t}\right) \left\|([dI_p + \Sigma_i(t)]^{-1/2} \Sigma_i(t) \alpha_i + x)\right\|_2^2 \times \exp\left(\frac{1}{2} - \frac{1}{2t}\right) \left\|x\right\|^2_2. $$

Thus, by substituting the expressions (12) and (13) into (11), we can obtain an analytical form for $b(x, t)$. See Appendix I for details.
C. SFS Based on Euler-Maruyama Discretization

Proposition 1 demonstrates that we can initiate from $X_0 = 0$ and update the values of $\{X_t : 0 < t \leq 1\}$ according to the Schrödinger-Föllmer SDE (4) in continuous time. This yields the desired distributional property, specifically that $X_t \sim \mu(t)$. Thus, to implement this sampling procedure computationally, it is sufficient to discretize the continuous process. We employ the Euler-Maruyama discretization for the SDE (4) with a fixed step size. Let

$$t_k = k \cdot s, \quad k = 0, 1, \ldots, K,$$

and set $Y_{t_0} = 0$. Then the Euler-Maruyama discretization of (4) has the form

$$Y_{t_{k+1}} = Y_{t_k} + sb(Y_{t_k}, t_k) + \sqrt{s} \epsilon_{k+1}, \quad k = 0, 1, \ldots, K - 1,$$

where $\{\epsilon_k\}_{k=1}^K$ are i.i.d. random vectors from $N(0, I_p)$ and

$$b(Y_{t_{k}}, t_k) = \frac{E_Z[\nabla f(Y_{t_k} + \sqrt{1-t_k} Z)]}{E_Z[f(Y_{t_k} + \sqrt{1-t_k} Z)]}, \quad Z \sim N(0, I_p).$$

The main computational task involved in updating the Euler-Maruyama discretization (14) is to compute the drift term $b$ defined in (15). The following points are worth noting.

(a) Recall the Radon-Nikodym derivative $f(x) = \frac{d\nu}{d\lambda}(x)$. The normalization constant of $\mu$ cancels out from the numerator and denominator of the drift term $b$. Consequently, the SFS is capable of sampling from unnormalized distributions.

(b) In general, calculating the drift term $b$ analytically is intractable when the target distribution $\mu$ exhibits a complex structure. Additionally, this calculation involves the derivative $\nabla f$, which may have a complicated form, and can be challenging to compute.

Since $g$ in (8) is proportional to $f$ up to a multiplicative constant independent of $x$, we can represent $b$ in terms of $g$ as

$$b(Y_{t_k}, t_k) = \frac{E_Z[\nabla g(Y_{t_k} + \sqrt{1-t_k} Z)]}{E_Z[g(Y_{t_k} + \sqrt{1-t_k} Z)]},$$

which eliminates any unknown constants. The pseudocode for implementing (14) is provided in Algorithm 1.

Algorithm 1 SFS $\mu = \exp(-V(x))/C$

1: Input: $V(x), K$. Initialize $s = 1/K$, $Y_{t_0} = 0$.
2: for $k = 0, 1, \ldots, K - 1$ do
3: Sample $\epsilon_k \sim N(0, I_p)$,
4: Compute the drift term $b(Y_{t_k}, t_k)$ by (16),
5: Update $Y_{t_{k+1}} = Y_{t_k} + sb(Y_{t_k}, t_k) + \sqrt{s} \epsilon_{k+1}$,
6: end for
7: Output: $\{Y_{t_k}\}_{k=1}^K$.

In general, however, for non-Gaussian mixture distributions (10) discussed previously, the drift term $b$ lacks a closed-form expression. Fortunately, it can be approximated with any desired accuracy using Monte Carlo methods. Let $Z_1, \ldots, Z_m$ represent i.i.d. samples from $N(0, I_p)$, with $m \geq 1$ large enough to ensure precision. Based on (16), we can approximate $b$ by

$$\hat{b}_m(Y_{t_k}, t_k) = \frac{1}{m} \sum_{j=1}^m [\nabla g(Y_{t_k} + \sqrt{1-t_k} Z_j)] \cdot [g(Y_{t_k} + \sqrt{1-t_k} Z_j)], \quad k = 0, \ldots, K - 1.$$  

(17)

We observe that (17) is a biased estimator of the drift term $b$, although both the numerator and denominator are unbiased estimators. The Euler-Maruyama discretization (14) then becomes

$$\tilde{Y}_{t_{k+1}} = \tilde{Y}_{t_k} + \hat{b}_m(\tilde{Y}_{t_k}, t_k) + \sqrt{s} \epsilon_{k+1}, \quad k = 0, 1, \ldots, K - 1,$$

where $\{\epsilon_k\}_{k=1}^K$ are i.i.d. samples from $N(0, I_p)$. Algorithm 2 presents the pseudocode for implementing SFS.

Algorithm 2 SFS for $\mu = \exp(-V(x))/C$ With Monte Carlo Estimation of the Drift Term

1: Input: $V(x), m, K$. Initialize $s = 1/K$, $\tilde{Y}_{t_0} = 0$.
2: for $k = 0, 1, \ldots, K - 1$ do
3: Sample $\epsilon_k \sim N(0, I_p)$,
4: Compute $\hat{b}_m$ according to (17),
5: Update $\tilde{Y}_{t_{k+1}} = \tilde{Y}_{t_k} + \hat{b}_m(\tilde{Y}_{t_k}, t_k) + \sqrt{s} \epsilon_{k+1}$,
6: end for
7: Output: $\{\tilde{Y}_{t_k}\}_{k=1}^K$.

III. THEORETICAL PROPERTIES

In this section, we establish non-asymptotic bounds on the Wasserstein distance between the law of the samples generated by SFS and the target distribution, using either Algorithms 1 or 2. To this end, we assume that the drift term $b(x, t)$ is Lipschitz continuous in $t$, i.e.,

$$\|b(x, t) - b(y, s)\|_2 \leq C_1 \left(\|x - y\|_2 + |t - s|^{1/2}\right),$$

$$x, y \in \mathbb{R}^p \text{ and } t, s \in [0, 1],$$

(3C)

where $C_1 > 0$ is a finite constant. By setting $t = s$ in (3C), one sees that (3C) implies (2C).

Remark 2: Since $f(x) \propto g(x)$ as defined in (8), the Lipschitz continuity of $g$ and its gradient $\nabla g$, along with $g$ being bounded below by a strictly positive constant, imply conditions (C1) and (C3). For further details, please refer to Appendix D for details.

Let $\nu_1$ and $\nu_2$ be two probability measures defined on $(\mathbb{R}^p, B(\mathbb{R}^p))$, and let $D(\nu_1, \nu_2)$ represent the collection of couplings $\nu$ on $(\mathbb{R}^p, B(\mathbb{R}^p))$ whose first and second marginal distributions are $\nu_1$ and $\nu_2$, respectively. The second-order Wasserstein distance is then defined as

$$W_2(\nu_1, \nu_2) = \inf_{\nu \in D(\nu_1, \nu_2)} \left(\int_{\mathbb{R}^p} \int_{\mathbb{R}^p} \|\theta_1 - \theta_2\|^2 d\nu(\theta_1, \theta_2)\right)^{1/2}.$$
A. Error Bounds for SFS in Algorithm 1

Let $Y_{tk}$ represent the value of the last iteration in Algorithm 1, assuming that the exact values of the drift term $b$ can be computed.

**Theorem 1:** Under Conditions (C1) and (C3), we have

$$W_2(\text{Law}(Y_{tk}), \mu) = O(\sqrt{ps}),$$  

where $s = 1/K$ is the step size.

**Remark 3:** The error bound in (18) is non-asymptotic, meaning that it holds for any values of the dimension $p$ and step size $s$. The $O(1)$ factor in this bound depends only on the constants from Conditions (C1) and (C3). This dependence is demonstrated in the proof of Theorem 1 in the Appendix. Similar remarks apply to Theorems 2-3 presented below.

**Remark 4:** Theorem 1 provides analytical guarantees for the efficacy of Algorithm 1, indicating that it is well-suited to the constants from Conditions (C1) and (C3). This dependence is demonstrated in the proof of Theorem 1 in the Appendix. Similar remarks apply to Theorems 2-3 presented below.

**Remark 5:** The convergence rate $\sqrt{s}$ represents the optimal strong convergence rate for the Euler-Maruyama discretization method applied to solve the SDE [49], [50]. This rate depends only on the square root of the ambient dimension $p$, avoiding exponential dependency on $p$. In high-dimensional settings where $p$ is large, the error can be controlled by setting the step size $s = o(1/p)$, ensuring that the number of SFS iterations $K = 1/s$ depends on $p$ super-linearly rather than exponentially. Thus, SFS does not suffer from the curse of dimensionality.

B. Error Bounds for SFS in Algorithm 2

Algorithm 2 addresses cases where the exact values of the drift term $b$ cannot be computed, and only Monte Carlo approximations to $b$ are available. To establish non-asymptotic error bounds here, we further assume that the potential $U(x, t)$ is strongly convex in $x$, i.e., there exists a finite constant $M > 0$ such that

$$U(x, t) - U(y, t) - \nabla U(y, t)\dot{y} \geq (M/2) \|x - y\|^2,$$  

(C4)

holds for all $x, y \in \mathbb{R}^p$ and $t \in [0, 1]$. Without loss of generality, assume $M < C_1$, where $C_1$ is given in (C3). Condition (C4) aligns with Condition H2 of [51] and Assumption 3.2 of [52] as used in analyses of stochastic gradient Langevin dynamics. This strong convexity condition on the potential $V$ is commonly assumed in the convergence analysis of Langevin algorithms [7], [8], [18], [19], [20], [21], [22], which have established non-asymptotic error bounds in terms of Wasserstein distance, Kullback-Leibler divergence, and total variation distance. To proceed, we further introduce the following assumption:

**Assumption 1:** There exist constants $\gamma, \xi > 0$ such that $g, \nabla g$ are $\gamma$-Lipschitz continuous, and $g \geq \xi$.

