ON SCALABLE AND FAST LANGEVIN-DYNAMICS-BASED SAMPLING ALGORITHMS

A Dissertation
Presented to
The Academic Faculty

By

Ruilin Li

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in the
School of Mathematics

Georgia Institute of Technology

May 2021

ON SCALABLE AND FAST LANGEVIN-DYNAMICS-BASED SAMPLING ALGORITHMS

Thesis committee:

Prof. Hongyuan Zha, Advisor
School of Data Science
The Chinese University of Hongkong, Shenzhen

Prof. Haomin Zhou, Co-Advisor
School of Mathematics
Georgia Institute of Technology

Prof. Molei Tao
School of Mathematics
Georgia Institute of Technology

Prof. Xiaojing Ye
Department of Mathematics and Statistics
Georgia State University

Prof. Cheng Mao
School of Mathematics
Georgia Institute of Technology

Date approved: April 26th

TO MY FAMILY

ACKNOWLEDGMENTS

This thesis could not have been successfully completed without the invaluable assistance of many individualss. The following list of acknowledgements is, by no means, exhaustive.

First and foremost, I would like to express my sincere gratitude to my advisor Prof. Hongyuan Zha. I am very lucky to have him as my supervisor. In the past five years, he led me into the beautiful world of machine learning and applied mathematics. His support, encouragement, and open-minded attitude continues to inspire me to pursue my own interest in research and in life.

Many thanks go to my co-advisor Prof. Haomin Zhou for his meticulous care which make my five-year Ph.D career productive, rich and colorful. I would like to express my gratitude to Prof. Molei Tao, for his hands-on advising and many rewarding discussion, all of which inspire my interest in sampling and dynamics. I am grateful to Prof. Xiaojing Ye, for his valuable guidance and help in the early years of my Ph.D study. I would like to appreciate Prof. Cheng Mao for accepting to serve on my dissertation committee and many insightful suggestions.

Great persons and colleagues I have had the pleasure to meet in these years, and that includes my fellow graduate students Fan Zhou, Hao Wu, Haodong Sun, Haoyan Zhai, Jaewoo Jung, Jiangning Chen, Jiaqi Yang, Qianli Hu, Qingqing Liu, Renyi Chen, Rundong Du, Shaojun Ma, Shijie Xie, Shu Liu, Weiwei Zhang, Xiao Liu, Xin Wang, Xin Xing, Yan Wang, Yanxi Hou, Yian Yao and more. I miss those helpful discussions and exciting moments.

Last but not least, my deepest gratitude goes to my parents and my wife for their unconditional support and love. They are the source of my strength and happiness. I dedicate this thesis to them.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

SUMMARY

SUMMARY

Langevin dynamics-based sampling algorithms are arguably among the most widely-used Markov Chain Monte Carlo (MCMC) methods. Two main directions of the modern study of MCMC methods are (i) How to scale MCMC methods to big data applications, and (ii) Tight convergence analysis of MCMC algorithms, with explicit dependence on various characteristics of target distribution, in a non-asymptotic manner.

This thesis continues the previous efforts in this two lines and consists of three parts. In the first part, we study stochastic gradient MCMC methods for large scale application. We propose a non-uniform subsampling of gradients scheme to approximately match the transition kernel of a base MCMC base with full gradient, aiming for better sample quality. The demonstration is based on underdamped Langevin dynamics.

In the second part, we consider an analog of Nesterov's accelerated algorithm in optimization for sampling. We derive a dynamics termed Hessian-Free-High-Resolution (HFHR) dynamics, from a high-resolution ordinary differential equation description of the Nesterov's accelerated algorithm. We then quantify the acceleration of HFHR over underdamped Langevin dynamics at both continuous dynamics level and discrete algorithm level.

In the third part, we study a broad family of bounded, contractive-SDE-based sampling algorithms via mean-square analysis. We show how to extend the applicability of classical mean-square analysis from finite time to infinite time. Iteration complexity in 2-Wasserstein distance is also characterized and when applied to Langevin Monte Carlo algorithm, we obtain an improved iteration complexity bound.

CHAPTER 1

INTRODUCTION

Sampling is an important problem in science and engineering, it arises naturally in Bayesian statistics [1, 2, 3], statistical physics [4], molecule dynamics [5], machine learning [6, 7], and computational biology [8]. Except for rare cases, direct sampling is difficult, if not impossible, and people often resort to approximate sampling, a standard approach of which is Markov Chain Monte Carlo (MCMC) methods [9]. By constructing a Markov chain that has the desired distribution as the invariant/equilibrium distribution of the Markov chain, one can obtain a sample of the desired distribution by recording states from the chain.

A classical example of the family of MCMC is Langevin dynamics, named after the famous French Physicist Paul Langevin. Langevin dynamics utilizes gradient information to guide a sampler to explore parameter spaces efficiently. There are two types of Langevin dynamics, one is called overdamped Langevin dynamics (OLD), and the other is called kinetic Langevin dynamics (abbreviated as ULD to comply with a convention of calling it underdamped Langevin dynamics). Suppose we we would like to sample from a Gibbs target distribution μ whose probability density function (w.r.t. Lebesgue measure in $\mathbb{R}^d$) is proportional to $\exp\big(-f(\boldsymbol{q})\big)$ where $f : \mathbb{R}^d \mapsto \mathbb{R}$ is a potential function, then OLD and ULD are respectively given by

$$\text{(OLD)} \qquad d\boldsymbol{q}_t = -\nabla f(\boldsymbol{q}_t)dt + \sqrt{2}d\boldsymbol{W}_t \tag{1.1}$$

$$\text{(ULD)} \quad \begin{cases} d\boldsymbol{q}_t = \boldsymbol{p}_t dt \\ d\boldsymbol{p}_t = -\gamma \boldsymbol{p}_t dt - \nabla f(\boldsymbol{q}_t)dt + \sqrt{2\gamma}d\boldsymbol{B}_t \end{cases} \tag{1.2}$$

where $\boldsymbol{q}_t \in \mathbb{R}^d$ is a position variable, $\boldsymbol{p}_t \in \mathbb{R}^d$ is a momentum variable, $\boldsymbol{W}_t, \boldsymbol{B}_t$ are i.i.d. Wiener processes in $\mathbb{R}^d$,$\gamma > 0$ is a friction coefficient. Under mild conditions [10] , OLD

converges to μ and ULD converges to

$$d\pi(\boldsymbol{q},\boldsymbol{p}) = d\mu(\boldsymbol{q})\nu(\boldsymbol{p})d\boldsymbol{p}, \text{ where } \nu(\boldsymbol{p}) = (2\pi)^{-\frac{d}{2}} \exp\left(-\frac{\|\boldsymbol{p}\|^2}{2}\right) \tag{1.3}$$

hence the $\boldsymbol{q}$ marginal in ULD follows the target distribution. Overdamped Langevin dynamics and underdamped Langevin dynamics are closely related, in fact, OLD is the overdamping limit of ULD as $\gamma \to \infty$ [11, 10], hence the name.

The following numerical algorithm for discretized overdamped Langevin dynamics is commonly known as Langevin Monte Carlo (LMC)/ unadjusted Langevin algorithm (ULA) [12]

$$\boldsymbol{q}_{k+1} = \boldsymbol{q}_k - \nabla f(\boldsymbol{q}_k)h + \sqrt{2h}\boldsymbol{\xi}_{k+1}, \quad k = 0, 1, 2, \cdots \tag{1.4}$$

where $\{\boldsymbol{\xi}_k\}_{k=1,2,\cdots}$ are independent d-dimension standard Gaussian random vectors. Theoretical investigation of LMC dates back to 90s [12] and one of the recommendations made by the authors of [12] is to avoid use LMC, or at least use it very cautiously since the ergodicity of LMC is very sensitive to the choice of step size h. LMC can be a transient chain if h is badly chosen, even when the continuous OLD is geometrically ergodic. These findings have strongly influenced subsequent research as the ensuing studies essentially focused on Metropolis-adjusted version of LMC, known as Metropolis adjusted Langevin algorithm (MALA) [13, 14, 15, 16].

Over the last decade, enabled by the dramatic increase of computing power, big data application with millions of data and complex models requiring millions or billions of parameters e.g. deep learning, are not uncommon. During the same period of time, there has been a resurgence of the studies on Langevin dynamics and its variants, due to new findings and deepened understandings. Two main lines of research are

- **Scalability** Design variants of Langevin dyamics/LMC that can scale to large data sets [17, 18, 19, 20, 21, 22, 23, 24, 25].

- **Convergence Rate** Quantitatively characterize the convergence rate of LMC and its

variants and the dependence on various factors, particularly on dimension, in an non-asymptotic manner [26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37].

This thesis continues the effort in these two directions. In the scalability direction, motivated by the great success of utilizing stochastic gradient (SG) [38] in the field of optimization [39, 40], people have also started to combine stochastic gradient with Langevin dynamics and proposed stochastic gradient Langevin dynamics (SGLD) [17], stochastic gradient Hamiltonian Monte Carlo (SGHMC) [18] which is closed related to stochastic gradient underdamped Langevin dyanmics (SGULD) [1] and many stochastic gradient version of variants of Langevin dynamics [19, 20, 21]. However, directly replacing the batch/full gradient by a (uniform) stochastic one without additional mitigation generally causes a gradient-based MCMC method to sample from a statistical distribution different from the target and can hence undermine the performance of downstream applications such as Bayesian inference, because the transition kernel of the MCMC method gets corrupted by the noise of subsampled gradient. In this thesis, we present a state-dependent non-uniform SG-MCMC algorithm termed Exponentially Weighted Stochastic Gradients method (EWSG), the demonstration of which is based on underdamped Langevin dynamics. The approach is based on designing the transition kernel of a SG-MCMC method to approximate the transition kernel of a full-gradient-based MCMC method. This approximation leads to non-uniform (in fact, exponential) weights that aim at capturing the entire state-variable distribution of the full-gradient-based MCMC method. EWSG differs from Variance Reduction (VR) techniques as it focuses on the entire distribution instead of just the variance; nevertheless, its reduced local variance is also proved. EWSG can also be viewed as an extension of the importance sampling idea, successful for SG-based optimizations, to sampling tasks.

Along the line of convergence rate, we propose an accelerated-gradient-based MCMC method in this thesis. It relies on a modification of the Nesterov's accelerated gradient

[1] SGULD is the same as the well-known SGHMC with $\hat{B} = 0$, see eq. (13) and Sec. 3.3 in [18] for details.

method for strongly convex functions (NAG-SC): We reformulate NAG-SC as a Hessian-Free High-Resolution ordinary differential equation, lift its high-resolution coefficient to be a hyperparameter α, and then inject appropriate noise and discretize the resulting diffusion process. The obtained diffusion process admits underdamped Langevin dynamics as a special case (when $\alpha = 0$). Accelerated sampling enabled by the new hyperparameter is theoretically quantified. At continuous-time level, for log-concave/log-strongly-concave target measures, exponential convergence in χ^2 divergence/2-Wasserstein distance is proved, with a rate analogous to the state-of-the-art results of underdamped Langevin dynamics, plus an additional acceleration. At discrete algorithm level, a dedicated discretization algorithm is proposed to simulate the Hessian-Free High-Resolution stochastic differential equation in a cost-efficient manner. For log-strong-concave target measures, the proposed algorithm achieves $\widetilde{\mathcal{O}}(\frac{\sqrt{d}}{\epsilon})$ iteration complexity in 2-Wasserstein distance, same as underdamped Langevin dynamics, but with a reduced constant.

In the same vein, we study a family of general bounded, contractive-SDE-based sampling algorithms. For this broad group of algorithms, we revisit the classical mean-square analysis framework for numerical stochastic differential equation. Classical mean-square analysis has global error bound only for finite time, we manage to extend it to infinite time for the class of sampling algorithms. Based on the improved mean-square analysis, we further obtain a general $\widetilde{\mathcal{O}}\left(C^{\frac{1}{p_2-\frac{1}{2}}}\frac{1}{\epsilon^{\frac{1}{p_2-\frac{1}{2}}}}\right)$ iteration complexity in 2-Wasserstein distance for the family of algorithms where C is a constant containing various information of the underlying SDE, e.g. dimension d. The iteration complexity bound not only reveals the dependence on tolerance ϵ, but also, somewhat surprisingly, shows that the dependence on the parameters of the underlying SDE can also be affected by the order of local strong error. When applied to Langevin Monte Carlo algorithm, we obtain a $\widetilde{\mathcal{O}}\left(\frac{\sqrt{d}}{\epsilon}\right)$ iteration complexity in 2-Wasserstein distance, which improves upon the previously best known $\widetilde{\mathcal{O}}\left(\frac{d}{\epsilon}\right)$ result, under the standard smoothness and strong-convexity assumptions, plus an additional linear growth condition on the third-order derivative of a potential function.

The rest of the thesis is organized as follows:

In Chapter 2, we show the motivation and derivation of EWSG, followed by a rigorous non-asymptotic analysis on the global error of EWSG. We present extensive numerical experiments, not only to demonstrate EWSG's effectiveness, but also to guide hyperparameter choices, and validate our theoretical analysis.

In Chapter 3, we discuss Hessian-Free-High-Resolution dynamics for sampling, demonstrate its derivation, theoretical results and a numerical algorithm for the discretized dynamics. We show the acceleration of the proposed algorithm via extensive experiments on both simulated and real-world applications.

In Chapter 4, we present the mean-square analysis for bounded, contractive-SDE-based sampling algorithms, derive global error bound without finite time constraint and iterations complexity bound in 2-Wasserstein distance accordingly. An application to Langevin Monte Carlo algorithm yields improved iteration complexity over previously best known result.

CHAPTER 2

EXPONENTIAL WEIGHTED STOCHASTIC GRADIENT METHODS FOR IMPROVING SAMPLING ACCURACY

Many Markov Chain Monte Carlo (MCMC) methods use physics-inspired evolution such as Langevin dynamics [9] to utilize gradient information for exploring posterior distributions over continuous parameter space efficiently. However, gradient-based MCMC methods are often limited by the computational cost of evaluating the gradient on large data sets. Motivated by the great success of stochastic gradient methods for optimization, stochastic gradient MCMC methods (SG-MCMC) for sampling have also been gaining increasing attention. When the accurate but expensive-to-evaluate batch gradients in a MCMC method are replaced by computationally cheaper estimates based on a subset of the data, the method is turned to a stochastic gradient version. Classical examples include SG (overdamped) Langevin Dynamics (SGLD) [17] and SG Hamiltonian Monte Carlo (SGHMC) [18], both designed for scalability suitable for machine learning tasks.

However, directly replacing the batch gradient by a (uniform) stochastic one without additional mitigation generally causes a MCMC method to sample from a statistical distribution different from the target, because the transition kernel of the MCMC method gets corrupted by the noise of subsampled gradient. In general, the additional noise is tolerable if the learning rate/step size is tiny or decreasing. However, when large steps are used for better efficiency, the extra noise is non-negligible and undermines the performance of downstream applications such as Bayesian inference.

In this section, we present a state-dependent non-uniform SG-MCMC algorithm termed **E**xponentially **W**eighted **S**tochastic **G**radients method (EWSG), which continues the efforts of uniform SG-MCMC methods for better scalability. Our approach is based on designing the transition kernel of a SG-MCMC method to match the transition kernel of a full-

gradient-based MCMC method. This matching leads to non-uniform (in fact, exponential) weights that aim at capturing the entire state-variable distribution of the full-gradient-based MCMC method, rather than providing unbiased gradient estimator and reducing its variance. Nevertheless, if focusing on the variance, the advantage of EWSG is the following: recall the stochasticity of a SG-MCMC method can be decomposed into the *intrinsic* randomness of MCMC and the *extrinsic* randomness introduced by gradient subsampling; in conventional uniform subsampling treatments, the latter randomness is independent of the former, and thus when they are coupled together, variances add up; EWSG, on the other hand, dynamically chooses the weight of each datum according to the current state of the MCMC, and thus the variances do not add up due to dependence. However, the gained accuracy is beyond reduced variance, as EWSG, when converged, samples from a distribution close to the invariant distribution of the full-gradient MCMC method (which has no variance contributed by the extrinsic randomness), because its transition kernel (of the corresponding Markov process) is close to that of the full-gradient-MCMC method. This is how better sampling accuracy can be achieved.

The main demonstration of EWSG is based on underdamped Langevin dynamics, although it works for other MCMC methods too (e.g., Appendix A.2, A.3). To concentrate on the role of non-uniform SG weights, we will work with constant step sizes only. The fact that EWSG has locally reduced variance than its uniform counterpart is rigorously shown in Theorem 2. Furthermore, a global non-asymptotic error analysis is given in Theorem 3 to quantify the convergence and improved accuracy of EWSG, as well as to provide insights about hyperparameter choices.

Practically, the non-uniform gradient subsampling of EWSG is efficiently implemented via a Metropolis-Hastings chain over the data index. A number of experiments on synthetic and real world data sets, across downstream tasks including Bayesian logistic regression and Bayesian neural networks, are conducted to demonstrate the effectiveness of EWSG and validate our theoretical results, despite the approximation used in the implementation.

In addition to improved accuracy, the convergence speed was empirically observed, in a fair comparison setup based on the same data pass, to be comparable to its uniform counterpart when hyper-parameters are appropriately chosen. The convergence (per data pass) was also seen to be clearly faster than a classical Variance Reduction (VR) approach (note: for sampling, not optimization), and EWSG hence provides a useful alternative to VR. Additional theoretical study of EWSG convergence speed is provided in Appendix A.5.

2.1 Related Work

Stochastic Gradient MCMC Methods Since the seminal work of SGLD [17], much progress [19, 20] has been made in the field of SG-MCMC. [23] theoretically justified the convergence of SGLD and offered practical guidance on tuning step size. [24] introduced a preconditioner and improved stability of SGLD. We also refer to [41] and [42] which will be discussed in Section 2.7. While these work were mostly based on 1st-order (overdamped) Langevin, other dynamics were considered too. For instance, [18] proposed SGHMC, which is closely related to underdamped Langevin dynamics [43, 44], and [21] put it in a more general framework. underdamped Langevin dynamics was recently shown to be faster than the 1st-order version in appropriate setups [28, 30] and began to gain more attention.

Variance Reduction For **optimization**, vanilla SG methods usually find approximate solutions quickly but the convergence slows down when an accurate solution is needed [45, 46]. SAG [47] improved the convergence speed of stochastic gradient methods to linear, which is the same as gradient descent methods with full gradient, at the expense of large memory overhead. SVRG [46] successfully reduced this memory overhead. SAGA [48] furthers improved convergence speed over SAG and SVRG. For **sampling**, [49] applied VR techniques to SGLD (see also [50, 51]). However, many VR methods have large memory overhead and/or periodically use the whole data set for gradient estimation calibration, and

hence can be resource-demanding.

EWSG is derived based on matching transition kernels of MCMC and improves the accuracy of the entire distribution rather than just the variance. However, it does have a consequence of variance reduction and thus can be implicitly regarded as a VR method. When compared to the classic work on VR for SG-MCMC [49], EWSG converges faster when the same amount of data pass is used, although its sampling accuracy is below that of VR for Gaussian targets (but well above vanilla SG; see Section 2.7.1). In this sense, EWSG and VR suit different application domains: EWSG can replace vanilla SG for tasks in which the priority is speed and then accuracy, as it keeps the speed but improves the accuracy; on the other hand, VR remains to be the heavy weapon for accuracy-demanding scenarios. Importantly, EWSG, as a generic way to improve SG-MCMC methods, can be combined with VR too (e.g., Appendix A.3); thus, they are not exclusive or competitors.

Importance Sampling (IS) IS employs nonuniform weights to improve SG methods for **optimization**. Traditional IS uses fixed weights that do not change along iterations, and the weight computation requires prior information of gradient terms, e.g., Lipschitz constants of gradient [52, 53, 54], which are usually unknown or difficult to estimate. Adaptive IS was also proposed in which the importance was re-evaluated at each iteration, whose computation usually required the entire data set per iteration and may also require information like the upper bound of gradient [55, 56].

For **sampling**, it is not easy to combine IS with SG [42]; the same paper is, to our knowledge, the closest to this goal and will be compared with in Section 2.7.3. EWSG can be viewed as a way to combine (adaptive) IS with SG for efficient sampling. It require no oracle about the gradient, nor any evaluation over the full data set. Instead, an inner-loop Metropolis chain maintains a random index that approximates a state-dependent non-uniform distribution (i.e. the weights/importance).

Other Mini-batch MCMC Methods Besides SG-MCMC methods, there are also many non-gradient-based MCMC methods using only a subset of data in each iteration. For example, austerity MH [57] formulates Metropolis-Hastings step as a statistical hypothesis testing problem and proposes to use only a subset of data to make statistically significant accept/reject decision. Using a subsampled unbiased estimator of the likelihood in a pseudo-marginal framework to accelerate the Metropolis-Hastings algorithm is proposed in [58]. A notable **exact** MCMC method is FlyMC [41], which introduces an auxiliary binary random variable for each datum and only the subset of data whose corresponding auxiliary binary indicator "light" up, are used in iteration. Some more recent advances on exact MCMC methods include [59, 60]. We also refer to [58] for an excellent review on subsampling MCMC methods.

2.2 Background and Notation

Underdamped Langevin Dynamics (ULD) [1] is

$$\begin{cases} d\boldsymbol{\theta} &= \boldsymbol{r}dt \\ d\boldsymbol{r} &= -(\nabla f(\boldsymbol{\theta}) + \gamma \boldsymbol{r})dt + \sigma d\boldsymbol{W} \end{cases} \tag{2.1}$$

where $\boldsymbol{\theta}, \boldsymbol{r} \in \mathbb{R}^d$ are state and momentum variables, V is a potential energy function which in our context (originated from cost minimization or Bayesian inference over many data) is the sum of many terms $f(\boldsymbol{\theta}) = \sum_{i=1}^{n} f_i(\boldsymbol{\theta})$, γ is a friction coefficient, σ is intrinsic noise amplitude, and $\boldsymbol{W}$ is a standard d-dimensional Wiener process. Under mild assumptions on V, Langevin dynamics admits a unique invariant distribution $\pi(\boldsymbol{\theta}, \boldsymbol{r}) \sim \exp\left(-\frac{1}{T}(f(\boldsymbol{\theta}) + \frac{\|\boldsymbol{r}\|^2}{2})\right)$ [10] and is in many cases geometric ergodic. T is the temperature of system determined via the fluctuation dissipation theorem $\sigma^2 = 2\gamma T$ [61].

We consider ULD instead of the overdamped version mainly for two reasons: (i) one

[1] In the field of machine learning, it is customary to use certain letters such as $\boldsymbol{\theta}, \boldsymbol{w}$, etc. to denote the parameter of interest , we follow the convention and rewrite ULD in $(\boldsymbol{\theta}, \boldsymbol{r})$

may think ULD is more complicated, and we'd like to show it is still easy to equip it with EWSG (EWSG can work for many MCMC methods; Appendix A.2 has an overdamped version); (ii) ULD can converge faster than overdamped Langevin for instance in high-dimensions (e.g.,[28, 30, 62]). Like the overdamped version, numerical integrators for ULD with well captured statistical properties of the continuous process have been extensively investigated (e.g, [12, 63]), and both the overdamped and underdamped integrators are friendly to derivations that will allow us to obtain explicit expressions of the non-uniform weights.

Terminology-wise, ∇f will be called the full/batch-gradient, $n\nabla f_I$ with random I will be called stochastic gradient (SG), and when I is uniform distributed it will be called a uniform SG/subsampling, otherwise non-uniform. When uniform SG is used to approximate the batch-gradient in underdamped Langevin, the method will be referred to as (vanilla) stochastic gradient underdamped Langevin dynamics (SGULD/SGHMC[2]), and it serves as a baseline in experiments.

2.3 An Illustration of Non-optimality of Uniform Subsampling

Uniform subsampling of gradients have long been the dominant way of stochastic gradient approximations mainly because it is intuitive, unbiased and easy to implement.

However, uniform gradient subsampling can introduce large noise, and is sub-optimal even in the family of unbiased stochastic gradient estimator, as the following Theorem 1 will show. One intuition is, consider for example cases where data size n is larger than dimension d. In such cases, $\{\nabla f_i\}_{i=1,2,\cdots,n} \subset \mathbb{R}^d$ are linearly dependent and hence it is likely that there exist probability distributions $\{p_i\}_{i=1,2,\cdots,n}$ other than the uniform one such that the gradient estimate is unbiased, however with smaller variance because linearly dependent terms need not to be all used. This is a motivation for us to develop non-uniform subsampling schemes (weights may be $\boldsymbol{\theta}$ dependent), although we will not require $n > d$

[2]To be consistent with existing literature, we will refer SGULD as SGHMC in the sequel.

later.

Theorem 1 *Suppose given $\boldsymbol{\theta} \in \mathbb{R}^d$, the errors of SG approximation $\boldsymbol{b}_i = n\nabla f_i(\boldsymbol{\theta}) - \nabla f(\boldsymbol{\theta}), 1 \leq i \leq n$ are i.i.d. absolutely continuous random vectors with possibly-$\boldsymbol{\theta}$-dependent density $p(\cdot|\boldsymbol{\theta})$ and $n > d$. We call $\boldsymbol{p} \in \mathbb{R}^n$ a sparse vector if the number of non-zero entries in $\boldsymbol{p}$ is no greater than $d+1$, i.e. $\|\theta\|_0 \leq d+1$. Then with probability 1, the optimal probability distribution $\boldsymbol{p}^\star$ that is unbiased and minimizes the trace of the covariance of $n\nabla f_I(\boldsymbol{\theta})$, i.e. $\boldsymbol{p}^\star$ which solves the following, is a sparse vector.*

$$\min_{\boldsymbol{p}} \text{Tr}(\mathbb{E}_{I\sim\boldsymbol{p}}[\boldsymbol{b}_I \boldsymbol{b}_I^T]) \quad \textit{s.t. } \mathbb{E}_{I\sim\boldsymbol{p}}[\boldsymbol{b}_I] = \mathbf{0}, \tag{2.2}$$

Proof: Denote the set of all n-dimensional probability vectors by Σ^n, the set of sparse probability vectors by $\mathcal{S}$, and the set of non-sparse (dense) probability vectors by $\mathcal{D} = \Sigma^n \setminus \mathcal{S}$. Denote $B = [\boldsymbol{b}_1, \cdots, \boldsymbol{b}_n]$, then the optimization problem can be written as

$$\min \sum_{i=1}^n p_i \|\boldsymbol{b}_i\|^2$$
$$\text{s.t.} \begin{cases} B\boldsymbol{p} = \mathbf{0} \\ \boldsymbol{p}^T \mathbf{1}_n = 1 \\ p_i \geq 0, i = 1, 2, \cdots, n \end{cases}$$

Note that the feasible region is always non-empty (take $\boldsymbol{p}$ to be a uniform distribution) and is also closed and bounded, hence this linear programming is always solvable. Denote the set of all minimizers by $\mathcal{M}$. Note that $\mathcal{M}$ depends on $\boldsymbol{b}_1, \cdots, \boldsymbol{b}_n$ and is in this sense random.

The Lagrange function is

$$L(\boldsymbol{p}, \boldsymbol{\lambda}, \mu, \boldsymbol{\omega}) = \boldsymbol{p}^T \boldsymbol{s} - \boldsymbol{\lambda}^T B\boldsymbol{p} - \mu(\boldsymbol{p}^T \mathbf{1}_n) - \boldsymbol{\omega}^T \boldsymbol{p}$$

where $\boldsymbol{s} = [\|\boldsymbol{b}_1\|^2, \|\boldsymbol{b}_2\|^2, \cdots, \|\boldsymbol{b}_n\|^2]^T$ and $\boldsymbol{\lambda}, \mu, \boldsymbol{\omega}$ are dual variables. The optimality condition reads as

$$\frac{\partial L}{\partial \boldsymbol{p}} = \boldsymbol{s} - B^T\boldsymbol{\lambda} - \mu\mathbf{1}_n - \boldsymbol{\omega} = \mathbf{0}$$

Dual feasibilty and complementary slackness require

$$\omega_i \leq 0, i = 1, 2, \cdots, n$$
$$\boldsymbol{\omega}^T\boldsymbol{p} = 0$$

Consider the probability of the event {a dense probability vector can solve the above minimization problem}, i.e., $\mathbb{P}(\mathcal{M} \cap \mathcal{D} \neq \emptyset)$. It is upper bounded by

$$\mathbb{P}(\mathcal{M} \cap \mathcal{D} \neq \emptyset) \leq \mathbb{P}(\boldsymbol{p} \in \mathcal{D} \text{ and } \boldsymbol{p} \text{ solves KKT condition})$$

Since $\boldsymbol{p} \in \mathcal{D}$, complementary slackness implies that at least $d+2$ entries in $\boldsymbol{\omega}$ are zero. Denote the indices of these entries by $\mathcal{J}$. For every $j \in \mathcal{J}$, by optimality condition, we have $s_j - \boldsymbol{\lambda}^T\boldsymbol{b}_j - \mu = 0$, i.e.,

$$\|\boldsymbol{b}_j\|^2 - \boldsymbol{\lambda}^T\boldsymbol{b}_j - \mu = 0$$

Take the first $d+1$ indices in $\mathcal{J}$, and note a geometric fact that $d+1$ points in a d-dimensional space must be on the surface of a hypersphere of at most $d-1$ dimension, which we denote by $\mathcal{S} = S^{q-1} + \boldsymbol{x}$ for some vector $\boldsymbol{x}$ and integer $q \leq d$. Because b_i's

distribution is absolutely continuous, we have

$$\begin{aligned}
&\mathbb{P}(\boldsymbol{p} \in \mathcal{D} \text{ and } \boldsymbol{p} \text{ solves KKT condition}) \\
\leq &\mathbb{P}(\boldsymbol{p} \in \mathcal{D} \text{ and } \boldsymbol{b}_j \in S, \forall j \in \mathcal{J}) \\
\leq &\mathbb{P}(\boldsymbol{b}_j \in S, \forall j \in \mathcal{J}) \\
= &\mathbb{P}(\boldsymbol{b}_{j_k} \in S, k = d+2, \cdots, |\mathcal{J}|) \\
= &\prod_{k=d+2}^{|\mathcal{J}|} \mathbb{P}(\boldsymbol{b}_{j_k} \in S) \qquad \text{(independence)} \\
= &0 \qquad \text{(absolute continuous)}
\end{aligned}$$

Hence $\mathbb{P}(\mathcal{M} \cap \mathcal{D} \neq \emptyset) = 0$ and

$$\begin{aligned}
1 &= \mathbb{P}(\mathcal{M} \neq \emptyset) \\
&= \mathbb{P}((\mathcal{M} \cap \mathcal{S}) \cup (\mathcal{M} \cap \mathcal{D}) \neq \emptyset) \\
&\leq \mathbb{P}(\mathcal{M} \cap \mathcal{S} \neq \emptyset) + \mathbb{P}(\mathcal{M} \cap \mathcal{D} \neq \emptyset) \\
&= \mathbb{P}(\mathcal{M} \cap \mathcal{S} \neq \emptyset)
\end{aligned}$$

Therefore we have

$$\mathbb{P}(\mathcal{M} \cap \mathcal{S} \neq \emptyset) = 1$$

■

Despite the sparsity of $\boldsymbol{p}^\star$, which seemingly suggests one only needs at most $d+1$ gradient terms per iteration when using SG methods, it is not practical because $\boldsymbol{p}^\star$ requires solving the linear programming problem (2.2) in Theorem 1, for which an entire data pass is needed. Nevertheless, this result motivates us to seek alternatives to uniform SG. The EWSG method we will develop indeed has reduced local variance with high probability, and at the same time remain efficiently implementable without having to use all data per parameter update; however, its practical implementation can be biased, but a global error

analysis (Theorem 3) shows that trading bias for variance can still be worthy.

2.4 Derivation of Exponential Weighted Stochastic Gradient

MCMC methods are characterized by their transition kernels. In traditional SG-MCMC methods, uniform SG is used, which is independent of the intrinsic randomness of MCMC methods (e.g. diffusion in ULD), as a result, the transition kernel of SG-MCMC is quite different from that with full gradient. Therefore, it is natural to ask - is it possible to couple these two originally independent randomness so that the transition kernels can be better matched and the sampling accuracy can be hence improved?

Consider Euler-Maruyama (EM) discretization[3] of Equation (2.1):

$$\begin{cases} \boldsymbol{\theta}_{k+1} &= \boldsymbol{\theta}_k + \boldsymbol{r}_k h \\ \boldsymbol{r}_{k+1} &= \boldsymbol{r}_k - (\nabla f(\boldsymbol{\theta}_k) + \gamma \boldsymbol{r}_k)h + \sigma\sqrt{h}\boldsymbol{\xi}_{k+1} \end{cases} \tag{2.3}$$

where h is step size and $\boldsymbol{\xi}_{k+1}$'s are i.i.d. d-dimensional standard Gaussian random variables. Denote the transition kernel of EM discretization with full gradient by $P^{EM}(\boldsymbol{\theta}_{k+1}, \boldsymbol{r}_{k+1}|\boldsymbol{\theta}_k, \boldsymbol{r}_k)$. Then, replace $\nabla f(\boldsymbol{\theta}_k)$ by a weighted SG $n\nabla f_{I_k}(\boldsymbol{\theta}_k)$, where I_k is the index chosen to approximate full gradient and has p.m.f. $\mathbb{P}(I_k = i|\boldsymbol{\theta}_k, \boldsymbol{r}_k) = p_i$. Denote the new transition

[3]EM is not the most accurate or robust discretization, see e.g., [12, 63], but since it may still be the most used method, demonstrations here will be based on EM. The same idea of EWSG can easily apply to most other discretizations such as GLA [63].

kernel by $\tilde{P}^{EM}(\boldsymbol{\theta}_{k+1}, \boldsymbol{r}_{k+1}|\boldsymbol{\theta}_k, \boldsymbol{r}_k)$. We have the following decomposition

$$
\begin{aligned}
&P^{EM}(\boldsymbol{\theta}_{k+1}, \boldsymbol{r}_{k+1}|\boldsymbol{\theta}_k, \boldsymbol{r}_k) \\
=&1_{\{\boldsymbol{\theta}_k + \boldsymbol{r}_k h\}}(\boldsymbol{\theta}_{k+1}) \frac{1}{Z} \exp\left(-\frac{\|\boldsymbol{r}_{k+1} - \boldsymbol{r}_k + (\nabla f(\boldsymbol{\theta}_k) + \gamma \boldsymbol{r}_k)h\|^2}{2\sigma^2 h}\right) \\
=&1_{\{\boldsymbol{\theta}_k + \boldsymbol{r}_k h\}}(\boldsymbol{\theta}_{k+1}) \frac{1}{Z} \exp\left(-\frac{\|\boldsymbol{x} + \sum_{i=1}^n \boldsymbol{a}_i\|^2}{2}\right) \\
=&1_{\{\boldsymbol{\theta}_k + \boldsymbol{r}_k h\}}(\boldsymbol{\theta}_{k+1}) \frac{1}{Z} \sum_{j=1}^n \underbrace{\frac{1}{n}\left(\exp\left(-\frac{\|\boldsymbol{x} + \sum_{i=1}^n \boldsymbol{a}_i\|^2}{2} + \frac{\|\boldsymbol{x} + n\boldsymbol{a}_i\|^2}{2}\right)\right)}_{p_i} \exp\left(-\frac{\|\boldsymbol{x} + n\boldsymbol{a}_i\|^2}{2}\right) \\
=&\tilde{P}^{EM}(\boldsymbol{\theta}_{k+1}, \boldsymbol{r}_{k+1}|\boldsymbol{\theta}_k, \boldsymbol{r}_k) \qquad (2.4)
\end{aligned}
$$

where Z is a normalization constant, $\boldsymbol{x} \triangleq \frac{\boldsymbol{r}_{k+1} - \boldsymbol{r}_k + h\gamma \boldsymbol{r}_k}{\sigma\sqrt{h}}$ and $\boldsymbol{a}_i \triangleq \frac{\sqrt{h}\nabla f_i(\boldsymbol{\theta}_k)}{\sigma}$. Motivated by Equation (2.4), if we were able to choose $p_i \propto \exp\left(-\frac{\|\boldsymbol{x} + \sum_{i=1}^n \boldsymbol{a}_i\|^2}{2} + \frac{\|\boldsymbol{x} + n\boldsymbol{a}_i\|^2}{2}\right)$, we would be able to recover the transition kernel of full gradient with that of stochastic gradient. However, Equation (2.4) is only formal and infeasible, because $\boldsymbol{x}$ is dependent of future state $\boldsymbol{\theta}_{k+1}$ which we do not know. To turn this idea into a practically feasible algorithm, we will fix $\boldsymbol{x}$ as a hyper-parameter and hope that the approximation is good enough so that $P^{EM}(\boldsymbol{\theta}_{k+1}, \boldsymbol{r}_{k+1}|\boldsymbol{\theta}_k, \boldsymbol{r}_k) \approx \tilde{P}^{EM}(\boldsymbol{\theta}_{k+1}, \boldsymbol{r}_{k+1}|\boldsymbol{\theta}_k, \boldsymbol{r}_k)$ still holds.

We refer to this choice of p_i Exponentially Weighted Stochastic Gradient (**EWSG**). Unlike Theorem 1, EWSG does not require $n > d$ to work. Note the idea of designing non-uniform weights of SG-MCMC to match the transition kernel of full gradient can be suitably applied to a wide class of gradient-based MCMC methods; for example, Appendix A.2 shows how EWSG can be applied to Langevin Monte Carlo (overdamped Langevin), and Appendix A.3 shows how it can be combined with VR. Therefore, EWSG complements a wide range of SG-MCMC methods.

The weight choice of EWSG is motivated by reproducing the transition kernel of a full-gradient MCMC method, hence we anticipate EWSG to be statistically more accurate

than a uniformly-subsampled stochastic gradient estimator. As a special but commonly interested accuracy measure, the smaller variance of EWSG is shown with high probability:

Theorem 2 *Assume* $\{\nabla f_i(\boldsymbol{\theta})\}_{i=1,2,\cdots,n}$ *are i.i.d random vectors and* $|\nabla f_i(\boldsymbol{\theta})| \leq R$ *for some constant* R *almost surely. Denote the uniform distribution over* $[n]$ *by* $\boldsymbol{p}^U$*, the exponentially weighted distribution by* $\boldsymbol{p}^E$*, and let* $\Delta = \mathrm{Tr}[\mathrm{cov}_{I\sim\boldsymbol{p}^E}[n\nabla f_I(\boldsymbol{\theta})|\boldsymbol{\theta}] - \mathrm{cov}_{I\sim\boldsymbol{p}^U}[n\nabla f_I(\boldsymbol{\theta})|\boldsymbol{\theta}]]$. *If* $\boldsymbol{x} = \mathcal{O}(\sqrt{h})$*, we have* $\mathbb{E}[\Delta] < 0$*, and* $\exists C > 0$ *independent of* n *or* h *such that* $\forall \epsilon > 0$*,*

$$\mathbb{P}(|\Delta - \mathbb{E}[\Delta]| \geq \epsilon) \leq 2\exp\left(-\frac{\epsilon^2}{nCh^2}\right).$$

Proof: Let $\boldsymbol{b}_i = n\nabla f_i$ and assume $\|\boldsymbol{b}_i\|_2 \leq R$ for some constant R. Denote $B = [\boldsymbol{b}_1, \boldsymbol{b}_2, \cdots, \boldsymbol{b}_n]$. For any probability distribution $\boldsymbol{p}$ over $\{1, \cdots, n\}$, we have

$$\begin{aligned}
&\mathrm{cov}_{I\sim\boldsymbol{p}}[\boldsymbol{b}_I|\boldsymbol{b}_1, \cdots, \boldsymbol{b}_n] \\
=&\sum_{i=1}^n p_i \boldsymbol{b}_i \boldsymbol{b}_i^T - \left(\sum_{i=1}^n p_i \boldsymbol{b}_i\right)\left(\sum_{i=1}^n p_i \boldsymbol{b}_i\right)^T \\
=&\sum_{i=1}^n p_i \boldsymbol{b}_i \boldsymbol{b}_i^T \sum_{i=1}^n p_i - \left(\sum_{i=1}^n p_i \boldsymbol{b}_i\right)\left(\sum_{i=1}^n p_i \boldsymbol{b}_i\right)^T \\
=&\sum_{i<j} (\boldsymbol{b}_i - \boldsymbol{b}_j)(\boldsymbol{b}_i - \boldsymbol{b}_j)^T p_i p_j
\end{aligned}$$

Therefore we let

$$\begin{aligned}
f(B) :=& \mathrm{Tr}\left[\sum_{i<j} (\boldsymbol{b}_i - \boldsymbol{b}_j)(\boldsymbol{b}_i - \boldsymbol{b}_j)^T p_i p_j - \sum_{i<j} (\boldsymbol{b}_i - \boldsymbol{b}_j)(\boldsymbol{b}_i - \boldsymbol{b}_j)^T \frac{1}{n^2}\right] \\
=& \sum_{i<j} \|\boldsymbol{b}_i - \boldsymbol{b}_j\|^2 p_i p_j - \sum_{i<j} \|\boldsymbol{b}_i - \boldsymbol{b}_j\|^2 \frac{1}{n^2} \qquad (\mathrm{Tr}[AB] = \mathrm{Tr}[BA])
\end{aligned}$$

and use it to compare the trace of covariance matrix of uniform- and nonuniform- subsamplings.

First of all,

$$
\begin{aligned}
&\mathbb{E}[f(B)] \\
=&\mathbb{E}[\|\boldsymbol{b}_i - \boldsymbol{b}_j\|^2] \sum_{i<j} \left(p_i p_j - \frac{1}{n^2} \right) \\
=&\mathbb{E}[\|\boldsymbol{b}_i - \boldsymbol{b}_j\|^2] \left(\sum_{i<j} p_i p_j - \frac{n-1}{2n} \right) \\
=&\mathbb{E}[\|\boldsymbol{b}_i - \boldsymbol{b}_j\|^2] \left(\frac{1 - \sum_{i=1}^n p_i^2}{2} - \frac{n-1}{2n} \right) \\
\leq&\mathbb{E}[\|\boldsymbol{b}_i - \boldsymbol{b}_j\|^2] \left(\frac{1 - \frac{1}{n}}{2} - \frac{n-1}{2n} \right) \\
=&0
\end{aligned}
$$

where the inequality is due to Cauchy-Schwarz and it is a strict inequality unless all p_i's are equal, which means uniform subsampling on average has larger variablity than a non-uniform scheme measured by the trace of covariance matrix.

Moreover, concentration inequality can help show $f(B)$ is negative with high probability if h is small. To this end, plug $\boldsymbol{x} = \mathcal{O}(\sqrt{h})$ in and rewrite

$$
p_i = \frac{1}{Z} \exp \left\{ Fh \left[\frac{\|\boldsymbol{y} + \frac{1}{n} \sum_{i=1}^n \boldsymbol{b}_i\|^2}{2} - \frac{\|\boldsymbol{y} + \boldsymbol{b}_i\|^2}{2} \right] \right\}
$$

where $\boldsymbol{y} = \frac{\sigma}{\sqrt{h}} \boldsymbol{x} = \mathcal{O}(1)$, $F = -\frac{1}{\sigma^2}$ and Z is the normalization constant. Denote the unnormalized probability by

$$
\widetilde{p}_i = \exp \left\{ Fh \left[\frac{\|\boldsymbol{y} + \frac{1}{n} \sum_{i=1}^n \boldsymbol{b}_i\|^2}{2} - \frac{\|\boldsymbol{y} + \boldsymbol{b}_i\|^2}{2} \right] \right\}
$$

and we have

$$f(B) = \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\|\boldsymbol{b}_i - \boldsymbol{b}_j\|^2\left(p_ip_j - \frac{1}{n^2}\right)$$
$$= \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\|\boldsymbol{b}_i - \boldsymbol{b}_j\|^2\frac{\widetilde{p}_i\widetilde{p}_j}{[\sum_{k=1}^{n}\widetilde{p}_k]^2} - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\|\boldsymbol{b}_i - \boldsymbol{b}_j\|^2\frac{1}{n^2}$$

To prove concentration results, it is useful to estimate

$$C_i = \sup_{\substack{\boldsymbol{b}_1,\cdots,\boldsymbol{b}_n\in B(\mathbf{0},R)\\ \widehat{\boldsymbol{b}}_i\in B(\mathbf{0},R)}} |f(\boldsymbol{b}_1,\cdots,\boldsymbol{b}_i,\cdots,\boldsymbol{b}_n)$$

$$-f(\boldsymbol{b}_1,\cdots,\widehat{\boldsymbol{b}}_i,\cdots,\boldsymbol{b}_n)|$$

where $B(\mathbf{0},R)$ is a ball centered at origin with radius R in $\mathbb{R}^d$.