**Theorem 2:** Assume (C1), (C3), (C4), and Assumption 1 hold. Then, for $s < 1/(2M + 1)$, the variance $\text{Var}(\tilde{b}_m(x, t))$ is bounded by

$$\text{Var}(\tilde{b}_m(x, t)) = O\left(\frac{p}{m}\right),$$

where $m$ is the number of samples used in Monte Carlo estimation (17).

**Remark 6:** Theorem 2 provides theoretical guarantees for Algorithm 2, including guidance on selecting $s$ and $m$. For convergence of the distribution of $Y_{tk}$, it is recommended to set the step size $s = o(1/p)$ and $m = p/o(1)$. In high-dimensional models with large $p$, a substantial number of random vectors are required from $N(0, I_p)$ to precisely estimate the drift term $b$.

**Remark 7:** Notably, the gradient estimator used in Algorithm 2 functions as a self-normalized importance sampling estimator, with its variance inherently linked to the variability of $g$ and the ambient space dimensionality. Specifically, for each $x \in \mathbb{R}^p$ and $t \in [0, 1]$, as defined in (17), we can express the estimator as

$$\tilde{b}_m(x, t) = \frac{1}{m} \sum_{j=1}^m \left[ g(x + \sqrt{1-t}Z_j) - g(x) - \nabla g(x)\sqrt{1-t}Z_j \right].$$

Under Assumption 1, it follows that

$$\text{Var}(\tilde{b}_m(x, t)) \leq \frac{2\mathbb{E}_Z \|\nabla g(x + \sqrt{1-t}Z)\|^2}{\xi^2} + \frac{2\mathbb{E}_Z \|\nabla g(x)\|^2}{\xi^2} = O\left(\frac{p\gamma^2}{\xi^2}\right).$$

Consequently, we observe that the variance of the estimator is proportional to the dimensionality $p$.

C. Regularization to Improve the Lower Bound on $f$

Theorem 1 is based on Assumptions (C1) and (C3), which hold if $g$ and $\nabla g$ are Lipschitz continuous and $g$ (or $f$) has a lower bound strictly greater than 0. Theorem 2 also requires that $g$ (or $f$) is bounded away from zero. However, this requirement does not hold if the target distribution has compact support. To fix this pity, we introduce a regularization on $\mu$ by mixing $\mu$ with $N(0, I_p)$, i.e., considering

$$\mu_\varepsilon = (1 - \varepsilon)\mu + \varepsilon N(0, I_p),$$

where $0 < \varepsilon < 1$. Since the density ratio of $\mu_\varepsilon$ over $N(0, I_p)$ is

$$f_\varepsilon = \frac{d\mu_\varepsilon}{dN(0, I_p)} = (1 - \varepsilon)f + \varepsilon,$$
it is easy to deduce that \( f_\varepsilon \geq \varepsilon > 0 \), and \( f_\varepsilon \) and \( \nabla f_\varepsilon \) are Lipschitz continuous as long as \( f \) and \( \nabla f \) are. Clearly, \( \mu_\varepsilon \) serves as a good approximation to \( \mu \) when \( \varepsilon \) is small. We can sample from \( \mu_\varepsilon \) using the Euler-Maruyama discretization of the SDE (4) with the drift term \( b(x,t) = \nabla \log Q_{1-\varepsilon} f_\varepsilon(x) \).

For a given \( \varepsilon \in (0,1) \), we denote the samples generated using Algorithm 1 and Algorithm 2 with \( \mu_\varepsilon \) as the target distribution by \( \{Y_{1,\varepsilon}(\varepsilon)\}_{K=0}^K \) and \( \{\tilde{Y}_{1,\varepsilon}(\varepsilon)\}_{K=0}^K \), respectively. We can then prove the following consistency results for both Algorithm 1 and Algorithm 2.

**Theorem 3:** Under Conditions (C1) and (C3), we have
\[
\lim_{K \to \infty, \varepsilon \to 0} W_2(\text{Law}(Y_{1,\varepsilon}(\varepsilon)), \mu) = 0.
\] (19)

Assume that (C1), (C3) and (C4) hold, and that \( g \) and \( \nabla g \) are Lipschitz continuous, we have
\[
\lim_{m,K \to \infty, \varepsilon \to 0} W_2(\text{Law}(\tilde{Y}_{1,\varepsilon}(\varepsilon)), \mu) = 0.
\] (20)

**Remark 8:** As in Theorem 1, the requirement on \( s = 1/K \) in both (19) and (20) is \( s = o(1/p) \). The requirement on \( m \) is similar to that in Theorem 2. As shown in the proof of Theorem 3 in the Appendix, to control the approximation error, we must choose \( \varepsilon = o(1/p) \).

### IV. Related Work

A considerable amount of research has focused on MCMC sampling algorithms derived from Langevin diffusion. The convergence properties of these Langevin sampling algorithms have been rigorously studied under three key assumptions: (a) the (strongly) convex potential assumption [7], [8], [18], [19], [20], [21], [22], (b) the dissipativity condition for the drift term [23], [24], [25], and (c) the local convexity condition for the potential function outside a ball [8], [26], [27], [28]. However, these conditions may not apply to models with multiple modes, such as Gaussian mixtures, where the potential functions are non-convex, and the Sobolev inequality may not hold. Additionally, the constant in the log-Sobolev inequality depends exponentially on the dimensionality [23], [30], [31], [32], indicating that the efficiency of Langevin samplers may be susceptible to the curse of dimensionality.

By contrast, SFS does not require the underlying Markov process to be ergodic. As a result, our findings in Theorem 1 do not depend on the conditions typically assumed for Langevin samplers; the error bounds rely solely on the square root of the ambient dimension. Notably, these convergence results are applicable to Gaussian mixtures, where the drift term of the Schrödinger-Föllmer diffusion can be computed analytically to optimize computational efficiency. In Theorem 2, however, where only a Monte Carlo approximation of the drift term is available, we assume strong convexity for the potential of the Schrödinger-Föllmer diffusion. We believe that the strong convexity condition on \( U(x,t) \) with respect to \( x \) is technically useful in the proofs but may not be essential for the efficacy of SFS. A recent study, expanding on our SFS framework, is presented in [53], addressing the challenge of achieving unbiased expectation estimates for a general probability measure that is absolutely continuous with respect to a standard Gaussian measure. This study employs multilevel Monte Carlo strategies to mitigate biases in the SFS approach for integral calculations.

The Schrödinger bridge has demonstrated complex connections with various domains, including statistical physics, optimal transport, and control theory [37]. Despite its theoretical potential, its application in statistical sampling remains limited in the literature. A Schrödinger bridge sampler was recently introduced in [54]. For a given distribution \( \mu \), the authors proposed iterative adjustments to the transition kernels of a reference Markov chain to produce a process whose terminal marginal distribution approximates \( \mu \). Another recent work explored the Schrödinger bridge problem in settings where only samples of the initial and target distributions are available [55], proposing an iterative procedure that uses constrained maximum likelihood estimation and importance sampling to estimate functions that solve the Schrödinger system. The algorithms in these studies draw on iterative proportional fitting or the Sinkhorn algorithm [56], [57]. In [48], [58], and [59], researchers addressed the problem of learning a generative model based on the Schrödinger-Föllmer diffusion with an unknown drift term, using deep neural networks to estimate the drift. The frameworks and methodologies in these previous works differ substantially from the present study. Specifically, our approach leverages the Schrödinger-Föllmer diffusion (2) as a direct and efficient tool for sampling from unnormalized distributions, distinguishing it from other approaches discussed here.

### V. Numerical Studies

In this work, we conducted a series of numerical experiments to evaluate the effectiveness of the SFS method. Our experiments included sampling from some one-dimensional and two-dimensional Gaussian mixture distributions, and Bayesian logistic regression, and Bayesian ridge regression. The SFS method is implemented in both R and Python, and code is available at https://github.com/Liao-Xu/SFS_R and https://github.com/Liao-Xu/SFS_py, respectively.

We compared SFS with several well-established sampling algorithms: the Metropolis-Hastings (MH) algorithm [1], [2], [60], [61], Hamiltonian Monte Carlo (HMC) [62], [63], Stochastic Gradient Hamiltonian Monte Carlo (SGHMC) [64], Unadjusted Langevin Algorithm (ULA) [7], [8], [18], [19], [21], Stochastic Gradient Langevin Dynamics (SGLD) [65], [66], [67], cyclical Stochastic Gradient Langevin Dynamics (cSGLD) [68], No U-Turn Sampler (NUTS) [69], [70], Haario Bardenet Adaptive Metropolis MCMC (ACMC) [71], [72], Sequential Monte Carlo (SMC) [73], [74], and Parallel Tempering (PT) [75], [76].

In our experiments, we use the R package mcmc [3], [15] for the MH algorithm and the R packages sde [77] and yuima [78] for the ULA. Furthermore, for our numerical studies, we employ the code from [64] for HMC and SGHMC, and the code from [68] for SGLD and cSGLD. For the implementation of NUTS and ACMC, we utilize the Python library PINTS [79]. In the context of Bayesian logistic regression, we use the Python library PyMC [80] for NUTS and HMC, the Python
library pyro [81] for SGLD. Moreover, we use the Python library PyMC [80] for SMC and the Python package ptemcee [75], [76] for PT.

A. One-Dimensional Gaussian Mixture Distribution

We first examined three one-dimensional Gaussian mixture distributions,

\[ f_1(x) = 0.5 \ N(x; -2, 0.5^2) + 0.5 \ N(x; 2, 0.5^2), \]  
\[ f_2(x) = 0.5N(x; -3, 0.5^2) + 0.5 \ N(x; 5, 0.5^2), \]  
\[ f_3(x) = 0.7 \ N(x; -9, 0.5^2) + 0.3N(x; 7, 0.5^2). \]  

From (21) to (23), the variance remains constant, but the centroids between two components become increasingly distant. As the distribution becomes more asymmetric about the origin, sampling becomes more challenging from models (21) to (23). We use the proposed SFS and other methods to generate samples, setting the sample size \( N = 5,000 \), and grid hyperparameter \( K = 100 \) in Algorithm 1.

Figure 1 illustrates the kernel density estimates (KDEs) from each method with distinct colors and line types, and the target density is shaded in grey. In cases where the centroids of Gaussians are close and symmetric, the proposed SFS, MH, and SGHMC perform comparable, while samples from other methods collapse on one mode as shown in Figure 1(a). In the case that the centroids of Gaussians move apart from each other and the target distribution becomes asymmetric, only samples from SFS can accurately represent the underlying target distribution while all other methods collapse on one mode, as shown in Figure 1(b) and (c).

B. Two-Dimensional Gaussian Mixture Distribution

In this section, we examine two-dimensional Gaussian mixture distributions,

\[ \alpha_i = \lambda_1 (\sin(2(i - 1)\pi/\kappa), \cos(2(i - 1)\pi/\kappa)), \]  
\[ \kappa = 4, 8, 16, \lambda_1 = 2, 4, 6, i = 1, \cdots, \kappa. \]  

For each setting, we specify the proportions \( \theta_i = 1/\kappa \) and the covariance matrices \( \Sigma_i = 0.03 \cdot I_p \), gradually increasing the distance between the centroids of Gaussian components. Additional results for more complex settings are available in the appendix. We set \( N = 20,000 \), \( K = 100 \) for Algorithm 1. We investigate the performance gap between Algorithms 1 and 2, which represents analytical solution and Monte Carlo approximation respectively. We set \( K = 100 \) and \( m = 100 \) in Algorithm 2. The centroids of the Gaussian components in model (24) form a circle pattern. We applied the proposed SFS alongside other methods to generate samples and visualized the kernel density estimation in Figure 2.

As shown in Figure 2, both Algorithm 1 and Algorithm 2 generate robust and effective samples that closely approximate the target distribution. Notably, only SFS effectively captures the underlying target distribution, while other methods converge on a limited number of modes as sampling complexity increases. In model (24), although samples from MH, SGLD, and SGHMC can approximate the density in simpler scenarios (the first row of Figure 2), these methods fail when the target distribution includes more components.

Overall, the proposed SFS consistently surpasses other algorithms, particularly in complex and challenging models. Moreover, while other methods suffer from mode collapsing in sophisticated models, SFS maintains its performance. This outcome indicates that SFS is an effective approach compared to methods such as MH, ULA, SGLD, SGHMC, cSGLD, NUTS, and ACMC, as illustrated through the visualization of two-dimensional Gaussian sampling.

C. SFS Scalability Demonstrated in High-Dimensional Gaussian Mixtures

We investigate the scalability of the SFS in the high-dimensional Gaussian mixture settings, varying the dimensionality varying from 5 to 1,000. For clarity, the Gaussian mixture means are arranged evenly around a circle in the \( p \)-dimensional space, with angles defined as \( \omega_i = \frac{2\pi i}{\kappa} \) for \( i = 0, 1, \ldots, \kappa - 1 \). Means \( \alpha \) are generated as:

\[ \alpha_{2j} = r \cdot \cos(\omega_i), \quad \alpha_{2j+1} = r \cdot \sin(\omega_i), \quad j = 0, 1, \ldots, \frac{p}{2}, \]  

where \( r = 2 \). Each Gaussian component has a covariance matrix \( \Sigma_i = 0.03 \cdot I_p \), and equal mixing coefficients \( \theta_i = \frac{1}{\kappa} \) are employed.

To evaluate performance, we compare SFS and its implementation in Algorithm 2 with advanced sampling methods such as Hamiltonian Monte Carlo (HMC), the No-U-Turn Sampler (NUTS), Sequential Monte Carlo (SMC), and Parallel Tempering (PT), for \( p = 20 \) and \( \kappa = 8 \). For intuitive visualization, KDE plots are generated along the first two dimensions, as shown in Figure 3. Results indicate that only SFS and SMC successfully capture all modes of the Gaussian mixture means, whereas the other methods experience mode collapse, even at 20 dimensions.

We further apply \( k \)-means clustering algorithm [82], [83], [84] to estimate the cluster proportions based on samples.
generated by each method, comparing the results to simulated ground truth proportions. Figure 4(a) presents the proportion errors across all methods, while Figure 4(b) displays the computational time costs. Both SFS and SFS implemented with Algorithm 2 outperform other methods in terms of the proportion estimates and computational time efficiency. Additionally, Algorithm 1 provides substantial computational gains due to its avoidance of Monte Carlo approximation, making it preferable for higher-dimensional models.

In a comparative analysis for \( p = 20 \), we find that only the SFS and SMC methods deliver satisfactory outcomes. Consequently, we extend the assessment to dimensions ranging from 5 to 200. Figure 5(a) illustrates that both SFS and SMC exhibit similar performance in terms of proportion errors. However, SFS maintains consistently low computational time across dimensions, while SMC shows a notable increase in computational time as the dimension \( p \) increases, particularly beyond \( p = 100 \), reaching approximately 0.5 seconds per sample at \( p = 200 \). We further extend SFS from 10 to 1,000, as depicted in Figure 5(b), showing that even at \( p = 1,000 \), SFS’s computational time remains below 1 minute, with proportion errors comparable to those observed at lower dimensions.

D. Bayesian Logistic Regression

Consider binary logistic regression with an independent and identically distributed sample \( \{x_i, y_i\}_{i=1}^n \), that is,

\[
P(y_i = 1|x_i) = \frac{\exp(x_i^\top \beta^*)}{1 + \exp(x_i^\top \beta^*)}, \quad i = 1, \ldots, n,
\]

where \( x_i \in \mathbb{R}^p \) is the covariate vector, \( y_i \in \{0, 1\} \) is the response variable and \( \beta^* = (\beta^*_1, \ldots, \beta^*_p)^\top \in \mathbb{R}^p \) is the vector of underlying regression coefficients. We generate samples for \( \beta^* \) from its posterior distribution. Following [18], [19], and [7], we set the prior distribution of \( \beta^* \) to be a Gaussian distribution with zero mean and covariance matrix \( \Sigma_{\beta^*} = (\sum_{i=1}^n x_i x_i^\top/n)^{-1} \), then the posterior distribution of \( \beta^* \) is expressed as \( \mu(\beta^*) = \exp\left( \sum_{i=1}^n (y_i x_i^\top \beta^* - \log(1 + \exp(x_i^\top \beta^*))) - \beta^*^\top \Sigma_{\beta^*}^{-1} \beta^*/2 \right) \).

The covariates vector \( \{x_i\}_{i=1}^n \) are i.i.d. from \( N(0, \Sigma) \), where \( \Sigma_{i,j} = 0.5^{|i-j|} \) for \( 1 \leq i, j \leq p \). Similar to [18], [19], and [7], the simulation setting is based on the binary logistic regression with \( n = 500, p \) varying from 2 to 10. In Algorithm 2, we set \( K = 200 \) and \( m = 1,000 \). Using the proposed SFS, along with MH, ULA, HMC, NUTS, SGLD, and SMC methods, we draw a random sample with a size \( N = 1,000 \). In terms of estimation accuracy, SFS consistently achieves the lowest
As shown in Figure 7(a), SFS exhibits superior performance in estimating coefficients across dimensions ranging from \( p = 2 \) to \( p = 20 \). Additionally, Figure 7(b) compares the computational time costs, with SFS ranking as the second most efficient algorithm, balancing accuracy and efficiency effectively.

**VI. Conclusion**

We introduce the SFS method for sampling from distributions using the Euler-Maruyama discretization of Schrödinger-Föllmer diffusion defined over the unit time interval \([0, 1]\).

A key advantage of SFS is that it does not require the underlying Markov process to be ergodic to sample effectively from the target distribution, distinguishing it from other MCMC samplers that rely on ergodicity. We establish non-asymptotic error bounds for the sampling distribution of the SFS in the Wasserstein distance under appropriate conditions. Specifically, when the drift term can be evaluated analytically, smoothness conditions on the target distribution are sufficient to ensure error bounds for samples generated in Algorithm 1. For cases where the drift term lacks an analytical expression, we propose using Monte Carlo estimation in Algorithm 2 and demonstrate the consistency of this approach. Our numerical experiments show that SFS can broaden the applicability of existing samplers, particularly for multi-modal distributions that may lack normalization. Thus, the proposed SFS represents a valuable addition to the suite of methods available for sampling from potentially unnormalized distributions.

Several directions warrant further exploration. For instance, while we establish convergence for samples from Algorithm 2 under a strong convexity condition on the potential, it would be beneficial to investigate the possibility of relaxing or even eliminating this requirement—a technically challenging but intriguing problem.

Additionally, applying SFS in Bayesian inference with more complex data structure that demand higher precision in estimation presents a promising area for future work.

**Appendix**

**Proofs**

In this section, we first provide some background on the Schrödinger-Föllmer diffusion, then we prove Proposition 1, Remark 2 and Theorems 1, 2 and 3. We note that Proposition 1 is a known result (see, e.g., [39], [40], [48]). We include a proof here for ease of reference.