Due to the mean value theorem, we have $C_i \le 2R\sup|\frac{\partial f}{\partial \boldsymbol{b}_i}|$. By symmetry, it suffices to compute $\sup|\frac{\partial f}{\partial \boldsymbol{b}_1}|$ to upper bound C_1. Note that

$$\frac{\partial\widetilde{p}_j}{\partial\boldsymbol{b}_1} = 2\widetilde{p}_jFh[\frac{1}{n}(\boldsymbol{y}+\frac{1}{n}\sum_{i=1}^{n}\boldsymbol{b}_i) - (\boldsymbol{y}+\boldsymbol{b}_j)\delta_{1j}] = \mathcal{O}(h)\widetilde{p}_j$$

where δ_{1j} is the Kronecker delta function. Thus

$$\begin{aligned}\frac{\partial f}{\partial\boldsymbol{b}_1} &= \sum_{j=1}(\boldsymbol{b}_1-\boldsymbol{b}_j)\frac{\widetilde{p}_1\widetilde{p}_j}{[\sum_{k=1}^{n}\widetilde{p}_k]^2} - \sum_{j=1}^{n}(\boldsymbol{b}_1-\boldsymbol{b}_j)\frac{1}{n^2} + \sum_{i,j=1}^{n}\|\boldsymbol{b}_1-\boldsymbol{b}_j\|^2\frac{\mathcal{O}(h)\widetilde{p}_i\widetilde{p}_j}{[\sum_{k=1}^{n}\widetilde{p}_k]^2}\\ &- 2\sum_{i,j=1}^{n}\|\boldsymbol{b}_1-\boldsymbol{b}_j\|^2\frac{\widetilde{p}_i\widetilde{p}_j}{\left[\sum_{k=1}^{n}\widetilde{p}_k\right]^3}\sum_{k=1}^{n}\widetilde{p}_k\mathcal{O}(h)\\ &= \widetilde{p}_1\sum_{j=1}^{n}(\boldsymbol{b}_1-\boldsymbol{b}_j)\frac{\widetilde{p}_j}{[\sum_{k=1}^{n}\widetilde{p}_k]^2} - \sum_{j=1}^{n}(\boldsymbol{b}_1-\boldsymbol{b}_j)\frac{1}{n^2} + \frac{\mathcal{O}(n^2)\mathcal{O}(h)}{\mathcal{O}(n^2)} + \frac{\mathcal{O}(n^2)}{\mathcal{O}(n^3)}\mathcal{O}(n)\mathcal{O}(h)\\ &= \mathcal{O}(\frac{h}{n}) + \mathcal{O}(h) + \mathcal{O}(h)\\ &= \mathcal{O}(h)\end{aligned}$$

where $\mathcal{O}(\frac{h}{n})$ in the 2nd last equation comes from the difference of the first two terms in the

3rd last equation. This estimation shows that $C_i \leq 2R\mathcal{O}(h) = \mathcal{O}(h)$.

Therefore, by McDiarmid's inequality, we conclude for any $\epsilon > 0$,

$$\mathbb{P}(|f - \mathbb{E}[f]| > \epsilon) \leq 2\exp\left(\frac{-2\epsilon^2}{\sum_{i=1}^n C_i^2}\right) = 2\exp\left(\frac{-2\epsilon^2}{n\mathcal{O}(h^2)}\right).$$

Any choice of $h(n) = o(n^{-1/2})$ will render this probability asymptotically vanishing as n grows, which means that f will be negative with high probability, which is equivalent to reduced variance per step. ■

It is not surprising that less non-intrinsic local variance correlates with better global statistical accuracy, which will be made explicit and rigorous in the next subsection.

2.5 Non-asymptotic Error Bound

We now establish a non-asymptotic global sampling error bound (in mean square distance between arbitrary test observables) of SG underdamped Langevin algorithms (the bound applies to both EWSG and other methods e.g., SGHMC). The main tool we will be using is the Poisson equation machinery [64, 25, 22]. A brief overview is the following:

Let $\boldsymbol{X} = \begin{pmatrix} \boldsymbol{\theta} \\ \boldsymbol{r} \end{pmatrix}$. The generator $\mathcal{L}$ of diffusion process Equation (2.1) is

$$\begin{aligned}\mathcal{L}(f(\boldsymbol{X}_t)) =& \lim_{h\to 0} \frac{\mathbb{E}[f(\boldsymbol{X}_{t+h})] - \mathbb{E}[f(\boldsymbol{X}_t)]}{h} \\ =& \boldsymbol{r}^T \nabla_{\boldsymbol{\theta}} f - (\gamma \boldsymbol{r} + \nabla f(\boldsymbol{\theta}))^T \nabla_{\boldsymbol{r}} f + \gamma \Delta_{\boldsymbol{r}} f.\end{aligned}$$

Given a test function $\phi(\boldsymbol{x})$, its posterior average is $\bar{\phi} = \int \phi(\boldsymbol{x})\pi(\boldsymbol{x})d\boldsymbol{x}$, approximated by its time average of samples $\widehat{\phi}_K = \frac{1}{K}\sum_{k=1}^K \phi(\boldsymbol{X}_k^E)$, where $\boldsymbol{X}_k^E$ is the sample path given by EM integrator. Then the Poisson equation $\mathcal{L}\psi = \phi - \bar{\phi}$ can be a useful tool for the weak convergence analysis of SG-MCMC. The solution ψ characterizes the difference between ϕ and its posterior average $\bar{\phi}$.

Our main theoretical result is the following:

Theorem 3 *Assume* $\mathbb{E}[\|\nabla f_i(\boldsymbol{\theta}_k^E)\|^l] < M_1, \mathbb{E}[\|\boldsymbol{r}_k^E\|^l] < M_2, \forall l = 1, 2, \cdots, 12, \forall i = 1, 2, \cdots, n$ *and* $\forall k \geq 0$. *Assume the Poisson equation solution* ψ *exists, and up to its 3rd-order derivatives are uniformly bounded* $\|D^l\psi\|_\infty < M_3, l = 0, 1, 2, 3$. *Then exist constants* $C_1, C_2, C_3 > 0$ *depending on* M_1, M_2, M_3, *such that*

$$\mathbb{E}\big(\widehat{\phi}_K - \bar{\phi}\big)^2 \leq C_1\frac{1}{T} + C_2\frac{h}{T}\frac{\sum_{k=0}^{K-1}\mathbb{E}[\mathrm{Tr}[\mathrm{cov}(n\nabla f_{I_k}|\mathcal{F}_k)]]}{K} + C_3h^2 \tag{2.5}$$

where $T = Kh$ *is the corresponding time in the underlying continuous dynamics,* I_k *is the index of the datum used to estimate gradient at* k*-th iteration, and* $\mathrm{cov}(n\nabla f_{I_k}|\mathcal{F}_k)$ *is the covariance of stochastic gradient at* k*-th iteration conditioned on the current sigma algebra* $\mathcal{F}_k$ *in the filtration.*

Proof: We rewrite the generator of underdamped Langevin with full gradient as

$$\mathcal{L}f(\boldsymbol{X}) = \boldsymbol{F}(\boldsymbol{X})^T\begin{bmatrix}\nabla_{\boldsymbol{\theta}}f(\boldsymbol{X})\\ \nabla_{\boldsymbol{r}}f(\boldsymbol{X})\end{bmatrix} + \frac{1}{2}A : \nabla\nabla f(\boldsymbol{X})$$

where

$$\boldsymbol{F}(\boldsymbol{X}) = \begin{bmatrix}\boldsymbol{r}\\ -\gamma\boldsymbol{r} - \nabla f(\boldsymbol{\theta})\end{bmatrix}, \quad A = GG^T \text{and} \quad G = \begin{bmatrix}O_{d\times d} & O_{d\times d}\\ O_{d\times d} & \sqrt{2\gamma}I_{d\times d}\end{bmatrix}$$

Rewrite the discretized underdamped Langevin with stochastic gradient in variable $\boldsymbol{X}$

$$\boldsymbol{X}_{k+1}^E - \boldsymbol{X}_k^E = h\boldsymbol{F}_k(\boldsymbol{X}_k^E) + \sqrt{h}G_k\boldsymbol{\eta}_{k+1}$$

where

$$\boldsymbol{F}_k(\boldsymbol{X}) = \begin{bmatrix} \boldsymbol{r} \\ -\gamma \boldsymbol{r} - n\nabla f_{I_k}(\boldsymbol{\theta}) \end{bmatrix}, \qquad G_k = G = \begin{bmatrix} O_{d\times d} & O_{d\times d} \\ O_{d\times d} & \sqrt{2\gamma} I_{d\times d} \end{bmatrix}$$

and $\boldsymbol{\eta}_{k+1}$ is a $2d$ dimensional standard Gaussian random vector. Note that this representation include both SGHMC and EWSG, for SGHMC I_k follows uniform distribution and for EWSG, I_k follows the MCMC-approximated exponentially weighted distribution.

Denote the generator associated with stochastic gradient underdamped Langevin at the k-th iteration by

$$\mathcal{L}_k f(\boldsymbol{X}) = \boldsymbol{F}_k(\boldsymbol{X})^T \begin{bmatrix} \nabla_{\boldsymbol{\theta}} f(\boldsymbol{X}) \\ \nabla_{\boldsymbol{r}} f(\boldsymbol{X}) \end{bmatrix} + \frac{1}{2} A : \nabla\nabla f(\boldsymbol{X})$$

and the difference of the generators of full gradient and stochastic gradient underdamped Langevin at k-th interation is denoted by

$$\begin{aligned} \Delta\mathcal{L}_k f(\boldsymbol{X}) = (\mathcal{L}_k - \mathcal{L}) f(\boldsymbol{X}) =& (\boldsymbol{F}_k(\boldsymbol{X}) - \boldsymbol{F}(\boldsymbol{X}))^T \begin{bmatrix} \nabla_{\boldsymbol{\theta}} f(\boldsymbol{X}) \\ \nabla_{\boldsymbol{r}} f(\boldsymbol{X}) \end{bmatrix} \\ =& \langle \nabla f(\boldsymbol{\theta}) - n\nabla f_{I_k}(\boldsymbol{\theta}), \nabla_{\boldsymbol{r}} f(\boldsymbol{X}) \rangle \end{aligned}$$

For brevity, we write $\phi_k = \phi(\boldsymbol{X}_k^E)$, $\boldsymbol{F}_k^E = \boldsymbol{F}_k(\boldsymbol{X}_k^E)$, $\psi_k = \psi(\boldsymbol{X}_k^E)$ and $D^l\phi_k = (D^l\psi)(\boldsymbol{X}_k^E)$ where $(D^l\psi)(z)$ is the l-th order derivative. We write $(D^l\psi)[\boldsymbol{s}_1, \boldsymbol{s}_2, \cdots, \boldsymbol{s}_l]$ for derivative evaluated in the direction $\boldsymbol{s}_j, j = 1, 2, \cdots, l$. Define

$$\boldsymbol{\delta}_k = \boldsymbol{X}_{k+1}^E - \boldsymbol{X}_k^E = h\boldsymbol{F}_k^E + \sqrt{h} G_k \boldsymbol{\eta}_{k+1}$$

Under the assumptions of Theorem 3, we show that the vector field F_k^E also has bounded momentum up to p-th order.

Lemma 4 *Under the assumption of Theorem 3, there exists a constant M such that up to $\frac{p}{2}$-th order moments of random vector field $\boldsymbol{F}_k^E$ are bounded*

$$\mathbb{E}\|\boldsymbol{F}_k^E\|_2^j \leq M,\ \forall j = 0, 1, 2, \cdots, \frac{p}{2},\ \forall k = 0, 1, 2 \cdots,$$

Proof: It suffices to bound the highest moment, as all other lower order moments are bounded by the highest one by Holder's inequality.

First notice that

$$\|\boldsymbol{F}_k^E\|_2 = \left\| \begin{bmatrix} \boldsymbol{r}_k^E \\ -\gamma \boldsymbol{r}_k^E - \nabla f_{I_k}(\boldsymbol{\theta}_k^E) \end{bmatrix} \right\|_2 \leq \sqrt{1+\gamma^2}\|\boldsymbol{r}_k^E\|_2 + \|\nabla f_{I_k}(\theta_k^E)\|_2$$

Hence

$$\begin{aligned}
\mathbb{E}\|\boldsymbol{F}_k^E\|_2^{\frac{p}{2}} \leq & \mathbb{E}\left(\sqrt{1+\gamma^2}\|\boldsymbol{r}_k^E\|_2 + \|\nabla f_{I_k}(\theta_k^E)\|_2\right)^{\frac{p}{2}} \\
= & \mathbb{E}\left\{\sum_{i=0}^{\frac{p}{2}} \binom{\frac{p}{2}}{i} \|\boldsymbol{r}_k^E\|_2^i \|\nabla f_{I_k}(\theta_k^E)\|_2^{\frac{p}{2}-i}\right\} \\
= & \sum_{i=0}^{\frac{p}{2}} \binom{\frac{p}{2}}{i} \mathbb{E}\left[\|\nabla f_{I_k}(\theta_k^E)\|_2^{\frac{p}{2}-i} \|\boldsymbol{r}_k^E\|_2^i\right] \\
\leq & \sum_{i=0}^{\frac{p}{2}} \binom{\frac{p}{2}}{i} \sqrt{\mathbb{E}\left[\|\nabla f_{I_k}(\theta_k^E)\|_2^{p-2i}\right]} \sqrt{\mathbb{E}\left[\|\boldsymbol{r}_k^E\|_2^{2i}\right]} \quad \text{(Cauchy-Schwarz inequality)}
\end{aligned}$$

By assumption, we know each $\mathbb{E}\left[\|\nabla f_{I_k}(\theta_k^E)\|_2^l\right], \mathbb{E}\|\boldsymbol{r}_k^E\|_2^l, l = 0, 1, \cdots, p$ is bounded, so we conclude there exists a constant $M > 0$ that bounds the $\frac{p}{2}$-th order moment of $\boldsymbol{F}_k^E, \forall k = 0, 1, \cdots,$ ■

Using Taylor's expansion for ψ, we have

$$\psi_{k+1} = \psi_k + D\psi_k[\boldsymbol{\delta}_k] + \frac{1}{2}D^2\psi_k[\boldsymbol{\delta}_k, \boldsymbol{\delta}_k] + \frac{1}{6}D^3\psi_k[\boldsymbol{\delta}_k, \boldsymbol{\delta}_k, \boldsymbol{\delta}_k] + R_{k+1}$$

where

$$R_{k+1} = \left(\frac{1}{6} \int_0^1 s^3 D^4 \psi(s\boldsymbol{X}_k^E + (1-s)\boldsymbol{X}_{k+1}^E) ds \right) [\boldsymbol{\delta}_k, \boldsymbol{\delta}_k, \boldsymbol{\delta}_k, \boldsymbol{\delta}_k]$$

is the remainder term. Therefore, we have

$$\begin{aligned} \psi_{k+1} =& \psi_k + h\mathcal{L}_k\psi_k + h^{\frac{1}{2}} D\psi_k[G_k\boldsymbol{\eta}_{k+1}] + h^{\frac{3}{2}} D^2\psi_k[\boldsymbol{F}_k^E, G_k\boldsymbol{\eta}_{k+1}] \\ &+ \frac{1}{2}h^2 D^2\psi_k[\boldsymbol{F}_k^E, \boldsymbol{F}_k^E] + \frac{1}{6} D^3\psi_k[\boldsymbol{\delta}_k, \boldsymbol{\delta}_k, \boldsymbol{\delta}_k] + r_{k+1} + R_{k+1} \end{aligned} \tag{2.6}$$

where

$$r_{k+1} = \frac{h}{2}\big(D^2\psi_k[G_k\boldsymbol{\eta}_{k+1}, G_k\boldsymbol{\eta}_{k+1}] - A : \nabla\nabla\psi_k \big)$$

Summing Equation (2.6) ove the first K terms, dividing by Kh and use Poisson equation, we have

$$\frac{1}{Kh}(\psi_K - \psi_0) = \frac{1}{K}\sum_{k=0}^{K-1}(\phi_k - \bar{\phi}) + \frac{1}{K}\sum_{k=0}^{K-1}\Delta\mathcal{L}_k\psi_k + \frac{1}{Kh}\sum_{i=1}^{3}(M_{i,K} + S_{i,K}), \tag{2.7}$$

where

$$M_{1,K} = \sum_{k=0}^{K-1} r_{k+1},\ M_{2,K} = h^{\frac{1}{2}}\sum_{k=0}^{K-1} D\psi_k[G_k\boldsymbol{\eta}_{k+1}],\ M_{3,K} = h^{\frac{3}{2}}\sum_{k=0}^{K-1} D^2\psi_k[\boldsymbol{F}_k^E, G_k\boldsymbol{\eta}_{k+1}],$$
$$S_{1,K} = \frac{h^2}{2}\sum_{k=0}^{K-1} D^2\psi_k[\boldsymbol{F}_k^E, \boldsymbol{F}_k^E],\ S_{2,K} = \sum_{k=0}^{K-1} R_{k+1},\ S_{3,K} = \frac{1}{6}\sum_{k=0}^{K-1} D^3\psi_k[\boldsymbol{\delta}_k, \boldsymbol{\delta}_k, \boldsymbol{\delta}_k]$$

Furthermore, it will be convenient to decompose

$$S_{3,K} = M_{0,K} + S_{0,K}$$

where

$$
\begin{aligned}
S_{0,K} =& h^2 \sum_{k=0}^{K-1} \left(hD^3\psi_k[\boldsymbol{F}_k^E, \boldsymbol{F}_k^E, \boldsymbol{F}_k^E] + 3D^3\psi_k[\boldsymbol{F}_k^E, G_k\boldsymbol{\eta}_{k+1}, G_k\boldsymbol{\eta}_{k+1}] \right) \\
M_{0,K} =& h^{\frac{3}{2}} \sum_{k=0}^{K-1} \left(D^3\psi_k[G_k\boldsymbol{\eta}_{k+1}, G_k\boldsymbol{\eta}_{k+1}, G_k\boldsymbol{\eta}_{k+1}] + 3hD^3\psi_k[\boldsymbol{F}_k^E, \boldsymbol{F}_k^E, G_k\boldsymbol{\eta}_{k+1}] \right)
\end{aligned}
$$

Rearrange terms in Equation (2.6), square on both sides, use Cauchy-Schwarz inequality and take expectation, we have

$$
\begin{aligned}
&\mathbb{E}\big(\hat{\phi}_K - \bar{\phi}\big)^2 \\
\leq& C \left[\mathbb{E}\frac{(\psi_K - \psi_0)^2}{(Kh)^2} + \frac{1}{K^2}\mathbb{E}\left(\sum_{k=0}^{K-1} (\Delta\mathcal{L}_k\psi_k) \right)^2 + \frac{1}{(Kh)^2}\sum_{i=0}^{2}\mathbb{E}S_{i,K}^2 + \frac{1}{(Kh)^2}\sum_{i=0}^{3}\mathbb{E}M_{i,K}^2 \right] \\
=& C \left[\mathbb{E}\frac{(\psi_K - \psi_0)^2}{T^2} + \frac{1}{K^2}\mathbb{E}\left(\sum_{k=0}^{K-1} (\Delta\mathcal{L}_k\psi_k) \right)^2 + \frac{1}{T^2}\sum_{i=0}^{2}\mathbb{E}S_{i,K}^2 + \frac{1}{T^2}\sum_{i=0}^{3}\mathbb{E}M_{i,K}^2 \right]
\end{aligned}
$$

where $T = kh$, the corresponding time of the underlying continuous dynamics.

We now show how each term is bounded. By boundedness of ψ, we have

$$
\mathbb{E}\frac{(\psi_K - \psi_0)^2}{T^2} \leq \frac{4\|\psi\|_\infty^2}{T^2} = \mathcal{O}(\frac{1}{T^2})
$$

The second term $\frac{1}{K^2}\mathbb{E}\big(\sum_{k=0}^{K-1} (\Delta\mathcal{L}_k\psi_k) \big)^2$ is critical in showing the advantage of EWSG, and we will show how to derive its bound in detail later.

The technique we use to bound $\frac{1}{T^2}\mathbb{E}S_{i,K}^2, i = 0, 1, 2$ are all similar, we will first show an upper bound for $|S_{i,K}|$ in terms of powers of $\|\boldsymbol{F}_k^E\|$, then take square and expectation, and finally expand squares and use Lemma 4 extensively to derive bounds. As a concrete example, we will show how to bound $\frac{1}{T^2}\mathbb{E}S_{0,K}^2$. Other bounds follow in a similar fashion and details are omitted.

To bound the term containing $S_{0,K}$, we first note that

$$
\begin{aligned}
|S_{0,K}| \leq & h^2 \sum_{k=0}^{K-1} \left(h|D^3\psi_k[\boldsymbol{F}_k^E, \boldsymbol{F}_k^E, \boldsymbol{F}_k^E]| + 3|D^3\psi_k[\boldsymbol{F}_k^E, G_k\boldsymbol{\eta}_{k+1}, G_k\boldsymbol{\eta}_{k+1}]| \right) \\
\leq & h^2\|D^3\psi\|_\infty \sum_{k=0}^{K-1} \left(h\|\boldsymbol{F}_k^E\|_2^3 + 3\|\boldsymbol{F}_k^E\|_2\|G_k\boldsymbol{\eta}_{k+1}\|_2^2 \right)
\end{aligned}
$$

Square both sides of the above inequality and take expectation, we obtain

$$
\begin{aligned}
& \frac{1}{T^2}\mathbb{E}|S_{0,K}|^2 \qquad (2.8) \\
\leq & \frac{h^4}{T^2}\|D^3\psi\|_\infty^2 \mathbb{E}\big(\sum_{k=0}^{K-1} h\|\boldsymbol{F}_k^E\|_2^3 + 3\|\boldsymbol{F}_k^E\|_2\|G_k\boldsymbol{\eta}_{k+1}\|_2^2 \big)^2 \\
\leq & \frac{h^4}{T^2}\|D^3\psi\|_\infty^2 K \sum_{k=0}^{K-1} \mathbb{E}(h\|\boldsymbol{F}_k^E\|_2^3 + 3\|\boldsymbol{F}_k^E\|_2\|G_k\boldsymbol{\eta}_{k+1}\|_2^2)^2 \quad \text{(Cauchy-Schwarz inequality)} \\
= & \frac{h^4}{T^2}\|D^3\psi\|_\infty^2 K \sum_{k=0}^{K-1} \mathbb{E}[h^2\|\boldsymbol{F}_k^E\|_2^6 + 6\|\boldsymbol{F}_k^E\|_2^4\|G_k\boldsymbol{\eta}_{k+1}\|_2^2 + 9\|\boldsymbol{F}_k^E\|_2^2\|G_k\boldsymbol{\eta}_{k+1}\|_4^2] \\
= & \frac{h^4}{T^2}\|D^3\psi\|_\infty^2 K \sum_{k=0}^{K-1} h^2\mathbb{E}\|\boldsymbol{F}_k^E\|_2^6 + 6\mathbb{E}\|\boldsymbol{F}_k^E\|_2^4\mathbb{E}\|G_k\boldsymbol{\eta}_{k+1}\|_2^2 + 9\mathbb{E}\|\boldsymbol{F}_k^E\|_2^2\mathbb{E}\|G_k\boldsymbol{\eta}_{k+1}\|_4^2 \\
= & \frac{1}{T^2}\mathcal{O}(K^2h^4) \\
= & \mathcal{O}(h^2)
\end{aligned}
$$

To bound the term containing $S_{1,K}$ and $S_{2,K}$, we have

$$
\begin{aligned}
|S_{1,K}| \leq & \frac{h^2}{2} \sum_{k=0}^{K-1} \|D^2\psi\|_\infty \|\boldsymbol{F}_k^E\|_2^2 \\
|S_{2,K}| \leq & \frac{1}{24}\|D^4\psi\|_\infty \sum_{k=0}^{K-1} \|\boldsymbol{\delta}_k\|_2^4 \leq \frac{1}{24}h^2\|D^4\psi\|_\infty \sum_{k=0}^{K-1} \|\sqrt{h}\boldsymbol{F}_k^E + G_k\boldsymbol{\eta}_{k+1}\|_2^4
\end{aligned}
$$

Then we can obtain the following bound in a similar fashion as in Equation (2.8)

$$\begin{aligned}\frac{1}{T^2}\mathbb{E}S_{1,K}^2 =&\mathcal{O}(h^2)\\ \frac{1}{T^2}\mathbb{E}S_{2,K}^2 =&\mathcal{O}(h^2)\end{aligned}$$

Now we will use martingale argument to bound $\frac{1}{T^2}\mathbb{E}M_{i,K}^2, i = 0, 1, 2, 3$. There are two injected randomness at k-th iteration, the Gaussian noise $\boldsymbol{\eta}_{k+1}$ and the stochastic gradient term determined by the stochastic index I_k. Denote the sigma algebra at k-th iteration by $\mathcal{F}_k$. For both SGHMC and EWSG we have

$$\boldsymbol{\eta}_{k+1} \perp \mathcal{F}_k \text{ and } I_k \perp \boldsymbol{\eta}_{k+1}$$

hence

$$\begin{aligned}\mathbb{E}[\boldsymbol{\eta}_{k+1}|\mathcal{F}_k] =&\mathbf{0}\\ \mathbb{E}[D^3\psi_k[G_k\boldsymbol{\eta}_{k+1}, G_k\boldsymbol{\eta}_{k+1}, G_k\boldsymbol{\eta}_{k+1}]|\mathcal{F}_k] =&0\\ \mathbb{E}[D^2\psi_k[\boldsymbol{F}_k^E, G_k\boldsymbol{\eta}_{k+1}]|\mathcal{F}_k] =&0\\ \mathbb{E}[D^3\psi_k[\boldsymbol{F}_k^E, \boldsymbol{F}_k^E, G_k\boldsymbol{\eta}_{k+1}]|\mathcal{F}_k] =&0\end{aligned}$$

Therefore, it is clear that $M_{i,K}, i = 0, 1, 2, 3$ are all martingales. Due to martingale

properties, we have

$$\begin{aligned}
\frac{1}{T^2}\mathbb{E}M_{0,K}^2 =& \frac{h^3}{T^2}\sum_{k=0}^{K-1}\mathbb{E}\big(D^3\psi_k[G_k\boldsymbol{\eta}_{k+1}, G_k\boldsymbol{\eta}_{k+1}, G_k\boldsymbol{\eta}_{k+1}] + 3hD^3\psi_k[\boldsymbol{F}_k^E, \boldsymbol{F}_k^E, G_k\boldsymbol{\eta}_{k+1}]\big)^2 \\
=& \frac{1}{T^2}\mathcal{O}(h^3K) = \mathcal{O}(\frac{h^2}{T}) \\
\frac{1}{T^2}\mathbb{E}M_{1,K}^2 =& \frac{1}{T^2}\sum_{k=0}^{K-1}\mathbb{E}r_{k+1}^2 = \frac{1}{T^2}\mathcal{O}(h^2K) = \mathcal{O}(\frac{h}{T}) \\
\frac{1}{T^2}\mathbb{E}M_{2,K}^2 =& \frac{h}{T^2}\sum_{k=0}^{K-1}\mathbb{E}(D\psi_k[G_k\boldsymbol{\eta}_{k+1}])^2 = \frac{1}{T^2}\mathcal{O}(hK) = \mathcal{O}(\frac{1}{T}) \\
\frac{1}{T^2}\mathbb{E}M_{3,K}^2 =& \frac{1}{T^2}h^3\sum_{k=0}^{K-1}\mathbb{E}(D^2\psi_k[\boldsymbol{F}_k^E, G_k\boldsymbol{\eta}_{k+1}])^2 = \frac{1}{T^2}\mathcal{O}(h^3K) = \mathcal{O}(\frac{h^2}{T})
\end{aligned}$$

We now collect all bounds derived so far and obtain

$$\begin{aligned}
\mathbb{E}\big(\hat{\phi}_K - \bar{\phi}\big)^2 \leq& C\left[\mathcal{O}(\frac{1}{T^2}) + \frac{1}{K^2}\mathbb{E}\left(\sum_{k=0}^{K-1}(\Delta\mathcal{L}_k\psi_k)\right)^2 + \mathcal{O}(h^2) + \mathcal{O}(\frac{h}{T}) + \mathcal{O}(\frac{1}{T}) + \mathcal{O}(\frac{h^2}{T})\right] \\
\leq& C\left[\mathcal{O}(\frac{1}{T}) + \frac{1}{K^2}\mathbb{E}\left(\sum_{k=0}^{K-1}(\Delta\mathcal{L}_k\psi_k)\right)^2 + \mathcal{O}(h^2)\right] \qquad (2.9)
\end{aligned}$$

In the above inequality, we use $\frac{1}{T^2} < \frac{1}{T}$ and $\frac{h}{T} \leq \frac{1}{T}$, $\frac{h^2}{T} \leq \frac{1}{T}$ as typically we assume $T \gg 1$ and $h \ll 1$ in non-asymptotic analysis.

Now we focus on the remaining term $\frac{1}{K^2}\mathbb{E}\big(\sum_{k=0}^{K-1}\Delta\mathcal{L}_k\psi_k\big)^2$. For SGHMC, we have that $\mathbb{E}[\Delta\mathcal{L}_k\psi_k|\mathcal{F}_k] = 0$, hence $\sum_{k=0}^{K-1}\Delta\mathcal{L}_k\psi_k$ is a martingale. By martingale property, we have

$$\frac{1}{K^2}\mathbb{E}\left(\sum_{k=0}^{K-1}\Delta\mathcal{L}_k\psi_k\right)^2 = \frac{1}{K^2}\sum_{k=0}^{K-1}\mathbb{E}(\Delta\mathcal{L}_k\psi_k)^2$$

For EWSG, $\sum_{k=0}^{K-1} \Delta\mathcal{L}_k\psi_k$ is no longer a martingale, but we still have the following

$$\begin{aligned}\frac{1}{K^2}\mathbb{E}\left(\sum_{k=0}^{K-1}\Delta\mathcal{L}_k\psi_k\right)^2 =& \frac{1}{K^2}\sum_{k=0}^{K-1}\mathbb{E}(\Delta\mathcal{L}_k\psi_k)^2 + \frac{2}{K^2}\sum_{i<j}\mathbb{E}(\Delta\mathcal{L}_i\psi_i)(\Delta\mathcal{L}_j\psi_j)\\ =& \frac{1}{K^2}\sum_{k=0}^{K-1}\mathbb{E}(\Delta\mathcal{L}_k\psi_k)^2 + \frac{2}{K^2}\sum_{i<j}\mathbb{E}[(\Delta\mathcal{L}_i\psi_i)\mathbb{E}[\Delta\mathcal{L}_j\psi_j|\mathcal{F}_j]]\end{aligned} \tag{2.10}$$

For the term $\mathbb{E}[\Delta\mathcal{L}_j\psi_j|\mathcal{F}_j]$, we have

$$\mathbb{E}[\Delta\mathcal{L}_j\psi_j|\mathcal{F}_j] = \mathbb{E}[\langle\nabla f(\boldsymbol{\theta}_j^E) - n\nabla f_{I_j}(\boldsymbol{\theta}_j^E), \nabla_{\boldsymbol{r}}\psi_j\rangle|\mathcal{F}_j] = \langle\mathbb{E}[\nabla f(\boldsymbol{\theta}_j^E) - n\nabla f_{I_j}(\boldsymbol{\theta}_j^E)|\mathcal{F}_j], \nabla_{\boldsymbol{r}}\psi_j\rangle$$

as $\psi_j \in \mathcal{F}_j$. Then by Cauchy-Schwarz inequality, boundedness of ψ and the fact $\|\nabla f(\boldsymbol{\theta}_j^E) - \mathbb{E}[n\nabla f_{I_j}(\boldsymbol{\theta}_j^E)|\mathcal{F}_j]\|_2 = \mathcal{O}(h)$ as shown in the proof of Theorem 2, we conclude $\mathbb{E}[\Delta\mathcal{L}_j\psi_j|\mathcal{F}_j] = \mathcal{O}(h)$.

Now plug the above result in Equation (2.10), we have

$$\begin{aligned}\frac{1}{K^2}\mathbb{E}\left(\sum_{k=0}^{K-1}\Delta\mathcal{L}_k\psi_k\right)^2 =& \frac{1}{K^2}\sum_{k=0}^{K-1}\mathbb{E}(\Delta\mathcal{L}_k\psi_k)^2 + \frac{2}{K^2}\sum_{i<j}\mathbb{E}[(\Delta\mathcal{L}_i\psi_i)\mathbb{E}[\Delta\mathcal{L}_j\psi_j|\mathcal{F}_j]]\\ =& \frac{1}{K^2}\sum_{k=0}^{K-1}\mathbb{E}(\Delta\mathcal{L}_k\psi_k)^2 + \frac{2}{K^2}\sum_{i<j}\mathbb{E}[\Delta\mathcal{L}_i\psi_i]\mathcal{O}(h)\\ =& \frac{1}{K^2}\sum_{k=0}^{K-1}\mathbb{E}(\Delta\mathcal{L}_k\psi_k)^2 + \frac{2}{K^2}\sum_{i<j}\mathcal{O}(h^2)\\ =& \frac{1}{K^2}\sum_{k=0}^{K-1}\mathbb{E}(\Delta\mathcal{L}_k\psi_k)^2 + \frac{2}{K^2}\sum_{i<j}\mathcal{O}(h^2)\\ =& \frac{1}{K^2}\sum_{k=0}^{K-1}\mathbb{E}(\Delta\mathcal{L}_k\psi_k)^2 + \mathcal{O}(h^2)\end{aligned}$$

Combine both cases of SGHMC and EWSG, we obtain

$$\frac{1}{K^2}\mathbb{E}\left(\sum_{k=0}^{K-1}\Delta\mathcal{L}_k\psi_k\right)^2 = \frac{1}{K^2}\sum_{k=0}^{K-1}\mathbb{E}(\Delta\mathcal{L}_k\psi_k)^2 + \mathcal{O}(h^2)$$

Note that $\mathcal{O}(h^2)$ term will later be combined with other error terms with the same order.

The final piece is to bound $\frac{1}{K^2}\sum_{k=0}^{K-1}\mathbb{E}(\Delta\mathcal{L}_k\psi_k)^2$, and we have

$$\begin{aligned}
&\frac{1}{K^2}\sum_{k=0}^{K-1}\mathbb{E}(\Delta\mathcal{L}_k\psi_k)^2 \\
=&\frac{1}{K^2}\sum_{k=0}^{K-1}\mathbb{E}\langle\nabla f(\boldsymbol{\theta}_k^E) - n\nabla f_{I_k}(\boldsymbol{\theta}_k^E), \nabla_{\boldsymbol{r}}\psi_k\rangle^2 \\
\leq&\frac{1}{K^2}\sum_{k=0}^{K-1}\mathbb{E}[\|\nabla f(\boldsymbol{\theta}_k^E) - n\nabla f_{I_k}(\boldsymbol{\theta}_k^E)\|_2^2 \cdot \|\nabla_{\boldsymbol{r}}\psi_k\|_2^2] \quad \text{(Cauchy-Schwarz inequality)} \\
\leq&\frac{M_3^2}{K^2}\sum_{k=0}^{K-1}\mathbb{E}[\|\nabla f(\boldsymbol{\theta}_k^E) - n\nabla f_{I_k}(\boldsymbol{\theta}_k^E)\|_2^2] \\
=&\frac{M_3^2}{K^2}\sum_{k=0}^{K-1}\mathbb{E}[\mathbb{E}[\|\nabla f(\boldsymbol{\theta}_k^E) - n\nabla f_{I_k}(\boldsymbol{\theta}_k^E)\|_2^2 \,|\, \mathcal{F}_k]] \\
\leq&\frac{2M_3^2}{K^2}\sum_{k=0}^{K-1}\mathbb{E}[\underbrace{\mathbb{E}[\,\|\nabla f(\boldsymbol{\theta}_k^E) - \mathbb{E}[n\nabla f_{I_k}(\boldsymbol{\theta}_k^E) \,|\, \mathcal{F}_k]\|^2 \,|\, \mathcal{F}_k]}_{Q_1}] \\
+&\underbrace{\mathbb{E}[\,\|\mathbb{E}[n\nabla f_{I_k}(\boldsymbol{\theta}_k^E) \,|\, \mathcal{F}_k] - n\nabla f_{I_k}(\boldsymbol{\theta}_k^E)\|_2^2 \,|\, \mathcal{F}_k]}_{Q_2}]
\end{aligned}$$

The term Q_1 captures the bias of stochastic gradient. For SGHMC, uniform gradient subsamping leads to an unbiased gradient estimator, so $Q_1 = 0$ for SGHMC. For EWSG, same as in the proof of Theorem 2, we have that

$$\mathbb{E}\left[\,\|\nabla f(\boldsymbol{\theta}_k^E) - \mathbb{E}[n\nabla f_{I_k}(\boldsymbol{\theta}_k^E) \,|\, \mathcal{F}_k]\|^2 \,|\, \mathcal{F}_k\right] = \mathcal{O}(h^2)$$

Combining two cases, we have

$$Q_1 = \mathcal{O}(h^2)$$

For a random vector $\boldsymbol{v}$ with mean $\mathbb{E}[\boldsymbol{v}] = \mathbf{0}$, we have

$$\mathbb{E}[\|\boldsymbol{v}\|^2] = \mathbb{E}\left[\mathrm{Tr}[\boldsymbol{v}\boldsymbol{v}^T]\right] = \mathrm{Tr}\left[\mathbb{E}[\boldsymbol{v}\boldsymbol{v}^T]\right] = \mathrm{Tr}\left[\mathrm{cov}(\boldsymbol{v})\right]$$

where $\mathrm{cov}(\boldsymbol{v})$ is the covariance matrix of random vector $\boldsymbol{v}$. Therefore, we have that

$$Q_2 = \mathrm{Tr}\left[\mathrm{cov}(n\nabla f_{I_k}|\mathcal{F}_k)\right],$$

i.e., Q_2 is the trace of the covariance matrix of stochastic gradient estimate conditioned on current filtration $\mathcal{F}_k$.

Combining Q_1 and Q_2, we have that

$$\begin{aligned}\frac{1}{K^2}\mathbb{E}\left(\sum_{k=0}^{K-1}\Delta\mathcal{L}_k\psi_k\right)^2 \leq& \frac{2M_3^2}{K^2}\sum_{k=0}^{K-1}\left[\mathbb{E}[\mathrm{Tr}[\mathrm{cov}(n\nabla f_{I_k}|\mathcal{F}_k)]] + \mathcal{O}(h^2)\right] \\ =& \frac{2M_3^2 h}{T}\frac{\sum_{k=0}^{K-1}\mathbb{E}[\mathrm{Tr}[\mathrm{cov}(n\nabla f_{I_k}|\mathcal{F}_k)]]}{K} + \mathcal{O}(\frac{h^3}{T})\end{aligned}$$

Now plug this bound into Equation (2.9) and we obtain

$$\mathbb{E}\big(\hat{\phi}_K - \bar{\phi}\big)^2 \leq C_1\frac{1}{T} + C_2\frac{h}{T}\frac{\sum_{k=0}^{K-1}\mathbb{E}\left[\mathrm{Tr}[\mathrm{cov}(n\nabla f_{I_k}|\mathcal{F}_k)]\right]}{K} + C_3h^2$$

for some constants $C_1, C_2, C_3 > 0$ depending on M_1, M_2, M_3. ■

Remark: (*interpreting the three terms in the bound*) Unlike a typical VR method which aims at finding unbiased gradient estimator with reduced variance, EWSG aims at bringing the entire density closer to that of a batch-gradient MCMC. As a consequence, its practical implementation may correspond to SG that has reduced variance but a small bias too. Equation (2.5) quantifies this bias-variance trade-off. How the extrinsic local variance and bias contribute to the global error is respectively reflected in the 2nd and 3rd terms, although the 3rd term also contains a contribution from the numerical discretization error. With or

without bias, the 3rd term remains $\mathcal{O}(h^2)$ because of this discretization error. However, for moderate T, the 2nd term is generally larger than the 3rd due to its lower order in h, which means reducing local variance can improve sampling accuracy even if at the cost of introducing a small bias. Since EWSG has a smaller local variance than uniform SG (Theorem 2, as a special case of improved overall statistical accuracy), its global performance is also favorable. The 1st term is for the convergence of the continuous process (Equation (2.1) in this case).

Remark: (*innovation and relation with the literature*) Theorem 3, to the best of our knowledge, is the first that incorporates the effects of both local bias and local variance of a SG approximation (previous SOTA bounds are only for unbiased SG). It still works when restricting to unbiased SG, and in this case our bound reduces to SOTA [25, 22]. Some more facts include: [64], being the seminal work and from which we adapt our proof, only discussed the batch gradient case, whereas our theory has additional (non-uniform) SG. [25, 22] studied the effect of SG, but the SG considered there did not use state-dependent weights, which would destroy several martingales used in their proofs. Unlike in [64] but like in [25, 22], our state space is not the compact torus but $\mathbb{R}^d$. Also, the time average $\widehat{\phi}_K$, to which our results apply, is a commonly used estimator, particularly when using a long time trajectory of Markov chain for sampling. However, if one is interested in an alternative of using an ensemble for sampling, techniques in [28, 65] might be useful to further bound difference between the law of $\boldsymbol{X}_k$ and the target distribution.

2.6 Practical Implementation

In EWSG, the probability of each gradient term is $p_i = \widehat{Z}^{-1} \exp\left\{-\frac{\|\boldsymbol{x}+\sum_{j=1}^n \boldsymbol{a}_j\|^2}{2} + \frac{\|\boldsymbol{x}+n\boldsymbol{a}_i\|^2}{2}\right\}$. Although the term $\|\boldsymbol{x} + \sum_{j=1}^n \boldsymbol{a}_j\|^2/2$ depends on the full data set, it is shared by all p_i's and can be absorbed into the normalization constant $\hat{Z}^{-1}$ (we still included it explicitly due to the needs of analyses in proofs); unique to each p_i is only the term $\|\boldsymbol{x} + n\boldsymbol{a}_i\|^2/2$. This motivates us to run a Metropolis-Hastings chain over the possible indices $i \in \{1, 2 \cdots, n\}$:

at each inner-loop step, a proposal of index j is uniformly drawn, and then accepted with probability $P(i \to j) =$

$$\min\left\{1, \exp\left(\frac{\|\boldsymbol{x} + n\boldsymbol{a}_j\|^2}{2} - \frac{\|\boldsymbol{x} + n\boldsymbol{a}_i\|^2}{2}\right)\right\}; \tag{2.11}$$

if accepted, the current index i is replaced by j. When the chain converges, the index will follow the distribution given by p_i. The advantage is, we avoid passing through the entire data sets to compute each p_i, but the index will still approximately sample from the non-uniform distribution.