**A. Background on Schrödinger-Föllmer Diffusion**

Let \( \Omega = C([0, 1], \mathbb{R}^p) \) denote the space of \( \mathbb{R}^p \)-valued continuous functions over the time interval \([0, 1]\). Define \( Z = (Z_t)_{t \in [0, 1]} \) as the canonical process on \( \Omega \), where \( Z_t(\omega) = \omega_t \) and \( \omega = (\omega_s)_{s \in [0, 1]} \in \Omega \). The canonical \( \sigma \)-field on \( \Omega \) is then generated as \( \mathcal{F} = \sigma(Z_t, t \in [0, 1]) = \{ \{ \omega : (Z_t(\omega))_{t \in [0, 1]} \in H \} : H \in \mathcal{B}(\mathbb{R}^p) \} \). Denote \( \mathcal{P}(\Omega) \) as the space of probability measures on the path space \( \Omega \), and \( \mathcal{W}_\infty \in \mathcal{P}(\Omega) \) as the Wiener measure whose initial marginal distribution is \( \delta_0 \). The law of reversible Brownian motion is then described by \( \mathcal{P} = \int \mathcal{W}_\infty \mathsf{d}x \), an unbounded measure on \( \Omega \). Note that \( \mathcal{P} \) has a marginal distribution aligning with the...
Lebesgue measure $\mathcal{L}$ at each $t$. [42] addressed the problem of identifying the most probable random evolution between two probability distributions $\tilde{\nu}, \tilde{\mu} \in \mathcal{P}(\mathbb{R}^p)$. This problem is known as the Schrödinger bridge problem (SBP). SBP can be reformulated as finding a probability law on the path space that interpolates between $\tilde{\nu}$ and $\tilde{\mu}$ and is close, in relative entropy, to the prior law of Brownian diffusion [37], [43]; specifically, it seeks a path measure $Q^* \in \mathcal{P}(\Omega)$ with marginal $Q^*_t = (Z_t)_{#} Q^* = Q^* \circ Z_{t}^{-1}$, $t \in [0, 1]$ such that

$$Q^* \in \arg \min D_{\text{KL}}(Q\|P),$$

and

$$Q_0 = \tilde{\nu}, Q_1 = \tilde{\mu},$$

where the relative entropy $D_{\text{KL}}(Q\|P) = \int \log \left(\frac{dQ}{dP}\right) dQ$ if $Q \ll P$ (i.e. $Q$ is absolutely continuous w.r.t. $P$), and $D_{\text{KL}}(Q\|P) = \infty$ otherwise. The following theorem characterizes the solution to SBP.

**Theorem 4**: [37] If $\tilde{\nu}, \tilde{\mu} \ll \mathcal{L}$, then SBP admits a unique solution $Q^* = f^*(Z_1)_{#} P$, where $f^*$ and $g^*$ are $\mathcal{L}$-measurable nonnegative functions satisfying the Schrödinger system

$$\begin{align*}
E \left[ f^*(Z_1) | Z_0 = x \right] &= \frac{dg}{dx}(x), \quad \mathcal{L} - \text{a.e.} \\
E \left[ g^*(Z_1) | Z_1 = y \right] &= \frac{df}{dy}(y), \quad \mathcal{L} - \text{a.e.}
\end{align*}$$

Furthermore, the pair $(Q^*_t, v^*_t)$ with

$$v^*_t(x) = \nabla_x \log E_\nu [g^*(Z_1) | Z_t = x]$$

solves the minimum action problem

$$\min_{\mu_t, v_t} \int_0^1 E_{\mu_t, v_t} \left[ ||v_t(z)||^2 \right] dt$$

s.t.

$$\begin{align*}
\partial_t \mu_t &= -\nabla \cdot (\mu_t v_t) + \Delta \mu_t, \quad \text{on } (0, 1) \times \mathbb{R}^p \\
\mu_0 &= \tilde{\nu}, \mu_1 = \tilde{\mu}.
\end{align*}$$

Let $K(s, x, t, y) = [2\pi(t-s)]^{-p/2} \exp\left(-\frac{||x-y||^2}{2(t-s)}\right)$ be the transition density of the Wiener process, $\tilde{q}(x)$ and $\tilde{p}(y)$ be the density of $\tilde{\nu}$ and $\tilde{\mu}$, respectively. Denote

$$\begin{align*}
f_0(x) &= f^*(x), \quad g_1(y) = g^*(y), \\
f_1(y) &= E_\nu [f^*(Z_1) | Z_1 = y] = \int K(0, x, 1, y)f_0(x)dx, \\
g_0(x) &= E_\nu [g^*(Z_1) | Z_0 = x] = \int K(0, x, 1, y)g_1(y)dy.
\end{align*}$$

Then, the Schrödinger system in Theorem 4 can also be characterized by

$$\begin{align*}
\tilde{q}(x) &= f_0(x)g_0(x), \quad \tilde{p}(y) = f_1(y)g_1(y)
\end{align*}$$

with the following forward and backward time harmonic equations [85].

$$\begin{align*}
\partial_t f_1(x) &= \frac{1}{2} \nabla f_1(x), \\
\partial_t g_1(x) &= -\frac{1}{2} \nabla g_1(x), \quad \text{on } (0, 1) \times \mathbb{R}^p.
\end{align*}$$

Let $q_t$ represent the marginal density of $Q^*_t$, i.e., $q_t(x) = \frac{dQ^*_t}{dx}(x)$, which can be expressed as the product of $q_t$ and $f_t$.

**Proof of Proposition 1**: It is well-known that the transition probability density of a standard $p$-dimensional Brownian motion is given by

$$\tilde{p}_{s,t}(x, y) = \frac{1}{(2\pi(t-s))^{p/2}} \exp\left(-\frac{1}{2(t-s)}||x-y||^2\right).$$

This implies that the diffusion process $\{X_t\}_{t \in [0, 1]}$ defined in (1) admits the transition probability density

$$p_{s,t}(x, y) = \tilde{p}_{s,t}(x, y) Q_{1-t}f(y) Q_{1-s}f(x).$$

It follows that for any measurable set $A \in \mathcal{B}(\mathbb{R}^p)$,

$$P(X_1 \in A) = \int_A p_{0,1}(0, y)dy = \int_A \tilde{p}_{0,1}(0, y) Q_{1}f(0)dy = \mu(A).$$

Therefore, $X_1$ is distributed as the probability distribution $\mu$. This completes the proof. \hfill $\Box$

**C. Drift Term $b(x, t)$ for Gaussian Mixture Distribution (10)**

We derive the expression of the drift term $b(x, t)$ in (10) for the Gaussian mixture distribution. **Proof**: For $Z \sim N(0, I_p)$, we have

$$E_Z f_S(x + \sqrt{1-t}Z) = E_{Y \sim N(x, (1-t)I_p)} f_S(Y) = \frac{1}{|\Sigma_t|^{1/2}} \exp \left( -\frac{1}{2} \left( Y - (Y - \alpha)\Sigma_t^{-1} (Y - \alpha) \right) \right)$$

$$f \exp \left( \frac{||y||^2}{2} - \frac{1}{2} ||y - \alpha||^2 \right) dy$$

$$= \frac{1}{(2\pi(1-t))^{p/2}|\Sigma_t|^{1/2}}$$

Authorized licensed use limited to the terms of the applicable license agreement with IEEE. Restrictions apply.
Moreover, since
\[
\begin{align*}
\gamma
\end{align*}
\]
exists a finite and positive constant
\[
R = \nabla\bigg|_{x,t} \sim Y \Sigma - i = 1^T \alpha_i - \frac{1}{2} \alpha_i \|x\|^2.
\]
Similarly, we have
\[
\begin{align*}
&= \Sigma_i^{-1} \alpha_i \exp \left( \frac{i y^2}{2} + \frac{(y - \alpha_i)^T \Sigma_i^{-1} (y - \alpha_i)}{2} \right) \frac{\exp \left( \frac{i y^2}{2} - \frac{(y - \alpha_i)^T \Sigma_i^{-1} (y - \alpha_i)}{2} \right)}{2} \\
&= \Sigma_i^{-1} \alpha_i \int \exp \left( \frac{i y^2}{2} - \frac{(y - \alpha_i)^T \Sigma_i^{-1} (y - \alpha_i)}{2} \right) d y.
\end{align*}
\]
Therefore, the analytical expression of \(b(x,t)\) in (11) is obtained by plugging the expressions (28) and (29) into (11).

\[\square\]

### D. Proof of Remark 2

**Proof:** Since \(g\) and \(\nabla g\) are Lipschitz continuous, there exists a finite and positive constant \(\gamma\) such that for all \(x, y \in \mathbb{R}^p\),
\[
|g(x) - g(y)| \leq \gamma \|x - y\|,
\]
and
\[
\|\nabla g(x) - \nabla g(y)\| \leq \gamma \|x - y\|.
\]

Moreover, since \(g\) has a lower bound greater than 0, there exists a finite and positive constant \(\xi\) such that
\[
g \geq \xi > 0.
\]

By (30) and (32), it yields that for all \(x \in \mathbb{R}^p\) and \(t \in [0,1]\),
\[
\|b(x,t)\|_2 = \frac{\|\nabla Q_{1-t}g(x)\|}{Q_{1-t}g(x)} \leq \frac{\gamma}{\xi}.
\]

Then, by (30)-(33), for all \(x, y \in \mathbb{R}^p\) and \(t \in [0,1]\),
\[
\begin{align*}
\|b(x,t) - b(y,t)\|_2 &= \left| \frac{\nabla Q_{1-t}g(x) - \nabla Q_{1-t}g(y)}{Q_{1-t}g(x) - Q_{1-t}g(y)} \right|_2 \\
&\leq \left| \frac{\nabla Q_{1-t}g(x) - \nabla Q_{1-t}g(y)}{Q_{1-t}g(x) - Q_{1-t}g(y)} \right|_2 + \|b(x,t)\|_2 \cdot \left| \frac{Q_{1-t}g(x) - Q_{1-t}g(y)}{Q_{1-t}g(x) - Q_{1-t}g(y)} \right| \\
&\leq \left( \frac{\gamma}{\xi} + \frac{\gamma^2}{\xi^2} \right) \|x - y\|_2.
\end{align*}
\]

Similarly, by (30)-(33), for all \(x, y \in \mathbb{R}^p\) and \(t \in [0,1]\),
\[
\begin{align*}
\|b(x,t) - b(x,s)\|_2 &= \left| \frac{\nabla Q_{1-t}g(x) - \nabla Q_{1-t}g(x)}{Q_{1-t}g(x) - Q_{1-t}g(x)} \right|_2 \\
&\leq \left( \frac{\gamma}{\xi} + \frac{\gamma^2}{\xi^2} \right) \|x - s\|_2.
\end{align*}
\]