In practice, we often only perform $M = 1$ step of the Metropolis index chain per integration step, especially if h is not too large. The rationale is, when h is small, the outer iteration evolves slower than the index chain, and as θ does not change much in, say, N outer steps, effectively $N \times M$ inner steps take place on almost the same index chain, which makes the index r.v. equilibrate better. Regarding the larger h case (where the efficacy of local variance reduction via non-uniform subsampling is more pronounced; see e.g., Theorem 3), $M = 1$ may no longer be optimal, but improved sampling with large h and $M = 1$ is still clearly observed in various experiments (Section 2.7).

Another hyper-parameter is $\boldsymbol{x}$, because p_i essentially depends on the future state $\boldsymbol{r}_{k+1}$ via $\boldsymbol{x}$, which we do not know, and yet we'd like to avoid expensive nonlinear solvers. Our heuristic recommendation is $\boldsymbol{x} = \frac{\sqrt{h}\gamma \boldsymbol{r}_k}{\sigma}$. Its intuition is, as long as $r_{k+1} - r_k$'s density is maximized at 0 (which will be the case at least for large k as r_k will converge to a Gaussian), this choice is a maximum likelihood estimator. This approximation appeared to be a good one in all our experiments with medium h and $M = 1$.

We further investigates hyperparameter selection in Section 2.7.1 and empirically shows that approximations due to M and $\boldsymbol{x}$ is not detrimental to our non-asymptotic theory in Section 2.5.

Practical EWSG is summarized in Algorithm 1. For simplicity of notation, we restrict

Algorithm 1 EWSG

Input: {the number of data terms n, gradient functions $V_i(\cdot), i = 1, 2, \cdots, n$, step size h, the number of data passes K, index chain length M, friction and noise coefficients γ and σ}
Initialize $\boldsymbol{\theta}_0, \boldsymbol{r}_0$ (arbitrarily, or use an informed guess)
for $k = 0, 1, \cdots, \lceil \frac{Kn}{M+1} \rceil$ **do**
  $i \leftarrow$ uniformly sampled from $1, \cdots, n$, compute and store $n\nabla f_i(\boldsymbol{\theta}_k)$
  $I \leftarrow i$
  for $m = 1, 2, \cdots, M$ **do**
    $j \leftarrow$ uniformly sampled from $1, \cdots, n$, compute and store $n\nabla f_j(\boldsymbol{\theta}_k)$
    $I \leftarrow j$ with probability in Equation (2.11)
  end for
  Evaluate $\tilde{V}(\boldsymbol{\theta}_k) = nV_I(\boldsymbol{\theta}_k)$
  Update $(\boldsymbol{\theta}_{k+1}, \boldsymbol{r}_{k+1}) \leftarrow (\boldsymbol{\theta}_k, \boldsymbol{r}_k)$ via one step of Euler-Maruyama integration using $\tilde{V}(\boldsymbol{\theta}_k)$
end for

the description to mini batch size $b = 1$, but an extension to $b > 1$ is straightforward. See Appendix A.1 in appendix. Practical EWSG has reduced variance but does not completely eliminate the extrinsic noise created by SG due to its approximations. A small bias was also created by these approximations, but its effect is dominated by the variance effect (see Section 2.5). In practice, if needed, one can combine EWSG with other VR technique to further improve accuracy. Appendix A.3 describes how EWSG can be combined with SVRG.

2.7 Numerical Examples

In this section, the proposed EWSG algorithm will be compared with SGHMC, SGLD [17], as well as several more recent popular approaches, including FlyMC [41], pSGLD [24], CP-SGHMC [42] (a method closest to the goal of applying IS idea to SG-based sampling) and SVRG-LD [49] (overdamped Langevin improved by VR). We conduct a detailed empirical study of EWSG on simple models in Section 2.7.1, with comparison and implication of two important hyper-parameters M and $\boldsymbol{x}$, and verification of the non-asymptotic theory (underdamped Langevin dynamics Theorem 3). We demonstrate EWSG for Bayesian lo-

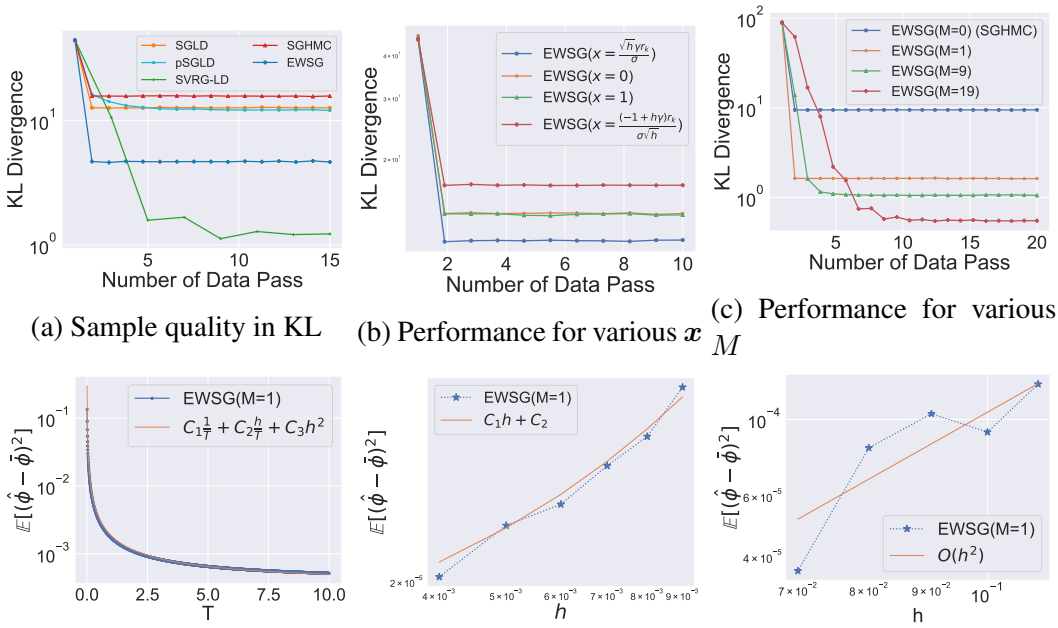
gistic regression on a large-scale data set in Section 2.7.2. We showcase a Bayesian Neural Network (BNN) example in Section 2.7.3. It serves only as a high-dimensional, multi-modal test case, and we do not intend to compare Bayesian and non-Bayesian neural nets. As FlyMC requires a tight lower bound of likelihood, known for only a few cases, it will only be compared against in Section 2.7.2 where such a bound is obtainable. CP-SGHMC requires heavy tuning on the number of clusters which differs across data sets/algorithms, so it will only be included in the BNN example, for which the authors empirically found a good hyper parameter for MNIST [42]. SVRG-LD is only compared in Section 2.7.1, because SG-MCMC methods can converge within only one data pass in Section 2.7.2, rendering control-variate based VR technique inapplicable, and it was suggested that VR leads to poor results for deep models (e.g., Section 2.7.3) [66]

For fair comparison, all algorithms use constant step sizes and are allowed fixed computation budget, i.e., for L data passes, all algorithms can only call gradient function nL times. All experiments are conducted on a machine with a 2.20GHz Intel(R) Xeon(R) E5-2630 v4 CPU and an Nvidia GeForce GTX 1080 GPU. If not otherwise mentioned, $\sigma = \sqrt{2\gamma}$ so only γ needs specification, the length of the index chain is set $M = 1$ for EWSG and the default values of two hyper-parameters required in pSGLD are set $\lambda = 10^{-5}$ and $\alpha = 0.99$, as suggested in [24].

2.7.1 Gaussian Examples

Consider sampling from a simple 2D Gaussian whose potential function is $f(\boldsymbol{\theta}) = \sum_{i=1}^{n} f_i(\boldsymbol{\theta}) = \sum_{i=1}^{n} \frac{1}{2}\|\boldsymbol{\theta} - \boldsymbol{c}_i\|^2$. We set $n = 50$ and randomize $\boldsymbol{c}_i$ from a two-dimensional standard normal $\mathcal{N}(\mathbf{0}, I_2)$. Due to the simplicity of $f(\boldsymbol{\theta})$, we can write the target density analytically and will use KL divergence $\mathrm{KL}(p\|q) = \int p(\boldsymbol{\theta}) \log \frac{p(\boldsymbol{\theta})}{q(\boldsymbol{\theta})} d\boldsymbol{\theta}$ to measure the difference between the target distribution and generated samples.

For each algorithm, we generate 10000 independent realizations for empirical estimation. All algorithms are run for 30 data passes with minibatch size of 1. Step size is tuned



(a) Sample quality in KL

(b) Performance for various $\boldsymbol{x}$

(c) Performance for various M

(d) MSE against time T (1^{st} and 2^{nd} terms in Equation (2.5))

(e) MSE against step size h with fixed finite T (2^{nd} term in Equation (2.5))

(f) MSE against step size h with $T \approx \infty$ (3^{rd} term in Equation (2.5))

Figure 2.1: Sampling from Gaussian target

from $5 \times \{10^{-1}, 10^{-2}, 10^{-3}, 10^{-4}\}$ and 5×10^{-3} is chosen for SGLD and pSGLD, 5×10^{-2} for SGHMC and EWSG and 5×10^{-4} for SVRG-LD. SGHMC and EWSG use $\gamma = 10$. Results are shown in Figure 2.1a and EWSG outperforms SGHMC, SGLD and pSGLD in terms of accuracy. Note SVRG-LD has the best accuracy[4] but the slowest convergence, and that is why EWSG is a useful alternative to VR: its light-weight suits situations with limited computational resources better.

Figure 2.1b shows the performance of several possible choices of the hyper-parameter $\boldsymbol{x}$, including the recommended option $\boldsymbol{x} = \sqrt{h}\gamma\boldsymbol{r}_k/\sigma$, and $\boldsymbol{x} = \mathbf{0}$, $\boldsymbol{x} = \mathbf{1}$, $\boldsymbol{x} = (-1 + h\gamma)\boldsymbol{r}_k/\sigma\sqrt{h}$ (which corresponds to $\boldsymbol{r}_{k+1} = \mathbf{0}$). Step size $h = 7 \times 10^{-2}$ is used for this experiment. The recommended option performs better than the others.

Another important hyper-parameter in EWSG is M. As the length of index chain M increases, the subsampling distribution approaches the ideal exponential weights. For finite M, however, some bias is introduced but variance is also reduced. This tradeoff is worth-

[4]For Gaussians, mean and variance completely determine the distribution, so appropriately reduced variance leads to great accuracy for the entire distribution.

while for reasonable T and h values according to Theorem 3, and considering that larger M means more gradient evaluations per step[5], there could be some M value that achieves the best balance between speed and accuracy. Figure 2.1c shows a fair comparison of four values of $M = 0, 1, 9, 19$, and the recommended $M = 1$ case converges as fast as SGHMC (when $M = 0$, EWSG does not run the Metropolis-Hastings index chain and hence degenerates to SGHMC) but improves its accuracy. It is also clear that as M increases, sampling accuracy gets improved.

As approximations are used in Algorithm 1, it is natural to ask if results of Theorem 3 still hold. We empirically investigate this question (using $M = 1$ and variance as the test function ϕ). Equation (2.5) in Theorem 3 is a nonasymptotic error bound consisting of three parts, namely an $\mathcal{O}(\frac{1}{T})$ term corresponding to the convergence at the continuous limit, an $\mathcal{O}(h/T)$ term coming from the SG variance, and an $\mathcal{O}(h^2)$ term due to bias and numerical error. Figure 2.1d plots the mean squared error (MSE) against time $T = Kh$ to confirm the 1st term. Figure 2.1e plots the MSE against h with fixed T in the small h regime (so that the 3rd term is negligible when compared to the 2nd) to confirm that the 2nd term scales like $\mathcal{O}(h)$.

For the 3rd term in Equation (2.5), we run sufficiently many iterations to ensure all chains are well-mixed, and Figure 2.1f confirms the final MSE to scale like $\mathcal{O}(h^2)$ even for large h (as the 2nd term vanishes due to $T \to \infty$). In this sense, despite the approximations introduced by the practical implementation, the performance of Algorithm 1 is still approximated by Theorem 3, even when $M = 1$. Theorem 3 can thus guide the choices of h and T in practice.

[5]in each iteration of the outer MCMC loop, EWSG consumes $M + 1$ data points, and hence in a fair comparison with fixed computation budget (e.g. E total gradient calls), EWSG runs $\frac{E}{M+1}$ iterations which is decreasing in M.

Table 2.1: Accuracy, log likelihood and wall time of various algorithms on test data after one data pass (mean $\pm$ std).

Method	SGLD	pSGLD	SGHMC	EWSG	FlyMC
Accuracy(%)	75.283 $\pm$ 0.016	75.126 $\pm$ 0.020	75.268 $\pm$ 0.017	**75.306 $\pm$ 0.016**	75.199 $\pm$ 0.080
Log Likelihood	-0.525 $\pm$ 0.000	-0.526 $\pm$ 0.000	-0.525 $\pm$ 0.000	**-0.523 $\pm$ 0.000**	**-0.523 $\pm$ 0.000**
Wall Time (s)	**3.085 $\pm$ 0.283**	4.312 $\pm$ 0.359	3.145 $\pm$ 0.307	3.755 $\pm$ 0.387	291.295 $\pm$ 56.368

2.7.2 Bayesian Logistic Regression

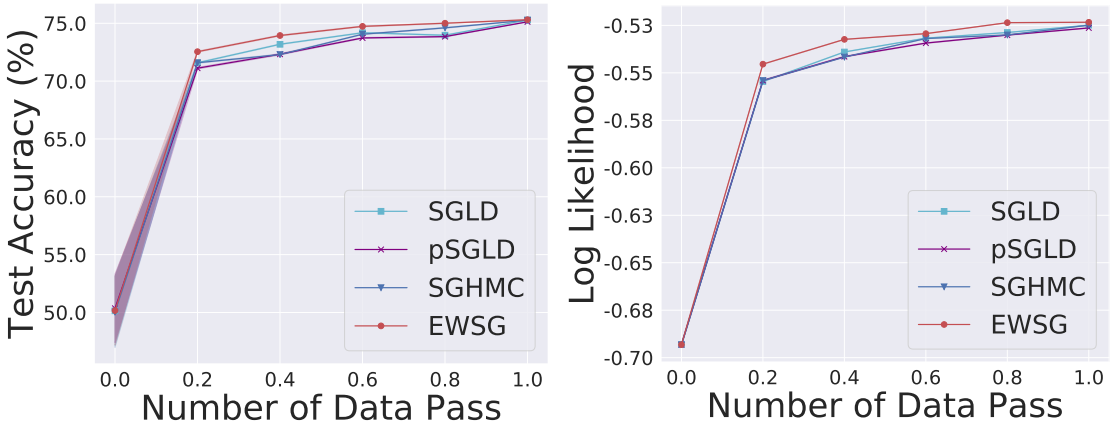
Consider Bayesian logistic regression for classification problems. The probabilistic model for predicting a label y_k given a feature vector x_k is $p(y_k = 1|\boldsymbol{x}_k, \boldsymbol{\theta}) = 1/(1+\exp(-\theta^T \boldsymbol{x}_k))$. We set a Gaussian prior with zero mean and covariance $\Sigma = 10I_d$ for $\boldsymbol{\theta}$. We conduct our experiments on Covertype data set[6], which contains 581,012 data points and 54 features. Given the large size of this data set, SG is needed to scale up MCMC methods. We use 80% of data for training and the rest 20% for testing.

The FlyMC algorithm[7] use a lower bound derived in [41] for likelihood function. For underdamped Langevin based algorithms, we set friction coefficient $\gamma = 50$. After tuning, we set the step size as $\{1, 3, 0.02, 5, 5\} \times 10^{-3}$ for SGULD, EWSG, SGLD, pSGLD and FlyMC. All algorithms are run for one data pass, with minibatch size of 50 (for FlyMC, it means 50 data are sampled in each iteration to switch state). 100 independent samples are drawn from each algorithm to estimate statistics. To further smooth out noise, all experiments are repeated 1000 times with different seeds.

Results are shown in Figure 2.2a, 2.2b and Table 2.1. EWSG outperforms others, except for log likelihood being comparable to FlyMC, which is an *exact* MCMC method. The wall time consumed by EWSG is only slightly more than that of SGLD and SGHMC, but fewer than pSGLD and order-of-magnitude fewer than FlyMC.

[6] https://archive.ics.uci.edu/ml/datasets/covertype

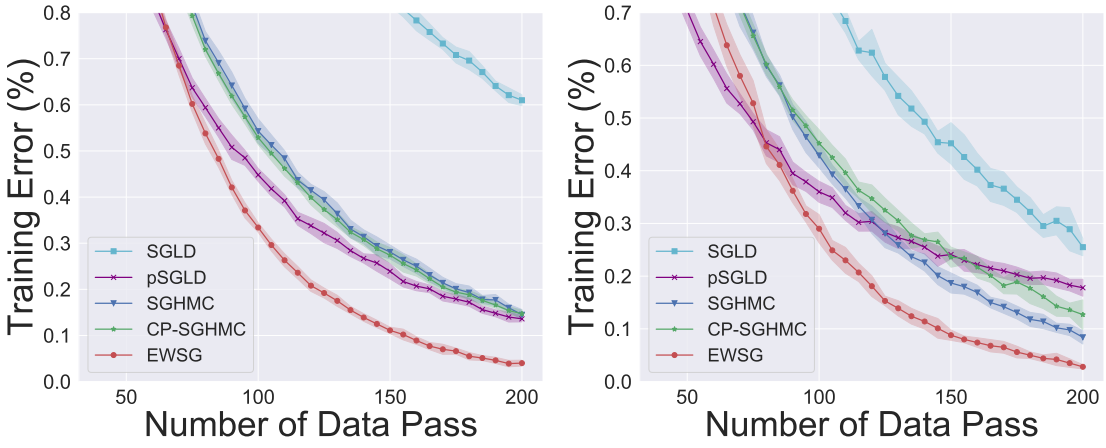
[7] https://github.com/HIPS/firefly-monte-carlo/tree/master/flymc



(a) Test Accuracy

(b) Test Log Likelihood

Figure 2.2: Bayesian logistic regression learning curve. The shaded area stands for one standard deviation.



(a) MLP architecture

(b) CNN architecture

Figure 2.3: Bayesian neural network learning curve. The shaded area stands for one standard deviation.

2.7.3 Bayesian Neural Network

Bayesian inference is compelling for deep learning [67]. Two popular architecture of neural nets are experimented – multilayer perceptron (MLP) and convolutional neural nets (CNN). In MLP, a hidden layer with 100 neurons followed by a softmax layer is used. In CNN, we use standard network configuration with 2 convolutional layers followed by 2 fully connected layers [68]. Both convolutional layers use 5×5 convolution kernel with 32 and 64 channels, 2×2 max pooling layers follow immediately after convolutional layer. The last two fully-connected layers each has 200 neurons. We set the standard normal as prior for all weights and bias.

We test algorithms on the MNIST data set, consisting of 60000 training data and 10000 test data, each datum is a 28×28 gray-scale image with one of the ten possible labels (digits $0 \sim 9$). For ULD based algorithms , we set friction coefficient $\gamma = 0.1$ in MLP and $\gamma = 1.0$ in CNN. In MLP, the step sizes are set $h = \{4, 2, 2\} \times 10^{-3}$ for EWSG, SGHMC and CP-SGHMC, and $h = \{0.001, 1\} \times 10^{-4}$ for SGLD and pSGLD, via grid search. For CP-SGHMC , (clustering-based preprocessing is conducted [42] before SGHMC) we use K-means with 10 clusters to preprocess the data set. In CNN, the step sizes are set $h = \{4, 2, 2\} \times 10^{-3}$ for EWSG, SGHMC and CP-SGHMC, and $h = \{0.02, 8\} \times 10^{-6}$ for SGLD and pSGLD, via grid search. All algorithms use minibatch size of 100 and are run for 200 epoches. For each algorithm, we generate 100 independent samples to make posterior prediction. To smooth out noise and obtain more significant results, we repeat experiments 10 times with different seeds.

The learning curve of training error is shown in Figure 2.3a and 2.3b. EWSG consistently improves over its uniform counterpart (i.e., SGHMC) and CP-SGHMC (an approximate IS SG-MCMC). Moreover, EWSG also outperforms two standard benchmarks SGLD and pSGLD. The improvement over baseline on MNIST data set is comparable to some of the early works [18, 24].

Note: in the MLP setup, the model has $d > 78400$ parameters whereas there are

$n = 60000$ data points, which shows EWSG does not require $n > d$ to work and can still outperform its uniform counterpart in the overparametrized regime (Theorem 1 demonstrates the underparametrized case only because the sparsity result is easy to understand, but EWSG doesn't only work for underparameterized models).

2.8 Conclusion

In this chapter, we proposed EWSG, which uses exponentially weighted subsampling of gradients to match the transition kernel of a base MCMC base with full gradient. The goal is better sample quality. Both local variance analysis and global non-asymptotic analysis are presented to demonstrate the advantage of EWSG theoretically. Empirical results also showed improved sampling/learning performance. We believe non-uniform stochastic gradient can be introduced to a large class of MCMC methods and capable for impactful algorithmic improvements.

CHAPTER 3

HESSIAN-FREE-HIGH-RESOLUTION NESTEROV ACCELERATION FOR SAMPLING

Optimization methods have been a major algorithmic machinery that drives both the theory and practice of machine learning in recent years. Since the seminal work of Nesterov [69], acceleration has played a key role in gradient-based optimization methods. One of the most notable examples is the Nesterov's Accelerated Gradient (NAG) method, an instance of a more general family of "momentum methods". NAG in fact consists of multiple methods, including NAG-C for convex functions, and NAG-SC for strongly convex functions, both of which have provably faster convergences than the vanilla gradient descent (GD) method in their corresponding setups [69, 70]. Although they are classical methods, significant new perspectives of acceleration have recently been studied, e.g., [71, 72, 73, 74, 75, 76]. This work will be based on NAG-SC, and 'NAG' from hereon will refer to NAG-SC unless confusion arises.

Approaches for sampling statistical distributions such as gradient-based Markov Chain Monte Carlo (MCMC) methods, at the same time, also remain of great importance in machine learning, primarily due to its link to statistical inference and the ability to capture uncertainty which is lacking in optimization-based methods. Although not entirely the same thing, there are profound and interesting interplays between optimization and sampling. For example, the perspective of viewing sampling as optimization in probability space dates back to late 90s [77], and is gaining increasing attention in machine learning commmunity [78, 79, 80, 81, 82, 83]. Discretized overdamped Langevin dynamics (OLD) [12] is commonly considered as the analog of GD in sampling, the convergence properties of its continuous dynamics and non-asymptotic analysis of discretization error are also widely studied [12, 84, 10, 26, 32, 33, 37, 29, 85, 86].

However, the notion of acceleration is less quantified in sampling compared to that in optimization, although attention has been rapidly building up. Along this direction, one line of research is based on diffusion processes, usually derived from the close connection between OLD and underdamped Langevin dynamics (ULD). For example, the convergence and nonasymptotics of discretized ULD have been studied in [28, 31, 35], and were demonstrated provably faster than discretized OLD in suitable setups. These are not only great progresses but also forming perspectives complementary to the extensive studies of the convergence of continuous ULD in the mathematical community [34, 87, 88, 89, 90, 91, 92, 93]. Another equally important line amounts to accelerating particle-based approaches for optimization in probability spaces [94, 95, 96], although we note there is no clear boundary between these two lines (e.g., [97]). More generally, it has been known that adding an irreversible part to the reversible dynamics of OLD[1] accelerates its convergence (e.g., [99, 100, 101, 102, 103]), and this work can be viewed to be under this umbrella, although discretization is also important and analyzed.

More precisely, we proposed an accelerated gradient-based MCMC algorithm termed HFHR, that is based on diffusion process and inspired by a simple yet natural motivation: how to appropriately inject noise to NAG algorithm in discrete time, so that it is turned into an algorithm for momentum-accelerated sampling? Note we don't want to add noise to the learning-rate$\to 0$ limit of NAG (which has been well studied [35]), as the discretization of this low-resolution ODE by a finite step size may not converge as fast as NAG with the same learning rate. However, we will still use continuous dynamics as intermediate steps and our roadmap is the following: first view NAG as the discretization of a high-resolution ordinary differential equation (ODE), then convert it to a stochastic differential equation (SDE) by injecting noise appropriately, and finally discretize the SDE appropriately to obtain a fast and efficient HFHR algorithm

More precisely, the first step combines ingredients from the existing literature to pre-

[1]For irreversibility-induced-acceleration *not* based on OLD, see e.g., [98] and references therein.

pare a non-asymptotic formulation for the later steps. The goal is to better account for NAG's behavior when a finite (not infinitesimal) learning rate is used. As pointed out in [76], a low-resolution limiting ODE [71], albeit being a milestone leading to a new venue of research (e.g, [72]), does not fully capture the acceleration enabled by NAG — for example, it can't distinguish between NAG and other momentum methods such as heavy ball [104]. The main reason is, the low-resolution ODE describes the $h \to 0$ limit of NAG, but in practice NAG uses a finite (nonzero) h. High-resolution ODE was thus proposed to include additional $\mathcal{O}(h)$ terms to account for the finite h effect [76], which led to a better characterization of NAG. The original form of the high-resolution ODE involves the Hessian of the objective function, which is computationally expensive to evaluate and store for high-dimensional problems, but this difficulty can be bypassed via a change of variable (e.g., [105, 106]), which allows us to derive a High-Resolution and Hessian-Free limiting ODE for NAG.

Then we make the following algorithmic innovations. One is to replace a specific coefficient in the HFHR ODE by a hyperparameter $\alpha \geq 0$, which, as we will demonstrate both theoretically and empirically, can lead to accelerated convergence of the eventual sampling algorithm The other is to add noise to the hyperparametrized HFHR ODE in a specific way, which turns it into an SDE suitable for sampling purposes. This SDE will be termed as HFHR dynamics. For obtaining an actual algorithm, the SDE needs to be discretized, and we'll just propose and analyze a relatively simple discretization for demonstrating that the accelerated convergence is not an artificial consequence of time-rescaling, which would not give acceleration after discretization with an appropriate step size. Meanwhile, we'd like to point out that our discretization is just one of the many possible schemes. It was known that high-order discretizations can improve statistical accuracy and even the speed of convergence (see e.g., [22, 107]; the analogue in (stochastic) optimization has also been studied, e.g., [108]), although such improvements often come with a cost of more computations per iteration. The discretization analyzed here is a relatively simple first-order

scheme that utilizes the structure of HFHR dynamics more than Euler-Maruyama does.

Our presentation is structured as follows. After detailing the construction of HFHR, we then theoretically analyze the convergence of HFHR, at both the continuous level (HFHR dynamics) and the discrete level (HFHR algorithm). We will first show that HFHR dynamics admits the target distribution as its invariant distribution (Theorem 5) and converges exponentially fast to it, as long as the target distribution satisfies a Poincaré's inequality (Theorem 6). Then we move on to a more specific setup of log-concave / log-strongly-concave target distributions, which is commonly considered in the literature [109, 110, 26, 31, 85, 36, 85], and demonstrate explicitly an additional acceleration of HFHR when compared to ULD (Theorem 9 / Theorem 10). For our discretized HFHR algorithm, a non-asymptotic error bound will be obtained (Theorem 11), which confirms that the acceleration of HFHR over ULD in continuous time carries through to the discrete territory, at least for log-strongly-concave target distributions. Finally, the theoretical analysis is complemented with experiment results, demonstrating the performance of HFHR on a series of representative target distributions as well as Bayesian neural network learning tasks, and numerically verifying the tightness of our theoretical results.

The main contribution of this article is the idea of adding noise to the NAG-SC optimization algorithm to turn it into a sampler, which provides a new perspective that is neither overdamped or underdamped Langevin. Theoretical analyses and numerical experiments are provided just to validate this new perspective. For instance, an example algorithm we provide (it's not unique as different discretizations can be used) achieves a $\tilde{\mathcal{O}}(\frac{\sqrt{d}}{\epsilon})$ iteration complexity in 2-Wasserstein distance, which has the same order as the best-known results for KLMC; meanwhile, we show theoretically that it has a prefactor that can be reduced by HFHR (Theorem 12 and Corollary 13), and accelerated convergence was empirically and consistently observed as well (Section 3.6).

3.1 Literature Review

Many celebrated approaches exist for establishing the exponential convergence (a.k.a. geometric ergodicity) of OLD, including the seminal work of [12], the ones using spectral gap (e.g., [26, Lemma 1]), synchronous coupling [84, p33-35][33, Proposition 1], functional inequalities such as Poincaré's inequality (PI) [10, Theorem 4.4] and logarithmic Sobolev inequality (LSI) [37, Theorem 1][35, Section 3.1]. There are also fruitful exponential convergence results for ULD, including the ones leveraging Lyapunov function [87, Theorem 3.2], hypocoercivity [91, 89, 90, 111], coupling [28, Theorem 5][31, Theorem 1][34, Theorem 2.3], modified Poincaré's inequality [88, Theorem 1] and spectral analysis [112, 92]. Generally speaking, due to technical difficulty related to lack of uniform ellipticity, the exponential convergence of ULD takes more effort to establish than OLD, especially when the potential f is not strongly convex.

However, studying the continuous processes is often not enough and they need to be discretized in order to be implemented as algorithms. The study of asymptotic convergence of discretized OLD dates back to at least the 1990s [113, 12]. The non-asymptotic convergence analysis of LMC discretization of OLD can be found in [26] and it shows the discretization achieves ϵ error, in total variation distance, in $\tilde{\mathcal{O}}(\frac{d}{\epsilon^2})$ steps. Following this, iteration complexity of discretized OLD was also quantified in different metrics, $\tilde{\mathcal{O}}(\frac{d}{\epsilon^2})$ in 2-Wasserstein distance [32] and $\tilde{\mathcal{O}}(\frac{d}{\epsilon})$ in KL divergence [29]. For discretized ULD, one has improved $\tilde{\mathcal{O}}(\frac{\sqrt{d}}{\epsilon})$ iteration complexity in 2-Wasserstein distance [28, 31] and $\tilde{\mathcal{O}}(\frac{\sqrt{d}}{\sqrt{\epsilon}})$ in KL divergence [35]. Better dimension dependence is generally conceived as a major advantage of ULD over OLD.

3.2 Terminology and Notations

The following conditions will be frequently referred to in various parts of this paper.

Assumption 1 ***(Standard Smoothness Condition)*** *Assume the potential function* $f : \mathbb{R}^d \mapsto$

$\mathbb{R}$ *is* $\mathcal{C}^2$ *and* L*-smooth, i.e., there exists a constant* $L > 0$ *such that* $\forall \boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^d$*, we have*

$$\|\nabla f(\boldsymbol{y}) - \nabla f(\boldsymbol{x})\| \leq L\|\boldsymbol{y} - \boldsymbol{x}\|.$$

(Note the above assumptions are equivalent to $\nabla^2 f \preceq LI$.)

Assumption 2 ***(Convexity)*** *The potential function* $f : \mathbb{R}^d \mapsto \mathbb{R}$ *is convex, if* $\forall \boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^d$*, we have*

$$f(\boldsymbol{y}) \geq f(\boldsymbol{x}) + \langle \nabla f(\boldsymbol{x}), \boldsymbol{y} - \boldsymbol{x} \rangle.$$

Assumption 3 ***(Strong-convexity)*** *The potential function* $f : \mathbb{R}^d \mapsto \mathbb{R}$ *is* m*-strongly-convex, if there exists* $m > 0$ *such that* $\forall \boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^d$*, we have*

$$f(\boldsymbol{y}) \geq f(\boldsymbol{x}) + \langle \nabla f(\boldsymbol{x}), \boldsymbol{y} - \boldsymbol{x} \rangle + \frac{m}{2}\|\boldsymbol{y} - \boldsymbol{x}\|^2.$$

Two metric/divergence we use to quantify convergence are χ^2 divergence and 2-Wasserstein distance

$$(\chi^2 \text{ divergence}) \qquad \chi^2(\mu_1 \| \mu_2) = \int \left(\frac{d\mu_1}{d\mu_2} - 1\right)^2 d\mu_2 \tag{3.1}$$

$$(\text{2-Wasserstein distance }) \qquad W_2(\mu_1, \mu_2) = \left(\inf_{\pi \in \Pi(\mu_1, \mu_2)} \mathbb{E}_{(\boldsymbol{X}, \boldsymbol{Y}) \sim \pi}[\|\boldsymbol{X} - \boldsymbol{Y}\|^2]\right)^{\frac{1}{2}} \tag{3.2}$$

where $\Pi(\mu_1, \mu_2)$ is all couplings of μ_1 and μ_2.

3.3 The Derivation of HFHR

HFHR is motivated by Nesterov Accelerated Gradient descent algorithm for Strongly Convex function (NAG-SC) in optimization [114]. It is obtained by formulating NAG-SC as a Hessian free high-resolution ODE (based on [76] and [105, 106]), lifting the high-resolution correction's coefficient as a free parameter, and adding appropriate Gaussian noises.

More precisely, let's start with NAG-SC algorithm:

$$\boldsymbol{x}_{k+1} = \boldsymbol{y}_k - s\nabla f(\boldsymbol{y}_k) \tag{3.3}$$

$$\boldsymbol{y}_{k+1} = \boldsymbol{x}_{k+1} + c(\boldsymbol{x}_{k+1} - \boldsymbol{x}_k) \tag{3.4}$$

where s is the learning rate (also known as step size), and $c = \frac{1-\sqrt{ms}}{1+\sqrt{ms}}$ is a constant chosen according to step size s and the strong convexity coefficient m of f, although the method also works for non-strongly-convex f.

A high-resolution ODE description of Equation (3.3) and (3.4) is obtained in [76, Section 2]

$$\ddot{\boldsymbol{y}} + \sqrt{s}\left(\frac{2(1-c)}{s(1+c)} + \nabla^2 f(\boldsymbol{y})\right)\dot{\boldsymbol{y}} + \frac{2}{1+c}\nabla f(\boldsymbol{y}) = \boldsymbol{0} \tag{3.5}$$

which can better account for the effect of non-infinitesimal s than the $s \to 0$ limit. However, in this original form, Hessian of f is involved, which is expensive to compute and store especially for high-dimensional problems.

To obtain a Hessian-free high-resolution ODE description of Equation (3.3) and (3.4), we first turn the iteration into a 'mechanical' version by introducing position variable $\boldsymbol{q}_k = \boldsymbol{y}_k$ and momentum variable $\boldsymbol{p}_k = \frac{(\boldsymbol{y}_k - \boldsymbol{x}_k)}{h}$. Replacing $\boldsymbol{x}_{k+1}$ in Equation (3.3) and the first $\boldsymbol{x}_{k+1}$ in Equation (3.4) by $\boldsymbol{q}_{k+1}$ and $\boldsymbol{p}_{k+1}$, the second $\boldsymbol{x}_{k+1}$ in Equation (3.4) by $\boldsymbol{q}_k - s\nabla f(\boldsymbol{q}_k)$, and the $\boldsymbol{x}_k$ in Equation (3.4) by $\boldsymbol{q}_k$ and $\boldsymbol{p}_k$, we obtain

$$\begin{cases} \boldsymbol{q}_{k+1} = \boldsymbol{q}_k + h\boldsymbol{p}_{k+1} - s\nabla f(\boldsymbol{q}_k) \\ \boldsymbol{p}_{k+1} = c\boldsymbol{p}_k - c\frac{s}{h}\nabla f(\boldsymbol{q}_k) \end{cases}$$

Now, choose γ, α and h as $h = \sqrt{cs}, \gamma = \frac{1-c}{h}, \alpha = \frac{s}{h}$. It is easy to see that $\gamma > 0$, $\alpha > 0$,

then NAG-SC exactly rewrites as

$$\begin{cases} \boldsymbol{q}_{k+1} = \boldsymbol{q}_k + h\boldsymbol{p}_{k+1} - h\alpha\nabla f(\boldsymbol{q}_k) \\ \boldsymbol{p}_{k+1} = \boldsymbol{p}_k - h\gamma\boldsymbol{p}_k - h\nabla f(\boldsymbol{q}_k) \end{cases}. \tag{3.6}$$

Similar ideas for bypassing the Hessian without introducing any approximation are well described in the literature (e.g., [105, 106]).

So far, both h and α are actually determined by the hyperparameter s of NAG-SC. However, if we now consider α as an independent variable and let $h \to 0$, we see Equation (3.6) is a 1st-order discretization (with step size h) of the dynamics

$$\begin{cases} \dot{\boldsymbol{q}} = \boldsymbol{p} - \alpha\nabla f(\boldsymbol{q}) \\ \dot{\boldsymbol{p}} = -\gamma\boldsymbol{p} - \nabla f(\boldsymbol{q}) \end{cases}. \tag{3.7}$$

Note α, if inherited from NAG-SC, should be $\alpha = \sqrt{s/c} = \mathcal{O}(h)$, which, in a low-resolution ODE will be discarded, and this eventually leads to ULD rather than HFHR. However, we now allow it to be a free parameter and will see that $\alpha \neq \mathcal{O}(h)$ can be advantageous.

Before quantify these advantages later on, we finish the derivation by appropriately injecting Gaussian noises to Equation (3.7). This is just like how OLD can be obtained by adding noise to gradient flow. The right amount and structure of noise turn the ODE into a Markov process that can serve our purpose of sampling, and the detailed form of our noise is described by the following:

$$\begin{cases} d\boldsymbol{q}_t = \boldsymbol{p}_t - \alpha\nabla f(\boldsymbol{q}_t) + \sqrt{2\alpha}d\boldsymbol{W}_t \\ d\boldsymbol{p}_t = -\gamma\boldsymbol{p}_t - \nabla f(\boldsymbol{q}_t) + \sqrt{2\gamma}d\boldsymbol{B}_t \end{cases}. \tag{3.8}$$

Here $\alpha \geq 0, \gamma > 0$ are constant parameters, and $\boldsymbol{W}_t, \boldsymbol{B}_t$ are independent standard Brownian motions in $\mathbb{R}^d$. This irreversible diffusion process will be named as **Hessian-Free**

High-Resolution(**HFHR**) dynamics. We will write it as HFHR(α, γ) to emphasize the dependence on α and γ when needed.

Note that HFHR can be viewed as a mixture of an ULD and a rescaled OLD

$$\underbrace{\begin{cases} \frac{d}{dt}\boldsymbol{q}_t = \boldsymbol{p}_t \\ \frac{d}{dt}\boldsymbol{p}_t = -\gamma \boldsymbol{p}_t - \nabla f(\boldsymbol{q}_t) + \sqrt{2\gamma} d\boldsymbol{B}_t \end{cases}}_{\text{ULD}}, \quad \underbrace{\begin{cases} \frac{d}{dt}\boldsymbol{q}_t = -\alpha \nabla f(\boldsymbol{q}_t) + \sqrt{2\alpha} d\boldsymbol{W}_t \\ \frac{d}{dt}\boldsymbol{p}_t = \boldsymbol{0} \end{cases}}_{\text{rescaled OLD}} \tag{3.9}$$

As both OLD and ULD have π as invariant distribution (for OLD, it is more precisely just the $\boldsymbol{q}$ marginal of π), it is not surprising that the invariant distribution of HFHR is also π as shown in the following theorem.

Theorem 5 *π is the invariant distribution of HFHR described in Equation* (3.8).

Proof: The Fokker-Plank equation of HFHR is given by

$$\partial_t \rho_t = -\nabla_{\boldsymbol{x}} \cdot \left(\begin{bmatrix} \boldsymbol{p} \\ -\nabla f(\boldsymbol{q}) \end{bmatrix} \rho_t \right) + \alpha \left(\nabla_{\boldsymbol{q}} \cdot (\nabla f(\boldsymbol{q}) \rho_t) + \Delta_{\boldsymbol{q}} \rho_t \right) + \gamma \left(\nabla_{\boldsymbol{p}} \cdot (\boldsymbol{p} \rho_t) + \Delta_{\boldsymbol{p}} \rho_t \right)$$

where $\nabla_{\boldsymbol{x}} = (\nabla_{\boldsymbol{q}}, \nabla_{\boldsymbol{p}})$. For $\pi \propto e^{-f(\boldsymbol{q}) - \frac{1}{2}\|\boldsymbol{p}\|^2}$, we have

$$\nabla_{\boldsymbol{x}} \cdot \left(\begin{bmatrix} \boldsymbol{p} \\ -\nabla f(\boldsymbol{q}) \end{bmatrix} \pi \right) = \left\langle \begin{bmatrix} \boldsymbol{p} \\ -\nabla f(\boldsymbol{q}) \end{bmatrix}, \nabla_{\boldsymbol{x}} \pi \right\rangle = 0,$$

$$\Delta_{\boldsymbol{q}} \pi = -\nabla_{\boldsymbol{q}} \cdot (\pi \nabla f(\boldsymbol{q}))$$

$$\Delta_{\boldsymbol{p}} \pi = -\nabla_{\boldsymbol{p}} \cdot (\pi \boldsymbol{p})$$

Therefore $\partial_t \pi = 0$ and hence π is the invariant distribution of HFHR. ∎

Remark: Note, however, that the 'decomposition' in Equation (3.9) is only formal, and the convergence of Equation (3.8) can be very different from that of ULD or OLD. In fact,

even for a linear system $\dot{x} = Ax + Bx$, its dynamics can already be very different from any finite composition of the flow maps of $\dot{x} = Ax$ and $\dot{x} = Bx$ unless $[A, B] = 0$. The high nonlinearity only makes the mixture of Equation (3.8) even more different from ULD and OLD.

Remark: The OLD part in Equation (3.9) is a time-rescaled version of the original OLD in Equation (1.1). If we only had the OLD part, choosing a large α would seemingly accelerate the convergence but this is meaningless from an algorithmic point of view, because if one discretizes the original OLD in Equation (1.1) using a step size h, the rescaled version should use a step size h/α. However, this is no longer the case when ULD and OLD are summed, as the nonlinear interaction between them will change both the convergence rate and stability limit in nonlinear ways. The acceleration enabled by HFHR is genuine; see Section 3.4&3.5.

3.4 Theoretical Analysis of the Continuous HFHR

In this section, we establish exponential convergence guarantees of HFHR in several different setups. We show the most general result in Theorem 6 which only requires target measures satisfying Poincaré's inequality (PI). Both Theorem 9 and 10 demonstrate additional acceleration to ULD respectively under log-concavity and log-strong-concavity assumption on target measures.

Before presenting theoretical results, we introduce a few notations that will be used in the main text as well as some proof.

$$L' = \sqrt{2}\max\left\{\sqrt{1+\alpha^2}\max\left\{\frac{1}{\sqrt{2}}, L\right\}, \sqrt{1+\gamma^2}\right\}$$

is the Lipschitz constant of the drift $\begin{bmatrix} \boldsymbol{p} - \alpha\nabla f(\boldsymbol{q}) \\ -\gamma\boldsymbol{p} - \nabla f(\boldsymbol{q}) \end{bmatrix}$ in HFHR dynamics, first appeared in Lemma 32

$$P = \begin{bmatrix} \gamma I & I \\ 0 & \sqrt{1+\alpha\gamma} I \end{bmatrix}$$

is a $2d \times 2d$ matrix with which we show contraction property of HFHR dynamics in Lemma 33. Denote the largest and the smallest singular value of P by

$$\sigma_{\max} = \sqrt{\frac{\alpha\gamma}{2} + \frac{\gamma^2}{2} + \frac{\sqrt{\alpha^2\gamma^2 - 2\alpha\gamma^3 + 4\alpha\gamma + \gamma^4 + 4}}{2} + 1},$$
$$\sigma_{\min} = s\sqrt{\frac{\alpha\gamma}{2} + \frac{\gamma^2}{2} - \frac{\sqrt{\alpha^2\gamma^2 - 2\alpha\gamma^3 + 4\alpha\gamma + \gamma^4 + 4}}{2} + 1},$$

and its condition number by

$$\kappa' = \frac{\sigma_{\max}}{\sigma_{\min}} = \sqrt{\frac{\frac{\alpha\gamma}{2} + \frac{\gamma^2}{2} + \frac{\sqrt{\alpha^2\gamma^2 - 2\alpha\gamma^3 + 4\alpha\gamma + \gamma^4 + 4}}{2} + 1}{\frac{\alpha\gamma}{2} + \frac{\gamma^2}{2} - \frac{\sqrt{\alpha^2\gamma^2 - 2\alpha\gamma^3 + 4\alpha\gamma + \gamma^4 + 4}}{2} + 1}}.$$

λ' characterizes the rate of exponential convergence of HFHR dynamics and is defined as

$$\lambda' = \min\left\{\frac{m}{\gamma} + \alpha m, \frac{\gamma^2 - L}{\gamma}\right\}$$

given that $\gamma^2 > L$.

We first show HFHR converges exponentially fast as long as μ satisfies a Poincaré's inequality.

Theorem 6 *Suppose $\alpha > 0$ and the target measure μ satisfies a Poincaré's inequality*

$$\int (g - \int g d\mu)^2 \leq \frac{1}{\lambda_{PI}(\mu)} \int \|\nabla g\|^2 d\mu \tag{3.10}$$

for any $g \in C^2(\mathbb{R}^d) \cap L^2(\mathbb{R}^d, \mu)$ with some positive constant $\lambda_{PI}(\mu) > 0$. Then we have

$$\chi^2(\rho_t, \pi) \leq \chi^2(\rho_0, \pi) \exp\left(-2\min\{\lambda_{PI}(\mu), 1\} \min\{\alpha, \gamma\} t\right)$$

where ρ_t is the joint law of $(\boldsymbol{q}_t, \boldsymbol{p}_t)$ in HFHR.

Proof: The Fokker-Planck equation of HFHR is given by

$$\partial_t \rho_t + \nabla \cdot (\rho_t \boldsymbol{J}) = 0, \quad \text{where } \boldsymbol{J} = \begin{pmatrix} \begin{bmatrix} \boldsymbol{p} - \alpha \nabla f(\boldsymbol{q}) - \alpha \nabla_{\boldsymbol{q}} \log \rho_t \\ -\gamma \boldsymbol{p} - \nabla f(\boldsymbol{q}) - \gamma \nabla_{\boldsymbol{p}} \log \rho_t \end{bmatrix} \end{pmatrix} \tag{3.11}$$

Since

$$\nabla \cdot \left(\rho_t \begin{bmatrix} -\nabla_{\boldsymbol{p}} \log \rho_t \\ \nabla_{\boldsymbol{q}} \log \rho_t \end{bmatrix} \right) = 0,$$

we then have

$$\boldsymbol{J} = \frac{\pi}{\rho_t} \begin{bmatrix} -\alpha I & I \\ -I & -\gamma I \end{bmatrix} \begin{bmatrix} \nabla_{\boldsymbol{q}} \frac{\rho_t}{\pi} \\ \nabla_{\boldsymbol{p}} \frac{\rho_t}{\pi} \end{bmatrix}.$$

By Lemma 27, π satisfies PI with constant $\lambda_{\text{PI}}(\pi) = \min\{\lambda_{\text{PI}}(\mu), 1\}$ as it is well known that ν satisfies Poincaré's inequality with $\lambda_{\text{PI}}(\nu) = 1$ [115, Theorem 3.20]. The time derivative of $\chi^2(\rho_t, \pi)$ is

$$\frac{d}{dt}\chi^2(\rho_t, \pi) = -\int 2(\frac{\rho_t}{\pi} - 1)\, \nabla \cdot (\rho_t \boldsymbol{J})\, d\boldsymbol{x} = 2\int \langle \nabla \frac{\rho_t}{\pi}, \boldsymbol{J} \rangle \rho_t \, d\boldsymbol{x} \leq -2\min\{\alpha, \gamma\} \int \|\nabla \frac{\rho_t}{\pi}\|^2 d\pi$$

Let $g = \frac{\rho_t}{\pi} - 1$, by Poincaré's inequality, we have $\int \|\frac{\rho_t}{\pi} - 1\|^2 d\pi \leq \frac{1}{\lambda_{\text{PI}}(\pi)} \int \|\nabla \frac{\rho_t}{\pi}\|^2 d\pi$. Therefore we have the time derivative of $\chi^2(\rho_t, \pi)$ is bounded by

$$\frac{d}{dt}\chi^2(\rho_t, \pi) \leq -2\lambda_{\text{PI}}(\pi) \min\{\alpha, \gamma\} \chi^2(\rho_t, \pi) = -2\min\{\lambda_{\text{PI}}(\mu), 1\} \min\{\alpha, \gamma\} \chi^2(\rho_t, \pi)$$

and the desired result follows by Gronwall's inequality. ∎

Remark: One always has $\chi^2(\mu_1 \| \mu_2) \geq \text{KL}(\mu_1 \| \mu_2) \geq \frac{1}{2}\|\mu_1 - \mu_2\|_{\text{TV}}^2$, due to the relation to KL divergence and Pinsker's inequality, hence exponential convergence in χ^2 divergence also implies that in KL divergence and total variation distance.

The Poincaré's inequality assumption holds for a large family of measures, including

log-concave ones as shown in the next two propositions. Therefore, Theorem 6 has broad applicability beyond just log-concave measures.

Proposition 7 *[91, Theorem A.19] If* $\lim_{\|\boldsymbol{x}\|\to\infty}\left(\frac{\|\nabla f(\boldsymbol{x})\|^2}{2}-\Delta f(\boldsymbol{x})\right)=\infty$*, then* μ *satisfies the Poincaré's inequality.*

Proposition 8 *[116] Every log-concave measure* $\mu\propto e^{-f(\boldsymbol{x})}$ *satisfies the Poincaré's inequality.*

We now focus on convex and L-smooth f (i.e., Assumption 1 and 2) for which we will obtain much tighter bounds. Proposition 8 ensures log-concave target distribution $\mu\propto e^{-f(\boldsymbol{q})}$ satisfies PI with some positive constant $\lambda_{\text{PI}}(\mu)\triangleq\lambda>0$. Then:

Theorem 9 *Under Assumption 1 and 2, additionally assume* $\gamma^2\geq\max\{2\lambda,L\}$*,* $\alpha\leq\frac{\gamma}{\lambda}-\frac{2}{\gamma}$*, we have*

$$\chi^2(\rho_t\|\pi)\leq e^{-(\frac{\sqrt{\lambda}}{2\gamma}+\frac{\sqrt{\lambda}}{16}\alpha)t}C$$

where $C=\left\{\chi^2(\rho_0\|\pi)+\mathbb{E}_\pi\left[\left\langle\nabla_{\boldsymbol{x}}\frac{\rho_0}{\pi},S\nabla_{\boldsymbol{x}}\frac{\rho_0}{\pi}\right\rangle\right]\right\}$ *is a constant determined by the initial condition,* ρ_t *is the law of* $(\boldsymbol{q}_t,\boldsymbol{p}_t)$*,* $\nabla_{\boldsymbol{x}}=(\nabla_{\boldsymbol{q}},\nabla_{\boldsymbol{p}})$ *and* $S=\frac{1}{\gamma}\begin{bmatrix}(\frac{2}{\gamma}+\alpha)I & I\\ I & \gamma I\end{bmatrix}$.

Proof: Two main tools we use in this proof are

- a carefully-crafted Lyapunov function, motivated by [35] and
- the Poincaré's inequality of the joint invariant distribution π.

More specifically, we will consider the following Lypunov function

$$\mathcal{L}(\rho_t)=\chi^2(\rho_t\|\pi)+\mathbb{E}_\pi[\left\langle\nabla_{\boldsymbol{x}}\frac{\rho_t}{\pi},S\nabla_{\boldsymbol{x}}\frac{\rho_t}{\pi}\right\rangle] \tag{3.12}$$

where $\nabla_{\boldsymbol{x}} = (\nabla_{\boldsymbol{q}}, \nabla_{\boldsymbol{p}})$ and $S = \begin{bmatrix} aI & bI \\ bI & dI \end{bmatrix} \in \mathbb{R}^{2d\times 2d}$ is a positive definite matrix to be determined later. Denote

$$\mathcal{L}_{\text{cross}}(\rho_t) = \mathbb{E}_\pi[\langle \nabla_{\boldsymbol{x}} \frac{\rho_t}{\pi}, S\nabla_{\boldsymbol{x}} \frac{\rho_t}{\pi} \rangle]. \tag{3.13}$$

It is well known that the standard Gaussian measure ν satisfies PI with PI constant $\lambda_{\text{PI}}(\nu) = 1$ [115]. Therefore, by Lemma 27, the joint invariant distribution π satisfies PI with PI constant $\lambda_{\text{PI}}(\pi) = \min\{1, \lambda_{\text{PI}}(\mu)\} = \min\{1, \lambda\}$. This dependence on λ, however, is undesirable when $\lambda \gg 1$ because it does not reflect fast convergence of HFHR.

In order to unify the two cases $\lambda >$ and $\lambda < 1$, we will work with a rescaled version of HFHR. To this end, we need to introduce a larger class of dynamics parametrized by an inverse temperature parameter β

$$\begin{cases} d\boldsymbol{q} = \big(\boldsymbol{p} - \alpha\nabla f(\boldsymbol{q})\big)\, dt + \sqrt{2\alpha\beta^{-1}} d\boldsymbol{B}_t^1 \\ d\boldsymbol{p} = \big(-\gamma\boldsymbol{p} - \nabla f(\boldsymbol{q})\big)\, dt + \sqrt{2\gamma\beta^{-1}} d\boldsymbol{B}_t^2 \end{cases}. \tag{3.14}$$

We refer it as tempered HFHR and denote it by tempered-HFHR(α, γ, β). It is easy to see that the plain HFHR described in Equation (3.8) is a tempered HFHR with $\beta = 1$, i.e. tempered-HFHR$(\alpha, \beta, 1)$.

Rescaling Since $\mu \propto e^{-f(\boldsymbol{q})}$ satisfies PI with PI constant λ, it is easy to see that $\tilde{\mu} \propto e^{-\tilde{f}(\boldsymbol{q})}$ where $f(\boldsymbol{q}) = \lambda\tilde{f}(\boldsymbol{q})$, satisfies PI with PI constant 1.

With the rescaled potential $\tilde{f}$, HFHR process rewrites as

$$\begin{cases} d\boldsymbol{q} = (\boldsymbol{p} - \alpha\lambda\nabla\tilde{f}(\boldsymbol{q}))dt + \sqrt{2\alpha} d\boldsymbol{B}_t^1 \\ d\boldsymbol{p} = (-\gamma\boldsymbol{p} - \lambda\nabla\tilde{f}(\boldsymbol{q}))dt + \sqrt{2\gamma} d\boldsymbol{B}_t^2 \end{cases}$$

Introduce rescaled velocity $\tilde{\boldsymbol{p}}$ via $\boldsymbol{q}(t) = \sqrt{\lambda}\tilde{\boldsymbol{p}}(t)$, then the SDE becomes

$$\begin{cases} d\boldsymbol{q} = (\sqrt{\lambda}\tilde{\boldsymbol{p}} - \alpha\lambda\nabla\tilde{f}(q))dt + \sqrt{2\alpha}d\boldsymbol{B}_t^1 \\ d\tilde{\boldsymbol{p}} = (-\gamma\tilde{\boldsymbol{p}} - \sqrt{\lambda}\nabla\tilde{f}(q))dt + \sqrt{2\gamma/\lambda}d\boldsymbol{B}_t^2 \end{cases}$$

Introduce rescaled dissipation parameters $\tilde{\alpha}, \tilde{\gamma}$ via $\alpha = \tilde{\alpha}\sqrt{\lambda}^{-1}$, $\gamma = \tilde{\gamma}\sqrt{\lambda}$. Then the SDE rewrites as

$$\begin{cases} d\boldsymbol{q} = (\sqrt{\lambda}\tilde{\boldsymbol{p}} - \tilde{\alpha}\sqrt{\lambda}\nabla\tilde{f}(\boldsymbol{q}))dt + \sqrt{2\tilde{\alpha}\sqrt{\lambda}^{-1}}d\boldsymbol{B}_t^1 \\ d\tilde{\boldsymbol{p}} = (-\tilde{\gamma}\sqrt{\lambda}\tilde{\boldsymbol{p}} - \sqrt{\lambda}\nabla\tilde{f}(\boldsymbol{q}))dt + \sqrt{2\tilde{\gamma}\sqrt{\lambda}^{-1}}d\boldsymbol{B}_t^2 \end{cases}$$

Rescale time via $\tau = \sqrt{\lambda}t$, then

$$\begin{cases} d\boldsymbol{q} = (\tilde{\boldsymbol{p}} - \tilde{\alpha}\nabla\tilde{f}(\boldsymbol{q}))d\tau + \sqrt{2\tilde{\alpha}\lambda^{-1}}d\boldsymbol{B}_\tau^1 \\ d\tilde{\boldsymbol{p}} = (-\tilde{\gamma}\tilde{\boldsymbol{p}} - \nabla\tilde{f}\boldsymbol{q}))d\tau + \sqrt{2\tilde{\gamma}\lambda^{-1}}d\boldsymbol{B}_\tau^2 \end{cases} \tag{3.15}$$

It is easy to see that the rescaled HFHR in Equation (3.15) is a tempered-HFHR($\tilde{\alpha}, \tilde{\gamma}, \lambda$).

Apply Lemma 28 to the Tempered HFHR In Equation (3.15), we have that $\tilde{f} \in C^2(\mathbb{R}^d)$ is convex and $\frac{L}{\lambda}$-smooth. Moreover, $\tilde{\mu} \propto e^{-\tilde{f}(\boldsymbol{q})}$ satisfies PI with PI constant 1.

Since

$$\begin{aligned} \tilde{\gamma}^2 \geq \max\{2, \frac{L}{\lambda}\} &\iff \gamma^2 \geq \max\{2\lambda, L\} \\ \tilde{\alpha} \leq \tilde{\gamma} - \frac{2}{\tilde{\gamma}} &\iff \alpha \leq \frac{\gamma}{\lambda} - \frac{2}{\gamma}, \end{aligned}$$

we can then apply Lemma 28 to the rescaled HFHR in Equation (3.15) and obtain the following result

$$\chi^2(\tilde{\rho}_\tau \| \tilde{\pi}) \leq e^{-(\frac{1}{2\tilde{\gamma}} + \frac{1}{16}\tilde{\alpha})\tau} \left\{ \chi^2(\tilde{\rho}_0 \| \tilde{\pi}) + \mathbb{E}_{\tilde{\pi}} \left[\langle \nabla_{\tilde{\boldsymbol{x}}} \frac{\tilde{\rho}_0}{\tilde{\pi}}, \tilde{S} \nabla_{\tilde{\boldsymbol{x}}} \frac{\tilde{\rho}_0}{\tilde{\pi}} \rangle \right] \right\}, \tag{3.16}$$

where $\tilde{\rho}_\tau(\boldsymbol{q}, \tilde{\boldsymbol{p}})$ is the law of $(\boldsymbol{q}_\tau, \tilde{\boldsymbol{p}}_\tau)$ at time τ, $\tilde{\pi} \propto e^{-\lambda \tilde{H}(\boldsymbol{q}, \tilde{\boldsymbol{p}})}$ with $\tilde{H}(\boldsymbol{q}, \tilde{\boldsymbol{p}}) = \tilde{f}(\boldsymbol{q}) + \frac{1}{2}\|\tilde{\boldsymbol{p}}\|^2$, $\nabla_{\tilde{\boldsymbol{x}}} = (\nabla_{\boldsymbol{q}}, \nabla_{\tilde{\boldsymbol{p}}})$ and $\tilde{S} = \begin{bmatrix} \tilde{a}I & \tilde{b}I \\ \tilde{b}I & \tilde{d}I \end{bmatrix}$ with $\tilde{a} = (\frac{2}{\tilde{\gamma}} + \tilde{\alpha})\tilde{b}, \tilde{d} = \tilde{\gamma}\tilde{b}, \tilde{b} = \frac{1}{\tilde{\gamma}\lambda}$ is a positive definite matrix.

Substitute Back We first write all measures with respect to the original position and momentum variables $(\boldsymbol{q}, \boldsymbol{p})$. Since $\begin{bmatrix} \boldsymbol{q} \\ \boldsymbol{p} \end{bmatrix} = \begin{bmatrix} I & 0 \\ 0 & \sqrt{\lambda} I \end{bmatrix} \begin{bmatrix} \boldsymbol{q} \\ \tilde{\boldsymbol{p}} \end{bmatrix} \triangleq P \begin{bmatrix} \boldsymbol{q} \\ \tilde{\boldsymbol{p}} \end{bmatrix}$, by change of variable formula we have

$$
\begin{aligned}
\rho_\tau(\boldsymbol{q}, \boldsymbol{p}) =& \tilde{\rho}_\tau(\boldsymbol{q}, \frac{1}{\sqrt{\lambda}}\boldsymbol{p})\lambda^{-\frac{d}{2}} \\
\pi(\boldsymbol{q}, \boldsymbol{p}) =& \tilde{\pi}(\boldsymbol{q}, \frac{1}{\sqrt{\lambda}}\boldsymbol{p})\lambda^{-\frac{d}{2}} \propto \exp\{-\lambda\left[\tilde{f}(\boldsymbol{q}) + \frac{1}{2}\|\frac{\boldsymbol{p}}{\sqrt{\lambda}}\|^2\right]\} = \exp\{f(\boldsymbol{q}) - \frac{1}{2}\|\boldsymbol{p}\|^2\}
\end{aligned}
$$

So $\pi(\boldsymbol{q}, \boldsymbol{p})$ is indeed the joint invariant distribution defined in Equation (1.3). Therefore,

$$
\begin{aligned}
\chi^2(\tilde{\rho}_\tau \| \tilde{\pi}) &= \int (\frac{\tilde{\rho}_\tau(\boldsymbol{q}, \tilde{\boldsymbol{p}})}{\tilde{\pi}(\boldsymbol{q}, \tilde{\boldsymbol{p}})} - 1)^2 \tilde{\pi}(\boldsymbol{q}, \tilde{\boldsymbol{p}}) d\boldsymbol{q} d\tilde{\boldsymbol{p}} \\
&= \int (\frac{\tilde{\rho}_\tau(\boldsymbol{q}, \frac{\boldsymbol{p}}{\sqrt{\lambda}})}{\tilde{\pi}(\boldsymbol{q}, \frac{\boldsymbol{p}}{\sqrt{\lambda}})} - 1)^2 \tilde{\pi}(\boldsymbol{q}, \frac{\boldsymbol{p}}{\sqrt{\lambda}})\lambda^{-\frac{d}{2}} d\boldsymbol{q} d\boldsymbol{p} \qquad (\boldsymbol{p} = \sqrt{\lambda}\tilde{\boldsymbol{p}}) \\
&= \int (\frac{\rho_\tau(\boldsymbol{q}, \boldsymbol{p})}{\pi(\boldsymbol{q}, \boldsymbol{p})} - 1)^2 \pi(\boldsymbol{q}, \boldsymbol{p}) d\boldsymbol{q} d\boldsymbol{p} \\
&= \chi^2(\rho_\tau \| \pi)
\end{aligned}
$$

Similar derivation, combined with chain rule, leads to

$$
\mathbb{E}_{\tilde{\pi}}\left[\langle \nabla_{\tilde{\boldsymbol{x}}} \frac{\tilde{\rho}_\tau}{\tilde{\pi}}, \tilde{S} \nabla_{\tilde{\boldsymbol{x}}} \frac{\tilde{\rho}_\tau}{\tilde{\pi}} \rangle\right] = \mathbb{E}_{\pi}\left[\langle P \nabla_{\boldsymbol{x}} \frac{\rho_\tau}{\pi}, \tilde{S} P \nabla_{\boldsymbol{x}} \frac{\rho_\tau}{\pi} \rangle\right]
$$

where $\nabla_{\boldsymbol{x}} = (\nabla_{\boldsymbol{q}}, \nabla_{\boldsymbol{p}})$.

Next we substitute back the original parameters $\gamma = \tilde{\gamma}\sqrt{\lambda}$ and $\alpha = \frac{\tilde{\alpha}}{\sqrt{\lambda}}$ and rewrite the

result in Equation (3.16) as

$$\chi^2(\rho_\tau\|\pi) \leq e^{-(\frac{\sqrt{\lambda}}{2\gamma}+\frac{\sqrt{\lambda}\alpha}{16})\tau}\left\{\chi^2(\rho_0\|\pi)+\mathbb{E}_\pi\left[\langle\nabla_{\boldsymbol{x}}P\frac{\rho_0}{\pi},\tilde{S}P\nabla_{\boldsymbol{x}}\frac{\rho_0}{\pi}\rangle\right]\right\},$$

Now write $S \triangleq P^T\tilde{S}P = \frac{1}{\gamma}\begin{bmatrix}(\frac{2}{\gamma}+\alpha)I & I\\ I & \gamma I\end{bmatrix}$ write τ as t we obtain

$$\chi^2(\rho_t\|\pi) \leq e^{-(\frac{\sqrt{\lambda}}{2\gamma}+\frac{\sqrt{\lambda}\alpha}{16})t}\left\{\chi^2(\rho_0\|\pi)+\mathbb{E}_\pi\left[\langle\nabla_{\boldsymbol{x}}P\frac{\rho_0}{\pi},SP\nabla_{\boldsymbol{x}}\frac{\rho_0}{\pi}\rangle\right]\right\} \tag{3.17}$$

■

Remark: Assumptions on the lower bound of γ such as $\gamma^2 > \max\{2\lambda, L\}$ are also made in existing works [31, 34]. Assumption $\alpha \leq \frac{\gamma}{\lambda} - \frac{2}{\gamma}$ is to ensure additional linear acceleration, i.e. a coefficient $\frac{\sqrt{\lambda}}{16}$ independent of α. For large α, there will still be additional acceleration, not necessarily linear in α, though.

When f is not only convex but also m-strongly convex, we have the following result:

Theorem 10 *Under Assumption 1, 3 and further suppose f is m-strongly convex, $\gamma^2 > L + m$ and $\alpha \leq \frac{\gamma^2-L-m}{m\gamma}$, denote the law of $\boldsymbol{q}_t$ by μ_t, then there exists a constant $\kappa' > 0$ depending only on α and γ, such that*

$$W_2(\mu_t,\mu) \leq \kappa' e^{-(\frac{m}{\gamma}+m\alpha)t}W_2(\mu_0,\mu).$$

Proof: Consider two copies of HFHR that are driven by the same Brownian motion

$$\begin{cases} d\boldsymbol{q}_t = (\boldsymbol{p}_t - \alpha\nabla f(\boldsymbol{q}_t))dt + \sqrt{2\alpha}d\boldsymbol{B}_t^1 \\ d\boldsymbol{p}_t = (-\gamma\boldsymbol{p}_t - \nabla f(\boldsymbol{q}_t))dt + \sqrt{2\gamma}d\boldsymbol{B}_t^2 \end{cases}, \quad \begin{cases} d\tilde{\boldsymbol{q}}_t = (\tilde{\boldsymbol{p}}_t - \alpha\nabla f(\tilde{\boldsymbol{q}}_t))dt + \sqrt{2\alpha}d\boldsymbol{B}_t^1 \\ d\tilde{\boldsymbol{p}}_t = (-\gamma\tilde{\boldsymbol{p}}_t - \nabla f(\tilde{\boldsymbol{q}}_t))dt + \sqrt{2\gamma}d\boldsymbol{B}_t^2 \end{cases},$$

where we set $(\tilde{\boldsymbol{q}}_0, \tilde{\boldsymbol{p}}_0) \sim \pi$, $\boldsymbol{p}_0 = \tilde{\boldsymbol{p}}_0$ and $\boldsymbol{q}_0$ such that

$$W_2^2(\mu_0, \mu) = \mathbb{E}\left[\|\boldsymbol{q}_0 - \tilde{\boldsymbol{q}}_0\|_2^2\right], \quad \boldsymbol{q}_0 \sim \mu_0$$

Denote $\begin{bmatrix} \boldsymbol{\phi}_t \\ \boldsymbol{\psi}_t \end{bmatrix} = P \begin{bmatrix} \boldsymbol{q}_t - \tilde{\boldsymbol{q}}_t \\ \boldsymbol{p}_t - \tilde{\boldsymbol{p}}_t \end{bmatrix}$ where $P = \begin{bmatrix} \gamma I & I \\ 0 & \sqrt{1+\alpha\gamma} I \end{bmatrix}$. By Lemma 33 and the assumption on α, γ, we have

$$\left\| \begin{bmatrix} \boldsymbol{\phi}_t \\ \boldsymbol{\psi}_t \end{bmatrix} \right\|^2 \leq e^{-2(\frac{m}{\gamma} + m\alpha)t} \left\| \begin{bmatrix} \boldsymbol{\phi}_0 \\ \boldsymbol{\psi}_0 \end{bmatrix} \right\|^2.$$

Therefore we obtain

$$\begin{aligned}
W_2^2(\mu_t, \mu) = \inf_{(\boldsymbol{q}_t, \tilde{\boldsymbol{q}}_t) \sim \Pi(\mu_t, \mu)} \mathbb{E}\|\boldsymbol{q}_t - \tilde{\boldsymbol{q}}_t\|^2 \leq & \inf_{(\boldsymbol{q}_t, \tilde{\boldsymbol{q}}_t) \sim \Pi(\mu_t, \mu), (\boldsymbol{p}_t, \tilde{\boldsymbol{p}}_t) \sim \Pi(\nu_t, \nu)} \mathbb{E} \left\| \begin{bmatrix} \boldsymbol{q}_t - \tilde{\boldsymbol{q}}_t \\ \boldsymbol{p}_t - \tilde{\boldsymbol{p}}_t \end{bmatrix} \right\|^2 \\
\leq & \mathbb{E}\|P^{-1}\|_2^2 \left\| \begin{bmatrix} \boldsymbol{\phi}_t \\ \boldsymbol{\psi}_t \end{bmatrix} \right\|^2 \\
\leq & \mathbb{E}\|P^{-1}\|_2^2 e^{-2(\frac{m}{\gamma} + m\alpha)t} \left\| \begin{bmatrix} \boldsymbol{\phi}_0 \\ \boldsymbol{\psi}_0 \end{bmatrix} \right\|^2 \\
\leq & (\kappa')^2 e^{-2(\frac{m}{\gamma} + m\alpha)t} \left\| \begin{bmatrix} \boldsymbol{q}_0 - \tilde{\boldsymbol{q}}_0 \\ \boldsymbol{p}_0 - \tilde{\boldsymbol{p}}_0 \end{bmatrix} \right\|^2 \\
= & (\kappa')^2 e^{-2(\frac{m}{\gamma} + m\alpha)t} W_2^2(\mu_0, \mu)
\end{aligned}$$

Taking square root yields the desired result. ∎

Respectively, Theorem 9 and 10 state that HFHR converges to the target distribution exponentially fast in log-concave and log-strongly-concave setup.

Before demonstrating the advantage of HFHR over ULD, we will first need to inspect the bound for HFHR when $\alpha = 0$, i.e. ULD, to ensure it is a tight bound for ULD. To this

Table 3.1: Comparison of convergence rate of HFHR and ULD with known dependence on parameters of dynamics. In log-strongly-concave setup, we write $m = \lambda$ due to Bakry-Émery condition [117] and denote condition number $\kappa = \frac{L}{m}$. $\rho > 0$ is the LSI constant assmued in [35]. The column of γ contains the values of γ corresponding to the best rate.

Dynamics	Setup		γ	Metric
	log-concave	log-strongly-concave		
ULD [31, Theorem 1]	N/A	$\frac{\sqrt{m}}{\sqrt{\kappa}+\sqrt{\kappa-1}}$	$2\sqrt{L}$	W_2
ULD [88, Theorem 1]	$\mathcal{O}(\sqrt{\lambda})$	$\mathcal{O}(\sqrt{m})$	$\sqrt{\lambda}$	χ^2
ULD [35, Theorem 1]	$\frac{\rho}{10}$	$\frac{\rho}{10}$	2	KL
HFHR (Theorem 9)	$\frac{\sqrt{\lambda}}{2\sqrt{L+2\lambda}} + \frac{\sqrt{\lambda}}{16}\alpha$	$\frac{1}{2\sqrt{\kappa+2}} + \frac{\sqrt{m}}{16}\alpha$	$\sqrt{L+2\lambda}$	χ^2
HFHR (Theorem 10)	N/A	$\frac{\sqrt{m}}{2\sqrt{\kappa}} + m\alpha$	$2\sqrt{L}$	W_2

end, we compare our bound with several existing convergence results for ULD with known dependence on the parameters of dynamics in Table 3.1.

In the log-concave setup, the $\mathcal{O}(\sqrt{\lambda})$ rate from [88] is optimal when $\lambda \to 0$ and can be realized by isotropic quadratic potential [88, Remark 1.1]. This new result is enabled by assuming growth condition on the Hessian of f and a compact embedding condition, in additional to PI assumption. Our result for ULD $\frac{\sqrt{\lambda}}{2\sqrt{L+2\lambda}} = \mathcal{O}(\sqrt{\lambda})$ is comparable to the optimal one in the same regime, if $L = \mathcal{O}(1)$. However, for large L, our result for ULD is in general weaker than the optimal one, but we can nonetheless pick $\alpha = \mathcal{O}(1)$ so that the rate of HFHR is still comparable to ULD. Our proof of Theorem 9 is motivated by a powerful machinery proposed in a recent work on ULD [35]. In [35], it is assumed that μ satisfies logarithmic Soblev inequality (LSI), which is known to be stronger than PI [118, 119], and does not necessarily hold for generic log-concave measures [120]. The rate $\mathcal{O}(\rho)$, however, is not directly comparable with other results as [35] works with a rescaled ULD [2].

In the log-strongly-concave setup, [31, Theorem 1] obtained exponential convergence result in 2-Wasserstein distance with rate $\frac{\sqrt{m}}{\sqrt{\kappa}+\sqrt{\kappa-1}}$ using a simple and elegant coupling ap-

[2]How a rescaling affects convergence rate can be found in [28, Lemma 8] and [31, Theorem 1].

proach, and showed this rate is optimal as it is achieved by the bivariate function $f(x, y) = \frac{m}{2}x^2 + \frac{L}{2}y^2$. In Theorem 10, we use the same coupling approach to obtain an (asymptotically) equivalent rate $\frac{\sqrt{m}}{2\sqrt{\kappa}}$.

After showing our convergence rate results for HFHR(0, γ) are comparable with optimal rates for ULD in many cases, the acceleration of HFHR immediately becomes evident. For example, if we push α to the upper bound specified in Theorem 10, we obtain rate $\mathcal{O}(\sqrt{L})$ in log-strongly-concave setup. Compared with the rate in [31], this is a speedup of order κ.

3.5 Discretization

We consider in this section the discretization of the proposed HFHR dynamics and work with constant step size h. Inspired by Strang's splitting for partial differential equations, which is known to have 2nd order accuracy [121, 122], we run the follow Strang-style splitting scheme for HFHR dynamics in each time interval $[kh, (k+1)h]$

$$\phi^{\frac{h}{2}} \circ \psi^h \circ \phi^{\frac{h}{2}}(\boldsymbol{x}_{kh}) \tag{3.18}$$

where $\boldsymbol{x}_{kh} = \begin{bmatrix} \boldsymbol{q}_{kh} \\ \boldsymbol{p}_{kh} \end{bmatrix}$, ϕ flow and ψ flow are respectively defined as

$$\phi : \begin{cases} d\boldsymbol{q} = \boldsymbol{p}dt \\ d\boldsymbol{p} = -\gamma \boldsymbol{p}dt + \sqrt{2\gamma}d\boldsymbol{B} \end{cases} \qquad \psi : \begin{cases} d\boldsymbol{q} = -\alpha \nabla f(\boldsymbol{q})dt + \sqrt{2\alpha}d\boldsymbol{W} \\ d\boldsymbol{p} = -\nabla f(\boldsymbol{q})dt \end{cases}$$

and $\phi^t(\boldsymbol{x}_0)/\psi^t(\boldsymbol{x}_0)$ means running ϕ/ψ flow for t time with $\boldsymbol{x}_0$ as its initial value.

Note that ϕ flow can be solved explicitly since the second equation is an Ornstein-Unlenbeck process and integrating the second equation followed by integrating the first

one gives us an explicit solution

$$\begin{cases} \boldsymbol{q}_t = \boldsymbol{q}_0 + \frac{1-e^{-\gamma t}}{\gamma}\boldsymbol{p}_0 + \sqrt{2\gamma}\int_0^t \frac{1-e^{-\gamma(t-s)}}{\gamma} d\boldsymbol{B}(s), \\ \boldsymbol{p}_t = e^{-\gamma t}\boldsymbol{p}_0 + \sqrt{2\gamma}\int_0^t e^{-\gamma(t-s)} d\boldsymbol{B}(s). \end{cases} \tag{3.19}$$

For an implementation of the stochastic integral part in Equation (3.19), denote $\boldsymbol{X} = \sqrt{2\gamma}\int_0^t \frac{1-e^{-\gamma(t-s)}}{\gamma} d\boldsymbol{B}(s)$ and $\boldsymbol{Y} = \sqrt{2\gamma}\int_0^t e^{-\gamma(t-s)} d\boldsymbol{B}(s)$, we have

$$\mathrm{Cov}(\boldsymbol{X}, \boldsymbol{Y}) = \begin{bmatrix} \frac{\gamma h + 4e^{-\gamma\frac{h}{2}} - e^{-\gamma h} - 3}{\gamma^2} I_d & \frac{(1-e^{-\gamma\frac{h}{2}})^2}{\gamma} I_d \\ \frac{(1-e^{-\gamma\frac{h}{2}})^2}{\gamma} I_d & (1-e^{-\gamma h}) I_d \end{bmatrix}.$$

Applying Cholesky decomposition, we obtain $\begin{bmatrix} \boldsymbol{X} \\ \boldsymbol{Y} \end{bmatrix} = M\boldsymbol{\xi}$ where M is the matrix square root of $\mathrm{Cov}(\boldsymbol{X}, \boldsymbol{Y})$, $\boldsymbol{\xi}$ is a $2d$ standard Gaussian random vector and can be readily simulated.

However, ψ flow is generally not explicitly solvable unless f is a quadratic function in $\boldsymbol{q}$. We hence approximate $\psi^h(\boldsymbol{x}_0)$ with one-step Euler-Maruyama integration

$$\psi^h(\boldsymbol{x}_0) \approx \widetilde{\psi}^h(\boldsymbol{x}_0) = \begin{cases} \boldsymbol{q}_h = -\alpha\nabla f(\boldsymbol{q}_0)h + \sqrt{2\alpha h}\boldsymbol{\eta} \\ \boldsymbol{p}_h = -\nabla f(\boldsymbol{q}_0)h \end{cases}$$

where $\boldsymbol{\eta}$ is a standard d-dimensional Gaussian random vector. Thus, one step of an implementable Strang's splitting for HFHR is hence

$$\phi^{\frac{h}{2}} \circ \widetilde{\psi}^h \circ \phi^{\frac{h}{2}} \tag{3.20}$$

We refer the algorithm depicted in Equation (3.20) as HFHR algorithm and the algorithm is detailed in Algorithm 2. The proposed Strang's splitting for HFHR dynamics has a strong error with order 1 as the following Theorem characterizes

Algorithm 2 HFHR Algorithm

Input: potential function f and its gradient ∇f, damping coefficients α and γ, step size h, initial condition $(\boldsymbol{q}_0, \boldsymbol{p}_0)$

$k = 0$ and initialize $\begin{bmatrix} \boldsymbol{q}_0 \\ \boldsymbol{p}_0 \end{bmatrix}$

while not converge **do**

Generate independent standard Gaussian random vectors $\boldsymbol{\eta}_{k+1} \in \mathbb{R}^d, \boldsymbol{\xi}^1_{k+1}, \boldsymbol{\xi}^2_{k+1} \in \mathbb{R}^{2d}$

Run $\phi^{\frac{h}{2}}$: $\begin{bmatrix} \boldsymbol{q}_1 \\ \boldsymbol{p}_1 \end{bmatrix} = \begin{bmatrix} \boldsymbol{q}_{kh} + \frac{1-e^{-\gamma\frac{h}{2}}}{\gamma}\boldsymbol{p}_{kh} \\ e^{-\gamma\frac{h}{2}}\boldsymbol{p}_{kh} \end{bmatrix} + M\boldsymbol{\xi}^1_{k+1}$

Run $\widetilde{\psi}^h$: $\begin{bmatrix} \boldsymbol{q}_2 \\ \boldsymbol{p}_2 \end{bmatrix} = \begin{bmatrix} \boldsymbol{q}_1 - \alpha\nabla f(\boldsymbol{q}_1)h + \sqrt{2\alpha h}\boldsymbol{\eta}_{k+1} \\ \boldsymbol{p}_1 - \nabla f(\boldsymbol{q}_1)h \end{bmatrix}$

Run $\phi^{\frac{h}{2}}$: $\begin{bmatrix} \boldsymbol{q}_3 \\ \boldsymbol{p}_3 \end{bmatrix} = \begin{bmatrix} \boldsymbol{q}_2 + \frac{1-e^{-\gamma\frac{h}{2}}}{\gamma}\boldsymbol{p}_2 \\ e^{-\gamma\frac{h}{2}}\boldsymbol{p}_2 \end{bmatrix} + M\boldsymbol{\xi}^2_{k+1}$

$\begin{bmatrix} \boldsymbol{q}_{(k+1)h} \\ \boldsymbol{p}_{(k+1)h} \end{bmatrix} \leftarrow \begin{bmatrix} \boldsymbol{q}_3 \\ \boldsymbol{p}_3 \end{bmatrix}$

$k \leftarrow k + 1$

end while

Theorem 11 ***[Discretization error of Algorithm 2** in L_2] Under Assumption 1, 3, and further suppose the operator $\nabla\Delta f$ grows at most linearly, i.e. $\left\|\nabla\Delta f(\boldsymbol{q})\right\| \le G\sqrt{1+\|\boldsymbol{q}\|^2}, \forall \boldsymbol{q} \in \mathbb{R}^d$. Assume without loss of generality that $\mathbf{0} \in \operatorname{argmin}_{\boldsymbol{x}\in\mathbb{R}^d} f(\boldsymbol{x})$. Also suppose γ in HFHR dynamics satisfy $\gamma^2 > L$.*

Then there exists $C > 0$, such that for $0 < h \le h_0 \triangleq \min\{\frac{1}{4\kappa' L'}, h_1, h_2, h_3\}$ where

$$
\begin{aligned}
h_1 =& \frac{\sqrt{\lambda'}}{4\sqrt{2}\kappa' L \max\{\alpha + 1.25, \gamma + 1\}(1.92 + 2.30\alpha L)}, \\
h_2 =& \frac{\lambda'}{16\sqrt{2}\kappa'(L+G)\max\{\alpha + 1.25, \gamma + 1\}(1.74 + 0.71\alpha)}, \\
h_3 =& \frac{\lambda'}{8\kappa' L \max\{\alpha + 1.25, \gamma + 1\}(1.92 + 2.30\alpha L)},
\end{aligned}
$$

we have

$$
\left(\mathbb{E}\|\boldsymbol{x}_k - \bar{\boldsymbol{x}}_k\|^2\right)^{\frac{1}{2}} \le Ch
$$

where $\bar{\boldsymbol{x}}_k$ is the k-th iterate of Algorithm 2 with step size h starting from $\boldsymbol{x}_0$, $\boldsymbol{x}_k$ is the solution of HFHR dynamics at time kh, starting from $\boldsymbol{x}_0$. This result holds uniformly for all $k \geq 0$ and k can go to ∞. In particular, $C = \mathcal{O}(\sqrt{d})$ and if $\gamma - \frac{L+m}{\gamma} \geq m\alpha$, then there exist $b_1, b_2 > 0$, both independent of α and are of order $\mathcal{O}(\sqrt{d})$, such that

$$C \leq \frac{b_1\alpha^3 + b_2}{\frac{m}{\gamma} + m\alpha}.$$

κ', L', λ' are constants depending only on L, m, γ, α.

Proof: Denote $t_k = kh$, the solution of the HFHR dynamics at time t by $\boldsymbol{x}_{0,\boldsymbol{x}_0}(t)$, the k-th iterates of the Strang's splitting method of HFHR dynamics by $\bar{\boldsymbol{x}}_{0,\boldsymbol{x}_0}(kh)$. Both $\boldsymbol{x}_{0,\boldsymbol{x}_0}(t)$ and $\bar{\boldsymbol{x}}_{0,\boldsymbol{x}_0}(kh)$ start from the same initial value $\boldsymbol{x}_0$. Let $P \triangleq \begin{bmatrix} \gamma I & I \\ 0 & \sqrt{1+\alpha\gamma} I \end{bmatrix}$, that transforms the solution of HFHR dynamics into $\boldsymbol{y}_{0,P\boldsymbol{x}_0}(t) = P\boldsymbol{x}_{0,\boldsymbol{x}_0}(t)$ and the Strang's splitting discretization of HFHR into $\bar{\boldsymbol{y}}_{0,P\boldsymbol{x}_0}(t) = P\bar{\boldsymbol{x}}_{0,\boldsymbol{x}_0}(t)$.

For the ease of notation, we write $\boldsymbol{y}_{0,\boldsymbol{y}_0}(t_k)$ as $\boldsymbol{y}_k$ and $\bar{\boldsymbol{y}}_{0,\boldsymbol{y}_0}(t_k)$ as $\bar{\boldsymbol{y}}_k$. We have the following identity

$$\begin{aligned}
\mathbb{E}\left\|\boldsymbol{y}_{k+1} - \bar{\boldsymbol{y}}_{k+1}\right\|^2 =& \mathbb{E}\left\|\boldsymbol{y}_{t_k,\boldsymbol{y}_k}(h) - \bar{\boldsymbol{y}}_{t_k,\bar{\boldsymbol{y}}_k}(h)\right\|^2 \\
=& \mathbb{E}\left\|\boldsymbol{y}_{t_k,\boldsymbol{y}_k}(h) - \boldsymbol{y}_{t_k,\bar{\boldsymbol{y}}_k}(h) + \boldsymbol{y}_{t_k,\bar{\boldsymbol{y}}_k}(h) - \bar{\boldsymbol{y}}_{t_k,\bar{\boldsymbol{y}}_k}(h)\right\|^2 \\
=& \underbrace{\mathbb{E}\left\|\boldsymbol{y}_{t_k,\boldsymbol{y}_k}(h) - \boldsymbol{y}_{t_k,\bar{\boldsymbol{y}}_k}(h)\right\|^2}_{①} + \underbrace{\mathbb{E}\left\|\boldsymbol{y}_{t_k,\bar{\boldsymbol{y}}_k}(h) - \bar{\boldsymbol{y}}_{t_k,\bar{\boldsymbol{y}}_k}(h)\right\|^2}_{②} \\
&+ 2\underbrace{\mathbb{E}\left\langle \boldsymbol{y}_{t_k,\boldsymbol{y}_k}(h) - \boldsymbol{y}_{t_k,\bar{\boldsymbol{y}}_k}(h), \boldsymbol{y}_{t_k,\bar{\boldsymbol{y}}_k}(h) - \bar{\boldsymbol{y}}_{t_k,\bar{\boldsymbol{y}}_k}(h)\right\rangle}_{③}
\end{aligned}$$

By Lemma 33, when $0 < h < \frac{1}{2\lambda'}$, term (1) can be upper bounded as

$$\begin{aligned}\mathbb{E}\left\|\boldsymbol{y}_{t_k,\boldsymbol{y}_k}(h) - \boldsymbol{y}_{t_k,\bar{\boldsymbol{y}}_k}(h)\right\|^2 \leq & e^{-2\lambda' h}\mathbb{E}\|\boldsymbol{y}_k - \bar{\boldsymbol{y}}_k\|^2 \\ \leq & \left(1 - 2\lambda' h + 2(\lambda')^2 h^2\right)\mathbb{E}\|\boldsymbol{y}_k - \bar{\boldsymbol{y}}_k\|^2 \\ \leq & \left(1 - \lambda' h\right)\mathbb{E}\|\boldsymbol{y}_k - \bar{\boldsymbol{y}}_k\|^2\end{aligned}$$

where the second inequality is due to $e^{-x} \leq 1 - x + \frac{x^2}{2}, \forall x > 0$.