### E. Preliminary Lemmas for Theorem 1

**First, we introduce Lemmas 1-2 in preparing for the proofs of Theorem 1.**

**Lemma 1:** Assume (C1) holds, then
\[
\mathbb{E}[\|X_t\|^2] \leq 2(C_0 + p) \exp(2C_0 t).
\]

**Proof:** By the definition of \(X_t\) in (4), we have \(\|X_t\|_2 \leq \int_0^t \|b(X_u,u)\|_2 du + \|B_t\|_2\). It follows that
\[
\begin{align*}
\|X_t\|^2 &\leq 2 \left( \int_0^t \|b(X_u,u)\|^2 du \right)^{1/2} + 2\|B_t\|_2 \\
&\leq 2 \int_0^t \|b(X_u,u)\|^2 du + 2\|B_t\|_2 \\
&\leq 2\int_0^t C_0 \|X_u\|^2 + 1 du + 2\|B_t\|_2,\end{align*}
\]
where the first inequality follows from the inequality \((a + b)^2 \leq 2a^2 + 2b^2\), the last inequality follows from condition (C1). Thus,
\[
\mathbb{E}[\|X_t\|^2] \leq 2C_0 \int_0^t C_0 \|X_u\|^2 + 1 du + 2\mathbb{E}[\|B_t\|_2^2] \\
\leq 2C_0 \int_0^t \mathbb{E}[\|X_u\|^2] du + 2(C_0 + p).
\]

By the Bellman-Gronwall inequality, we have
\[
\mathbb{E}[\|X_t\|^2] \leq 2(C_0 + p) \exp(2C_0 t).
\]

This completes the proof. \(\square\)

**Lemma 2:** Assume (C1) holds, then for any \(0 \leq t_1 \leq t_2 \leq 1\),
\[
\mathbb{E}[\|X_{t_2} - X_{t_1}\|^2] \leq 4C_0 \exp(2C_0 t_2 - t_1)^2 + 2C_0(2(t_2 - t_1))^2 + 2p(t_2 - t_1).
\]

Authorized licensed use limited to the terms of the applicable license agreement with IEEE. Restrictions apply.
Proof: By the definition of $X_t$ in (4), we have
$$
\|X_{t_2} - X_{t_1}\|_2 \leq \int_{t_1}^{t_2} \|b(X_u, u)\|_2 du + \|B_{t_2} - B_{t_1}\|_2.
$$
Therefore,
$$
\|X_{t_2} - X_{t_1}\|_2^2 \\
\leq 2 \left( \int_{t_1}^{t_2} \|b(X_u, u)\|_2^2 du \right)^2 + 2\|B_{t_2} - B_{t_1}\|_2^2 \\
\leq 2(t_2 - t_1) \int_{t_1}^{t_2} \|b(X_u, u)\|_2^2 du + 2\|B_{t_2} - B_{t_1}\|_2^2 \\
\leq 2(t_2 - t_1) \int_{t_1}^{t_2} C_0 \|X_u\|_2^2 + 1|du| + 2\|B_{t_2} - B_{t_1}\|_2^2,
$$
where the last inequality follows from Condition (C1). Hence,
$$
E\|X_{t_2} - X_{t_1}\|_2^2 \\
\leq 2(t_2 - t_1) \int_{t_1}^{t_2} C_0 (E\|X_u\|_2^2 + 1) du + 2E\|B_{t_2} - B_{t_1}\|_2^2 \\
\leq 4C_0 \exp(2C_0)(C_0 + \rho)(t_2 - t_1)^2 + 2C_0 (t_2 - t_1)^2 + 2\rho(t_2 - t_1),
$$
where the last inequality follows from Lemma 1. This completes the proof.

\[\square\]

F. Proof of Theorem 1

Proof: By the definition of $Y_{t_k}$ and $X_{t_k}$, we have
$$
\|Y_{t_k} - X_{t_k}\|_2^2 \\
\leq \|Y_{t_k-1} - X_{t_k-1}\|_2^2 + \left( \int_{t_{k-1}}^{t_k} \|b(X_u, u) - b(Y_{t_k-1}, t_{k-1})\|_2 du \right)^2 \\
+ 2\|Y_{t_k-1} - X_{t_k-1}\|_2 \left( \int_{t_{k-1}}^{t_k} \|b(X_u, u) - b(Y_{t_k-1}, t_{k-1})\|_2 du \right) \\
\leq \|Y_{t_k-1} - X_{t_k-1}\|_2^2 + 4C_1^2 \|Y_{t_k-1} - X_{t_k-1}\|_2 \left( \int_{t_{k-1}}^{t_k} \|b(X_u, u) - b(Y_{t_k-1}, t_{k-1})\|_2 du \right) \\
+ 8C_1^2 \|Y_{t_k-1} - X_{t_k-1}\|_2 \left( \int_{t_{k-1}}^{t_k} \|X_{t_k-1} - X_{t_k-1}\|_2^2 du + \|u - t_{k-1}\|_2 \right) \\
\leq \|Y_{t_k-1} - X_{t_k-1}\|_2^2 + 4C_1^2 \|Y_{t_k-1} - X_{t_k-1}\|_2 \left( \int_{t_{k-1}}^{t_k} \|X_{t_k-1} - X_{t_k-1}\|_2^2 du + \|u - t_{k-1}\|_2 \right)
$$
where the second inequality holds due to $2ab \leq a^2 + b^2$, the third inequality follows from Condition (C3). Then,
$$
E\|Y_{t_k} - X_{t_k}\|_2^2 \\
\leq (1 + s + 8C_1^2 (s^2 + s^3))E\|Y_{t_k-1} - X_{t_k-1}\|_2^2 \\
+ 8C_1^2 \left( \int_{t_{k-1}}^{t_k} \|X_{t_k-1} - X_{t_k-1}\|_2^2 du + 4C_1^2 (s^2 + s^3) \right) \\
\leq (1 + s + 8C_1^2 (s^2 + s^3))E\|Y_{t_k-1} - X_{t_k-1}\|_2^2 \\
+ h(s) + 4C_1^2 (s^2 + s^3),
$$
where $h(s) = 8C_1^2 (s^2 + s^3)4C_0(C_0 + p) \exp(2C_0)s^2 + 2C_0s^2 + 2ps$, and the last inequality (34) follows from Lemma 2. Owing to $Y_{t_0} = X_{t_0} = 0$, we can conclude that
$$
E\|Y_{t_K} - X_{t_K}\|_2^2 \leq \frac{(1 + s + 8C_1^2 (s^2 + s^3))K - 1}{s + 8C_1^2 (s^2 + s^3)} \left[ h(s) + 4C_1^2 (s^2 + s^3) \right] = O(ps).
$$
Therefore,
$$
W_2(\text{Law}(Y_{t_K}, \mu)) = O(\sqrt{ps}).
$$
This completes the proof.

G. Preliminary Lemmas for Theorems 2

First, we introduce Lemmas 3-6 in preparing for the proofs of Theorem 2.

Lemma 3: (Lemma 1 in [20] and Lemma 2 in [22]). Denote $\Delta_k = X_{t_k} - Y_{t_k}$ and $\omega_k = b(Y_{t_k}, t_k) - b(X_{t_k}, t_k)$ with $k = 0, 1, \ldots, K$. Assume Conditions (C2) and (C4) hold, and $s < 2/(C_1 + M) < 1$, then
$$
\|\Delta_k - s\omega_k\|_2 \leq \rho\|\Delta_k\|_2,
$$
where $\rho = 1 - sM \in (0, 1)$.

Proof: By (C2) and (C4), we have
$$
(x - y)^T(\nabla U(x, t) - \nabla U(y, t)) \\
\geq \frac{M C_1}{M + C_1} \|x - y\|_2^2 + \frac{1}{M + C_1} \|\nabla U(x, t) - \nabla U(y, t)\|_2^2
$$
for all $x, y \in \mathbb{R}^p$. Therefore, by some simple calculations, we can get
$$
\|\Delta_k - s\omega_k\|_2^2 \\
= \|\Delta_k\|_2^2 - 2s\Delta_k^T\omega_k + s^2 \|\omega_k\|_2^2 \\
\leq \|\Delta_k\|_2^2 - \frac{2sMC_1}{M + C_1} \|\Delta_k\|_2 - \frac{2s}{M + C_1} \|\omega_k\|_2^2 \\
+ \frac{1}{M + C_1} \|\omega_k\|_2^2 \\
= \left( 1 - \frac{2sMC_1}{M + C_1} \right) \|\Delta_k\|_2^2 + s \left( 1 - \frac{2}{M + C_1} \right) \|\omega_k\|_2^2.
$$
By (C4), we have $\|\omega_k\|_2 \geq M \|\Delta_k\|_2$. Due to $s \leq 2/(M + C_1)$, then it yields
$$
\|\Delta_k - s\omega_k\|_2 \leq (1 - sM)^2 \|\Delta_k\|_2^2.
$$
As $C_1 > M$, then $2/(M + C_1) < 1/M$. Thus, $\rho = 1 - sM \in (0, 1)$. This completes the proof.

Lemma 4: If Assumption 1 holds, then
$$
\sup_{x \in \mathbb{R}^p} \mathbb{E} \left[ \|b(x, t) - \tilde{b}_m(x, t)\|_2^2 \right] = O \left( \frac{P}{m} \right).
$$

Proof: Denote two independent sets of independent copies of $Z \sim N(0, I_p)$, that is, $Z = \{Z_1, \ldots, Z_m\}$ and $Z' = \{Z'_1, \ldots, Z'_m\}$. For notation convenience, we denote
$$
d = \mathbb{E}Z \nabla g(x + \sqrt{1 - \tilde{t}Z}) ,
$$
$$
d_m = \sum_{i=1}^{m} \nabla g(x + \sqrt{1 - \tilde{t}Z_i}) ,
$$
e = \mathbb{E}Z \nabla g(x + \sqrt{1 - \tilde{t}Z}) ,
$$
e_m = \sum_{i=1}^{m} \nabla g(x + \sqrt{1 - \tilde{t}Z_i}) ,
$$
d'_m = \sum_{i=1}^{m} \nabla g(x + \sqrt{1 - \tilde{t}Z'_i}) ,
$$
e'_m = \sum_{i=1}^{m} \nabla g(x + \sqrt{1 - \tilde{t}Z'_i}) .
$$

Authorized licensed use limited to the terms of the applicable license agreement with IEEE. Restrictions apply.
Due to \(d - d_m = \mathbb{E}[d'_m - d_m|Z]\), then \(\|d - d_m\|_2^2 \leq \mathbb{E}[\|d'_m - d_m\|_2^2|Z]\).