For term (2), we have by Lemma 36 that

$$\mathbb{E}\left\|\boldsymbol{y}_{t_k,\bar{\boldsymbol{y}}_k}(h) - \bar{\boldsymbol{y}}_{t_k,\bar{\boldsymbol{y}}_k}(h)\right\|^2 \leq \sigma_{\max}^2\,\mathbb{E}\left\|\boldsymbol{x}_{t_k,\bar{\boldsymbol{x}}_k}(h) - \bar{\boldsymbol{x}}_{t_k,\bar{\boldsymbol{x}}_k}(h)\right\|^2 \leq \sigma_{\max}^2\, C_2^2 h^3$$

where $\sigma_{\max}$ is the largest singular value of matrix P.

For term (3), we have by Lemma 30 that

$$\begin{aligned}& 2\mathbb{E}\left\langle \boldsymbol{y}_{t_k,\boldsymbol{y}_k}(h) - \boldsymbol{y}_{t_k,\bar{\boldsymbol{y}}_k}(h), \boldsymbol{y}_{t_k,\bar{\boldsymbol{y}}_k}(h) - \bar{\boldsymbol{y}}_{t_k,\bar{\boldsymbol{y}}_k}(h)\right\rangle \\ =& 2\mathbb{E}\left\langle \boldsymbol{y}_k - \bar{\boldsymbol{y}}_k + \boldsymbol{z}, \boldsymbol{y}_{t_k,\bar{\boldsymbol{y}}_k}(h) - \bar{\boldsymbol{y}}_{t_k,\bar{\boldsymbol{y}}_k}(h)\right\rangle \\ =& \underbrace{2\mathbb{E}\left\langle \boldsymbol{y}_k - \bar{\boldsymbol{y}}_k, \boldsymbol{y}_{t_k,\bar{\boldsymbol{y}}_k}(h) - \bar{\boldsymbol{y}}_{t_k,\bar{\boldsymbol{y}}_k}(h)\right\rangle}_{(3a)} + \underbrace{2\mathbb{E}\left\langle \boldsymbol{z}, \boldsymbol{y}_{t_k,\bar{\boldsymbol{y}}_k}(h) - \bar{\boldsymbol{y}}_{t_k,\bar{\boldsymbol{y}}_k}(h)\right\rangle}_{(3b)}\end{aligned}$$

For term (3a), by the tower property of conditional expectation, we have

$$
\begin{aligned}
&2\mathbb{E}\left\langle \boldsymbol{y}_k - \bar{\boldsymbol{y}}_k, \boldsymbol{y}_{t_k,\bar{\boldsymbol{y}}_k}(h) - \bar{\boldsymbol{y}}_{t_k,\bar{\boldsymbol{y}}_k}(h)\right\rangle \\
=&2\mathbb{E}\left[\mathbb{E}\left[\left\langle \boldsymbol{y}_k - \bar{\boldsymbol{y}}_k, \boldsymbol{y}_{t_k,\bar{\boldsymbol{y}}_k}(h) - \bar{\boldsymbol{y}}_{t_k,\bar{\boldsymbol{y}}_k}(h)\right\rangle \middle| \mathcal{F}_k\right]\right] \\
=&2\mathbb{E}\left\langle \boldsymbol{y}_k - \bar{\boldsymbol{y}}_k, \mathbb{E}\left[\boldsymbol{y}_{t_k,\bar{\boldsymbol{y}}_k}(h) - \bar{\boldsymbol{y}}_{t_k,\bar{\boldsymbol{y}}_k}(h)\middle| \mathcal{F}_k\right]\right\rangle \\
\leq&2\sqrt{\mathbb{E}\|\boldsymbol{y}_k - \bar{\boldsymbol{y}}_k\|^2}\sqrt{\mathbb{E}\left\|\mathbb{E}\left[\boldsymbol{y}_{t_k,\bar{\boldsymbol{y}}_k}(h) - \bar{\boldsymbol{y}}_{t_k,\bar{\boldsymbol{y}}_k}(h)\middle| \mathcal{F}_k\right]\right\|^2} \\
\leq&2\sqrt{\mathbb{E}\|\boldsymbol{y}_k - \bar{\boldsymbol{y}}_k\|^2}\sqrt{\sigma_{\max}^2\mathbb{E}\left\|\mathbb{E}\left[\boldsymbol{x}_{t_k,\bar{\boldsymbol{x}}_k}(h) - \bar{\boldsymbol{x}}_{t_k,\bar{\boldsymbol{x}}_k}(h)\middle| \mathcal{F}_k\right]\right\|^2} \\
\leq&2\sqrt{\mathbb{E}\|\boldsymbol{y}_k - \bar{\boldsymbol{y}}_k\|^2}\sqrt{\sigma_{\max}^2 C_1^2 h^4} \\
\leq&2\sigma_{\max}C_1\sqrt{\mathbb{E}\|\boldsymbol{y}_k - \bar{\boldsymbol{y}}_k\|^2}h^2.
\end{aligned}
$$

For term (3b), when $0 < h < \frac{1}{4L''}$ we have by Lemma 30 and Lemma 36

$$
\begin{aligned}
&2\mathbb{E}\left\langle \boldsymbol{z}, \boldsymbol{y}_{t_k,\bar{\boldsymbol{y}}_k}(h) - \bar{\boldsymbol{y}}_{t_k,\bar{\boldsymbol{y}}_k}(h)\right\rangle \\
\leq&2\sqrt{\mathbb{E}\|\boldsymbol{z}\|^2}\sqrt{\mathbb{E}\left\|\boldsymbol{y}_{t_k,\bar{\boldsymbol{y}}_k}(h) - \bar{\boldsymbol{y}}_{t_k,\bar{\boldsymbol{y}}_k}(h)\right\|^2} \\
=&2\sqrt{\mathbb{E}\|\boldsymbol{z}\|^2}\sqrt{\mathbb{E}\left[\mathbb{E}\left[\left\|\boldsymbol{y}_{t_k,\bar{\boldsymbol{y}}_k}(h) - \bar{\boldsymbol{y}}_{t_k,\bar{\boldsymbol{y}}_k}(h)\right\|^2\middle| \mathcal{F}_k\right]\right]} \\
=&2\sqrt{\mathbb{E}\|\boldsymbol{z}\|^2}\sqrt{\sigma_{\max}^2\mathbb{E}\left[\mathbb{E}\left[\left\|\boldsymbol{x}_{t_k,\bar{\boldsymbol{x}}_k}(h) - \bar{\boldsymbol{x}}_{t_k,\bar{\boldsymbol{x}}_k}(h)\right\|^2\middle| \mathcal{F}_k\right]\right]} \\
\leq&2\sigma_{\max}\sqrt{\tilde{C}\mathbb{E}\|\boldsymbol{y}_k - \bar{\boldsymbol{y}}_k\|^2 h^2}\sqrt{C_2^2 h^3} \\
\leq&2\sigma_{\max}C_2\sqrt{\tilde{C}}\sqrt{\mathbb{E}\|\boldsymbol{y}_k - \bar{\boldsymbol{y}}_k\|^2}h^{\frac{5}{2}}
\end{aligned}
$$

where $\tilde{C} = 2\left(L''\right)^2 = 2(\kappa')^2\left(L'\right)^2$ is from Lemma 30 and Lemma 32.

Recall both C_1 and C_2 depend on $\|\boldsymbol{x}_k\|$ and we would like to upper bound this term. To this end, consider $\tilde{\boldsymbol{x}}(t)$, a solution of HFHR dynamics with initial value $\tilde{\boldsymbol{x}}_0$ that follows the invariant distribution $\tilde{\boldsymbol{x}}_0 \sim \pi$ and realizes $W_2(\pi_0, \pi)$, i.e., $\mathbb{E}\|\tilde{\boldsymbol{x}}_0 - \boldsymbol{x}_0\|^2 = W_2^2(\pi_0, \pi)$.

Denote $\tilde{\boldsymbol{x}}_k = \tilde{\boldsymbol{x}}(kh)$ and $e_k = \left(\mathbb{E}\|\boldsymbol{y}_k - \bar{\boldsymbol{y}}_k\|^2\right)^{\frac{1}{2}}$, we then have

$$
\begin{aligned}
\mathbb{E}\|\bar{\boldsymbol{x}}_k\|^2 =& \mathbb{E}\|\boldsymbol{x}_k + \bar{\boldsymbol{x}}_k - \boldsymbol{x}_k\|^2 \\
\leq& 2\mathbb{E}\|\boldsymbol{x}_k\|^2 + 2\mathbb{E}\|\bar{\boldsymbol{x}}_k - \boldsymbol{x}_k\|^2 \\
\leq& 4\mathbb{E}\|\tilde{\boldsymbol{x}}_k\|^2 + 4\mathbb{E}\|\tilde{\boldsymbol{x}}_k - \boldsymbol{x}_k\|^2 + 2\mathbb{E}\|\bar{\boldsymbol{x}}_k - \boldsymbol{x}_k\|^2 \\
=& 4\mathbb{E}\|\tilde{\boldsymbol{x}}_k\|^2 + 4\mathbb{E}\left\|P^{-1}P(\tilde{\boldsymbol{x}}_k - \boldsymbol{x}_k)\right\|^2 + 2\mathbb{E}\left\|P^{-1}P(\bar{\boldsymbol{x}}_k - \boldsymbol{x}_k)\right\|^2 \\
\leq& 4\left(\int_{\mathbb{R}^d} \|\boldsymbol{q}\|^2 \, d\mu + d\right) + \frac{4}{\sigma_{\min}^2}\mathbb{E}\left\|P(\tilde{\boldsymbol{x}}_k - \boldsymbol{x}_k)\right\|^2 + \frac{2}{\sigma_{\min}^2}\mathbb{E}\|\bar{\boldsymbol{y}}_k - \boldsymbol{y}_k\|^2 \\
\overset{(i)}{\leq}& 4\left(\int_{\mathbb{R}^d} \|\boldsymbol{q}\|^2 \, d\mu + d\right) + \frac{4}{\sigma_{\min}^2}e^{-2\lambda' kh}\mathbb{E}\left\|P(\tilde{\boldsymbol{x}}_0 - \boldsymbol{x}_0)\right\|^2 + \frac{2}{\sigma_{\min}^2}e_k^2 \\
\leq& 4\left(\int_{\mathbb{R}^d} \|\boldsymbol{q}\|^2 \, d\mu + d\right) + 4\kappa^2 W_2^2(\pi_0, \pi) + \frac{2}{\sigma_{\min}^2}e_k^2 \\
\triangleq& Fe_k^2 + G
\end{aligned}
$$

where (i) is due to Lemma 33. Recall from Lemma 36, we have

$$
\begin{aligned}
C_1 &\leq A_1\sqrt{\mathbb{E}\|\bar{x}_k\|^2} + B_1 \leq A_1\sqrt{F}e_k + (A_1\sqrt{G} + B_1) \triangleq U_1 e_k + V_1 \\
C_2 &\leq A_2\sqrt{\mathbb{E}\|\bar{x}_k\|^2} + B_2 \leq A_2\sqrt{F}e_k + (A_2\sqrt{G} + B_2) \triangleq U_2 e_k + V_2
\end{aligned}
$$

where

$$
\begin{aligned}
A_1 =& (L+G)\max\{\alpha + 1.25, \gamma + 1\}(1.74 + 0.71\alpha) \\
B_1 =& (L+G)\max\{\alpha + 1.25, \gamma + 1\}\left[0.5\alpha + (1.26\sqrt{\alpha} + 1.14\alpha\sqrt{\alpha} + 2.32\sqrt{\gamma})\sqrt{hd}\right] \\
A_2 =& L\max\{\alpha + 1.25, \gamma + 1\}(1.92 + 2.30\alpha L)\sqrt{h} \\
B_2 =& L\max\{\alpha + 1.25, \gamma + 1\}(2.60\sqrt{\alpha} + 3.34\sqrt{\gamma}h)\sqrt{d}
\end{aligned}
$$

Combine the above and bounds for terms (1), (2), (3a) and (3b), we then obtain

$$
\begin{aligned}
&e_{k+1}^2 \\
\leq&(1-\lambda' h)e_k^2 + \sigma_{\max}^2 C_2^2 h^3 + 2\sigma_{\max} C_1 e_k h^2 + 2\sigma_{\max} C_2 \sqrt{\tilde{C}} e_k h^{\frac{5}{2}} \\
\leq&(1-\lambda' h)e_k^2 + \sigma_{\max}^2 2(U_2^2 e_k^2 + V_2^2) h^3 + 2\sigma_{\max}(U_1 e_k + V_1) e_k h^2 + 2\sigma_{\max}(U_2 e_k + V_2)\sqrt{\tilde{C}} e_k h^{\frac{5}{2}} \\
=&\left(1 - \lambda' h + 2\sigma_{\max}^2 U_2^2 h^3 + 2\sigma_{\max} U_1 h^2 + 2\sigma_{\max} U_2 \sqrt{\tilde{C}} h^{\frac{5}{2}}\right) e_k^2 \\
&+ \left(2\sigma_{\max} V_1 + 2\sigma_{\max} V_2 \sqrt{\tilde{C}h}\right) e_k h^2 + 2\sigma_{\max}^2 V_2^2 h^3 \\
\leq&\left(1 - \lambda' h + 2\sigma_{\max}^2 U_2^2 h^3 + 2\sigma_{\max} U_1 h^2 + 2\sigma_{\max} U_2 \sqrt{\tilde{C}} h^{\frac{5}{2}}\right) e_k^2 + \frac{\lambda'}{8} h e_k^2 \\
&+ \frac{2\left(2\sigma_{\max} V_1 + 2\sigma_{\max} V_2 \sqrt{\tilde{C}h}\right)^2}{\lambda'} h^3 + 2\sigma_{\max}^2 V_2^2 h^3 \\
=&\left(1 - \frac{7}{8}\lambda' h + 2\sigma_{\max}^2 U_2^2 h^3 + 2\sigma_{\max} U_1 h^2 + 2\sigma_{\max} U_2 \sqrt{\tilde{C}} h^{\frac{5}{2}}\right) e_k^2 \\
&+ \left(\frac{2\left(2\sigma_{\max} V_1 + 2\sigma_{\max} V_2 \sqrt{\tilde{C}h}\right)^2}{\lambda'} + 2\sigma_{\max}^2 V_2^2\right) h^3 \\
\overset{(i)}{\leq}&(1 - \frac{1}{2}\lambda' h)e_k^2 + \left(\frac{2\left(2\sigma_{\max} V_1 + 2\sigma_{\max} V_2 \sqrt{\tilde{C}h}\right)^2}{\lambda'} + 2\sigma_{\max}^2 V_2^2\right) h^3 \\
\triangleq&(1 - \frac{1}{2}\lambda' h)e_k^2 + Kh^3
\end{aligned}
$$

where (i) is due to $h < \min\{h_1, h_2, h_3\}$ and

$$
\begin{aligned}
h_1 =& \frac{\sqrt{\lambda'}}{4\sqrt{2}\kappa' L \max\{\alpha + 1.25, \gamma + 1\}(1.92 + 2.30\alpha L)}, \\
h_2 =& \frac{\lambda'}{16\sqrt{2}\kappa'(L+G)\max\{\alpha + 1.25, \gamma + 1\}(1.74 + 0.71\alpha)}, \\
h_3 =& \frac{\lambda'}{8\kappa' L \max\{\alpha + 1.25, \gamma + 1\}(1.92 + 2.30\alpha L)}.
\end{aligned}
$$

Unfolding the above inequality, we arrive at

$$\begin{aligned}
e_k^2 \leq & \left(1-\frac{\lambda'}{2}h\right)^k e_0^2 + \left(1+(1-\frac{\lambda'}{2}h)+\cdots+(1-\frac{\lambda'}{2}h)^{k-1}\right) Kh^3 \\
\overset{(i)}{\leq} & Kh^3 \sum_{i=0}^{\infty} \left(1-\frac{\lambda'}{2}h\right)^i \\
= & \frac{2K}{\lambda'}h^2
\end{aligned}$$

where (i) is due to $e_k = 0$. Therefore

$$\left(\mathbb{E}\|\boldsymbol{x}_k - \bar{\boldsymbol{x}}_k\|^2\right)^{\frac{1}{2}} = \left(\mathbb{E}\left\|P^{-1}(\boldsymbol{y}_k - \bar{\boldsymbol{y}}_k)\right\|^2\right)^{\frac{1}{2}} \leq \frac{1}{\sigma_{\min}} e_k \leq \frac{1}{\sigma_{\min}}\sqrt{\frac{2K}{\lambda'}}h$$

Collecting all the constants and we have

$$\begin{aligned}
& \frac{1}{\sigma_{\min}}\sqrt{\frac{2K}{\lambda'}} \\
\leq & \frac{8\kappa'}{\lambda'}(L+G)\max\{\alpha+1.25, \gamma+1\}(1.74+0.71\alpha)\left(\sqrt{\int_{\mathbb{R}^d}\|\boldsymbol{q}\|^2 d\mu + d} + \kappa' W_2(\pi_0,\pi)\right) \\
& + \frac{4\kappa'}{\lambda'}(L+G)\max\{\alpha+1.25, \gamma+1\}\left(0.5\alpha + (1.26\sqrt{\alpha}+1.14\alpha\sqrt{\alpha}+2.32\sqrt{\gamma})\sqrt{d}\right) \\
& + \frac{8\kappa'}{\sqrt{\lambda'}}\left(\frac{\sqrt{\kappa' L'}}{\sqrt{\lambda'}}+1\right) L\max\{\alpha+1.25, \gamma+1\}(1.92+2.30\alpha L)\times \\
& \left(\sqrt{\int_{\mathbb{R}^d}\|\boldsymbol{q}\|^2 d\mu + d} + \kappa' W_2(\pi_0,\pi)\right) \\
& + \frac{4\kappa'}{\sqrt{\lambda'}}\left(\frac{\sqrt{\kappa' L'}}{\sqrt{\lambda'}}+1\right) L\max\{\alpha+1.25, \gamma+1\}(2.60\sqrt{\alpha}+3.34\sqrt{\gamma})\sqrt{d} \\
\triangleq & C
\end{aligned}$$

It is clear that in terms of the dependence on dimension d, we have $C = \mathcal{O}(\sqrt{d})$. In the regime where $\frac{\gamma^2 - L}{\gamma} \geq \frac{m}{\gamma} + m\alpha$, then $\lambda' = \frac{m}{\gamma} + m\alpha$. Recall the definition of κ' and there

exist $A', B' > 0$ such that $\kappa' \leq A'\sqrt{\alpha} + B'$. It follows that

$$C \leq \frac{a_1\alpha^3 + a_2\alpha^{\frac{5}{2}} + a_3\alpha^2 + a_4\alpha^{\frac{3}{2}} + a_5\alpha + a_6\alpha^{\frac{1}{2}} + a_7}{\lambda'} \leq \frac{b_1\alpha^3 + b_2}{\lambda'} = \frac{b_1\alpha^3 + b_2}{\frac{m}{\gamma} + m\alpha}$$

for some positive constants $a_1, a_2, a_3, a_4, a_5, a_6, a_7, b_1, b_2 > 0$ and independent of α, in particular, we have $b_1 = \mathcal{O}(\sqrt{d}), b_2 = \mathcal{O}(\sqrt{d})$. ■

Remark: The linear growth condition on $\nabla\Delta f$ is a mild assumption, and in some sense, even weaker than classical conditions on the existence of solutions to SDE. For example, a linear growth condition is assumed in [10] to ensure a global, unique solution exists for an SDE. If we assume the potential function is a monomial $f(x) = f_p(x) = x^p, p \in \mathbb{Z}_+$, then the linear growth condition on $\nabla\Delta f$ is met whenever $p \leq 4$ whereas classical condition on the existence of solutions to SDE only apply when $p \leq 2$.

Remark: The proof is based on the powerful mean-squared analysis framework [123], we extend the framework and carefully keep track of all constants to study the dependence on some important parameters such as step size h and dimension d. As a side note, the proof technique in [31, Theorem 2] does not directly apply to Algorithm 2 since the diffusion on position variable in HFHR lowers the regularity of sample paths, which is critical in their proof.

Two results follow immediately from Theorem 11. Theorem 12 characterizes the difference between the law of position variable $\boldsymbol{q}$ in Algorithm 2 and the target distribution μ in 2-Wasserstein distance. Corollary 13 gives the iteration complexity of Algorithm 2 to reach ϵ-accuracy to the target distribution.

Theorem 12 *Under the same assumption as in Theorem 11 and further assume* $\gamma - \frac{L+m}{\gamma} \geq m\alpha$*, if we start from* $(\boldsymbol{q}_0, \boldsymbol{p}_0) \sim \pi_0$*, then there exists* $h_0, C > 0$ *(same as that in Theorem 11 and* $C = \mathcal{O}(\sqrt{d})$*) such that when* $0 < h < h_0$*, we have*

$$W_2(\mu_k, \mu) \leq \sqrt{2}\kappa' e^{-(\frac{m}{\gamma} + m\alpha)kh} W_2(\pi_0, \pi) + \sqrt{2}Ch \tag{3.21}$$

where μ_k is the law of the q marginal of the k-th iterate in Algorithm 2, μ is the q marginal of the invariant distribution π.

Proof: Denote the k-th iterate of the Strang's splitting method of HFHR by $\bar{\boldsymbol{x}}_k$ with time step h, the solution of HFHR dynamics at time hk by $\boldsymbol{x}_k$. Both $\bar{\boldsymbol{x}}_k$ and $\boldsymbol{x}_k$ start from $\boldsymbol{x}_0 = \begin{bmatrix} \boldsymbol{q}_0 \\ \boldsymbol{p}_0 \end{bmatrix}$. Also denote the solution of HFHR dynamics starting from $\tilde{\boldsymbol{x}}_0$ at time kh by $\tilde{\boldsymbol{x}}_k$ where $\tilde{\boldsymbol{x}}_0 = \begin{bmatrix} \tilde{\boldsymbol{q}}_0 \\ \boldsymbol{p}_0 \end{bmatrix}$, $(\tilde{\boldsymbol{q}}_0, \tilde{\boldsymbol{p}}_0) \sim \pi$ and $\mathbb{E}\left\| \begin{bmatrix} \boldsymbol{q}_0 - \tilde{\boldsymbol{q}}_0 \\ \boldsymbol{p}_0 - \tilde{\boldsymbol{p}}_0 \end{bmatrix} \right\|^2 = W_2^2(\pi_0, \pi)$. Since π is the invariant distribution of HFHR dynamics, it follows that $\tilde{\boldsymbol{x}}_k \sim \pi$.

By Lemma 33 and Theorem 11, we have

$$
\begin{aligned}
W_2^2(\mu_k, \mu) =& \inf_{\xi \in \Pi(\mu_k, \mu)} \mathbb{E}_{(\boldsymbol{q}_1, \boldsymbol{q}_2) \sim \xi} \|\boldsymbol{q}_1 - \boldsymbol{q}_2\|^2 \\
\leq& \inf_{\xi \in \Pi(\pi_k, \pi)} \mathbb{E}_{(\boldsymbol{x}_1, \boldsymbol{x}_2) \sim \xi} \|\boldsymbol{x}_1 - \boldsymbol{x}_2\|^2 \\
\leq& \mathbb{E}\|\bar{\boldsymbol{x}}_k - \tilde{\boldsymbol{x}}_k\|^2 \\
\leq& 2C^2h^2 + 2\mathbb{E}\left\|P^{-1}P(\boldsymbol{x}_k - \tilde{\boldsymbol{x}}_k)\right\|^2 \\
\leq& 2C^2h^2 + 2\|P^{-1}\|_2^2\mathbb{E}\left\|P(\boldsymbol{x}_k - \tilde{\boldsymbol{x}}_k)\right\|^2 \\
\leq& 2C^2h^2 + 2\|P^{-1}\|_2^2 e^{-2\lambda' kh}\mathbb{E}\left\|P(\boldsymbol{x}_0 - \tilde{\boldsymbol{x}}_0)\right\|^2 \\
\leq& 2C^2h^2 + 2(\kappa')^2 e^{-2\lambda' kh} W_2^2(\pi_0, \pi)
\end{aligned}
$$

Take square root on both sides and apply $\sqrt{a^2 + b^2} \leq a + b$, we obtain

$$
W_2(\mu_k, \mu) \leq \sqrt{2}Ch + \sqrt{2}\kappa' e^{-\lambda' kh} W_2(\pi_0, \pi).
$$

■

Corollary 13 *Under the same assumption as in Theorem 11 and further assume $\gamma - \frac{L+m}{\gamma} \geq m\alpha$, if we start from $(\boldsymbol{q}_0, \boldsymbol{p}_0) \sim \pi_0$, then there exists $h_0, C > 0$ (same as that in Theorem*

11 and $C = \mathcal{O}(\sqrt{d})$) such that for any target accuracy $\epsilon > 0$, if we choose $h = h^\star \triangleq \min\{h_0, \frac{\epsilon}{2\sqrt{2}C}\}$, then after

$$k = k^\star \triangleq \frac{1}{\frac{m}{\gamma} + m\alpha} \max\{\frac{1}{h_0}, \frac{2\sqrt{2}C}{\epsilon}\} \log \frac{2\sqrt{2}\kappa' W_2(\pi_0, \pi)}{\epsilon} \tag{3.22}$$

steps, we have $W_2(\mu_k, \mu) \leq \epsilon$. When high accuracy is needed, i.e., $\epsilon < 2\sqrt{2}Ch_0$, the iteration complexity becomes $k^\star = 2\sqrt{2}\frac{C}{\frac{m}{\gamma}+m\alpha}\frac{1}{\epsilon}\log\frac{2\sqrt{2}\kappa' W_2(\pi_0,\pi)}{\epsilon} = \tilde{\mathcal{O}}(\frac{\sqrt{d}}{\epsilon})$. Recall from Theorem 11 that $C \leq \frac{b_1\alpha^3+b_2}{\frac{m}{\gamma}+m\alpha}$ when $\gamma - \frac{L+m}{\gamma} \geq m\alpha$, so the minimizer of the upper bound of $\frac{C}{\frac{m}{\gamma}+m\alpha}$

$$\alpha^\star = \operatorname*{argmin}_{\alpha \geq 0} \frac{b_1\alpha^3 + b_2}{(\frac{m}{\gamma} + m\alpha)^2} > 0,$$

which implies there exists a positive $\alpha^\star$ that minimizes the iteration complexity of Algorithm 2, in particular, Algorithm 2 with $\alpha = \alpha^\star$ has better iteration complexity than ULD (HFHR with $\alpha = 0$) does.

Proof: By Theorem 12, we have

$$W_2(\mu_k, \mu) \leq \sqrt{2}Ch + \sqrt{2}\kappa' e^{-\lambda' kh} W_2(\pi_0, \pi).$$

Given any target accuracy $\epsilon > 0$, if we run the Strang's splitting method of HFHR with $h^\star = \min\{h_0, \frac{\epsilon}{2\sqrt{2}C}\}$, then after $k^\star = \frac{1}{\lambda'}\max\{\frac{1}{h_0}, \frac{2\sqrt{2}C}{\epsilon}\}\log\frac{2\sqrt{2}\kappa' W_2(\pi_0,\pi)}{\epsilon}$, we have

$$W_2(\mu_{k^\star}, \mu) \leq \sqrt{2}Ch + \sqrt{2}\kappa' e^{-\lambda' kh} W_2(\mu_0, \mu) \leq \frac{\epsilon}{2} + \frac{\epsilon}{2} = \epsilon.$$

Recall $C = \mathcal{O}(\sqrt{d})$, when high accuracy is needed, e.g. $\epsilon < 2\sqrt{2}Ch_0$, the iteration complexity to reach ϵ-accuracy under 2-Wasserstein distance is $k^\star = \mathcal{O}(\frac{\sqrt{d}}{\epsilon}\log\frac{1}{\epsilon}) = 2\sqrt{2}\frac{C}{\lambda'}\frac{1}{\epsilon}\log\frac{2\sqrt{2}\kappa' W_2(\pi_0,\pi)}{\epsilon} = \tilde{\mathcal{O}}(\frac{\sqrt{d}}{\epsilon})$. Recall from Theorem 11, $C \leq \frac{b_1\alpha^3+b_2}{\frac{m}{\gamma}+m\alpha}$, we have

$$\frac{C}{\lambda'} \leq \frac{b_1\alpha^3 + b_2}{(\frac{m}{\gamma} + m\alpha)^2}$$

Denote $g(\alpha) = \frac{b_1\alpha^3+b_2}{(\frac{m}{\gamma}+m\alpha)^2}$, simple calculation shows that $g'(0) < 0$ and $\lim_{\alpha\to\infty} g(\alpha) = +\infty$, hence

$$\alpha' = \operatorname*{argmin}_{\alpha\geq 0} g(\alpha)$$

exists and is positive. ■

Remark: The result of Theorem 12 agrees with Theorem 2 for a discretized algorithm of ULD termed as 1st-order KLMC in [31], in that both results have an exponentially decaying first term due to converging continuous dynamics, and a second term caused by discreitzation error that scale linearly in $h\sqrt{d}$. Consequently, the iteration complexity result $\tilde{\mathcal{O}}(\frac{\sqrt{d}}{\epsilon})$ in Corollary 13 matches that in the discussion after Theorem 2 of [31]. It is worth mentioning that an even-faster discretized algorithm for ULD with $\tilde{\mathcal{O}}(\frac{\sqrt{d}}{\sqrt{\epsilon}})$ iteration complexity, termed 2nd-order KLMC was proposed in [31]. However, 2nd-order KLMC requires the hessian of potential functions to work which can be computationally expensive.

Remark: When high-accuracy is needed, the iteration complexity depends linearly in $\frac{C}{\lambda'}$ and $\frac{C}{\lambda'}$ is upper bounded by $\frac{b_1\alpha^3+b_2}{(\frac{m}{\gamma}+m\alpha)^2}$. As shown in Figure 3.1, the upper bounded, hence in some sense the overall iteration complexity is minimized for some $\alpha^\star > 0$, better than that of underdamped Langevin dynamic, which corresponds to the case $\alpha = 0$.

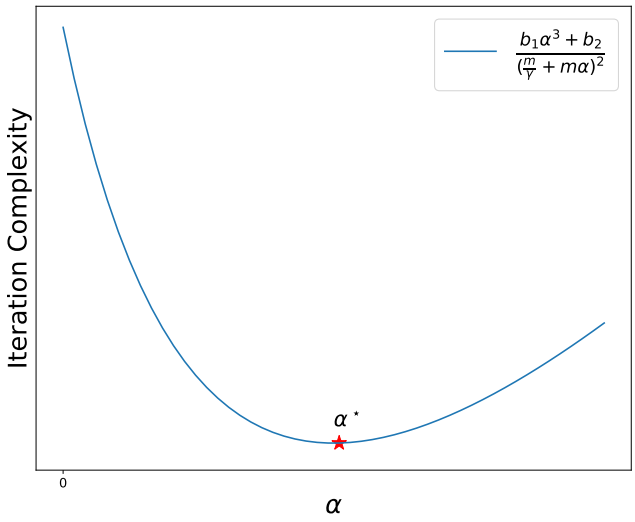


Figure 3.1: Illustration of the effect of α on iteration complexity

Inspecting the role of α in Equation (3.21), we see that α clearly increases the rate of exponential decay, at the same time, may also increase discretization error. However, the net effect of having a positive $\alpha > 0$, at least for some $\alpha^\star$, is reduced iteration complexity, and hence enables a more efficient discrete algorithm then ULD algorithm, as illustrated in Corollary 13 and its remarks. The improved efficiency of Algorithm 2 over ULD algorithm shows that the acceleration of HFHR can be carried from continuous dynamics realm to the discrete algorithm regime. Appendix B.11 presents a case study that analytically demonstrates this point, and the same conclusion has been repeatedly observed in numerical experiments.

3.6 Numerical Experiments

This section empirically studies the performance of HFHR algorithm and compares it with ULD algorithm (1st-order KLMC algorithm in [26]). In Section 3.6.1, we test both algorithms on a collection of target distributions with simple but representative functions as potential f and cover strongly convex (low/high dimensional Gaussian, small/large strongly-convex coefficient m), convex and non-convex cases (bi-modal, perturbed Gaussian and 2D Rosenbrock's function [124]). In Section 3.6.2, we work with Gaussian distribution and numerically verify the tightness of our theoretical results on HFHR dynamics. A non-linear potential is studied in Section 3.6.3 to verify the tightness of Theorem 12 and demonstrate improved iteration complexity of HFHR algorithm over ULD algorithm. In Section 3.6.4, we demonstrate how HFHR performs in a downstream learning task based on a high-dimensional, non-convex, multi-modal Bayesian neural network model. In all experiments, we use the same γ and step size h for ULD and HFHR. All experiments are conducted on a machine with a 2.20GHz Intel(R) Xeon(R) E5-2630 v4 CPU and an Nvidia GeForce GTX 1080 GPU.

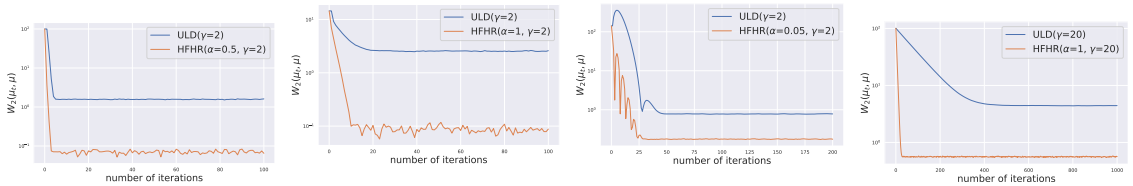
Table 3.2: Test functions. We use the shorthand notation $G^d_{m,\kappa}(\boldsymbol{x}) = \frac{m}{2}(\kappa x_d^2 + \sum_{i=1}^{d-1} x_i^2)$. Letters 'S', 'C' and 'N' represent strongly convex, convex and non-convex respectively.

f	f_1	f_2	f_3	f_4	f_5	f_6	f_7	f_8
Expresion	$\frac{1}{2}x^2$	$G^2_{0.1,10}$	$G^2_{10,10}$	$G^{100}_{1,100}$	$\frac{1}{4}x^4$	$\frac{5x^2+\sin(10x)}{10}$	$5(x^4-2x^2)$	$\frac{(x-1)^2+10(y-x^2)^2}{2}$
Convexity	S	S	S	S	C	N	N	N

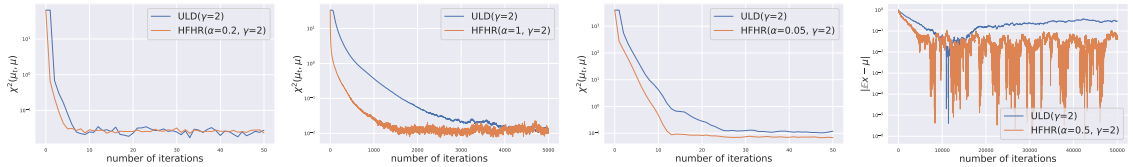
3.6.1 Simple Target Distributions

In this subsection, we test 8 target distributions with simple, yet representative potential functions, summarized in Table 3.2. They are classified into two groups, Gaussians and non-Gaussians. For Gaussian, smoothness coefficient L is available, hence we take $\gamma = 2\sqrt{L}$ in accordance with Table 3.1. To be consistent with Theorem 11, we measure closeness to the target distribution using W_2 which has closed-form expression for two Gaussians. For non-Gaussians, we empirically set $\gamma = 2$ and measure sample quality by χ^2 divergence with density approximated by histogram. Note approximating the density corresponding to $f_8(x, y)$ by uniform-mesh-based histogram is either inaccurate or requiring the mesh to be very fine, and we thus report the error in the x component $|\mathbb{E}x - \mu|$ instead, where μ is the true mean of x-component. Step size h is tuned so that it is near the *stability limit* of ULD algorithm. In each experiment, we sample 10000 independent realizations for each algorithm

Figure 3.2 shows the experiment results. Each result contains two phases. In the first phase, error has a trend of smoothly decreasing, corresponding to the exponentially decay bound in Equation (3.21); in the second phase, the curve is noisy since algorithms are saturated by discretization error. Across all experiments, the additional acceleration of HFHR over ULD is clearly seen in the first phase and can be very significant, for example in Figure 3.2d and 3.2f.



(a) 1D Gaussian (b) 2D Gaussian($m = 0.1$) (c) 2D Gaussian($m = 10$) (d) 100D Gaussian



(e) Convex (f) Perturbed (g) Bi-modal (h) 2D Rosenbrock

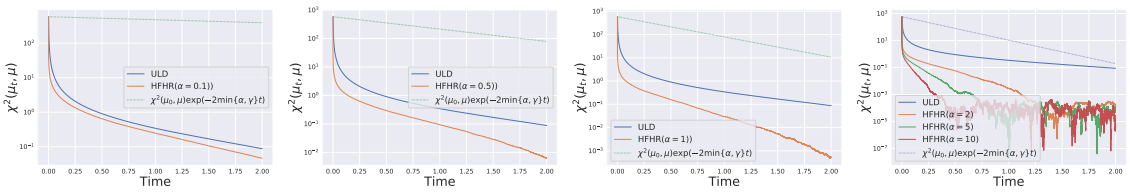
Figure 3.2: (a) $f_1\,(h = 2)$. (b) $f_2\,(h = 0.2)$. (c) $f_3\,(h = 2.5)$. (d)$f_4\,(h = 0.2)$. (e) $f_5\,(h = 0.5)$. (f) $f_6\,(h = 0.001)$. (g) $f_7\,(h = 0.1)$. (h) $f_8\,(h = 0.005)$. y-axes are in log scale.

3.6.2 A Case Study on Gaussian: Empirical Performances versus Theoretical Guarantees for HFHR Dynamics

In this subsection, we numerically verify the tightness of theoretical results for the continuous HFHR dynamics in Section 3.4. To this end, we work with Gaussian distributions because (i) Gaussians have log-strongly-convex densities and hence satisfy all the assumptions in our theory; (ii) Gaussians are analytically tractable, all parameters needed for the theory, including Lipschitz constant L, strongly-convex coefficient m, etc., are known to us, also the metrics used in our theory (e.g. 2-Wasserstein distance, χ^2 divergence) have closed form expression for Gaussians; (iii) Gaussians are widely used in various machine learning applications.

The first verification starts with Theorem 6. We experiment with 1-dimensional standard Gaussian distribution and use Algorithm 2 with tiny step size ($h = 10^{-4}$, small enough so that results about the continuous dynamics can be probed) to simulate HFHR dynamics and compare with the theoretical bound in Theorem 6. We set $\gamma = 2$ which corresponds to the critical damping of ULD and generate 10000 to estimate χ^2 divergence to the target distribution.

We run Algorithm 2 with $\alpha \in \{0.1, 0.5, 1, 2, 5, 10\}$ and plot the results as well as the



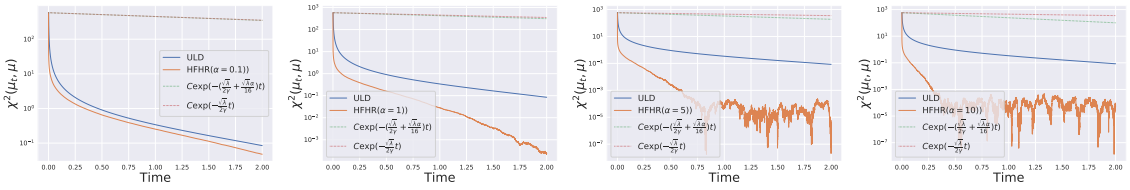
(a) $\alpha = 0.1$ (b) $\alpha = 0.5$ (c) $\alpha = 1$ (d) $\alpha \geq 2$

Figure 3.3: Illustration of the consistency between the theoretical bound in Theorem 6 and experiment results.

upper bound in Theorem 6 in Figure 3.3. In all four figures, HFHR outperforms ULD in terms of convergence speed, even when α is small, e.g. Figure 3.3a, and the gap between convergence speeds becomes larger as we increase α, e.g. Figure 3.3a $\sim$ 3.3c. Note also the bound in Theorem 6 may not be tight due to its generality. For example, the rate of the exponential convergence bound in Theorem 6 reads $-2\min\{\alpha, \gamma\}$ and is -2γ whenever $\alpha \geq \gamma$. However, as Figure 3.3d shows, when $\alpha \geq 2$, larger α still practically introduces acceleration. On a side note, we observe that the convergence of HFHR and ULD in χ^2 divergence consists of two phases, the convergence in the first phase is super-linear and gradually transitions to roughly linear in the second phase. Theorem 6 characterizes the linear convergence rate in the second phase and appears to be almost tight in Figure 3.3d (note the line of HFHR($\alpha = 2$) is nearly parallel to the upper bound).

Next we empirically verify Theorem 9. The experiment setup is identical to that for Figure 3.3. The results together with the theoretical bounds in Theorem 9 for both HFHR and ULD are plotted in Figure 3.4. In all four plots, HFHR converges faster than ULD. In particular, in Figure 3.4c and 3.4d, HFHR reaches 10^{-3}-accuracy in χ^2 divergence in less than 0.75 unit time while ULD just reaches 10^{-1}-accuracy in the experiment time frame. In addition, we make two observations: (i) In Figure 3.4a and 3.4b, the two upper bounds for HFHR and ULD respectively in Theorem 9 are hardly distinguishable because α is small, however, the actual acceleration created by α is clearly manifested, even when α is tiny, i.e., $\alpha = 0.1$ in Figure 3.4a. Therefore, in practice, HFHR dynamics can introduce even more acceleration than Theorem 9 ensures; (ii) Note that the Lipschitz constant $L = 1$ and Poincaré constant $\lambda = 1$, so the assumption $\gamma^2 \geq \max\{2\lambda, L\}$ is satisfied and $\alpha \leq \frac{\gamma}{\lambda} - \frac{2}{\gamma}$

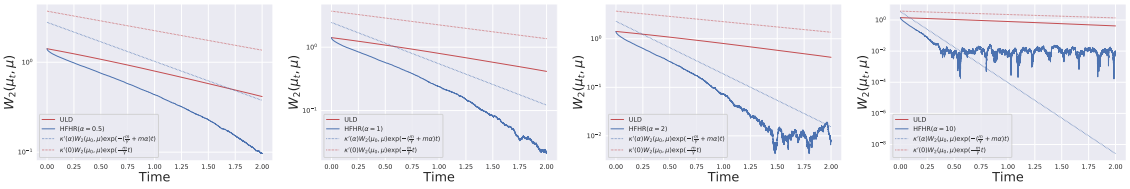
is satisfied when $\alpha \leq 1$. In Figure 3.4c and 3.4d, α is large and breaks the assumption of α in Theorem 9, but also creates significant acceleration. This observation demonstrates the applicability of HFHR is wider than conditions needed by Theorem 9. In fact, Theorem 9 only requires convexity but Gaussian targets correspond to strongly convex potentials, so its bound being not tight should not be surprising.



(a) $\alpha = 0.1$ (b) $\alpha = 1$ (c) $\alpha = 5$ (d) $\alpha = 10$

Figure 3.4: Illustration of the consistency between the theoretical bound in Theorem 9 and experiment results.

Finally, we verify Theorem 10. The experimental setup is the same as that for Figure 3.3. We plot the W_2 errors of HFHR and ULD, together with their respective upper bounds in Theorem 10 in Figure 3.5. We observe acceleration of HFHR over ULD in all four subfigures of Figure 3.5. In addition, our theoretical upper bounds are nearly parallel to the performance of actual HFHR algorithm, which empirically demonstrates that our theory in Theorem 10 is tight up to a constant. Note that the two assumptions in Theorem 10, i.e., $\gamma^2 > L + m$ are satisfied when $\alpha \leq 1$. In Figure 3.5c and 3.5d, the choices of α can break the assumption, nevertheless, Algorithm 2 empirically works well and creates significant acceleration over ULD.



(a) $\alpha = 0.5$ (b) $\alpha = 1$ (c) $\alpha = 2$ (d) $\alpha = 10$

Figure 3.5: Illustration of the consistency between the theoretical bound in Theorem 10 and experiment results.

3.6.3 A Case Study on a Nonlinear Problem: Empirical Performances versus Theoretical Guarantees for HFHR Algorithm

Numerical verification of the tightness of theoretical results for Algorithm 2 in Section 3.5 is conducted is this subsection. A key feature of HFHR algorithm is that its discretization error/iteration complexity has a $\mathcal{O}(\sqrt{d})$ dependence on the ambient dimension (same as ULD algorithm and better than OLD algorithm's $\mathcal{O}(d)$ dependence). If we use standard Gaussian as potential, HFHR dynamics become decoupled across dimensions, and hence its discretization error having a $\mathcal{O}(\sqrt{d})$ dependence would be a natural consequence as we use 2-Wasserstein distance to quantify statistical accuracy.

To inspect a more interesting example, we consider the following potential, the sum of a quadratic function and a log-sum-exp function, i.e.,

$$f(\boldsymbol{x}) = \log\left(e^{x_1} + \cdots + e^{x_d}\right) + \frac{1}{2}\|\boldsymbol{x}\|^2, \tag{3.23}$$

which couples all dimensions in HFHR dynamics. Moreover, the new potential f is still a strongly convex function and satisfies the assumption in Theorem 12. When the target measure is non-Gaussian, we no longer have a closed form expression for 2-Wasserstein distance and it is computationally very expensive to approximate 2-Wasserstein distance by samples. Instead, we use the error of mean as a surrogate because

$$\left\|\mathbb{E}_{\mu_k}\boldsymbol{q} - \mathbb{E}_{\mu}\boldsymbol{q}\right\| \leq W_2(\mu_k, \mu)$$

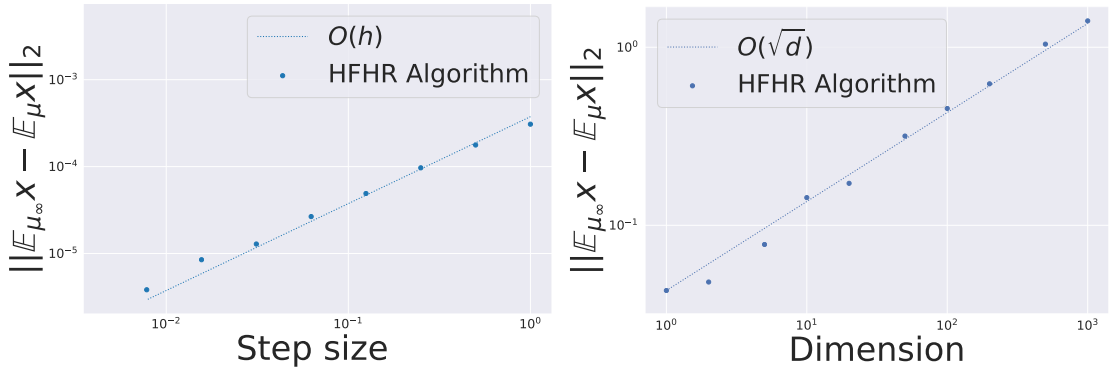
and hence the bound in Equation (3.21) also applies to the error in mean, so does the iteration complexity bound in Equation (3.22).

We first verify Theorem 12, which says the sampling error is upper bounded by two terms, the first corresponding to the exponential decay of the continuous dynamics, whose rate is characterized in Theorem 10 and already numerically verified previously, and the

second term corresponding to discretization error. We theoretically proved in Theorem 12 that the discretization error is linear in step size h and square root of dimension $\sqrt{d}$.

To numerically verify the linear dependence on h, we work with $d = 2$ and ran ULD algorithm with tiny step size ($h = 0.0005$) to obtain 10^8 independent realizations and use them to estimate $\mathbb{E}_\mu \boldsymbol{q}$. We then set $\gamma = 2, \alpha = 1$ and sample from the potential in Equation 3.23 using Algorithm 2, with $h \in \{2^k | -7 \leq k \leq 0\}$. For each h, we run $\frac{T}{h}$ (with $T = 50$) iterations in Algorithm 2 to ensure the Markov chains are well-mixed and the contribution to final error from exponential decay is order-of-magnitude smaller than discretization error. The results are plotted in Figure 3.6a.

We experimentally observe a clear linear dependence on step size h and the observation strongly support our results on discretization error in Theorem 12.



(a) Linear dependence of discretization error of Algorithm 2 on h

(b) Linear dependence of discretization error of Algorithm 2 on $\sqrt{d}$

Figure 3.6: Illustration of the consistency between the theoretical bound in Theorem 12 and experiment results.

To numerically verify the $\mathcal{O}(\sqrt{d})$ dependence on dimension d, we extensively experiment with $d \in \{1, 2, 5, 10, 20, 50, 100, 200, 500, 1000\}$, for each d, we run 1000 independent realizations of ULD algorithm until converged with tiny step size ($h = 0.005$) and use their sample mean as the true mean. We fix $\gamma = 2, \alpha = 1, h = 0.1, T = 10$ and for each d, we run 1000 independent realizations of HFHR algorithm for $\frac{T}{h} = 100$ iterations. Experiment results are plotted in Figure 3.6b. A clear linear dependence of error on $\sqrt{d}$ is

shown in Figure 3.6b, demonstrating the bound obtained in Theorem 12 is tight in terms of the dependence in d.

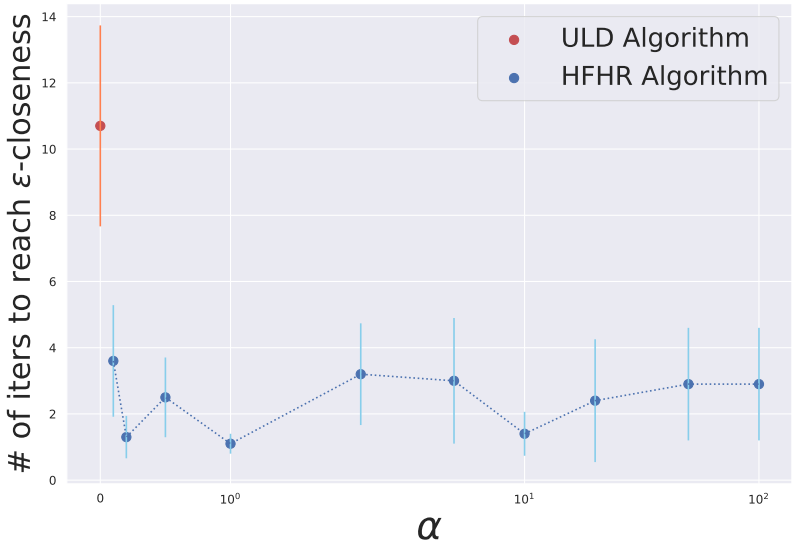


Figure 3.7: Improvement of Algorithm 2 over ULD algorithm in iteration complexity. (vertical bar stands for one standard deviation.)

The final experiment in this subsection is to compare Algorithm 2 with ULD algorithm in terms of iteration complexity. The goal is to demonstrate that the genuine acceleration of HFHR is not an artifact due to time rescaling, which would disappear after discretization as the stability limit changes accordingly. To do so, we push both ULD and HFHR to their respective largest h values that still allow monotone converge at a large scale, and compare their mixing times. For Gaussian targets, this is already analytically studied in Appendix B.11 (see e.g., Fig.B.1). Thus, we again work with the potential in Equation (3.23). For general nonlinear problems like this one, Remark 3.5 and the illustration of Figure 3.1 suggest that with appropriately chosen α, HFHR algorithm can effectively reduce the prefactor of iteration complexity, which implies reduced iteration complexity. To just provide one empirical verification of this improvement over ULD algorithm, we choose $d = 10$ and use the error of mean $\left\|\mathbb{E}_{\mu_k}\boldsymbol{q} - \mathbb{E}_{\mu}\boldsymbol{q}\right\|$ to measure sampling accuracy. The benchmark, i.e., $\mathbb{E}_{\mu}\boldsymbol{q}$, is obtained from 1000 independent realizations of ULD algorithm with tiny step size $(h = 0.005)$, ran for long enough to ensure the

Markov chains are well-mixed. The initial measure is chosen as a Dirac measure at $(\mathbf{1}_d, \mathbf{0}_d)$, where $\mathbf{1}_d, \mathbf{0}_d$ are d-dimensional vectors filled with 1 and 0 respectively. We pick threshold $\epsilon = 0.1$, then for each $\alpha \in \{0, 0.1, 0.2, 0.5, 1, 2, 5, 10, 20, 50, 100\}$, we try all combinations of $(\gamma, h) \in \{0.1, 0.2, 0.5, 1, 2, 5, 10, 20, 50, 100\} \times \{5, 1, 0.5, 0.1, 0.05, 0.01, 0.005\}$ for Algorithm 2 (ULD algorithm when $\alpha = 0$), and empirically find the best combination that requires the fewest iterations to meet $\left\|\mathbb{E}_{\mu_k}\boldsymbol{q} - \mathbb{E}_{\mu}\boldsymbol{q}\right\| \leq \epsilon$. We find that $h = 5$ already surpasses the stability limit of ULD algorithm, hence the range of step size in the experiment covers the largest step size that can be practically used for ULD algorithm We repeat the experiment with 10 different random seeds for better statistical significance and the results are shown in Figure 3.7. When $\alpha > 0$, HFHR algorithm consistently outperforms ULD algorithm In particular when $\alpha = 1$, empirically the best α we found in this experiment, HFFR algorithm reaches the specified ϵ-closeness nearly $6\times$ times faster than ULD algorithm This empirical study corroborates that the acceleration HFHR dynamics creates also carries to its discretization, and the acceleration of HFHR algorithm over ULD algorithm can be significant.

3.6.4 Bayesian Neural Network

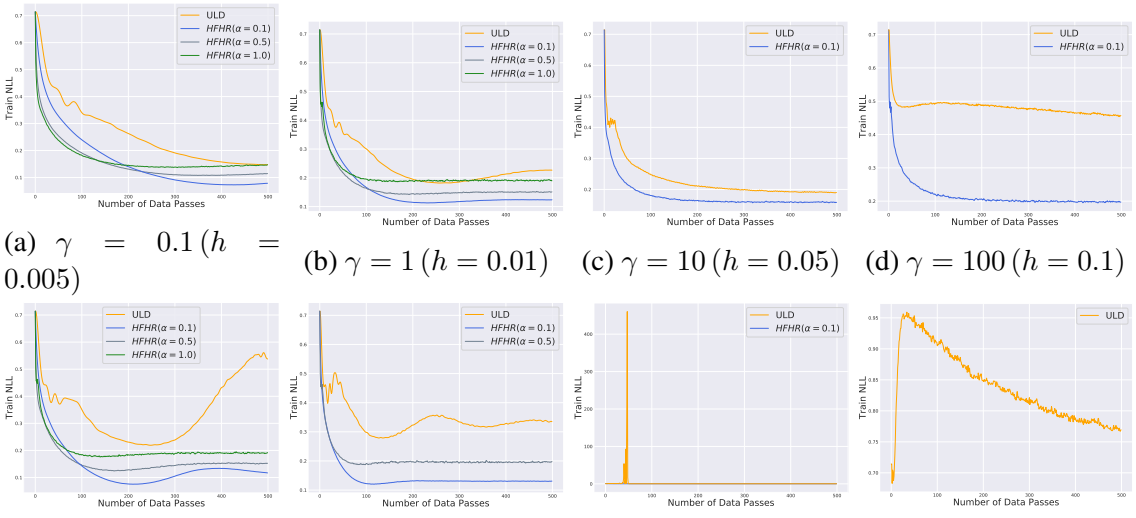
We now consider a Bayesian neural network (BNN) which is a compelling learning model [67] and at the same time a high-dimensional and multi-modal example. The specific network considered here has a fully-connected architecture with [22, 10, 2] neurons in each layer and ReLU activation. Standard Gaussian prior is used for all parameters. We compare both ULD and HFHR on Parkinson's disease data set from UCI machine learning repository [125].

Choices of hyper-parameter for Algorithm 2 and ULD algorithm are extensively investigated. For each pair $(\gamma, \alpha) \in \{0.1, 0.5, 1, 5, 10, 50, 100\}^2$, we empirically tune the step size to the stability limit of ULD algorithm, simulate 10,000 independent realizations of the Markov chain, and use the ensemble to conduct Bayesian posterior prediction. HFHR

will then use the same step size. For each γ, we plot the negative log likelihood of HFHR algorithm (with different α choices) and ULD algorithm on training and test data in Figure 3.8. Large α causing numerical instability are not drawn.

From Figure 3.8, we find that HFHR converges significantly faster than ULD in a wide range of setups. In general, the (strongly) log-concave assumption required in Theorem 9 and Theorem 11 may not hold for complex models. However, the BNN experiment shows the acceleration of HFHR over ULD still holds for highly complex models such as BNN, even when there is no obvious theoretical guarantee.

This demonstrates the applicability and effectiveness of HFHR as a beyond-log-concave, general sampling algorithm



(a) $\gamma = 0.1\,(h = 0.005)$ (b) $\gamma = 1\,(h = 0.01)$ (c) $\gamma = 10\,(h = 0.05)$ (d) $\gamma = 100\,(h = 0.1)$

(e) $\gamma = 0.1\,(h = 0.01)$ (f) $\gamma = 1\,(h = 0.02)$ (g) $\gamma = 10\,(h = 0.1)$ (h) $\gamma = 100\,(h = 0.2)$

Figure 3.8: Training Negative Log-Likelihood (NLL) for various γ. Top row: step sizes are below the stability limit of ULD algorithm; Bottom row: a further increased step size wouild go above the stability limit of ULD algorithm

3.7 Conclusion

This chapter proposes HFHR, an accelerated gradient-based MCMC method for sampling. To demonstrate the acceleration enabled by HFHR, the geometric ergodicity of HFHR (both the continuous and the discretized versions) is quantified, and its convergence is

provably faster than Underdamped Langevin Dynamics, which by itself is often already considered as an acceleration of Overdamped Langevin Dynamics. As HFHR is based on a new perspective, which is to turn NAG-SC optimizer with **finite** learning rate into a sampler, there are a number of interesting directions in which this work can be extended. Besides further theoretical investigations that aim at refining the error bounds, examples also include the followings: to scale HFHR up to large data sets, full gradient may be replaced by stochastic gradient (SG) — how to quantify, and hence optimize the performance of SG-HFHR? Can the generalization ability of HFHR trained learning models (e.g., BNN) be quantified, and how does it compare with that by LMC and/or KLMC? We plan to study these in follow-up works.

CHAPTER 4
NON-ASYMPTOTIC ANALYSIS OF BOUNDED CONTRACTIVE-SDE-BASED SAMPLING ALGORITHMS VIA MEAN-SQUARE ANALYSIS

4.1 Introduction

Many sampling algorithms are based on stochastic differential equations (SDE). By setting a target distribution as the invariant distribution of a SDE, and running an appropriate numerical algorithm that simulate the SDE for long enough, the iterates of the numerical algorithm will approximately follow the target distribution and can be used for downstream applications such as Bayesian inference. Such examples include Langevin Monte Carlo algorithm (LMC) [12], Metropolis-Adjusted Langevin Algorithm (MALA) [15] and Kinetic Langevin Monte Carlo algorithm (KLMC) [31], etc.

Quantitatively characterizing the sampling error of numerical algorithms, particularly in a non-asymptotic manner, is usually critical for people to evaluate and compare various sampling algorithms, guide hyperparameter selection in practice and better understand the nature of these sampling algorithms. Many approaches have been proposed to this end, for example, Lyapunov analysis [29, 35], viewing sampling as optimization in probability space [29, 126] and numerical analysis [26, 32, 31, 28]. These studies lead to fruitful results on non-asymptotic error bounds with explicit dependence on various parameters of the underlying SDE, e.g. dimension, condition number of the potential function of the target, etc. A classical and powerful framework– mean-square analysis [123] in the study of numerical SDE, however, is largely unexplored for modern non-asymptotic analysis of sampling algorithms. This is partly because traditional mean-square analysis only has bound for finite time, i.e. $t \in [0, T]$ and does not extend to regime $t \to \infty$, in addition, the dependence of the error bound on the parameters of underlying SDE is implicit and

it is hard to evaluation the performance of a numerical algorithm in e.g. high-dimension problems.

In this chapter, we study a broad family of bounded numerical algorithms for discretized SDE, whose underlying SDE have contraction property. For this type of algorithms, we revisit mean-square analysis, and show that we manage to extend the the bound of global error to infinite time, i.e. $t \in \mathbb{R}$. Same as in classical mean-square analysis, we show global error is only half order lower than the order of local strong error (p_2). We further obtain the $\widetilde{\mathcal{O}}\left(C^{\frac{1}{p_2-\frac{1}{2}}}\frac{1}{\epsilon^{\frac{1}{p_2-\frac{1}{2}}}}\right)$ iteration complexity in 2-Wasserstein distance for the family of algorithms where C is a constant containing various information of the underlying SDE, e.g. dimension d. The iteration complexity bound not only reveals the dependence on tolerance ϵ, but also, somewhat surprisingly, shows that the dependence on the parameters of the underlying SDE is also determined by the order of local strong error.

As an application of the general iteration complexity result, we study the widely used Langevin Monte Carlo algorithm (LMC) for sampling from a Gibbs distribution $\mu \propto \exp\big(-f(\boldsymbol{x})\big)$, which is an Euler-Maruyama discretization of Langevin dynamics. Under the standard smoothness and strong-convexity assumptions, plus an additional linear growth condition on the third-order derivative of f, we obtain a $\widetilde{\mathcal{O}}\left(\frac{\sqrt{d}}{\epsilon}\right)$ iteration complexity in 2-Wasserstein distance, which improves upon the previously best known $\widetilde{\mathcal{O}}\left(\frac{d}{\epsilon}\right)$ result [33].

4.2 Background

Consider a general SDE

$$d\boldsymbol{x}_t = \boldsymbol{b}(t, \boldsymbol{x}_t)dt + \boldsymbol{\sigma}(t, \boldsymbol{x}_t)d\boldsymbol{B}_t \tag{4.1}$$

where $\boldsymbol{b} \in \mathbb{R}^d$ is a drift term, $\boldsymbol{\sigma} \in \mathbb{R}^{l\times d}$ is a diffusion coefficient matrix and $\boldsymbol{B}_t$ is a d-dimensional Wiener process. Under mild condition [10], there exists a unique strong

solution $\boldsymbol{x}_t$ to Equation (4.1). Some SDEs admit invariant distribution and their solutions are geometrically ergodic, e.g. Ornstein-Unlenbeck process, Langevin dynamics and kinetic Langevin dynamics. Such SDE are desired for sampling purposes, because one can set a target distribution as the invariant distribution of an SDE, by solving the solution $\boldsymbol{x}_t$ of the SDE and pushing time t to infinity, one can then (approximately) sample from the target distribution. Except for a few known cases, however, explicit solutions of Equation (4.1) are elusive and people resort to numerical schemes to simulate/integrate SDE, such examples include but not limited to Euler-Maruyama method, Milstein methods and Runge-Kutta method. At k-th iteration, a typical numerical algorithm takes a previous iterate $\bar{\boldsymbol{x}}_{k-1}$ and a step size h, and outputs a new iterate $\bar{\boldsymbol{x}}_k$ as an approximation of the solution $\boldsymbol{x}_t$ of Equation (4.1) at time $t = kh$.

A classical and powerful framework to quantify the *global* discretization error of a numerical algorithm for Equation (4.1), i.e.,

$$e_k = \left\{\mathbb{E}\|\boldsymbol{x}_{kh} - \bar{\boldsymbol{x}}_k\|\right\}^{\frac{1}{2}}$$

is mean-square analysis [123]. Mean-square analysis studies how *local* error propagate to global error, in particular, if one-step (local) weak error and strong error (both the solution $\boldsymbol{x}_t$ and the numerical algorithm start from initial value $\boldsymbol{x}$) satisfy

$$\begin{aligned} \|\mathbb{E}\boldsymbol{x}_h - \mathbb{E}\bar{\boldsymbol{x}}_h\| \leq & C_1 \left(1 + \mathbb{E}\|\boldsymbol{x}\|^2\right)^{\frac{1}{2}} h^{p_1}, \quad \text{(local weak error)} \\ \left(\mathbb{E}\|\boldsymbol{x}_h - \bar{\boldsymbol{x}}_h\|^2\right)^{\frac{1}{2}} \leq & C_2 \left(1 + \mathbb{E}\|\boldsymbol{x}\|^2\right)^{\frac{1}{2}} h^{p_2}, \quad \text{(local strong error)} \end{aligned} \tag{4.2}$$

over a time interval $[0, Kh]$ for some constants $C_1, C_2 > 0$, $p_2 \geq \frac{1}{2}$ and $p_1 \geq p_2 + \frac{1}{2}$, then the global error can be bounded by

$$e_k \leq C \left(1 + \mathbb{E}\|\boldsymbol{x}_0\|^2\right)^{\frac{1}{2}} h^{p_2 - \frac{1}{2}}, \; k = 1, 2, \cdots, K \tag{4.3}$$

for some $C > 0$. This result roughly says the global error is half-order lower than the local strong error. Despite the nice result in Equation (4.3), there are two limitations that prevent directly employing mean-square analysis in the non-asymptotic analysis of sampling algorithms. First, the bound of global error in Equation (4.3) only holds in finite time because the constant C can grow exponentially as K increases, rendering the bound useless when $K \to \infty$. Second, the classical mean-square analysis does not keep track of the dependence of C on various parameters, e.g. dimension, condition number of the potential function of the target distribution, which are the main focus of current research in the field.

4.3 Mean-Square Analysis of Bounded Contractive-SDE-Based Algorithms

In this section, we study a family of bounded algorithms for contractive SDE and show that we can lift the finite time limitation of classical mean-square analysis for this type of algorithms. One bottleneck in classical mean-square analysis is that local error (Equation (4.2)) typically depends on initial values. These initial values are iterates of numerical algorithm, changing from iteration to iteration and can be unbounded, which poses challenges when pushing time limit to infinity.

We note that if a numerical algorithm is bounded, then aforementioned technical difficulty can be easily bypassed. A numerical algorithm is bounded if all of its iterates meet

$$\mathbb{E}\|\bar{\boldsymbol{x}}_k\|^2 \leq U,\ k = 0, 1, \cdots$$

for some constant $U > 0$. The upper bound U may depend on the parameters of the underlying SDE, e.g. dimension, Lipschitz constant of drift and noise diffusion, and may also depend on the initial value $\boldsymbol{x}_0$ of the algorithm.

A bounded numerical algorithm makes sense only if the underlying SDE is also bounded, for which we impose the following sufficient condition:

Definition 14 *A stochastic differential equation is contractive if there exists a constant*

$\beta > 0$*, such that the solution of the SDE satisfy*

$$\left(\mathbb{E}\|\boldsymbol{x}_t - \boldsymbol{y}_t\|^2\right)^{\frac{1}{2}} \leq \|\boldsymbol{x} - \boldsymbol{y}\| \exp(-\beta t),\ \forall \boldsymbol{x}, \boldsymbol{y} \tag{4.4}$$

where $\boldsymbol{x}, \boldsymbol{y}$ *are initial values of* $\boldsymbol{x}_t, \boldsymbol{y}_t$*.*

It is easy to see that Equation (4.4) leads to bounded solution of the SDE. If initial value $\boldsymbol{y}$ is chosen to follow the invariant distribution of Equation (4.1) i.e., $\boldsymbol{y} \sim \mu$, then $\boldsymbol{y}_t \sim \mu$ and

$$\begin{aligned}
\mathbb{E}\|\boldsymbol{x}_t\|^2 \leq & \mathbb{E}\|\boldsymbol{x}_t - \boldsymbol{y}_t\|^2 + \mathbb{E}\|\boldsymbol{y}_t\|^2 \\
\leq & \mathbb{E}_{\boldsymbol{y}\sim\mu}\|\boldsymbol{x} - \boldsymbol{y}\|^2 \exp(-2\beta t) + \mathbb{E}\|\boldsymbol{y}_t\|^2 \\
\leq & 2\|\boldsymbol{x}\|^2 \exp(-2\beta t) + \left(1 + 2\exp(-2\beta t)\right) \int_{\mathbb{R}^d} \|\boldsymbol{w}\|^2 \, d\mu \\
\leq & 2\|\boldsymbol{x}\|^2 + 3\int_{\mathbb{R}^d} \|\boldsymbol{w}\|^2 \, d\mu.
\end{aligned}$$

One important and famous example of contractive SDE is Langevin dynamics which we will detail in Section 4.4.

We have the following result for the family of bounded numerical algorithms for contractive SDE:

Theorem 15 *Suppose Equation* (4.1) *is contractive with rate* β *and there is a numerical algorithm* $\mathcal{A}$ *with step size* h *simulating the solution* $\boldsymbol{x}_t$ *of the SDE, whose iterates are denoted by* $\bar{\boldsymbol{x}}_k, k = 0, 1, \cdots$*. If there exists* $h_0 > 0, C_0, C_1, C_2 > 0, p_1 \geq 1, \frac{1}{2} < p_2 \leq p_1 - \frac{1}{2}$ *such that two solutions* $\boldsymbol{x}_t, \boldsymbol{y}_t$ *of Equation* (4.1) *starting from* $\boldsymbol{x}, \boldsymbol{y}$ *satisfy*

$$\boldsymbol{x}_t - \boldsymbol{y}_t = \boldsymbol{x} - \boldsymbol{y} + \boldsymbol{z} \text{ and } \mathbb{E}\|\boldsymbol{z}\|^2 \leq C_0\|\boldsymbol{x} - \boldsymbol{y}\|^2 h, \quad \forall \boldsymbol{x}, \boldsymbol{y}, 0 < h \leq h_0, \tag{4.5}$$

and the algorithm $\mathcal{A}$ has local weak(strong) error of order $p_1(p_2)$

$$\|\mathbb{E}\boldsymbol{x}_h - \mathbb{E}\bar{\boldsymbol{x}}_h\| \leq C_1 h^{p_1}, \quad \left(\mathbb{E}\|\boldsymbol{x}_h - \bar{\boldsymbol{x}}_h\|^2\right)^{\frac{1}{2}} \leq C_2 h^{p_2}, \forall 0 < h \leq h_0 \tag{4.6}$$

where $\boldsymbol{x}_h$ is a solution of Equation (4.1) *with some initial value $\boldsymbol{x}$ and $\bar{\boldsymbol{x}}_h$ is the result of applying $\mathcal{A}$ to $\boldsymbol{x}$ for one step. If the solution of SDE $\boldsymbol{x}_t$ and algorithm $\mathcal{A}$ both start from $\boldsymbol{x}_0$, then for $0 < h \leq \min\{\frac{1}{4\beta}, h_0\}$, the global error $\boldsymbol{e}_k$ at k-th iteration is bounded as*

$$e_k \leq C h^{p_2 - \frac{1}{2}}, \quad k = 0, 1, 2, \cdots \tag{4.7}$$

where $C = \frac{2C_2}{\sqrt{\beta}}\left(1 + \frac{\sqrt{2}\left(\frac{C_1}{C_2} + C_0\right)}{\sqrt{\beta}}\right)$.

Proof: We write the solution of an SDE by $\boldsymbol{x}_{t_0,\boldsymbol{x}_{t_0}}(t_0 + t)$ when the dependence on initialization needs highlight. Denote $t_k = kh$ and $\boldsymbol{x}_{t_k} = \boldsymbol{x}_k$ for better readability.

We have the following decomposition

$$\begin{aligned}
e_{k+1}^2 =& \mathbb{E}\|\boldsymbol{x}_{k+1} - \bar{\boldsymbol{x}}_{k+1}\|^2 \\
=& \mathbb{E}\left\|\boldsymbol{x}_{t_k,\boldsymbol{x}_{t_k}}(t_{k+1}) - \boldsymbol{x}_{t_k,\bar{\boldsymbol{x}}_k}(t_{k+1}) + \boldsymbol{x}_{t_k,\bar{\boldsymbol{x}}_k}(t_{k+1}) - \bar{\boldsymbol{x}}_{k+1}\right\|^2 \\
=& \underbrace{\mathbb{E}\left\|\boldsymbol{x}_{t_k,\boldsymbol{x}_{t_k}}(t_{k+1}) - \boldsymbol{x}_{t_k,\bar{\boldsymbol{x}}_k}(t_{k+1})\right\|^2}_{①} + \underbrace{\mathbb{E}\left\|\boldsymbol{x}_{t_k,\bar{\boldsymbol{x}}_k}(t_{k+1}) - \bar{\boldsymbol{x}}_{k+1}\right\|^2}_{②} \\
&+ 2\underbrace{\mathbb{E}\langle \boldsymbol{x}_{t_k,\boldsymbol{x}_{t_k}}(t_{k+1}) - \boldsymbol{x}_{t_k,\bar{\boldsymbol{x}}_k}(t_{k+1}), \boldsymbol{x}_{t_k,\bar{\boldsymbol{x}}_k}(t_{k+1}) - \bar{\boldsymbol{x}}_{k+1}\rangle}_{③}.
\end{aligned} \tag{4.8}$$

Term ① is taken care of the contraction property

$$\mathbb{E}\left\|\boldsymbol{x}_{t_k,\boldsymbol{x}_{t_k}}(t_{k+1}) - \boldsymbol{x}_{t_k,\bar{\boldsymbol{x}}_k}(t_{k+1})\right\|^2 \leq e_k^2 \exp(-\beta h). \tag{4.9}$$

Term (2) is dealt with by the bound on local strong error

$$\mathbb{E}\left\|\boldsymbol{x}_{t_k,\bar{\boldsymbol{x}}_k}(t_{k+1}) - \bar{\boldsymbol{x}}_{k+1}\right\|^2 \leq C_2^2 h^{2p_2}. \tag{4.10}$$

Term (3) requires more efforts to cope with, and by the decomposition in Equation (4.5) we have

$$\begin{aligned}
&\mathbb{E}\langle \boldsymbol{x}_{t_k,\boldsymbol{x}_{t_k}}(t_{k+1}) - \boldsymbol{x}_{t_k,\bar{\boldsymbol{x}}_k}(t_{k+1}), \boldsymbol{x}_{t_k,\bar{\boldsymbol{x}}_k}(t_{k+1}) - \bar{\boldsymbol{x}}_{k+1}\rangle \\
=&\mathbb{E}\langle \boldsymbol{x}_k - \bar{\boldsymbol{x}}_k, \boldsymbol{x}_{t_k,\bar{\boldsymbol{x}}_k}(t_{k+1}) - \bar{\boldsymbol{x}}_{k+1}\rangle + \mathbb{E}\langle \boldsymbol{z}, \boldsymbol{x}_{t_k,\bar{\boldsymbol{x}}_k}(t_{k+1}) - \bar{\boldsymbol{x}}_{k+1}\rangle \\
\overset{(i)}{=}&\mathbb{E}\langle \boldsymbol{x}_k - \bar{\boldsymbol{x}}_k, \mathbb{E}[\boldsymbol{x}_{t_k,\bar{\boldsymbol{x}}_k}(t_{k+1}) - \bar{\boldsymbol{x}}_{k+1}|\mathcal{F}_k]\rangle + \mathbb{E}\langle \boldsymbol{z}, \boldsymbol{x}_{t_k,\bar{\boldsymbol{x}}_k}(t_{k+1}) - \bar{\boldsymbol{x}}_{k+1}\rangle \\
\overset{(ii)}{\leq}& e_k \left(\mathbb{E}\left\|\mathbb{E}[\boldsymbol{x}_{t_k,\bar{\boldsymbol{x}}_k}(t_{k+1}) - \bar{\boldsymbol{x}}_{k+1}|\mathcal{F}_k]\right\|^2\right)^{\frac{1}{2}} + \left(\mathbb{E}\|\boldsymbol{z}\|^2\right)^{\frac{1}{2}} \left(\mathbb{E}\left\|\boldsymbol{x}_{t_k,\bar{\boldsymbol{x}}_k}(t_{k+1}) - \bar{\boldsymbol{x}}_{k+1}\right\|^2\right) \\
\overset{(iii)}{\leq}& e_k\, C_1 h^{p_1} + C_0 e_k \sqrt{h}\, C_2 h^{p_2} \\
\overset{(iv)}{\leq}& (C_1 + C_0 C_2) e_k h^{p_2+\frac{1}{2}} \\
\overset{(v)}{\leq}& \frac{\beta}{4} e_k^2 h + \frac{(C_1 + C_0 C_2)^2}{\beta} h^{2p_2}
\end{aligned} \tag{4.11}$$

where (i) uses the tower property of conditional expectation and $\mathcal{F}_k$ is the filtration at k-th iteration, (ii) uses Cauchy-Schwarz inequality, (iii) is due to local weak error, local strong error and Equation (4.5), (iv) is due to $p_1 \geq p_2 + \frac{1}{2}$, and (v) is again due to Cauchy-Schwarz inequality.

Now plug Equation (4.9), (4.10) and (4.11) in Equation (4.8), we obtain

$$\begin{aligned}
e_{k+1}^2 \leq& e_k^2 \exp(-\beta h) + \frac{\beta}{2} e_k^2 h + \left(C_2^2 + \frac{2(C_1 + C_0 C_2)^2}{\beta}\right) h^{2p_2} \\
\overset{(i)}{\leq}& \left(1 - \beta h + \frac{\beta^2 h^2}{2}\right) e_k^2 + \frac{\beta h}{2} e_k^2 + \left(C_2^2 + \frac{2(C_1 + C_0 C_2)^2}{\beta}\right) h^{2p_2} \\
\leq& \left(1 - \frac{\beta h}{4}\right) e_k^2 + \left(C_2^2 + \frac{2(C_1 + C_0 C_2)^2}{\beta}\right) h^{2p_2}
\end{aligned}$$

where (i) is due to the assumption $0 < h \leq \frac{1}{4\beta}$ and $e^{-x} \leq 1 - x + \frac{x^2}{2}$ for $0 < x < 1$. Unfolding the above inequality gives us

$$
\begin{aligned}
& e_{k+1}^2 \\
\leq & \left(1 - \frac{\beta h}{4}\right)^{k+1} e_0^2 + \left(1 + \left(1 - \frac{\beta h}{4}\right) + \cdots + \left(1 - \frac{\beta h}{4}\right)^k\right) \left(C_2^2 + \frac{2(C_1 + C_0 C_2)^2}{\beta}\right) h^{2p_2} \\
\leq & \frac{4}{\beta} \left(C_2^2 + \frac{2(C_1 + C_0 C_2)^2}{\beta}\right) h^{2p_2 - 1}.
\end{aligned}
$$

Taking square root on both sides and using $\sqrt{a^2 + b^2} \leq a + b$ yields

$$
e_{k+1} \leq \frac{2C_2}{\sqrt{\beta}} \left(1 + \frac{\sqrt{2}\left(\frac{C_1}{C_2} + C_0\right)}{\sqrt{\beta}}\right) h^{p_2 - \frac{1}{2}}.
$$

■

Remark: Equation (4.5) is only a mild condition, and can be shown for stochastic differential equations with Lipschitz-continuous drift and diffusion terms, see, e.g. [123, Lemma 1.3]. Boundness of numerical algorithms is implicitly assumed in Equation (4.6).

Following Theorem 15, we have a non-asymptotic error bound of the sampling error in 2-Wasserstein distance.

Theorem 16 *Under the same assumption and with the same notation of Theorem 15, we have*

$$
W_2(\mathit{Law}(\bar{\boldsymbol{x}}_k), \mu) \leq \sqrt{2} e^{-\beta k h} W_2(\mathit{Law}(\boldsymbol{x}_0), \mu) + \sqrt{2} C h^{p_2 - \frac{1}{2}}
$$

for $0 < h \leq \min\{\frac{1}{4\beta}, h_0\}$.

Proof: Let $\boldsymbol{y}_0 \sim \mu$ and $(\boldsymbol{x}_0, \boldsymbol{y}_0)$ are coupled such that $\mathbb{E}\|\boldsymbol{x}_0 - \boldsymbol{y}_0\|^2 = W_2^2(\mathrm{Law}(\boldsymbol{x}_0), \mu)$. Denote the solution of Equation (4.1) starting from $\boldsymbol{x}_0, \boldsymbol{y}_0$ by $\boldsymbol{x}_t, \boldsymbol{y}_t$ respectively, and $t_k =$

kh. We have

$$\begin{aligned} W_2^2(\text{Law}(\bar{\boldsymbol{x}}_k), \mu) \leq & \mathbb{E}\left\| \bar{\boldsymbol{x}}_k - \boldsymbol{y}_{t_k} \right\|^2 \\ \leq & 2\mathbb{E}\left\| \bar{\boldsymbol{x}}_k - \boldsymbol{x}_{t_k} \right\|^2 + 2\mathbb{E}\left\| \boldsymbol{x}_{t_k} - \boldsymbol{y}_{t_k} \right\|^2 \\ \overset{(i)}{\leq} & 2e_k^2 + 2\mathbb{E}\|\boldsymbol{x}_0 - \boldsymbol{y}_0\|^2 \exp\left(-2\beta t_k\right) \\ = & 2e_k^2 + 2\exp\left(-2\beta t_k\right) W_2^2(\text{Law}(\boldsymbol{x}_0), \mu) \end{aligned}$$

where (i) is due to the contraction assumption on Equation (4.1).

Taking square roots on both sides, we obtain

$$\begin{aligned} W_2(\text{Law}(\bar{\boldsymbol{x}}_k), \mu) \leq & \sqrt{2e_k^2 + 2\exp\left(-2\beta t_k\right) W_2^2(\text{Law}(\boldsymbol{x}_0), \mu)} \\ \leq & \sqrt{2}\exp\left(-\beta k h\right) W_2(\text{Law}(\boldsymbol{x}_0), \mu) + \sqrt{2} e_k \end{aligned}$$

Invoking the conclusion of Theorem 15 completes the proof. ■

As a natural corollary of Theorem 16, we can characterize the iteration complexity of a general bounded contractive-SDE-based algorithm as shown in Theorem 17

Corollary 17 *Under the same assumption and with the same notation of Theorem 15, after*

$$k \geq k^\star = \max\{4, \frac{1}{\beta h_0}, \frac{1}{\beta}\left(\frac{2C}{\epsilon}\right)^{\frac{1}{p_2 - \frac{1}{2}}}\} \log \frac{2\sqrt{2} W_2(\textit{Law}(\boldsymbol{x}_0), \mu)}{\epsilon}$$

iterations of algorithm $\mathcal{A}$, it is guaranteed that $W_2(\textit{Law}(\bar{\boldsymbol{x}}_k), \mu) \leq \epsilon$. In particular, when high accuracy is needed, i.e., $\epsilon < 2C\left(\min\{\frac{1}{4\beta}, h_0\}\right)^{p_2 - \frac{1}{2}}$, we have

$$k^\star = \frac{(2C)^{\frac{1}{p_2 - \frac{1}{2}}}}{\beta} \frac{1}{\epsilon^{\frac{1}{p_2 - \frac{1}{2}}}} \log \frac{2\sqrt{2} W_2(\textit{Law}(\boldsymbol{x}_0), \mu)}{\epsilon} = \widetilde{\mathcal{O}}\left(\frac{C^{\frac{1}{p_2 - \frac{1}{2}}}}{\beta} \frac{1}{\epsilon^{\frac{1}{p_2 - \frac{1}{2}}}}\right)$$

Proof: Given any tolerance $\epsilon > 0$, we know from Theorem 16 that if k is large enough and

h is small enough such that

$$\sqrt{2}\exp\left(-\beta k h\right) W_2(\text{Law}(\boldsymbol{x}_0), \mu) \leq \frac{\epsilon}{2}. \tag{4.12}$$

$$Ch^{p_2-\frac{1}{2}} \leq \frac{\epsilon}{2} \tag{4.13}$$

we then have $W_2(\text{Law}(\bar{\boldsymbol{x}}_k), \mu) \leq \epsilon$. Solving Inequality (4.12) yields

$$k \geq \frac{1}{\beta h} \log \frac{2\sqrt{2} W_2(\text{Law}(\boldsymbol{x}_0), \mu)}{\epsilon} \triangleq k^\star \tag{4.14}$$

To minimize the lower bound, we want pick step size h as large as possible. Besides $h \leq \min\{\frac{1}{4\beta}, h_0\}$, Equation (4.13) poses further constraint on h, hence we have

$$h \leq \min\{\frac{1}{4\beta}, h_0, \left(\frac{\epsilon}{2C}\right)^{\frac{1}{p_2-\frac{1}{2}}}\}.$$

Plug the upper bound of h in Equation (4.14), we have

$$k^\star = \max\{4, \frac{1}{\beta h_0}, \frac{1}{\beta}\left(\frac{2C}{\epsilon}\right)^{\frac{1}{p_2-\frac{1}{2}}}\} \log \frac{2\sqrt{2} W_2(\text{Law}(\boldsymbol{x}_0), \mu)}{\epsilon}.$$

When high accuracy is needed, i.e., $\epsilon < 2C\left(\min\{\frac{1}{4\beta}, h_0\}\right)^{p_2-\frac{1}{2}}$, we have

$$k^\star = \frac{(2C)^{\frac{1}{p_2-\frac{1}{2}}}}{\beta} \frac{1}{\epsilon^{\frac{1}{p_2-\frac{1}{2}}}} \log \frac{2\sqrt{2} W_2(\text{Law}(\boldsymbol{x}_0), \mu)}{\epsilon} = \widetilde{\mathcal{O}}\left(\frac{C^{\frac{1}{p_2-\frac{1}{2}}}}{\beta} \frac{1}{\epsilon^{\frac{1}{p_2-\frac{1}{2}}}}\right)$$

■

Corollary 17 states how iteration complexity depends on the order of local (strong) error (i.e., p_2) of a numerical algorithm. Clearly, the higher order error an algorithm yields locally, the better the iteration complexity of the algorithm is, in term of the dependence on tolerance/accuracy ϵ. What is probably somewhat surprising, is that Corollary 17 reveals that a high-order numerical algorithm can also improve on the iteration complexity's

dependence on various parameters of the underlying SDE, e.g., dimension d, condition number of potential function κ, which are all contained in C.

As an application of Theorem 16 and Corollary 17, we will work with Langevin dynamics and its discretization Langevin Monte Carlo algorithm, and derive tight bounds on its iteration complexity in the next section.

4.4 Application to Langevin Monte Carlo Algorithm

Assumption 4 *(L-**smoothness**) Assume f is L-smooth, i.e. there exists $L > 0$ such that*

$$\left\|\nabla f(\boldsymbol{x}) - \nabla f(\boldsymbol{y})\right\| \leq L\|\boldsymbol{x} - \boldsymbol{y}\|, \quad \forall \boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^d.$$

Assumption 5 *(**Strongly-convex potential**) Suppose f is m-strongly-convex, i.e., there exists a constant $m > 0$ such that*

$$f(\boldsymbol{y}) - f(\boldsymbol{x}) - \langle \nabla f(\boldsymbol{x}), \boldsymbol{y} - \boldsymbol{x} \rangle \geq \frac{m}{2}\|\boldsymbol{y} - \boldsymbol{x}\|^2, \quad \forall \boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^d.$$

If f also satisfies Assumption 4, we denote $\kappa \triangleq \frac{L}{m}$, known as the condition number of f.