Under Assumption 1, we have
\[
\begin{align*}
\mathbb{E}\|d - d_m\|_2^2 & \leq \mathbb{E}\left[\mathbb{E}[\|d'_m - d_m\|_2^2|Z]\right] = \mathbb{E}\|d'_m - d_m\|_2^2 \\
& = \left(1 - t\right)^2 \mathbb{E}_{Z, Z'_1}\|g(x + \sqrt{1 - t}Z_1) - g(x + \sqrt{1 - t}Z'_1)\|_2^2 \\
& \leq \frac{(1 - t)^2}{m} \mathbb{E}_{Z, Z'_1}\|Z_1 - Z'_1\|_2^2 \\
& \leq \frac{2\gamma^2}{m}.
\end{align*}
\]
Similarly, we also have
\[
\begin{align*}
\mathbb{E}\|e - e_m\|_2^2 & \leq \mathbb{E}[\|e'_m - e_m\|_2^2 \\
& = \left(1 - t\right)^2 \mathbb{E}_{Z, Z'_1}\|g(x + \sqrt{1 - t}Z_1) - g(x + \sqrt{1 - t}Z'_1)\|_2^2 \\
& \leq \frac{(1 - t)^2}{m} \mathbb{E}_{Z, Z'_1}\|Z_1 - Z'_1\|_2^2 \\
& \leq \frac{2\gamma^2}{m}.
\end{align*}
\]
By (35) and (36), it follows that
\[
\begin{align*}
\sup_{x \in \mathbb{R}^p, t \in [0,1]} \mathbb{E}\|d - d_m\|_2^2 & \leq \frac{2\gamma^2}{m}, \\
\sup_{x \in \mathbb{R}^p, t \in [0,1]} \mathbb{E}\|e - e_m\|_2^2 & \leq \frac{2\gamma^2}{m}.
\end{align*}
\]
Using Assumption 1 again, through some simple calculations, it yields that
\[
\begin{align*}
\mathbb{E}\|b(x, t) - \tilde{b}_m(x, t)\|_2 & = \mathbb{E}[\|\tilde{b}_m(x, t) - e_m\|_2] \\
& \leq \|\tilde{b}_m\|_2 \|e - e_m\|_2 \\
& = \|\tilde{b}_m\|_2 \|e - e_m\|_2 \\
& \leq \frac{\|d\|_2 \|e\|_2}{\|\tilde{b}_m\|} + \frac{\|d - d_m\|_2}{\|\tilde{b}_m\|} \\
& \leq \frac{\gamma^2}{\|e\|} + \frac{\|d - d_m\|_2}{\|\tilde{b}_m\|}.
\end{align*}
\]
By (37)-(38), we have
\[
\begin{align*}
\sup_{x \in \mathbb{R}^p, t \in [0,1]} \mathbb{E}\|b(x, t) - \tilde{b}_m(x, t)\|_2 & = O\left(\frac{p}{m}\right).
\end{align*}
\]

**Lemma 5:** Assume that \(g\) is \(\gamma\)-Lipschitz continuous and there exists a constant \(\xi > 0\) such that \(g \geq \xi > 0\). Then, for \(k = 0, 1, \ldots, K\),
\[
\mathbb{E}\|\tilde{Y}_k\|_2^2 \leq \frac{6\gamma^2}{\xi^2} + 3p.
\]

**Proof:**
Define \(\Theta_{k,t} = \tilde{Y}_k + (t - t_k)\tilde{b}_m(\tilde{Y}_k, t_k)\) and \(\tilde{Y} = \Theta_{k+1,t} + B_t - B_{t_k}\), where \(t_k \leq t \leq t_{k+1}\) with \(k = 0, 1, \ldots, K - 1\).
Since \(g\) is \(\gamma\)-Lipschitz continuous and \(g \geq \xi > 0\), for all \(x \in \mathbb{R}^p\) and \(t \in [0,1]\),
\[
\begin{align*}
\|b(x, t)\|_2^2 & \leq \frac{\gamma^2}{\xi^2}, \\
\|\tilde{b}_m(x, t)\|_2^2 & \leq \frac{\gamma^2}{\xi^2}.
\end{align*}
\]
By (39), we have
\[
\begin{align*}
\|\Theta_{k,t}\|_2^2 &= \|\tilde{Y}_k\|_2^2 + (t - t_k)\|\tilde{b}_m(\tilde{Y}_k, t_k)\|_2^2 \\
& \leq (1 + s)\|\tilde{Y}_k\|_2^2 + \frac{(s + s^2)\gamma^2}{\xi^2}.
\end{align*}
\]
Furthermore, we have
\[
\begin{align*}
\mathbb{E}\|\tilde{Y}_k\|_2^2 & \leq (1 + s)\|\tilde{Y}_k\|_2^2 + \frac{(s + s^2)\gamma^2}{\xi^2} + sp.
\end{align*}
\]
Since \(\tilde{Y}_0 = 0\), by induction, we have
\[
\begin{align*}
\mathbb{E}\|\tilde{Y}_{k+1}\|_2^2 & \leq (1 + s)\mathbb{E}\|\tilde{Y}_k\|_2^2 + \frac{(s + s^2)\gamma^2}{\xi^2} + sp.
\end{align*}
\]

**H. Proof of Theorem 2**

**Proof:** Let \(\Delta_k = X_{t_k} - \tilde{Y}_{t_k}\). Then,
\[
\begin{align*}
\Delta_{k+1} &= \Delta_k + (X_{t_{k+1}} - X_{t_k}) - (\tilde{Y}_{t_k} + 1 - \tilde{Y}_{t_k}) \\
& = \Delta_k - s [\tilde{b}_m(\tilde{Y}_{t_k}, t_k) - \tilde{b}_m(\tilde{Y}_{t_k} + \Delta_k, t_k)] + \int_{t_k}^{t_{k+1}} [b(X_t, t) - \tilde{b}_m(X_t, t)] dt.
\end{align*}
\]
Combining Lemma 3 with the triangle inequality, we have
\[
\begin{align*}
\|\Delta_k - s [\tilde{b}_m(\tilde{Y}_{t_k}, t_k) - \tilde{b}_m(\tilde{Y}_{t_k} + \Delta_k, t_k)]\|_{L_2} & \leq \|\Delta_k - s [\tilde{b}(\tilde{Y}_{t_k}, t_k) - \tilde{b}(\tilde{Y}_{t_k} + \Delta_k, t_k)]\|_{L_2}.
\end{align*}
\]

This completes the proof.  \(\square\)
Moreover,\[ \| \Delta_k - s \left[ b_m(\bar{Y}_k, t) - \bar{b}_m(\bar{Y}_k, t) \right] \|_{L_2} \leq s \rho \| \Delta_k \|_{L_2} + s \cdot O \left( \sqrt{\frac{p}{m}} \right). \] (40)

Therefore, by Lemma 6,\[ \| \Delta_k + s \left[ b_m(\bar{Y}_k, t) - \bar{b}_m(\bar{Y}_k, t) \right] \|_{L_2} \leq \rho \| \Delta_k \|_{L_2} + s \cdot O \left( \sqrt{\frac{p}{m}} \right). \]

From the definition of $\rho$ in Lemma 3, we can get\[ \| \Delta_{k+1} \|_{L_2} \leq \rho^{k+1} \| \Delta_0 \|_{L_2} + O(\sqrt{sp}) + O \left( \sqrt{\frac{p}{m}} \right). \]

Furthermore, since we set $X_{t_0} = \bar{Y}_{t_0} = 0$, we have\[ W_2(\text{Law}(\bar{Y}_k), \mu) \leq \rho^K \| \Delta_0 \|_{L_2} + O(\sqrt{sp}) + O \left( \sqrt{\frac{p}{m}} \right). \]

\[ \square \]

1. Proof of Theorem 3

Proof: By the triangle inequality, we have\[ W_2(\text{Law}(Y_{tK}(\varepsilon)), \mu) \leq W_2(\mu, \mu_\varepsilon) + W_2(\text{Law}(Y_{tK}(\varepsilon)), \mu_\varepsilon) \]

and\[ W_2(\text{Law}(\bar{Y}_{tK}(\varepsilon)), \mu) \leq W_2(\mu, \mu_\varepsilon) + W_2(\text{Law}(\bar{Y}_{tK}(\varepsilon)), \mu_\varepsilon). \]

First, we show that $W_2(\mu, \mu_\varepsilon)$ will converge to zero as $\varepsilon$ goes to zero. Let $Y \sim \mu$ and $Z \sim N(0, I_p)$, and let $\tau$ be a Bernoulli random variable with $P(\tau = 1) = 1 - \varepsilon$ and $P(\tau = 0) = \varepsilon$. Assume that $Y$, $Z$, and $\tau$ are mutually independent. Then $(Y, (1 - \tau)Z + \tau Y)$ is a coupling of $(\mu, \mu_\varepsilon)$. Denote the joint distribution of $(Y, (1 - \tau)Z + \tau Y)$ by $\pi$. Then, we have\[ \int_{\mathbb{R}^p \times \mathbb{R}^p} \| x - y \|_2^2 d\pi = \mathbb{E} \| Y - ((1 - \tau)Z + \tau Y) \|_2^2. \]

Therefore, it follows that\[ \lim \limits_{\varepsilon \to 0} W_2(\mu, \mu_\varepsilon) = 0. \] (42)

Similar to the proof of Theorems 1-2, we have\[ \lim \limits_{K \to \infty} W_2(\text{Law}(Y_{tK}(\varepsilon)), \mu_\varepsilon) = 0, \] (43)

\[ \lim \limits_{m, K \to \infty} W_2(\text{Law}(\bar{Y}_{tK}(\varepsilon)), \mu_\varepsilon) = 0. \] (44)

Combining (42) with (43) and (44), we have\[ \lim \limits_{K \to \infty} W_2(\text{Law}(Y_{tK}(\varepsilon)), \mu) = 0, \]

\[ \lim \limits_{m, K \to \infty} W_2(\text{Law}(\bar{Y}_{tK}(\varepsilon)), \mu) = 0. \]

This completes the proof. \[ \square \]
Gaussian mixture distributions, between the neighboring modes gradually increases. From the first row to the second row. The distance between the neighboring modes gradually increases from the top row to the bottom row.