Assumption 6 *(**3rd-order derivatives grow at most linearly**) Assume the operator $\nabla(\Delta f)$ grows at most linearly, i.e., there exists a constant $G > 0$ such that*

$$\left\|\nabla(\Delta f(\boldsymbol{x}))\right\| \leq G\left(1 + \|\boldsymbol{x}\|\right).$$

In addition, for normalization purpose, we assume without loss of generality, the origin is a local minimizer of f, i.e. $\nabla f(\mathbf{0}) = \mathbf{0}$.

We first show in Lemma 18 that Langevin dynamics is a member of the family of contractive SDE, and with a contraction rate of strong-convexity coefficient $\beta = m$.

Lemma 18 *Suppose Assumption 5 holds. Then two copies of overdamped Langevin dynamics have the following contraction property*

$$\left\{\mathbb{E}\|\boldsymbol{y}_t - \boldsymbol{x}_t\|^2\right\}^{\frac{1}{2}} \leq \|\boldsymbol{y} - \boldsymbol{x}\| \exp(-mt)$$

where $\boldsymbol{x}, \boldsymbol{y}$ are the initial values of $\boldsymbol{x}_t, \boldsymbol{y}_t$.

Proof: $\boldsymbol{x}_t, \boldsymbol{y}_t$ are respectively the solutions to

$$\begin{aligned} d\boldsymbol{x}_t =& -\nabla f(\boldsymbol{x}_t)dt + \sqrt{2}d\boldsymbol{B}_t \\ d\boldsymbol{y}_t =& -\nabla f(\boldsymbol{y}_t)dt + \sqrt{2}d\boldsymbol{B}_t \end{aligned}$$

where $\boldsymbol{B}_t$ is a standard d-dimensional Brownian motion. Denote $L_t = \frac{1}{2}\mathbb{E}\|\boldsymbol{y}_t - \boldsymbol{x}_t\|^2$ and take time derivative, we obtain

$$\frac{d}{dt}L_t = -\mathbb{E}\langle \boldsymbol{y}_t - \boldsymbol{x}_t, \nabla f(\boldsymbol{y}_t) - \nabla f(\boldsymbol{x}_t)\rangle \overset{(i)}{\leq} -m\mathbb{E}\|\boldsymbol{y}_t - \boldsymbol{x}_t\|^2 = -2mL_t$$

where (i) is due to the strong-convexity assumption made on f. We then obtain $L_t \leq L_0 \exp(-2mt)$ and it follows that

$$\left\{\mathbb{E}\|\boldsymbol{y}_t - \boldsymbol{x}_t\|^2\right\}^{\frac{1}{2}} \leq \|\boldsymbol{y} - \boldsymbol{x}\| \exp(-mt)$$

■

Next, we will need to work out the constants C_0, C_1, C_2 needed in Theorem 15. Before proceed with the derivation, we introduce the following lemma on the growth of overdamped Langevin dynamics, which turns out to be very useful in quantifying local weak error and local strong error.

Lemma 19 *Suppose Assumption 4 and 5 hold, then when $0 \leq h \leq \frac{1}{4\kappa L}$, the solution of*

overdamped Langevin dynamics $\boldsymbol{x}_t$ over satisfies

$$\mathbb{E}\|\boldsymbol{x}_h - \boldsymbol{x}\|^2 \leq 6\left(d + \frac{m}{2}\|\boldsymbol{x}\|^2\right)h$$

where $\boldsymbol{x}$ is the initial value at $t = 0$.

Proof: We have

$$\begin{aligned}
\mathbb{E}\|\boldsymbol{x}_h - \boldsymbol{x}\|^2 =& \mathbb{E}\left\|-\int_0^h \nabla f(\boldsymbol{x}_t)dt + \sqrt{2}\int_0^h d\boldsymbol{B}_t\right\|^2 \\
\leq& 2\mathbb{E}\left\|\int_0^h \nabla f(\boldsymbol{x}_t)dt\right\|^2 + 4\mathbb{E}\left\|\int_0^h d\boldsymbol{B}_t\right\|^2 \\
\overset{(i)}{=}& 2\mathbb{E}\left\|\int_0^h \nabla f(\boldsymbol{x}_t)dt\right\|^2 + 4hd \\
\leq& 2\mathbb{E}\left[\left(\int_0^h \left\|\nabla f(\boldsymbol{x}_t) - \nabla f(\boldsymbol{x})\right\| dt + \int_0^h \left\|\nabla f(\boldsymbol{x})\right\| dt\right)^2\right] + 4hd \\
\leq& 2\mathbb{E}\left[\left(L\int_0^h \|\boldsymbol{x}_t - \boldsymbol{x}\| dt + h\left\|\nabla f(\boldsymbol{x})\right\|\right)^2\right] + 4hd \\
\leq& 4\mathbb{E}\left[L^2\left(\int_0^h \|\boldsymbol{x}_t - \boldsymbol{x}\| dt\right)^2 + h^2\left\|\nabla f(\boldsymbol{x})\right\|^2\right] + 4hd \\
\overset{(ii)}{\leq}& 4hd + 4h^2\left\|\nabla f(\boldsymbol{x})\right\|^2 + 4L^2 h\int_0^h \mathbb{E}\|\boldsymbol{x}_t - \boldsymbol{x}\|^2 dt
\end{aligned}$$

where (i) is due to Ito's isometry, (ii) is due to Cauchy-Schwarz inequality. By Gronwall's inequality, we obtain

$$\mathbb{E}\|\boldsymbol{x}_h - \boldsymbol{x}\|^2 \leq 4h\left(d + h\left\|\nabla f(\boldsymbol{x})\right\|^2\right)\exp\left\{4L^2h^2\right\}.$$

Since $\left\|\nabla f(\boldsymbol{x})\right\| = \left\|\nabla f(\boldsymbol{x}) - \nabla f(\boldsymbol{0})\right\| \leq L\|\boldsymbol{x}\|$, when $0 < h \leq \frac{1}{4\kappa L}$, we finally reach at

$$\mathbb{E}\|\boldsymbol{x}_h - \boldsymbol{x}\|^2 \leq 4e^{\frac{1}{4}}\left(d + 2hL^2\|\boldsymbol{x}\|^2\right)h \leq 6\left(d + \frac{m}{2}\|\boldsymbol{x}\|^2\right)h$$

■

We are now ready to compute $C_0, C_1, C)2$. As the following result shows, we have $C_0 = \frac{\sqrt{m}}{2}$.

Lemma 20 *Suppose Assumption 4 and 5 hold. Let $\boldsymbol{x}_t, \boldsymbol{y}_t$ be two solutions of overdamped Langevin dynamics starting from $\boldsymbol{x}, \boldsymbol{y}$ respectively, for $0 < h \leq \frac{1}{4\kappa L}$, we have the following representation*

$$\boldsymbol{x}_h - \boldsymbol{y}_h = \boldsymbol{x} - \boldsymbol{y} + \boldsymbol{z}$$

with

$$E\|\boldsymbol{z}\|^2 \leq \frac{m}{4}\|\boldsymbol{x} - \boldsymbol{y}\|^2 h$$

Proof: Let $\boldsymbol{z} = (\boldsymbol{x}_h - \boldsymbol{y}_h) - (\boldsymbol{x} - \boldsymbol{y}) = -\int_0^h \nabla f(\boldsymbol{x}_s) - \nabla f(\boldsymbol{y}_s) ds$. Ito's lemma readily implies that

$$\begin{aligned}
\mathbb{E}\|\boldsymbol{x}_h - \boldsymbol{y}_h\|^2 =& \|\boldsymbol{x} - \boldsymbol{y}\|^2 - 2\mathbb{E}\int_0^h \langle \boldsymbol{x}_s - \boldsymbol{y}_s, \nabla f(\boldsymbol{x}_s) - \nabla f(\boldsymbol{y}_s)\rangle ds \\
\overset{(i)}{\leq}& \|\boldsymbol{x} - \boldsymbol{y}\|^2 - 2m\int_0^h \mathbb{E}\|\boldsymbol{x}_s - \boldsymbol{y}_s\|^2 ds \\
\leq& \|\boldsymbol{x} - \boldsymbol{y}\|^2
\end{aligned}$$

where (i) is due to strong-convexity of f. We then have that

$$\begin{aligned}
\mathbb{E}\|\boldsymbol{z}\|^2 =& \left\| \mathbb{E}\left[\int_0^h \nabla f(\boldsymbol{x}_s) - \nabla f(\boldsymbol{y}_s) ds \right] \right\|^2 \\
\leq& \left(\int_0^h \left\| \mathbb{E}\left[\nabla f(\boldsymbol{x}_s) - \nabla f(\boldsymbol{y}_s) \right] \right\| ds \right)^2 \\
\leq& \int_0^h 1^2 ds \int_0^h \left\| \mathbb{E}\left[\nabla f(\boldsymbol{x}_s) - \nabla f(\boldsymbol{y}_s) \right] \right\|^2 ds \\
\leq& h \int_0^h \mathbb{E} \left\| \nabla f(\boldsymbol{x}_s) - \nabla f(\boldsymbol{y}_s) \right\|^2 ds \\
\leq& L^2 h \int_0^h \mathbb{E}\|\boldsymbol{x}_s - \boldsymbol{y}_s\|^2 ds \\
\leq& L^2 \|\boldsymbol{x} - \boldsymbol{y}\|^2 h^2 \\
\overset{(i)}{\leq}& \frac{m}{4} \|\boldsymbol{x} - \boldsymbol{y}\|^2 h
\end{aligned}$$

where (i) is due to $h \leq \frac{1}{4\kappa L}$. ■

The local strong error and local weak error are bounded in Lemma 21 and 22 respectively.

Lemma 21 *Suppose Assumption 4 and 5 hold. Denote the one-step iteration of LMC algorithm with step size h by $\bar{\boldsymbol{x}}_h$ and the solution of overdamped Langevin dynamics at time $t = h$ by $\boldsymbol{x}_h$. Both the discrete algorithm and the continuous dynamics start from the same initial value $\boldsymbol{x}$. If $0 \leq h \leq \frac{1}{4\kappa L}$, then the local strong error of LMC algorithm satisfies*

$$\left\{ \mathbb{E}\|\bar{\boldsymbol{x}}_h - \boldsymbol{x}_h\|^2 \right\}^{\frac{1}{2}} \leq \widetilde{C}_2 h^{\frac{3}{2}}$$

with $\widetilde{C}_2 = 2L \left(d + \frac{m}{2} \|\boldsymbol{x}\|^2 \right)^{\frac{1}{2}}$.

Proof: We have for $0 \leq h \leq \frac{1}{4\kappa L}$,

$$\begin{aligned}
\mathbb{E}\|\bar{\boldsymbol{x}}_h - \boldsymbol{x}_h\|^2 =& \mathbb{E}\left\|\int_0^h \nabla f(\boldsymbol{x}_s) - \nabla f(\boldsymbol{x}) ds\right\|^2 \\
\leq& \mathbb{E}\left(\int_0^h \left\|\nabla f(\boldsymbol{x}_s) - \nabla f(\boldsymbol{x})\right\| ds\right)^2 \\
\leq& L^2 \mathbb{E}\left(\int_0^h \|\boldsymbol{x}_s - \boldsymbol{x}\| ds\right)^2 \\
\overset{(i)}{\leq}& L^2 h \int_0^h \mathbb{E}\|\boldsymbol{x}_s - \boldsymbol{x}\|^2 ds \\
\overset{(ii)}{\leq}& 3L^2 \left(d + \frac{m}{2}\|\boldsymbol{x}\|^2\right) h^3
\end{aligned}$$

where (i) is due to Cauchy-Schwartz inequality and (ii) is due to Lemma 19. Taking square roots on both side completes the proof. ■

Lemma 22 *Suppose Assumption 4, 5 and 6 hold. Denote the one-step iteration of LMC algorithm with step size h by $\bar{\boldsymbol{x}}_h$ and the solution of overdamped Langevin dynamics at time $t = h$ by $\boldsymbol{x}_h$. Both the discrete algorithm and the continuous dynamics start from the same initial value $\boldsymbol{x}$. If $0 \leq h \leq \frac{1}{4\kappa L}$, then the local weak error of LMC algorithm satisfies*

$$\|\mathbb{E}\bar{\boldsymbol{x}}_h - \mathbb{E}\boldsymbol{x}_h\| \leq \widetilde{C}_1 h^2$$

with $\widetilde{C}_1 = \frac{1}{2}\left(\sqrt{m}\left(L + \frac{G}{L}\right)\sqrt{d + \frac{m}{2}\|\boldsymbol{x}\|^2} + \|\boldsymbol{x}\| + G\right)$.

Proof: By Ito's lemma, we have

$$d\nabla f(\boldsymbol{x}_t) = -\nabla^2 f(\boldsymbol{x}_t)\nabla f(\boldsymbol{x}_t)dt + \nabla(\Delta f(\boldsymbol{x}_t))dt + \sqrt{2}\int_0^t \nabla^2 f(\boldsymbol{x}_t) d\boldsymbol{B}_t.$$

It follows that

$$
\begin{aligned}
&\|\mathbb{E}\bar{\boldsymbol{x}}_h - \mathbb{E}\boldsymbol{x}_h\| \\
=&\left\|\mathbb{E}\int_0^h \nabla f(\boldsymbol{x}_s) - \nabla f(\boldsymbol{x}) ds\right\| \\
=&\left\|\mathbb{E}\left\{\int_0^h\int_0^s -\nabla^2 f(\boldsymbol{x}_r)\nabla f(\boldsymbol{x}_r) + \nabla(\Delta f(\boldsymbol{x}_r)) dr ds + \sqrt{2}\int_0^h\int_0^s \nabla^2 f(\boldsymbol{x}_r) d\boldsymbol{B}_r ds\right\}\right\| \\
=&\left\|\mathbb{E}\left\{\int_0^h\int_0^s -\nabla^2 f(\boldsymbol{x}_r)\nabla f(\boldsymbol{x}_r) + \nabla(\Delta f(\boldsymbol{x}_r)) dr ds\right\}\right\| \\
\leq&\int_0^h\int_0^s \mathbb{E}\left\|\nabla^2 f(\boldsymbol{x}_r)\nabla f(\boldsymbol{x}_r)\right\| dr ds + \int_0^h\int_0^s \mathbb{E}\left\|\nabla(\Delta f(\boldsymbol{x}_r))\right\| dr ds \\
\leq& L\int_0^h\int_0^s \mathbb{E}\left\|\nabla f(\boldsymbol{x}_r)\right\| dr ds + \int_0^h\int_0^s \mathbb{E}\left\|\nabla(\Delta f(\boldsymbol{x}_r))\right\| dr ds \\
\overset{(i)}{\leq}&(L^2+G)\int_0^h\int_0^s \mathbb{E}\|\boldsymbol{x}_r\| dr ds + \frac{G}{2}h^2 \\
\leq&(L^2+G)\int_0^h\int_0^s \mathbb{E}\|\boldsymbol{x}_r - \boldsymbol{x}\| dr ds + \frac{h^2}{2}\|\boldsymbol{x}\| + \frac{G}{2}h^2 \\
\overset{(ii)}{\leq}&(L^2+G)\int_0^h\int_0^s \sqrt{\mathbb{E}\|\boldsymbol{x}_r - \boldsymbol{x}\|^2} dr ds + \frac{h^2}{2}\|\boldsymbol{x}\| + \frac{G}{2}h^2 \\
\overset{(iii)}{\leq}&(L^2+G)\int_0^h\int_0^s \sqrt{6\left(d + \frac{m}{2}\|\boldsymbol{x}\|^2\right) r} dr ds + \frac{h^2}{2}\|\boldsymbol{x}\| + \frac{G}{2}h^2 \\
\leq&\left((L^2+G)\sqrt{d + \frac{m}{2}\|\boldsymbol{x}\|^2}\sqrt{h} + \frac{1}{2}\|\boldsymbol{x}\| + \frac{1}{2}G\right)h^2 \\
\overset{(iv)}{\leq}&\frac{1}{2}\left(\sqrt{m}\left(L + \frac{G}{L}\right)\sqrt{d + \frac{m}{2}\|\boldsymbol{x}\|^2} + \|\boldsymbol{x}\| + G\right)h^2
\end{aligned}
$$

where (i) is due to Assumption 6, (ii) is due to Jensen's inequality, (iii) is due to Lemma 19 and (iv) is due to $h \leq \frac{1}{4\kappa L}$ ∎

Note that the bound for local strong/weak error depends on initial value, which changes from iteration to iteration. It would be helpful if we have a uniform bound on all iterates of LMC. To this end, we provide such a bound in Lemma 23.

Lemma 23 *Suppose Assumption 4 and 5 hold. Denote the iterates of LMC by $\bar{\boldsymbol{x}}_k$. If*

$0 \leq h \leq \frac{1}{4\kappa L}$ *we then have the iterates of LMC algorithm are uniformly upper bounded by*

$$\mathbb{E}\|\bar{\boldsymbol{x}}_k\|^2 \leq \|\boldsymbol{x}_0\|^2 + \frac{8d}{7m}, \quad \forall k \geq 0$$

Proof: We have

$$\begin{aligned}
\mathbb{E}\|\bar{\boldsymbol{x}}_{k+1}\|^2 =& \mathbb{E}\left\|\bar{\boldsymbol{x}}_k - h\nabla f(\bar{\boldsymbol{x}}_k) + \sqrt{2h}\boldsymbol{\xi}_{k+1}\right\|^2 \\
\overset{(i)}{=}& \mathbb{E}\|\bar{\boldsymbol{x}}_k\|^2 + h^2\mathbb{E}\left\|\nabla f(\bar{\boldsymbol{x}}_k)\right\|^2 + 2hd - 2h\mathbb{E}\langle\bar{\boldsymbol{x}}_k, \nabla f(\bar{\boldsymbol{x}}_k)\rangle \\
\overset{(ii)}{\leq}& \mathbb{E}\|\bar{\boldsymbol{x}}_k\|^2 + h^2L^2\mathbb{E}\|\bar{\boldsymbol{x}}_k\|^2 + 2hd - 2h\mathbb{E}\langle\bar{\boldsymbol{x}}_k, \nabla f(\bar{\boldsymbol{x}}_k)\rangle \\
\overset{(iii)}{\leq}& \mathbb{E}\|\bar{\boldsymbol{x}}_k\|^2 + h^2L^2\mathbb{E}\|\bar{\boldsymbol{x}}_k\|^2 + 2hd - 2mh\mathbb{E}\|\bar{\boldsymbol{x}}_k\|^2 \\
\overset{(iv)}{\leq}& (1 - \frac{7}{4}mh)\mathbb{E}\|\bar{\boldsymbol{x}}_k\|^2 + 2hd
\end{aligned}$$

where (i) is due to the independence between $\boldsymbol{\xi}_{k+1}$ and $\bar{\boldsymbol{x}}_k$, (ii) is due to Assumption 4, (iii) is due to the property of m-strongly-convex functions, $\langle\nabla f(\boldsymbol{y}) - \nabla f(\boldsymbol{x}), \boldsymbol{y} - \boldsymbol{x}\rangle \geq m\|\boldsymbol{y} - \boldsymbol{x}\|^2 \ \forall \boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^d$, and (iv) uses the assumption $h \leq \frac{1}{4\kappa L}$.

Unfolding the inequality, we obtain

$$\mathbb{E}\|\bar{\boldsymbol{x}}_k\|^2 \leq (1 - \frac{7}{4}mh)^k\mathbb{E}\|\bar{\boldsymbol{x}}_0\|^2 + 2hd\left(1 + \frac{7}{4}mh + \cdots + (\frac{7}{4}mh)^{k-1}\right) \leq \|\boldsymbol{x}_0\|^2 + \frac{8d}{7m}$$

■

Now, combine Lemma 23 with (conditional expectation version of) Lemma 21 and 22, we obtain C_1 and C_2, namely

$$\begin{aligned}
\widetilde{C}_1 \leq& \frac{1}{2}\left(\sqrt{m}\left(L + \frac{G}{L}\right)\sqrt{d + \frac{m}{2}(\|\boldsymbol{x}_0\|^2 + \frac{8d}{7m})} + \sqrt{\|\boldsymbol{x}_0\|^2 + \frac{8d}{7m}} + G\right) \\
\leq& \frac{1}{2}\left(\left[m\left(L + \frac{G}{L}\right) + 1\right]\sqrt{\frac{2d}{m} + \|\boldsymbol{x}_0\|^2} + G\right) \triangleq C_1
\end{aligned}$$

and

$$\widetilde{C}_2 \leq 2L\left(d+\frac{m}{2}\left(\|\boldsymbol{x}_0\|^2+\frac{8d}{7m}\right)\right)^{\frac{1}{2}} \leq 2\sqrt{m}L\sqrt{\frac{2d}{m}+\|\boldsymbol{x}_0\|^2} \triangleq C_2$$

With all the necessary ingredients, we now invoke Theorem 16 and obtain the following result:

Theorem 24 *Suppose assumption 4, 5 and 6 hold. If we run LMC from $\boldsymbol{x}_0$, then after k iterations, we have*

$$W_2(\mathit{Law}(\bar{\boldsymbol{x}}_k),\mu) \leq \sqrt{2}e^{-mkh}W_2(\mathit{Law}(\boldsymbol{x}_0),\mu)+\sqrt{2}C_{\mathit{LMC}}h, \quad 0<h\leq\frac{1}{4\kappa L}, k\in\mathbb{N}$$

where $C_{\mathit{LMC}} = \frac{L}{\sqrt{m}}\sqrt{\frac{2d}{m}+\|\boldsymbol{x}_0\|^2}\left(7+\sqrt{2}\frac{m\left(L+\frac{G}{L}\right)+1}{mL}\right)+\frac{\sqrt{2}G}{m^{\frac{3}{2}}} = \mathcal{O}(\sqrt{d})$.

Proof: We collection all constants here in the proof for easier reference

$$\begin{aligned}
&\beta = m,\ h_0 = \frac{1}{4\kappa L},\ C_0 = \frac{\sqrt{m}}{2}\\
&C_1 = \frac{1}{2}\left(\left[m\left(L+\frac{G}{L}\right)+1\right]\sqrt{\frac{2d}{m}+\|\boldsymbol{x}_0\|^2}+G\right)\\
&C_2 = 2\sqrt{m}L\sqrt{\frac{2d}{m}+\|\boldsymbol{x}_0\|^2}.
\end{aligned}$$

Then the constant in Theorem 15 for LMC algorithm is

$$C_{\text{LMC}} = \frac{L}{\sqrt{m}}\sqrt{\frac{2d}{m}+\|\boldsymbol{x}_0\|^2}\left(7+\sqrt{2}\frac{m\left(L+\frac{G}{L}\right)+1}{mL}\right)+\frac{\sqrt{2}G}{m^{\frac{3}{2}}}$$

Assuming L, m, G are all constants then clearly $C_{\text{LMC}} = \mathcal{O}(\sqrt{d})$. Then applying Theorem 16 to LMC, we have

$$W_2(\text{Law}(\bar{\boldsymbol{x}}_k),\mu) \leq \sqrt{2}e^{-mkh}W_2(\text{Law}(\boldsymbol{x}_0),\mu)+\sqrt{2}C_{\text{LMC}}h \tag{4.15}$$

for $0<h\leq\frac{1}{4\kappa L}$. ■

Note that the discretization error term $\sqrt{2}C_{\text{LMC}}h$ is tight (up to a constant) in terms of the dependence on the order of step size h and dimension d. For example, consider a standard d-dimensional Gaussian as our target, then $f(\boldsymbol{x}) = \frac{1}{2}\|\boldsymbol{x}\|^2, \boldsymbol{x} \in \mathbb{R}^d$ with $L = m = 1, G = 0$. For simplicity, suppose LMC algorithm starts from $\boldsymbol{x}_0 = \mathbf{0}$. We then have $C_{\text{LMC}} = (7+2\sqrt{d})\sqrt{2d}$ and the discretization error is $\sqrt{2}C_{\text{LMC}}h = (4+4\sqrt{2})\sqrt{d}h$. We have explicit expression for the solution $\boldsymbol{x}_t$ of the linear Langevin equation (Ornstein-Unlenbeck process) and the iterates $\bar{\boldsymbol{x}}_k$ of LMC

$$
\begin{aligned}
\boldsymbol{x}_t =& \sqrt{2}\int_0^t \exp\left(-(t-s)\right) d\boldsymbol{B}_s \sim \mathcal{N}\left(\mathbf{0}, (1-e^{-2t})I\right), \\
\bar{\boldsymbol{x}}_k =& \sqrt{2h}\left(\boldsymbol{\xi}_k + (1-h)^2\boldsymbol{\xi}_{k-1} + \cdots + (1-h)^{2(k-1)}\boldsymbol{\xi}_1\right) \sim \mathcal{N}\left(\mathbf{0}, \frac{2}{2-h}\left(1-(1-h)^{2k}\right)I\right).
\end{aligned}
$$

When $0 < h < \frac{1}{L} = 1$, the discretization error of LMC is

$$
\begin{aligned}
W_2(\text{Law}(\bar{\boldsymbol{x}}_k), \text{Law}(\boldsymbol{x}_{kh})) =& \sqrt{d}\left(\sqrt{\frac{2}{2-h}\left(1-(1-h)^{2k}\right)} - \sqrt{1-e^{-2kh}}\right) \\
\overset{(i)}{\leq}& \sqrt{d}\sqrt{1-e^{-2kh}}\left(\sqrt{\frac{2}{2-h}} - 1\right) \\
\leq& \frac{1}{2}\sqrt{d}h
\end{aligned}
$$

where (i) is due to $1 - x < e^{-x}$. The discretization error has the same order of dependence in dimension d and step size h. Therefore, our quantification of the global discretization error of LMC is already tight.

In the same vein, we apply Corollary 17 to LMC and obtain the following characterization of iteration complexity for LMC.

Theorem 25 *Suppose assumption 4, 5 and 6 hold. If we run LMC from $\boldsymbol{x}_0$, then when*

$$
k > k^{\star}_{LMC} = \max\{4\kappa^2, \frac{2C_{LMC}}{m}\frac{1}{\epsilon}\}\log\frac{2\sqrt{2}W_2(Law(\boldsymbol{x}_0), \mu)}{\epsilon},
$$

Table 4.1: Comparison of iteration complexity results in 2-Wassertein distance of LMC with L-smooth and m-strongly-convex potential.

	Iteration Complexity	Additional Assumption
[27, Theorem 1]	$\widetilde{O}\left(\frac{d}{\epsilon^2}\right)$	N/A
[29, Theorem 1]	$\widetilde{O}\left(\frac{d}{\epsilon^2}\right)$	N/A
[126, Corollary 10]	$\widetilde{O}\left(\frac{d}{\epsilon^2}\right)$	N/A
[33, Theorem 8]	$\widetilde{O}\left(\frac{d}{\epsilon}\right)$	$\left\|\nabla^2 f(\boldsymbol{x}) - \nabla^2 f(\boldsymbol{y})\right\| \leq \widetilde{L}\|\boldsymbol{x}-\boldsymbol{y}\|$
This work (Theorem 24)	$\widetilde{O}\left(\frac{\sqrt{d}}{\epsilon}\right)$	Assumption 6

it is guarantees that $W_2(\mathit{Law}(\bar{\boldsymbol{x}}_k), \mu) \leq \epsilon$. C_{LMC} is the same as defined in Theorem 24. When high accuracy is needed, i.e., $\epsilon \leq \frac{C_{\mathit{LMC}}}{2m\kappa^2}$, we further have

$$k^{\star}_{\mathit{LMC}} = \frac{2C_{\mathit{LMC}}}{m}\frac{1}{\epsilon}\log\frac{2\sqrt{2}W_2(\mathit{Law}(\boldsymbol{x}_0),\mu)}{\epsilon} = \widetilde{O}\left(\frac{\sqrt{d}}{\epsilon}\right).$$

Proof: The proof is straight by Corollary 17 and the computation of C_{LMC} from Theorem 24. ∎

The $\widetilde{O}\left(\frac{\sqrt{d}}{\epsilon}\right)$ iteration complexity in 2-Wasserstein distance improves upon the previous ones [27, 29, 33, 126]. A brief comparison is summarized in Table 4.1. We note that a recent work [86] establishes $\widetilde{O}(\sqrt{d})$ iteration complexity in 2-Wasserstein distance for Metropolis-Adjusted Langevin Algorithm (MALA) under warm start assumption, and the dimension dependence is optimal. In view of Corollary 17, LMC hence has the same dimension dependence as MALA.

4.5 Conclusion

In this chapter, we revisit the classical mean-square analysis for a family of bounded, contractive-SDE-based numerical algorithms and extend the global error bound of mean-square analysis from finite time to infinite time. The global error is further used to derive a $\widetilde{\mathcal{O}}\left(C^{\frac{1}{p_2-\frac{1}{2}}}\frac{1}{\epsilon^{\frac{1}{p_2-\frac{1}{2}}}}\right)$ iteration complexity in 2-Wasserstein distance. The iteration complexity bound unveils how a high-order numerical algorithm can help improve dependence on

various parameters, e..g. dimension. When applied to Langevin Monte Carlo algorithm, we obtain an improved $\widetilde{\mathcal{O}}\left(\frac{\sqrt{d}}{\epsilon}\right)$ under the standard smoothness and strongly-convexity assumption, plus an addition linear growth condition on the third-order derivative of the potential function.

Appendices

APPENDIX A

SUPPLEMENTARY MATERIALS OF CHAPTER 2

A.1 Mini Batch Version of EWSG

When mini batch size $b > 1$, for each mini batch $\{i_1, i_2, \cdots, i_b\}$, we use $\frac{n}{b}\sum_{j=1}^{b}\nabla f_{i_j}$ to approximate full gradient ∇f, and assign the mini batch $\{i_1, i_2, \cdots, i_b\}$ probability $p_{i_1 i_2, \cdots, i_b}$. We can easily extend the transition probability of $b = 1$ to general b, simply by replacing $n\nabla f_i$ with $\frac{n}{b}\sum_{j=1}^{b}\nabla f_{i_j}$ and end up with

$$\tilde{P}(\boldsymbol{\theta}_{k+1}, \boldsymbol{r}_{k+1}|\boldsymbol{\theta}_k, \boldsymbol{r}_k) = \delta(\boldsymbol{\theta}_{k+1} = \boldsymbol{\theta}_k + \boldsymbol{r}_k h)\times$$

$$\sum_{i_1, i_2, \cdots, i_b} p_{i_1 i_2 \cdots i_b} \Phi\left(\boldsymbol{x} + n\boldsymbol{a}_{i_1 i_2 \cdots i_b}\right) \frac{1}{\sigma\sqrt{h}}$$

where

$$\boldsymbol{x} = \frac{\boldsymbol{r}_{k+1} - \boldsymbol{r}_k + h\gamma\boldsymbol{r}_k}{\sigma\sqrt{h}}, \quad \boldsymbol{a}_{i_1 i_2 \cdots i_b} = \frac{\sqrt{h}}{\sigma}\frac{1}{b}\sum_{j=1}^{b}\nabla f_{i_j}(\boldsymbol{\theta}_k)$$

Therefore, to match the transition probability of underdamped Langevin dynamics with stochastic gradient and full gradient, we let $p_{i_1 i_2 \cdots i_b} =$

$$\frac{1}{Z}\exp\left\{\frac{1}{2}\left[\|\boldsymbol{x} + n\boldsymbol{a}_{i_1 i_2 \cdots i_b}\|^2 - \|\boldsymbol{x} + \sum_{i_1 i_2 \cdots i_b}\boldsymbol{a}_{i_1 i_2 \cdots i_b}\|^2\right]\right\}$$

where Z is a normalization constant.

To sample multidimensional random data indices $I_1, \cdots, I_b$ from $p_{i_1 i_2 \cdots i_b}$, we again use a Metropolis chain, whose acceptance probability only depends on $a_{i_1 i_2 \cdots i_b}$ and $a_{j_1 j_2 \cdots j_b}$ but not the full gradient.

A.2 EWSG Version for Overdamped Langevin

Overdamped Langevin equation is the following SDE

$$d\boldsymbol{\theta}_t = -\nabla f(\boldsymbol{\theta}_t)dt + \sqrt{2}d\boldsymbol{B}_t$$

where $V(\boldsymbol{\theta}) = \sum_{i=1}^{n} V_i(\boldsymbol{\theta})$ and B_t is a d-dimensional Brownian motion. The Euler-Maruyama discretization is

$$\boldsymbol{\theta}_{k+1} = \boldsymbol{\theta}_k - h\nabla f(\boldsymbol{\theta}_k) + \sqrt{2h}\boldsymbol{\xi}_{k+1}$$

where $\boldsymbol{\xi}_{k+1}$ is a d-dimensional random Gaussian vector. When stochastic gradient is used, the above numerical schedme turns to

$$\boldsymbol{\theta}_{k+1} = \boldsymbol{\theta}_k - h\nabla f_{I_k}(\boldsymbol{\theta}_k) + \sqrt{2h}\boldsymbol{\xi}_{k+1}$$

where I_k is the datum index used in k-th iteration to estimate the full gradient.

Denote $\boldsymbol{x} = \frac{\boldsymbol{\theta}_{k+1}-\boldsymbol{\theta}_k}{\sqrt{2h}}$ and $\boldsymbol{a}_i = \frac{\sqrt{h}\nabla f_i(\boldsymbol{\theta}_k)}{\sqrt{2}}$. If we set

$$p_i = \mathbb{P}(I_k = i) \propto \exp\Big\{ - \frac{\|\boldsymbol{x} + \sum_{j=1}^{n} \boldsymbol{a}_j\|^2}{2} + \frac{\|\boldsymbol{x} + n\boldsymbol{a}_i\|^2}{2} \Big\}$$

and follow the same steps in the derivation of EWSG for ULD, we will see the transition kernel of full gradient and the transition kernel of stochastic gradient are matched up.

A.3 Variance Reduction (VR)

We have seen that when step size h is large, EWSG still introduces extra variance. To further mitigate this inaccuracy, we provide in this section a complementary variance reduction technique.

Locally (i.e., conditioned on the state of the system at the current step), we have increased variance

$$\begin{aligned}\operatorname{cov}[\boldsymbol{r}_{k+1}|\boldsymbol{r}_k] &= \mathbb{E}[\operatorname{cov}[\boldsymbol{r}_{k+1}|I]] + \operatorname{cov}[\mathbb{E}[\boldsymbol{r}_{k+1}|I]] \\ &= h(\Sigma_{k+1}^2 + h\operatorname{cov}[n\nabla f_I(\boldsymbol{\theta}_k)]) \end{aligned} \tag{A.1}$$

where $\Sigma_{k+1}^2 = \frac{1}{h}\mathbb{E}[\operatorname{cov}[\boldsymbol{r}_{k+1}|I]]$. The extra randomness due to the randomness of the index I enters the parameter space through the coupling of $\boldsymbol{\theta}$ and $\boldsymbol{r}$ and eventually deviates the stationary distribution from that of the original dynamics. Adopting the perspective of modified equation [127, 128, 129], we model this as an enlarged diffusion coefficient. To correct for this enlargement and still sample from the correct distribution, we can either, in each step, shrink the size of intrinsic noise to $\Sigma_k \in \mathbb{R}^{d\times d}$ such that $\sigma^2 I = \Sigma_k^2 + h\text{cov}[n\nabla f_I(\boldsymbol{\theta}_{k-1})]$, or alternatively increase the dissipation. More precisely, due to the matrix version fluctuation dissipation theorem $\Sigma^2 = 2\Gamma T$, one could instead increase the friction coefficient $\Gamma \in \mathbb{R}^{d\times d}$ rather than shrinking the intrinsic noise. The second approach is computationally more efficient because it no longer requires square-rooting / Cholesky decomposition of (possibly large-scale) matrices. Therefore, in each step, we set

$$\Gamma_k = \frac{1}{2T}(\sigma^2 I + h\text{cov}[n\nabla f_I(\boldsymbol{\theta}_{k-1})]).$$

Accurately computing $\operatorname{cov}[n\nabla f_I(\boldsymbol{\theta}_{k-1})]$ is expensive as it requires running I through $1, \cdots, n$, which defeats the purpose of introducing a stochastic gradient. To downscale the computation cost from $\mathcal{O}(n)$ to $\mathcal{O}(1)$, we use an SVRG type estimation of the this variance instead. More specifically, we periodically compute $\operatorname{cov}[n\nabla f_I(\boldsymbol{\theta}_{k-1})]$ only every L data passes, in an outer loop. In every iteration of an inner loop, which integrates the Langevin, an estimate of $\operatorname{cov}[n\nabla f_I(\boldsymbol{\theta}_{k-1})]$ is updated in an SVRG fashion. sof See Algorithm 3 for detailed description. We refer variance reduced variant of EWSG as EWSG-VR.

To demonstrate the performance of EWSG-VR, we reuse the setup of simple Gaus-

Algorithm 3 EWSG-VR

Input: {number of data terms n, gradient functions $\nabla f_i(\cdot)$, step size h, number of data passes K, period of variance calibration L, index chain length M, friction and noise coefficients γ and σ}
initialize $\boldsymbol{\theta}_0, \boldsymbol{r}_0, \gamma_0 = \gamma$
initialize inner loop index $k = 0$
for $l = 1, 2, \cdots, K$ **do**
  if $(l-1) \mod L = 0$ **then**
    compute $\boldsymbol{m}_1 \leftarrow \mathbb{E}_I[n\nabla f_I(\boldsymbol{\theta}_k)]$, $\boldsymbol{m}_2 \leftarrow \mathbb{E}_I[n^2 \nabla f_I(\boldsymbol{\theta}_k)\nabla f_I(\boldsymbol{\theta}_k)^T]$
    $\boldsymbol{\omega} \leftarrow \boldsymbol{\theta}_k$
  else
    for $t = 1, 2, \cdots, \lceil \frac{n}{M+1} \rceil$ **do**
      $i \leftarrow$ uniformly sampled from $1, \cdots, n$, compute and store $n\nabla f_i(\boldsymbol{\theta}_k)$
      for $m = 1, 2, \cdots, M$ **do**
        $j \leftarrow$ uniformly sampled from $1, \cdots, n$, compute and store $n\nabla f_j(\boldsymbol{\theta}_k)$
        $i \leftarrow j$ with probability in Equation 2.11
      end for
      update $(\boldsymbol{\theta}_{k+1}, \boldsymbol{r}_{k+1}) \leftarrow (\boldsymbol{\theta}_k, \boldsymbol{r}_k)$ according to Equation 2.3, using $n\nabla f_i(\boldsymbol{\theta}_k)$ as gradient and Γ_k as friction
      $\boldsymbol{m}_1 \leftarrow \boldsymbol{m}_1 + \nabla f_i(\boldsymbol{\theta}_k) - \nabla f_i(\boldsymbol{\omega})$
      $\boldsymbol{m}_2 \leftarrow \boldsymbol{m}_2 + n\nabla f_i(\boldsymbol{\theta}_k)\nabla f_i(\boldsymbol{\theta}_k)^T - n\nabla f_i(\boldsymbol{\omega})\nabla f_i(\boldsymbol{\omega})^T$
      covar $\leftarrow \boldsymbol{m}_2 - \boldsymbol{m}_1\boldsymbol{m}_1^T$
      $\Gamma_{k+1} \leftarrow \frac{1}{2T}(\sigma^2 \mathrm{I} + h\,\text{covar})$
      $k \leftarrow k + 1$
    end for
  end if
end for

sian example in subsection 2.7.1. As shown in Algorithm 3, the only hyper-parameter of EWSG-VR additional to EWSG is the period of variance calibration, for which we set $L = 1$. All other hyper-parameters (e.g. step size h, friction coefficient γ) are set the same as EWSG. We also run underdamped Langevin dynamics with full gradient (FG) using the same hyper-parameters of EWSG. We plot the KL divergence in Figure A.1. We see that EWSG-VR further reduces variance and achieves better statistical accuracy measured in KL divergence. Although EWSG-VR periodically use full data set to calibrate variance estimation, it is still significantly faster than the full gradient version. Note that KL divergence of SGLD, pSGLD and SGHMC are too large so that we can not even see them in Figure A.1

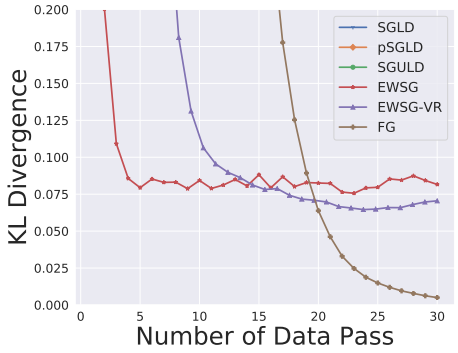
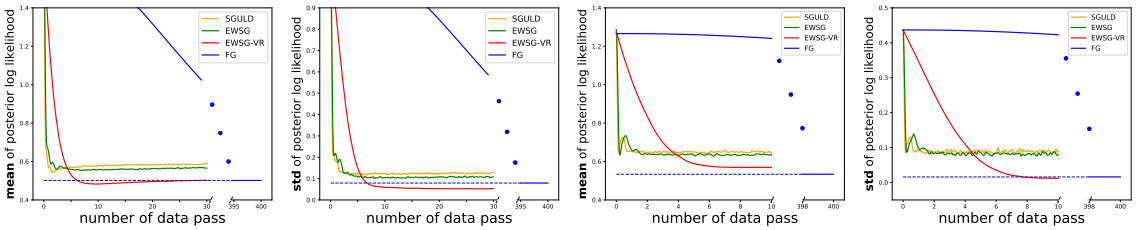


Figure A.1: KL divergence

We also consider applying EWSG-VR to Bayesian logistic regression problems. We run experiments on two standard classification data sets `parkinsons` [1], `pima`[2] from UCI repository [130].



(a) parkinsons (b) pima

Figure A.2: Posterior prediction of mean (*left*) and standard deviation (*right*) of log likelihood on test data set generated by SGHMC, EWSG and EWSG-VR on two Bayesian logistic regression tasks. Statistics are computed based on 1000 independent simulations. Minibatch size $b = 1$ for all methods except FG. $M = 1$ for EWSG and EWSG-VR.

From Figure A.2, we see stochastic gradient methods (SGHMC, EWSG and EWSG-VR) only take tens of data passes to converge while full gradient version (FG) requires hundreds of data passes to converge. Compared with SGHMC, EWSG produces closer results to FG for which we treat as ground truth, in terms of statistical accuracy. With variance reduction, EWSG-VR is able to achieve even better performance, significantly improving the accuracy of the prediction of mean and standard deviation of log likelihood. It, however, converges slower than EWSG without VR.

One downside of EWSG-VR is that it periodically use whole data set to calibrate vari-

[1] https://archive.ics.uci.edu/ml/datasets/parkinsons
[2] https://archive.ics.uci.edu/ml/datasets/diabetes

ance estimation, so it may not be suitable for very large data sets (e.g. Covertype data set used in subsection 2.7.2) for which stochastic gradient methods could converge within one data pass.