**Fig. 8.** KDE with marginal distribution plots for the 16-mode Gaussian mixture distributions simulation. The distance between the neighboring modes gradually increases from the top row to the bottom row.

**Fig. 9.** KDE with marginal distribution plots for the 25-mode Gaussian mixture distributions. From the first row to the second row. The distance between the neighboring modes gradually increases.

**ADDITIONAL EXPERIMENTS**

*J. Two-Dimensional Gaussian Mixture Distribution*

In this part, we consider three sets of two-dimensional Gaussian mixture distributions,

\[
\alpha = (\lambda_2 (-3, -1, 1, 3))^\top \times (\lambda_2 (-3, -1, 1, 3)),
\]

\[
\kappa = 16, \lambda_3 = 1, 1.5, 2, i = 1, \ldots, \kappa,
\]

\[
\alpha = (\lambda_3 (-2, -1, 0, 1, 2))^\top \times (\lambda_3 (-2, -1, 0, 1, 2)),
\]

\[
\kappa = 25, \lambda_4 = 2, 3, i = 1, \ldots, \kappa,
\]

\[
\alpha = (\lambda_4 (-3, -2, -1, 0, 1, 2, 3))^\top \times (\lambda_4 (-3, -2, -1, 0, 1, 2, 3)),
\]

\[
\kappa = 49, \lambda_4 = 2, 3, i = 1, \ldots, \kappa.
\]

(45) \quad (46) \quad (47)

We set the proportions \( \theta_i = 1/\kappa \) and the covariance matrices \( \Sigma_i = 0.03 \cdot I_2 \) across these settings. The centroids of Gaussian components form a square matrix shape across these scenarios for each setting. We set \( N = 20,000 \) and \( K = 200 \) for these settings in Algorithm 1.

As illustrated in Figures 8 to 10, only the samples from SFS succeed in estimating the underlying target distribution while all other methods collapse on one or a few modes when the models become more difficult to sample from.

For models (45)-(47), when the centroids form a square matrix shape, only the MH algorithm can perform as well as SFS in simpler cases (the first row of Figures 8 to 10), while all the other methods collapse on one or few modes. When the models become more difficult to sample from, mode collapsing for other methods becomes even worse. In all the simulations, we observe that only samples generated via SFS can accurately produce a density estimation that matches the underlying mixture distribution.

In general, varying from models (45) to (47), the proposed method SFS consistently outperforms other algorithms, particularly in more complex and challenging models. Moreover, while other methods suffer from mode collapsing in sophisticated models, SFS maintains its performance. Therefore, we can draw the conclusion that SFS is an effective method in comparison to algorithms such as MH, ULA, SGLD, SGHMC, cSGLD, NUTS, and ACMC, as demonstrated through the visualization of two-dimensional Gaussian sampling.

**ACKNOWLEDGMENT**

The authors would like to thank the editor and the reviewers for their valuable feedback, which has improved this manuscript. The numerical calculations for this study were conducted with the Supercomputing Center, Wuhan University.

**REFERENCES**

[1] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller, “Equation of state calculations by fast computing machines,” *J. Chem. Phys.*, vol. 21, no. 6, pp. 1087–1092, Jun. 1953.

[2] W. K. Hastings, “Monte Carlo sampling methods using Markov chains and their applications,” *Biometrika*, vol. 57, no. 1, pp. 97–109, 1970.

[3] L. Tierney, “Markov chains for exploring posterior distributions,” *Ann. Statist.*, vol. 22, no. 4, pp. 1701–1728, Dec. 1994.

[4] S. Geman and D. Geman, “Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-6, no. 6, pp. 721–741, Nov. 1984.

[5] A. E. Gelfand and A. F. M. Smith, “Sampling-based approaches to calculating marginal densities,” *J. Amer. Stat. Assoc.*, vol. 85, no. 410, pp. 398–409, Jun. 1990.

[6] G. O. Roberts and R. L. Tweedie, “Exponential convergence of Langevin distributions and their discrete approximations,” *Bernoulli*, vol. 2, no. 4, pp. 341–363, Dec. 1996.

[7] A. S. Dalalyan, “Theoretical guarantees for approximate sampling from smooth and log-concave densities,” *J. Roy. Stat. Soc., Ser. B, Stat. Methodol.*, vol. 79, no. 3, pp. 651–676, Jun. 2017.

[8] A. Durmus and É. Moulines, “Nonasymptotic convergence analysis for the unadjusted Langevin algorithm,” *Ann. Appl. Probab.*, vol. 27, no. 3, pp. 1551–1587, Jun. 2017.

[9] E. A. J. F. Peters and G. de With, “Rejection-free Monte Carlo sampling for general potentials,” *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 85, no. 2, Feb. 2012, Art. no. 026703.

[10] A. Bouchard-Côté, S. J. Vollmer, and A. Doucet, “The bouncy particle sampler: A nonreversible rejection-free Markov chain Monte Carlo method,” *J. Amer. Stat. Assoc.*, vol. 113, no. 522, pp. 855–867, Apr. 2018.

[11] J. Bierkens, P. Fearnhead, and G. Roberts, “The zig-zag process and super-efficient sampling for Bayesian analysis of big data,” *Ann. Statist.*, vol. 47, no. 3, pp. 1288–1320, Jun. 2019.

[12] G. M. Martin, D. T. Frazier, and C. P. Robert, “Computing bayes: Bayesian computation from 1763 to the 21st century,” 2020, arXiv:2004.06425.

[13] W. Changye and C. P. Robert, “Markov chain Monte Carlo algorithms for Bayesian computation, a survey and some generalisation,” in *Case Studies in Applied Bayesian Data Science*. Cham, Switzerland: Springer, 2020, pp. 89–119.

[14] D. B. Dunson and J. E. Johndrow, “The Hastings algorithm at fifty,” *Biometrika*, vol. 107, no. 1, pp. 1–23, Mar. 2020.

[15] S. Brooks, A. Gelman, G. Jones, and X.-L. Meng, *Handbook Markov Chain Monte Carlo*. Boca Raton, FL, USA: CRC Press, 2011.

[16] D. Bakry, P. Cattiaux, and A. Guillin, “Rate of convergence for ergodic continuous Markov processes: Lyapunov versus Poincaré,” *Interdiscip. Top.*, vol. 2012, no. 522, pp. 855–867, 2012.

[17] P. Cattiaux and A. Guillin, “Trends to equilibrium in total variation distance,” *Annales De L’Institut Henri Poincaré, Probabilités Et Statistiques*, vol. 45, no. 1, pp. 117–145, Feb. 2009.
A. S. Dalalyan and É. Moulines, “High-dimensional Bayesian inference via the unadjusted Langevin algorithm,” Bernoulli, vol. 25, no. 4A, pp. 2854–2882, Nov. 2019.

A. Durmus and E. Moulines, “Sampling from a strongly log-concave distribution with the unadjusted Langevin algorithm,” 2016, arXiv:1605.01539.

A. Dalalyan, “Further and stronger analogy between sampling and optimization: Langevin Monte Carlo and gradient descent,” in Proc. Conf. Learn. Theory, 2017, pp. 678–689.

X. Cheng and P. Bartlett, “Convergence of Langevin MCMC in KL-divergence,” in Proc. Mach. Learn. Res., Algorithmic Learn. Theory, vol. 83, 2018, pp. 186–211.

A. S. Dalalyan and A. Karagulyan, “User-friendly guarantees for the Langevin Monte Carlo with inaccurate gradient,” Stochastic Processes Appl., vol. 129, no. 12, pp. 5278–5311, Dec. 2019.

M. Raginsky, A. Rakhlin, and M. Telgarsky, “Non-convex learning via stochastic gradient Langevin dynamics: A nonasymptotic analysis,” in Proc. Conf. Learn. Theory, 2017, pp. 1674–1703.

W. Mou, N. Flammarion, M. J. Wainwright, and P. L. Bartlett, “Improved bounds for discretization of Langevin diffusions: Near-optimal rates without convexity,” Bernoulli, vol. 28, no. 3, pp. 1577–1601, Aug. 2022.

Y. Zhang, Ō. D. Akyildiz, T. Damoulas, and S. Sabanis, “Nonasymptotic estimates for stochastic gradient Langevin dynamics under local conditions in nonconvex optimization,” Appl. Math. Optim., vol. 87, no. 2, p. 25, Apr. 2023.

X. Cheng, N. S. Chatterji, Y. Abbasi-Yadkori, P. L. Bartlett, and M. I. Jordan, “Sharp convergence rates for Langevin dynamics in the nonconvex setting,” 2018, arXiv:1805.01648.

Y.-A. Ma, Y. Chen, C. Jin, N. Flammarion, and M. I. Jordan, “Sampling can be faster than optimization,” Proc. Nat. Acad. Sci. USA, vol. 116, no. 42, pp. 20881–20885, Oct. 2019.

N. Bou-Rabee, A. Eberle, and R. Zimmer, “Coupling and convergence for Hamiltonian Monte Carlo,” Ann. Appl. Probab., vol. 30, no. 3, pp. 1209–1250, Jun. 2020.

O. Mangoubi, N. S. Pillai, and A. Smith, “Does Hamiltonian Monte Carlo mix faster than a random walk on multimodal densities?” 2018, arXiv:1808.03230.

G. Menez and A. Schlichting, “Poincaré and logarithmic sobolev inequalities by decomposition of the energy landscape,” Ann. Probab., vol. 42, no. 5, pp. 1809–1884, Sep. 2014.

F.-Y. Wang, “Log-sobolev inequalities: Different roles of ric and Hess,” Ann. Probab., vol. 37, no. 4, pp. 1587–1604, Jul. 2009.

J. K. Hale, Asymptotic Behavior of Dissipative Systems, Providence, RI, USA: American Mathematical Soc., 2010.

S. Chib and E. Greenberg, “Understanding the metropolis-hastings algorithm,” J. Amer. Statist. Assoc., vol. 90, no. 431, pp. 909–920, 1995.

M. Welling and Y. W. Teh, “Bayesian learning via stochastic gradient Langevin dynamics,” 2011, arXiv:1101.0089.

C. P. Robert and G. Casella, “The metropolis-hastings algorithm,” in Handbook of Statistics, vol. 199, 2019.

S. Duane, A. D. Kennedy, B. J. Pendleton, and D. Roweth, “Hybrid Monte Carlo,” Phys. Lett. B, vol. 195, pp. 216–222, Sep. 1987.

R. Neal, “MCMC using Hamiltonian dynamics,” in Handbook of Markov Chain Monte Carlo, vol. 2. New York, NY, USA: Chapman & Hall, 2011, p. 2.

T. Chen, E. B. Fox, and C. Guestrin, “Stochastic gradient Hamiltonian Monte Carlo,” in Proc. Int. Conf. Mach. Learn. (ICML), vol. 32, 2014, pp. 1683–1691.

M. Welling and Y. W. Teh, “Bayesian learning via stochastic gradient Langevin dynamics,” in Proc. Int. Conf. Mach. Learn. (ICML), Jan. 2012, pp. 1591–1598.

S. Patterson and Y. W. Teh, “Stochastic gradient Riemannian Langevin dynamics on the probability simplex,” in Proc. Adv. Neural Inf. Process. Syst., Dec. 2013, pp. 3102–3110.

R. Zhang, C. Li, J. Zhang, C. Chen, and A. G. Wilson, “Cyclical stochastic gradient MCMC for Bayesian deep learning,” in Proc. Int. Conf. Learn. Represent., Jan. 2019, pp. 1–27.

Authorized licensed use limited to the terms of the applicable license agreement with IEEE. Restrictions apply.
[69] M. D. Hoffman and A. Gelman, “The no-U-turn sampler: Adaptively setting path lengths in Hamiltonian Monte Carlo,” J. Mach. Learn. Res., vol. 15, no. 1, pp. 1593–1623, 2014.

[70] M. Betancourt, “A conceptual introduction to Hamiltonian Monte Carlo,” 2017, arXiv:1701.02434.

[71] R. H. Johnstone et al., “Uncertainty and variability in models of the cardiac action potential: Can we build trustworthy models?” J. Mol. Cellular Cardiol., vol. 96, pp. 49–62, Jul. 2016.

[72] H. Haario, E. Saksman, and J. Tamminen, “An adaptive Metropolis algorithm,” Bernoulli, vol. 7, no. 2, pp. 223–242, 2001.

[73] S. Minson, M. Simons, and J. Beck, “Bayesian inversion for finite fault earthquake source models 1-theory and algorithm,” Geophys. J. Int., vol. 194, no. 3, pp. 1701–1726, Sep. 2013.

[74] J. Ching and Y.-C. Chen, “Transitional Markov chain Monte Carlo method for Bayesian model averaging,” J. Eng. Mech., vol. 133, no. 7, pp. 816–832, Jul. 2007.

[75] W. D. Vousden, W. M. Farr, and I. Mandel, “Dynamic temperature selection for parallel tempering in Markov chain Monte Carlo simulations,” Monthly Notices Roy. Astronomical Soc., vol. 455, no. 2, pp. 1919–1937, Nov. 2016.

[76] D. Foreman-Mackey, D. W. Hogg, D. Lang, and J. Goodman, “Emcee: The MCMC hammer,” Publications Astronomical Soc. Pacific, vol. 125, no. 925, p. 306, Mar. 2013.

[77] S. M. Iacus, Simulation and Inference for Stochastic Differential Equations: With R Examples. Cham, Switzerland: Springer, 2009.

[78] S. M. Iacus and N. Yoshida, Simulation and Inference for Stochastic Processes with YUIMA. Cham, Switzerland: Springer, 2018.

[79] M. Clerx et al., “Probabilistic inference on noisy time series (PINTS),” Simulation and Inference for Stochastic Differential Equations with YUIMA. Cham, Switzerland: Springer, 2018.

[80] O. Abril-Pla et al., “PyMC: A modern, and comprehensive probabilistic programming framework in Python,” PeerJ Comput. Sci., vol. 9, Sep. 2023, Art. no. e1516.

[81] E. Bingham et al., “Pyro: Deep universal probabilistic programming,” J. Mach. Learn. Res., vol. 20, no. 28, pp. 1–6, 2019.

[82] J. MacQueen et al., “Some methods for classification and analysis of multivariate observations,” in Proc. 5th Berkeley Symp. Math. Statist. Probab., vol. 1, pp. 23–45, 1967.

[83] E. W. Forgy, “Cluster analysis of multivariate data: Efficiency versus interpretability of classifications,” Biometrics, vol. 21, pp. 768–769, Sep. 1965.

[84] S. Lloyd, “Least squares quantization in PCM,” IEEE Trans. Inf. Theory, vol. IT-28, no. 2, pp. 129–137, Mar. 1982.

[85] Y. Chen, T. T. Georgiou, and M. Pavon, “Stochastic control liaisons: Richard sinkhorn meets gapped Monge on a Schrodinger bridge,” SIAM Rev., vol. 63, no. 2, pp. 249–313, Jan. 2021.

**Jian Huang** received the B.Sc. degree in mathematics from Wuhan University, Wuhan, China, in 1985, and the Ph.D. degree in statistics from the University of Washington, Seattle, USA, in 1994. He is currently a Chair Professor in applied statistics and financial mathematics with the Department of Applied Mathematics, The Hong Kong Polytechnic University, Hong Kong, China. He has authored or co-authored over 200 publications, including The Annals of Statistics, Journal of the American Statistical Association, The American Journal of Human Genetics, PNAS, Econometrica, Statistical Science, and Journal of Machine Learning Research. His current research interests include semiparametric models, statistical genetics, analysis of high-dimensional data, machine learning, and statistical computing.

Dr. Huang was an Associate Editor of The Annals of Statistics and Statistical Sinica. He was a fellow of the American Statistical Association (2009) and a Clarivate Analytics Highly Cited Researcher (2015, 2016, 2017, and 2018).

**Yuling Jiao** received the B.Sc. degree in applied mathematics from Shangqiu Normal University, Shangqiu, China, in 2008, and the Ph.D. degree in applied mathematics from Wuhan University, Wuhan, China, in 2014. He is currently a Full Professor with the School of Artificial Intelligence, Wuhan University. He has authored or co-authored over 50 research articles, including SIAM Journal on Numerical Analysis, SIAM Journal on Scientific Computing, Journal of Machine Learning Research, IEEE TRANSACTIONS ON SIGNAL PROCESSING, and IEEE SIGNAL PROCESSING LETTERS. His current research interests include signal/image processing, inverse problem, statistical computing, and deep learning.

**Lican Kang** received the B.Sc. degree in applied mathematics from Jiangxi Normal University, Nanchang, China, in 2015, and the Ph.D. degree in statistics from Wuhan University, Wuhan, China, in 2021. He is currently an Incoming Assistant Professor at the Institute for Math and AI, Wuhan University, Wuhan, China. He has authored over ten research papers published in journals such as the SIAM Journal on Control and Optimization, Journal of Machine Learning Research, IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS, and INFORMATION SCIENCES. His current research endeavors are primarily focused on machine learning and statistical computing.

**Xu Liao** received the B.Sc. degree from the Department of Statistics, Xi’an Jiaotong University, Xi’an, China, in 2019, and the Ph.D. degree in biostatistics and health data science from the Duke-NUS Medical School, National University of Singapore, Singapore, in 2024. She has published in journals, including Nature Communications, Nucleic Acids Research, IEEE TRANSACTIONS ON INFORMATION THEORY, and IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS. Her current research interests include statistical genomics, trajectory inference, dimension reduction, and deep learning.

**Jin Liu** received the Graduate degree from the School of Electronics and Information, Dalian University of Technology, in 2004, and the M.S. and Ph.D. degrees in statistics from the Department of Statistics and Actuarial Science, University of Iowa, in 2007 and 2011, respectively. He is currently an Associate Professor with the School of Data Science, The Chinese University of Hong Kong, Shenzhen. He has authored or co-authored over 50 research articles, including Nature Communications, Nucleic Acids Research, and Journal of the American Statistical Association. His current research interests include statistical genetics/genomics, bioinformatics, and machine learning. His research interests include solving practical problems for single-cell/spatial omics, developing MR methods to draw causal inference with practical considerations.

**Yanyan Liu** received the B.Sc. and Ph.D. degrees in mathematics from Wuhan University, Wuhan, China, in 1989 and 2001, respectively. She is currently a Full Professor with the School of Mathematics and Statistics, Wuhan University. She has authored or co-authored more than 50 research articles, including Biometrics, Bioinformatics, Statistica Sinica, Lifetime Data Analysis, Journal of Nonparametric Statistics, Biometrical Journal, Scandinavian Journal of Statistics, Computational Statistics & Data Analysis, and Journal of Machine Learning Research. Her current research interests include statistical analysis for high-dimensional data, machine learning, statistical computing, distributed algorithm, and model average for large scale data.