A.4 Additional Experiments

A.4.1 A Misspecified Gaussian Case

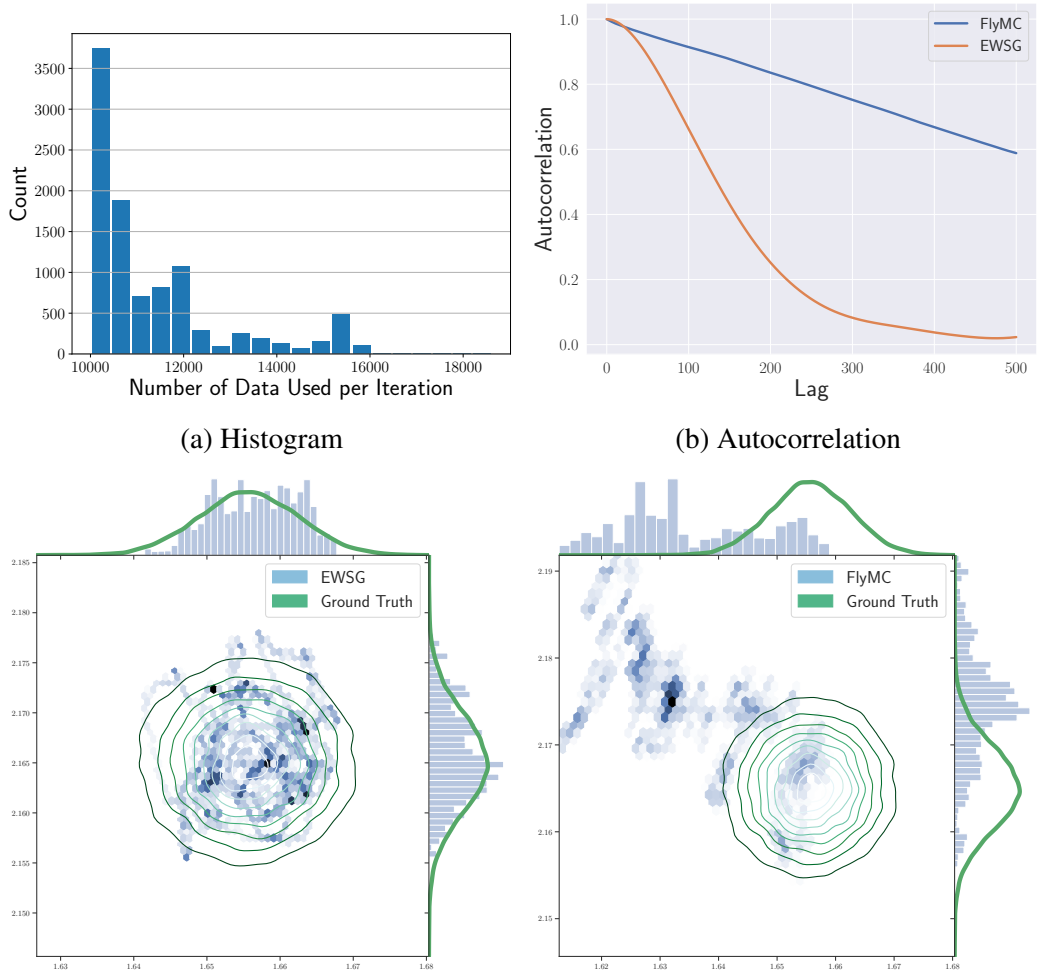
In this subsection, we follow the same setup as in [58] and study a misspecified Gaussian model where one fits a one-dimensional normal distribution $p(\theta) = \mathcal{N}(\theta|\mu_0, \sigma_0^2)$ to 10^5 i.i.d points drawn according to $X_i \sim \log \mathcal{N}(0, 1)$, and flat prior is assigned $p(\mu_0, \log \sigma_0) \propto 1$. It was shown in [58] that FlyMC algorithm behaves erratically in this case, as "bright" data points with large values are rarely updated and they drive samples away from the target distribution. Consequently the chain mixes very slowly. One important commonality FlyMC shares with EWSG is that in each iteration, both algorithms select a subset of data in a non-uniform fashion. Therefore, it is interesting to investigate the performance of EWSG in this misspecified model.

For FlyMC[3], a tight lower bound based on Taylor's expansion is used to minimize "bright" data points used per iteration. At each iteration, 10% data points are resampled and turned "on/off" accordingly and the step size is adaptively adjusted. FlyMC algorithm is run for 10000 iterations. Figure A.3a shows the histogram of number of data points used in each iteration for FlyMC algorithm. On average, FlyMC consumes 10.9% of all data points per iteration. For fair comparison, the minibatch size of EWSG is hence set $10^5 \times 10.9\% = 10900$ and we run EWSG for 1090 data passes. We set step size $h = 1 \times 10^{-4}$ and friction coefficient $\gamma = 300$ for EWSG. An isotropic random walk Metropolis Hastings (MH) is also run for sufficiently long and serves as the ground truth.

Figure A.3b shows the autocorrelation of three algorithms. The autocorrelation of FlyMC decays very slowly, samples that are even 500 iterations away still show strong

[3] https://github.com/rbardenet/2017JMLR-MCMCForTallData

correlation. The autocorrelation of EWSG, on the other hand, decays much faster, suggesting EWSG explores parameter space efficiently than FlyMC does. Figure A.3c and A.3d show the samples (the first 1000 samples are discarded as burn-in) generated by EWSG and FlyMC respectively. The samples of EWSG center around the mode of the target distribution while the samples of FlyMC are still far away from the true posterior. The experiment shows EWGS works quite well even in misspecified models, and hence is an effective candidate in combining importance sampling with scalable Bayesian inference.



(a) Histogram

(b) Autocorrelation

(c) Samples of EWSG

(d) Samples of FlyMC

Figure A.3: (a) Histogram of data used in each iteration for FlyMC algorithm. (b) Autocorrelation plot of FlyMC, EWSG and MH. (c) Samples of EWSG. (d) Samples of FlyMC.

A.4.2 Additional Results of BNN Experiment

We report the test error of various SG-MCMC methods after 200 epochs in Table A.1. For both MLP and CNN architecture, EWSG outperforms its uniform counterpart SGHMC as well as other benchmarks SGLD, pSGLD and CP-SGHMC. The results clearly demonstrate the effectiveness of the proposed EWSG on deep models.

Table A.1: Test error (mean $\pm$ standard deviation) after 200 epoches.

Method	Test Error(%), MLP	Test Error(%), CNN
SGLD	1.976 ± 0.055	0.848 ± 0.060
pSGLD	1.821 ± 0.061	0.860 ± 0.052
SGHMC	1.833 ± 0.073	0.778 ± 0.040
CP-SGHMC	1.835 ± 0.047	0.772 ± 0.055
EWSG	$\mathbf{1.793 \pm 0.100}$	$\mathbf{0.753 \pm 0.035}$

A.4.3 Additional Experiment on BNN: Tuning M

In each iteration of EWSG, we run an index Markov chain of length M and select a "good" minibatch to estimate gradient, therefore EWSG essentially uses $b \times (M+1)$ data points per iteration where b is minibatch size. How does EWSG compare with its uniform gradient subsampling counterpart with a larger minibatch size ($b \times (M+1)$)?

We empirically answer this question in the context of BNN with MLP architecture. We use the same step size for SGHMC and EWSG and experiment a large range of values of minibatch size b and index chain length M. Each algorithm is run for 200 data passes and 10 independent samples are drawn to estimate test error. The results are shown in Table A.2. We find that EWSG beats SGHMC with larger minibatch in 8 out of 9 comparison groups, which suggests in general EWSG could be a better way to consuming data compared to increasing minibatch size and may shed light on other areas where stochastic gradient methods are used (e.g. optimization).

Table A.2: Test errors of EWSG (top of each cell) and SGHMC (bottom of each cell) after 200 epoches. b is minibatch size for EWSG, and minibatch size of SGHMC is set as $b \times (M+1)$ to ensure the same number of data used per parameter update for both algorithms. Step size is set $h = \frac{10}{b(M+1)}$ as suggested in [18], different from that used to produce Table A.1. Results with smaller test error is highlighted in boldface.

b	$M+1=2$	$M+1=5$	$M+1=10$
100	**1.86%** 1.94%	**1.83%** 1.92%	**1.80%** 1.97%
200	1.90% **1.87%**	**1.87%** 1.97%	**1.80%** 2.07%
500	**1.79%** 1.97%	**2.01%** 2.17%	**2.36%** 2.37%

A.5 EWSG does not necessarily change the speed of convergence significantly

Changing the weights of stochastic gradient from uniform to non-uniform, as we saw, can increase the statistical accuracy of the sampling; however, it does not necessarily increase or decrease the speed of convergence to the (altered) limiting distribution. Numerical examples already demonstrated this fact, but on the theoretical side, we note the non-asymptotic bound provided by Theorem 3 may not be tight in terms of the speed of convergence due to its generality. Therefore, here we quantify the convergence speed on a simple quadratic example:

Consider $V_i(\theta) = \frac{1}{n}(\theta - \mu_i)^2/2$ where μ_i's are constant scalars. Assume without loss of generality that $\sum_i \mu_i = 0$, and thus $V(\theta) = \sum_{i=1}^n V_i(\theta) = \theta^2/2 +$ some constant. We will show the convergence speed of $\mathbb{E}\theta$ is comparable for uniform and a class of non-uniform SG-MCMC (including EWSG) applied to second-order Langevin equation (overdamped Langevin will be easier and thus omitted):

Theorem 26 *Consider, for $0 < \gamma < 2$, respectively SGHMC and EWSG,*

$$\begin{cases} \theta'_{k+1} &= \theta'_k + hr'_k \\ r'_{k+1} &= r'_k - h\gamma r'_k - h(\theta'_k - \mu_{I'_k}) + \sqrt{h}\sigma\xi'_{k+1} \end{cases}$$

and

$$\begin{cases} \theta_{k+1} &= \theta_k + hr_k \\ r_{k+1} &= r_k - h\gamma r_k - h(\theta_k - \mu_{I_k}) + \sqrt{h}\sigma\xi_{k+1} \end{cases},$$

where I'_k are i.i.d. uniform random variable on $[n]$, I_k are $[\theta, r]$ dependent random variable on $[n]$ satisfying $\mathbb{P}(I_k = i) = 1/n + \mathcal{O}(h^p)$, and ξ_{k+1}, ξ'_{k+1} are standard i.i.d. Gaussian random variables. Denote by $\bar{\theta'}_k = \mathbb{E}\theta'_k$, $\bar{r'}_k = \mathbb{E}r'_k$, $\bar{\theta}_k = \mathbb{E}\theta_k$, $\bar{r}_k = \mathbb{E}r_k$, $x'_k = [\bar{\theta'}_k, \bar{r'}_k]^T$, and $x_k = [\bar{\theta}_k, \bar{r}_k]^T$, then

$$x'_k = (I + Ah)^k x'_0, \quad \textit{where } A = \begin{bmatrix} 0 & 1 \\ -1 & -\gamma \end{bmatrix}, \tag{A.2}$$

for small enough h, $\|x'_k\|$ converges to 0 exponentially with $k \to \infty$, and x_k converges at a comparable speed in the sense that $\|x_k - x'_k\| = \mathcal{O}(h^p)$ if $x_0 = x'_0$.

Proof: Taking the expectation of the $[\theta', r']$ iteration and using the fact that $\sum_i \mu_i = 0$ and hence $\mathbb{E}\mu_{I'_k} = 0$, one easily obtains (A.2). The geometric convergence of x'_k thus follows from the fact that eigenvalues of $I + Ah$ have less than 1 modulus for small enough h.

Let $e_k = [0, \mathbb{E}\mu_{I_k}]^T$ and then

$$e_k = [0, \sum_{i=1}^{n} \mathbb{P}(I_k = i)\mu_i]^T = [0, \mathcal{O}(h^p)]^T$$

Now we take the expectation of both sides of the $[\theta, r]$ iteration and obtain $x_{k+1} = (I + Ah)x_k + he_k$. Therefore

$$\begin{aligned} x_k =& (I + Ah)^k x_0 + (I + Ah)^{k-1} he_0 + \cdots + (I + Ah)he_{k-2} + he_{k-1} \\ =& x'_k + h\big((I + Ah)^{k-1} e_0 + \cdots + (I + Ah)e_{k-2} + e_{k-1}\big) \end{aligned}$$

To bound the difference, note $I + Ah$ is diagonalizable with complex eigenvalues $\lambda_{1,2}$

satisfying

$$|\lambda_1| = |\lambda_2| = \sqrt{1 - h\gamma + h^2} = 1 - \gamma h/2 + \mathcal{O}(h^2).$$

Projecting e_j to the corresponding eigenspaces via $e_j = v_{1,j} + v_{2,j}$, we can get

$$\begin{aligned}
&h\|(I + Ah)^{k-1}e_0 + \cdots + e_{k-1}\| \\
\leq &h\left(\|(I + Ah)^{k-1}e_0\| + \cdots + \|e_{k-1}\|\right) \\
= &h\left(|\lambda_1|^{k-1}\|v_{1,0}\| + |\lambda_2|^{k-1}\|v_{2,0}\| + \cdots + \|v_{1,k-1}\| + \|v_{2,k-1}\|\right) \\
\leq &hCh^p(|\lambda_1|^{k-1} + \cdots + 1) \\
= &hCh^p\frac{1 - |\lambda_1|^k}{1 - |\lambda_1|} \\
\leq &hCh^p\frac{1}{1 - |\lambda_1|} \\
\leq &\hat{C}h^p
\end{aligned}$$

for some constant C and $\hat{C}$. ■

Important to note is, although this is already a nonlinear example for EWSG (as nonlinearity enters through the μ_{I_k} term), it is a linear example for SGHMC. For the fully nonlinear cases, a tight quantification of EWSG's convergence speed remains to be an open theoretical challenge (a loose quantification is already given by the general Theorem 3).

APPENDIX B

SUPPLEMENTARY MATERIALS OF CHAPTER 3

B.1 Poincaré's Inequalities for Product Measure

Lemma 27 *Suppose $\mathcal{X}_1 = \mathcal{X}_2 = \mathbb{R}^d$, and measures $\mu_1 \in \mathcal{P}(\mathcal{X}_1), \mu_2 \in \mathcal{P}(\mathcal{X}_2)$ satisfy Poincaré's inequality with constant $\lambda_{PI}(\mu_1), \lambda_{PI}(\mu_2)$. Then the product measure $\mu = \mu_1 \otimes \mu_2 \in \mathcal{P}(\mathcal{X}_1 \times \mathcal{X}_2)$ satisfies Poincaré's inequality with constant $\lambda_{PI}(\mu) = \min\{\lambda_{PI}(\mu_1), \lambda_{PI}(\mu_2)\}$.*

Proof: For any smooth function $f(x_1, x_2)$, denote $g(x_1) = \int f d\mu_2$ and it follows that $\int g d\mu_1 = \int f d\mu$. We have

$$
\begin{aligned}
&\int (f - \int f d\mu)^2 d\mu \\
=&\int (f - g + g - \int f\mu)^2 d\mu \\
=&\int (f-g)^2 d\mu + 2\int (f-g)(g - \int f d\mu) d\mu + \int (g - \int f d\mu)^2 d\mu \\
=&\int (\int (f-g)^2 d\mu_2) d\mu_1 + \int (g - \int f d\mu)^2 d\mu_1 \\
\leq&\frac{1}{\lambda_{\text{PI}}(\mu_2)} \int (\int \|\nabla_{x_2} f\|^2 d\mu_2) d\mu_1 + \frac{1}{\lambda_{\text{PI}}(\mu_1)} \int \|\nabla_{x_1} g\|^2 d\mu_1 \\
=&\frac{1}{\lambda_{\text{PI}}(\mu_2)} \int \|\nabla_{x_2} f\|^2 d\mu + \frac{1}{\lambda_{\text{PI}}(\mu_1)} \int \| \int \nabla_{x_1} f d\mu_2 \|^2 d\mu_1 \\
\overset{(i)}{\leq}&\frac{1}{\lambda_{\text{PI}}(\mu_2)} \int \|\nabla_{x_2} f\|^2 d\mu + \frac{1}{\lambda_{\text{PI}}(\mu_1)} (\int (\int \|\nabla_{x_1} f\|^2 d\mu_1)^{\frac{1}{2}} d\mu_2)^2 \\
\overset{(ii)}{\leq}&\frac{1}{\lambda_{\text{PI}}(\mu_2)} \int \|\nabla_{x_2} f\|^2 d\mu + \frac{1}{\lambda_{\text{PI}}(\mu_1)} \int \|\nabla_{x_1} f\|^2 d\mu \\
\leq&\frac{1}{\min\{\lambda_{\text{PI}}(\mu_1), \lambda_{\text{PI}}(\mu_2)\}} \int \|\nabla f\|^2 d\mu
\end{aligned}
$$

where (i) is due to Minkowski's inequality and (ii) is due to Holder's inequality. ■

B.2 Tempered HFHR with Unit PI Constant

Lemma 28 *Under Assumption 1, 2 and suppose* $\gamma^2 \geq \max\{2, L\}$ *and* $\alpha \leq \gamma - \frac{2}{\gamma}$. *Then the tempered HFHR*(α, γ, β) *in Equation* (3.14) *converges to* $\pi \propto e^{-\beta H(\boldsymbol{q},\boldsymbol{p})}$ *where* $H(\boldsymbol{q}, \boldsymbol{p}) = f(\boldsymbol{q}) + \frac{1}{2}\|\boldsymbol{p}\|^2$. *Moreover, if the joint invariant distribution* π *satisfies PI with PI constant* $\lambda_{PI}(\pi) = \beta$, *we have the following exponential convergence*

$$\chi^2(\rho_t\|\pi) \leq e^{-(\frac{1}{2\gamma}+\frac{1}{16}\alpha)t}\left\{\chi^2(\rho_0\|\pi) + \mathbb{E}_\pi\left[\langle\nabla_{\boldsymbol{x}}\frac{\rho_0}{\pi}, S\nabla_{\boldsymbol{x}}\frac{\rho_0}{\pi}\rangle\right]\right\},$$

where $\chi^2(\mu,\nu) = \int(\frac{d\mu}{d\nu}-1)^2 d\nu$ *and* ρ_t *is the joint law of* $(\boldsymbol{q}_t, \boldsymbol{p}_t)$ *of tempered HFHR*(α, γ, β) *at time* t, $\nabla_{\boldsymbol{x}} = (\nabla_{\boldsymbol{q}}, \nabla_{\boldsymbol{p}})$ *and* $S \in \mathbb{R}^{2d\times 2d}$ *is a symmetric matrix, more specifically,*

$$S = \begin{bmatrix} aI & bI \\ bI & dI \end{bmatrix} \text{ with } a = (\frac{2}{\gamma}+\alpha)b, d = \gamma b, b = \frac{1}{\gamma\beta}.$$

Proof: Denote the eigenvalues of $\nabla^2 f$ by $\eta_i, i = 1, 2, \cdots, d$. By convexity assumption on f and L-smoothness assumption on ∇f, we have $0 \leq \eta_i \leq L, i = 1, 2, \cdots, d$.

By assumption, $\kappa \propto e^{-f(\boldsymbol{q})}$ satisfies PI with PI constant 1, it is easy to see $\mu \propto e^{-\beta f(\boldsymbol{q})}$ satisfies PI with PI constant β. It is well known that standard Gaussian measure satisfies PI with constant 1, so $\nu \propto e^{-\frac{\beta}{2}\|\boldsymbol{p}\|^2}$ satisfies PI with PI constant β. By Lemma 27, we know π satisfies PI with constant $\lambda_{\text{PI}}(\pi) = \min\{\lambda_{\text{PI}}(\mu), \lambda_{\text{PI}}(\nu)\} = \beta$.

Consider the following Lyapunov function, χ^2 divergence augmented by the cross term

$$\mathcal{L}(\rho_t) = \chi^2(\rho_t\|\pi) + \mathcal{L}_{\text{cross}}(\rho_t)$$

where $\mathcal{L}_{\text{cross}}(\rho_t)$ is defined in Equation (3.13) with

$$S = \begin{bmatrix} aI & bI \\ bI & dI \end{bmatrix} \text{ with } a = (\frac{2}{\gamma}+\alpha)b, d = \gamma b, b = \frac{1}{\gamma\beta} \tag{B.1}$$

By direct computation and Lemma 29, we have

$$\frac{d}{dt}\mathcal{L}(\rho_t) \leq -\mathbb{E}_\pi\left[\left\langle \nabla_{\boldsymbol{x}}\frac{\rho_t}{\pi}, (\beta^{-1}D + M_{\text{cross}})\nabla_{\boldsymbol{x}}\frac{\rho_t}{\pi}\right\rangle\right] \tag{B.2}$$

where $D \triangleq \begin{bmatrix} 2\alpha I & 0 \\ 0 & 2\gamma I \end{bmatrix}$ and M_{cross} is the matrix from Equation (B.6) in Lemma 29.

Denote $A \triangleq \beta^{-1}D + M_{\text{cross}}$, then we have

$$\begin{aligned} A =& \begin{bmatrix} 2\frac{\alpha}{\beta}I & 0 \\ 0 & 2\frac{\gamma}{\beta}I \end{bmatrix} + \begin{bmatrix} 2bI + 2a\alpha\nabla^2 f(\boldsymbol{q}) & (b\alpha - a)\nabla^2 f(\boldsymbol{q}) + b\gamma I + dI \\ -(a - b\alpha)\nabla^2 f(\boldsymbol{q}) + b\gamma I + dI & -2b\nabla^2 f(\boldsymbol{q}) + 2d\gamma I + 2\gamma I \end{bmatrix} \\ =& 2\beta^{-1}\begin{bmatrix} \alpha I & 0 \\ 0 & \gamma I \end{bmatrix} + 2a\alpha\begin{bmatrix} \nabla^2 f(\boldsymbol{q}) & 0 \\ 0 & 0 \end{bmatrix} + \frac{2}{\gamma\beta}\begin{bmatrix} I & \gamma I - \frac{1}{\gamma}\nabla^2 f(\boldsymbol{q}) \\ \gamma I - \frac{1}{\gamma}\nabla^2 f(\boldsymbol{q}) & \gamma^2 I - \nabla^2 f(\boldsymbol{q}) \end{bmatrix} \end{aligned}$$

Denote

$$E = \frac{1}{\lambda_{\text{PI}}(\pi)}I + S = \frac{1}{\beta}I + S = \frac{1}{\beta}I + \frac{1}{\gamma\beta}\begin{bmatrix} (\frac{2}{\gamma} + \alpha)I & I \\ I & \gamma I \end{bmatrix}$$

The rest of the proof is dedicated to matrix analysis of A and E. Denote $P \succeq Q$ if $P - Q$ is a positive semi-definite matrix and we will frequently use the following property of block matrix

$$\det\left(\begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}\right) = \det\left(A_{11}A_{22} - A_{12}A_{21}\right) \text{ if } A_{21}A_{22} = A_{22}A_{21}. \tag{B.3}$$

First since $\alpha \leq \gamma - \frac{2}{\gamma}$, we have

$$E = \frac{1}{\beta}I + \frac{1}{\gamma\beta}\begin{bmatrix} (\alpha + \frac{2}{\gamma})I & I \\ I & \gamma I \end{bmatrix} \preceq \frac{1}{\beta}I + \frac{1}{\gamma\beta}\begin{bmatrix} \gamma I & I \\ I & \gamma I \end{bmatrix} = \frac{1}{\beta}\begin{bmatrix} 2I & \frac{1}{\gamma}I \\ \frac{1}{\gamma}I & 2I \end{bmatrix} \tag{B.4}$$

Since $2a\alpha \begin{bmatrix} \nabla^2 f(\boldsymbol{q}) & 0 \\ 0 & 0 \end{bmatrix} \succeq 0$, we have

$$A \succeq 2\beta^{-1} \begin{bmatrix} \alpha I & 0 \\ 0 & \gamma I \end{bmatrix} + \frac{2}{\gamma\beta} \begin{bmatrix} I & \gamma I - \frac{1}{\gamma}\nabla^2 f(\boldsymbol{q}) \\ \gamma I - \frac{1}{\gamma}\nabla^2 f(\boldsymbol{q}) & \gamma^2 I - \nabla^2 f(\boldsymbol{q}) \end{bmatrix} \tag{B.5}$$

Now we consider the following difference

$$\begin{aligned} & A - (\frac{1}{2\gamma} + \frac{1}{16}\alpha)E \\ \succeq & \underbrace{\frac{2}{\gamma\beta} \begin{bmatrix} I & \gamma I - \frac{1}{\gamma}\nabla^2 f(\boldsymbol{q}) \\ \gamma I - \frac{1}{\gamma}\nabla^2 f(\boldsymbol{q}) & 2\gamma^2 I - \nabla^2 f(\boldsymbol{q}) \end{bmatrix} - \frac{1}{\gamma\beta} \begin{bmatrix} I & \frac{1}{2\gamma} I \\ \frac{1}{2\gamma} I & I \end{bmatrix}}_{B} \\ & + 2\beta^{-1} \begin{bmatrix} \alpha I & 0 \\ 0 & 0 \end{bmatrix} - \frac{\frac{1}{16}\alpha}{2\beta} \begin{bmatrix} I & \frac{1}{2\gamma} I \\ \frac{1}{2\gamma} I & I \end{bmatrix} \end{aligned}$$

where the matrix inequality is due to assumption $\gamma^2 \geq L$, Equation (B.4) and (B.5).

For B, we have

$$B = \frac{1}{\gamma\beta} \begin{bmatrix} I & 2\gamma I - \frac{2}{\gamma}\nabla^2 f(\boldsymbol{q}) - \frac{1}{2\gamma} I \\ 2\gamma I - \frac{2}{\gamma}\nabla^2 f(\boldsymbol{q}) - \frac{1}{2\gamma} I & 4\gamma^2 I - 2\nabla^2 f(\boldsymbol{q}) - I \end{bmatrix}$$

Using Equation (B.3) and diagonalization of $\nabla^2 f(\boldsymbol{q})$, we know the eigenvalues of B are the collections of eigenvalues of the following 2×2 matrices

$$B_i = \frac{1}{\gamma\beta} \begin{bmatrix} 1 & 2\gamma I - \frac{2\eta_i}{\gamma} - \frac{1}{2\gamma} \\ 2\gamma - \frac{2\eta_i}{\gamma} - \frac{1}{2\gamma} I & 4\gamma^2 - 2\eta_i - 1 \end{bmatrix}$$

Notice

$$\det(B_i) = \frac{1}{\gamma^2\beta^2}(1 - \frac{1}{4\gamma^2} + \frac{6\eta_i\gamma^2 - 2\eta_i - 4\eta_i^2}{\gamma^2}) \geq \frac{1}{\gamma^2\beta^2}(1 - \frac{1}{4\gamma^2}) > 0$$

hence each B_i is positive definite and the smaller eigenvalue of each B_i is

$$\begin{aligned}
\lambda^-(B_i) =& \frac{1}{2\gamma\beta}\left[4\gamma^2 - 2\eta_i - \sqrt{1 - \frac{1}{4\gamma^2} + \frac{6\eta_i\gamma^2 - 2\eta_i - 4\eta_i^2}{\gamma^2}}\right] \\
=& \frac{1}{2\gamma\beta}\left[4\gamma^2 - 2\eta_i - \sqrt{(1 - \frac{1}{4\gamma^2}) + \frac{(2\eta_i\gamma^2 - 2\eta_i) + (4\eta_i\gamma^2 - 4\eta_i^2)}{\gamma^2}}\right] \\
\overset{(i)}{\geq}& \frac{1}{2\gamma\beta}\left[2\gamma^2 - \sqrt{1 + 2\gamma^2 - 2 + 4L}\right] \\
\overset{(ii)}{\geq}& \frac{1}{2\gamma\beta}(2\gamma^2 - \sqrt{6}\gamma) \\
\geq& \frac{\gamma}{8\beta}
\end{aligned}$$

where $(i), (ii)$ follow by $0 \leq \eta_i \leq L$ and the assumption $\gamma^2 \geq \max\{2, L\}$.

Therefore, the smallest eigenvalue of B is also lower bounded

$$\lambda^{\min}(B) = \min_{i=1,2,\cdots,d} \lambda^-(B_i) \geq \frac{\gamma}{8\beta}.$$

and this is equivalent to $B \succeq \frac{8\gamma}{\beta} I$.

We now obtain

$$\begin{aligned}
A - (\frac{1}{2\gamma} + \frac{1}{16}\alpha)E \succeq& 2\beta^{-1}\begin{bmatrix} \alpha I & 0 \\ 0 & 0 \end{bmatrix} + \frac{\gamma}{8\beta} I - \frac{\frac{1}{16}\alpha}{2\beta}\begin{bmatrix} I & \frac{1}{2\gamma} I \\ \frac{1}{2\gamma} I & I \end{bmatrix} \\
=& \underbrace{\frac{1}{8\beta}\begin{bmatrix} (4\alpha - \frac{1}{4}\alpha + \gamma) I & -\frac{\alpha}{8\gamma} I \\ -\frac{\alpha}{8\gamma} I & (\gamma - \frac{\alpha}{4}) I \end{bmatrix}}_{F}
\end{aligned}$$

By Equation (B.3), it is easy to see that the eigenvalues of F are identical to the eigenvalues (ignoring multiplicity) of the following 2×2 matrix $\tilde{F} = \frac{1}{8\beta}\begin{bmatrix} 4\alpha - \frac{1}{4}\alpha + \gamma & -\frac{\alpha}{8\gamma} \\ -\frac{\alpha}{8\gamma} & \gamma - \frac{\alpha}{4} \end{bmatrix}$ and we have

$$\det(\tilde{F}) = \frac{1}{(8\beta)^2}\left(-\frac{15\alpha^2}{16} - \frac{\alpha^2}{64\gamma^2} + \frac{7\alpha\gamma}{2} + \gamma^2\right) \overset{(i)}{\geq} \frac{1}{(8\beta)^2}\left(\frac{1}{16}\gamma^2 - \frac{1}{64} + \frac{7\alpha\gamma}{2}\right) \overset{(ii)}{>} 0$$

where (i) and (ii) follow by the assumption $\alpha \leq \gamma - \frac{2}{\gamma} < \gamma$ and $\gamma^2 > 2$. Therefore $\tilde{F}$ is a positive definite matrix and all of its eigenvalues are positive, and hence the eigenvalues of F are also positive, equivalently $F \succeq 0$.

We now have established the relation $A - (\frac{1}{2\gamma} + \frac{1}{16}\alpha)E \succeq 0$ and return to the time derivative in Equation (B.2)

$$\begin{aligned}
\frac{d}{dt}\mathcal{L}(\rho_t) \leq & -\mathbb{E}_\pi\left[\left\langle \nabla_{\boldsymbol{x}}\frac{\rho_t}{\pi}, A\nabla_{\boldsymbol{x}}\frac{\rho_t}{\pi}\right\rangle\right] \\
\leq & -(\frac{1}{2\gamma} + \frac{1}{16}\alpha)\mathbb{E}_\pi\left[\left\langle \nabla_{\boldsymbol{x}}\frac{\rho_t}{\pi}, E\nabla_{\boldsymbol{x}}\frac{\rho_t}{\pi}\right\rangle\right] \\
= & -(\frac{1}{2\gamma} + \frac{1}{16}\alpha)\left[\frac{1}{\lambda_{\text{PI}}(\pi)}\mathbb{E}_\pi\left[\|\nabla_{\boldsymbol{x}}\frac{\rho_t}{\pi}\|^2\right] + \mathcal{L}_{\text{cross}}(\rho_t)\right] \\
\overset{(i)}{\leq} & -(\frac{1}{2\gamma} + \frac{1}{16}\alpha)\left[\mathbb{E}_\pi[(\frac{\rho_t}{\pi} - 1)^2] + \mathcal{L}_{\text{cross}}(\rho_t)\right] \\
= & -(\frac{1}{2\gamma} + \frac{1}{16}\alpha)\mathcal{L}(\rho_t)
\end{aligned}$$

where (i) is due to Poincaré's inequality.

By Gronwall's inequality, $\mathcal{L}$ has exponential decay $\mathcal{L}(\rho_t) \leq e^{-(\frac{1}{2\gamma} + \frac{1}{16}\alpha)t}\mathcal{L}(\rho_0)$ and since S is positive definite, we further have

$$\chi^2(\rho_t\|\pi) \leq \mathcal{L}(\rho_t) \leq e^{-(\frac{1}{2\gamma} + \frac{1}{16}\alpha)t}\mathcal{L}(\rho_0),$$

■

B.3 Time Derivative of M_{cross}

Lemma 29

$$\frac{d}{dt}\mathcal{L}_{cross}(\rho_t) \leq -\beta^{-1}\int \langle \nabla_{\boldsymbol{x}}\frac{\rho_t}{\pi}, M_{cross}\nabla_{\boldsymbol{x}}\frac{\rho_t}{\pi}\rangle d\pi$$

where ρ_t is the law of $(\boldsymbol{q}_t, \boldsymbol{p}_t)$ of tempered-HFHR(α, γ, β) and

$$M_{cross} = \begin{bmatrix} 2a\alpha\nabla^2 f(\boldsymbol{q}) + 2bI & -a\nabla^2 f(\boldsymbol{q}) + b\alpha\nabla^2 f(\boldsymbol{q}) + b\gamma I + dI \\ -a\nabla^2 f(\boldsymbol{q}) + b\alpha\nabla^2 f(\boldsymbol{q}) + b\gamma I + dI & -2b\nabla^2 f(\boldsymbol{q}) + 2d\gamma I \end{bmatrix} \tag{B.6}$$

Proof: For better readability, we collect some notations used in the proof here

$$\mu(\boldsymbol{q}) \propto e^{-\beta f(\boldsymbol{q})}, \quad \nu(\boldsymbol{p}) \propto e^{-\frac{\beta}{2}\|\boldsymbol{p}\|^2}, \quad \pi(\boldsymbol{q},\boldsymbol{p}) \propto e^{-\beta H(\boldsymbol{q},\boldsymbol{p})}$$

where $H(\boldsymbol{q},\boldsymbol{p}) = f(\boldsymbol{q}) + \frac{1}{2}\|\boldsymbol{p}\|^2$ and write tempered HFHR again for reference

$$\begin{cases} d\boldsymbol{q} = \big(\boldsymbol{p} - \alpha\nabla f(\boldsymbol{q})\big)\, dt + \sqrt{2\alpha\beta^{-1}} d\boldsymbol{B}_t^1 \\ d\boldsymbol{p} = \big(-\gamma\boldsymbol{p} - \nabla f(\boldsymbol{q})\big)\, dt + \sqrt{2\gamma\beta^{-1}} d\boldsymbol{B}_t^2 \end{cases}.$$

The Fokker-Planck equation of tempered HFHR in Equation (3.14) is given by

$$\partial_t\rho_t + \nabla\cdot(\rho_t\boldsymbol{J}) = 0, \quad \text{where } \boldsymbol{J} = \left(\begin{bmatrix} \boldsymbol{p} - \alpha\nabla f(\boldsymbol{q}) - \alpha\beta^{-1}\nabla_{\boldsymbol{q}}\log\rho_t \\ -\gamma\boldsymbol{p} - \nabla f(\boldsymbol{q}) - \gamma\beta^{-1}\nabla_{\boldsymbol{p}}\log\rho_t \end{bmatrix}\right) \tag{B.7}$$

Since $\nabla\cdot\left(\rho_t\begin{bmatrix} -\nabla_{\boldsymbol{p}}\log\rho_t \\ \nabla_{\boldsymbol{q}}\log\rho_t \end{bmatrix}\right) = 0$, we can then further simplify $\boldsymbol{J}$ to

$$\boldsymbol{J} = -\beta^{-1}\frac{\pi}{\rho_t}A\begin{bmatrix} \nabla_{\boldsymbol{q}}\frac{\rho_t}{\pi} \\ \nabla_{\boldsymbol{p}}\frac{\rho_t}{\pi} \end{bmatrix}, \tag{B.8}$$

where $A = \begin{bmatrix} \alpha I & -I \\ I & \gamma I \end{bmatrix}$.

The functional derivative w.r.t. ρ_t is

$$\frac{\delta \mathcal{L}_{\text{cross}}(\rho_t)}{\delta \rho_t} = 2(\nabla_{\boldsymbol{x}})^\star (S \nabla_{\boldsymbol{x}} \frac{\rho_t}{\pi})$$

where $(\nabla_{\boldsymbol{x}})^\star$ is the adjoint operator with respect to $\mathbb{E}_\pi[\langle \cdot, \cdot \rangle]$ and

$$(\nabla_{\boldsymbol{x}})^\star = -\nabla_{\boldsymbol{x}}^T - \nabla_{\boldsymbol{x}}^T \log \pi = ((\nabla_{\boldsymbol{q}})^\star, (\nabla_{\boldsymbol{p}})^\star) = (-\nabla_{\boldsymbol{q}}^T + \beta(\boldsymbol{\nabla} f(\boldsymbol{q}))^T, -\nabla_{\boldsymbol{p}}^T + \beta \boldsymbol{p}^T).$$

The time derivative of the $\mathcal{L}_{\text{cross}}(\rho_t)$ is

$$\begin{aligned}
\frac{d}{dt}\mathcal{L}_{\text{cross}}(\rho_t)(\rho_t) &= \int \frac{\delta \mathcal{L}_{\text{cross}}(\rho_t)}{\delta \rho_t} \partial_t \rho_t d\boldsymbol{x} \\
&= -\int \frac{\delta \mathcal{L}_{\text{cross}}(\rho_t)}{\delta \rho_t} \nabla_{\boldsymbol{x}} \cdot (\rho_t \boldsymbol{J}) d\boldsymbol{x} \\
&= \int \langle \nabla_{\boldsymbol{x}} \frac{\delta \mathcal{L}_{\text{cross}}(\rho_t)}{\delta \rho_t}, \boldsymbol{J} \rangle \rho_t d\boldsymbol{x} \\
&= -2\beta^{-1} \int \langle \nabla_{\boldsymbol{x}} (\nabla_{\boldsymbol{x}})^\star (S \nabla_{\boldsymbol{x}} \frac{\rho_t}{\pi}), \begin{bmatrix} \alpha I & -I \\ I & \gamma I \end{bmatrix} \nabla_{\boldsymbol{x}} \frac{\rho_t}{\pi} \rangle d\pi \\
&= -2\beta^{-1} \int \langle \nabla_{\boldsymbol{x}} (\nabla_{\boldsymbol{x}})^\star (S \nabla_{\boldsymbol{x}} h), \begin{bmatrix} \alpha I & -I \\ I & \gamma I \end{bmatrix} \nabla_{\boldsymbol{x}} h \rangle d\pi \qquad (h \triangleq \frac{\rho_t}{\pi})
\end{aligned} \tag{B.9}$$

For the term in Equation (B.9), we have

$$
\begin{aligned}
&- 2\beta^{-1}\mathbb{E}_{\pi}[\left\langle \nabla_{\boldsymbol{x}}((\nabla_{\boldsymbol{x}})^{\star} S \nabla_{\boldsymbol{x}} h), \begin{bmatrix} \alpha I & -I \\ I & \gamma I \end{bmatrix} \nabla_{\boldsymbol{x}} h \right\rangle] \\
= &- 2a\beta^{-1}\mathbb{E}_{\pi}[\left\langle \nabla_{\boldsymbol{x}}(\nabla_{\boldsymbol{q}})^{\star} \nabla_{\boldsymbol{q}} h, \begin{bmatrix} \alpha I & -I \\ I & \gamma I \end{bmatrix} \nabla_{\boldsymbol{x}} h \right\rangle] && \text{(B.10)} \\
&- 2b\beta^{-1}\mathbb{E}_{\pi}[\left\langle \nabla_{\boldsymbol{x}}((\nabla_{\boldsymbol{q}})^{\star} \nabla_{\boldsymbol{p}} h + (\nabla_{\boldsymbol{p}})^{\star} \nabla_{\boldsymbol{q}} h), \begin{bmatrix} \alpha I & -I \\ I & \gamma I \end{bmatrix} \nabla_{\boldsymbol{x}} h \right\rangle] && \text{(B.11)} \\
&- 2d\beta^{-1}\mathbb{E}_{\pi}[\left\langle \nabla_{\boldsymbol{x}}(\nabla_{\boldsymbol{p}})^{\star} \nabla_{\boldsymbol{p}} h, \begin{bmatrix} \alpha I & -I \\ I & \gamma I \end{bmatrix} \nabla_{\boldsymbol{x}} h \right\rangle] && \text{(B.12)}
\end{aligned}
$$

For the cross term in Equation (B.11), we have

$$
\begin{aligned}
&- 2b\beta^{-1}\mathbb{E}_{\pi}[\left\langle \nabla_{\boldsymbol{x}}((\nabla_{\boldsymbol{q}})^{\star} \nabla_{\boldsymbol{p}} h + (\nabla_{\boldsymbol{p}})^{\star} \nabla_{\boldsymbol{q}} h), \begin{bmatrix} \alpha I & -I \\ I & \gamma I \end{bmatrix} \nabla_{\boldsymbol{x}} h \right\rangle] \\
= &- 2b\beta^{-1}\mathbb{E}_{\pi}[\left\langle \nabla_{\boldsymbol{x}}((\nabla_{\boldsymbol{q}})^{\star} \nabla_{\boldsymbol{p}} h + (\nabla_{\boldsymbol{p}})^{\star} \nabla_{\boldsymbol{q}} h), \begin{bmatrix} \alpha I & 0 \\ 0 & \gamma I \end{bmatrix} \nabla_{\boldsymbol{x}} h \right\rangle] \\
&- 2b\beta^{-1}\mathbb{E}_{\pi}[\left\langle \nabla_{\boldsymbol{x}}((\nabla_{\boldsymbol{q}})^{\star} \nabla_{\boldsymbol{p}} h + (\nabla_{\boldsymbol{p}})^{\star} \nabla_{\boldsymbol{q}} h), \begin{bmatrix} 0 & -I \\ I & 0 \end{bmatrix} \nabla_{\boldsymbol{x}} h \right\rangle] \\
= &- 2b\alpha\beta^{-1}\mathbb{E}_{\pi}[\left\langle \nabla_{\boldsymbol{q}}((\nabla_{\boldsymbol{q}})^{\star} \nabla_{\boldsymbol{p}} h + (\nabla_{\boldsymbol{p}})^{\star} \nabla_{\boldsymbol{q}} h), \nabla_{\boldsymbol{q}} h \right\rangle] && \text{(B.13)} \\
&- 2b\gamma\beta^{-1}\mathbb{E}_{\pi}[\left\langle \nabla_{\boldsymbol{p}}((\nabla_{\boldsymbol{q}})^{\star} \nabla_{\boldsymbol{p}} h + (\nabla_{\boldsymbol{p}})^{\star} \nabla_{\boldsymbol{q}} h), \nabla_{\boldsymbol{p}} h \right\rangle] && \text{(B.14)} \\
&- 2b\beta^{-1}\mathbb{E}_{\pi}[\left\langle \nabla_{\boldsymbol{x}}((\nabla_{\boldsymbol{q}})^{\star} \nabla_{\boldsymbol{p}} h + (\nabla_{\boldsymbol{p}})^{\star} \nabla_{\boldsymbol{q}} h), \begin{bmatrix} 0 & -I \\ I & 0 \end{bmatrix} \nabla_{\boldsymbol{x}} h \right\rangle] && \text{(B.15)}
\end{aligned}
$$

For the term in Equation (B.13), we have

$$
\begin{aligned}
& -2b\alpha\beta^{-1}\mathbb{E}_{\pi}[\left\langle \nabla_{\boldsymbol{q}}((\nabla_{\boldsymbol{q}})^{\star}\nabla_{\boldsymbol{p}}h+(\nabla_{\boldsymbol{p}})^{\star}\nabla_{\boldsymbol{q}}h),\nabla_{\boldsymbol{q}}h\right\rangle] \\
= & -2b\alpha\beta^{-1}\mathbb{E}_{\pi}[\left\langle \nabla_{\boldsymbol{q}}(\nabla_{\boldsymbol{q}})^{\star}\nabla_{\boldsymbol{p}}h,\nabla_{\boldsymbol{q}}h\right\rangle +\left\langle \nabla_{\boldsymbol{q}}(\nabla_{\boldsymbol{p}})^{\star}\nabla_{\boldsymbol{q}}h,\nabla_{\boldsymbol{q}}h\right\rangle] \\
= & -2b\alpha\beta^{-1}\mathbb{E}_{\pi}[\left\langle \nabla_{\boldsymbol{p}}h,\nabla_{\boldsymbol{q}}(\nabla_{\boldsymbol{q}})^{\star}\nabla_{\boldsymbol{q}}h\right\rangle +\left\langle (\nabla_{\boldsymbol{q}})^{\star}\nabla_{\boldsymbol{q}}\nabla_{\boldsymbol{q}}h,\nabla_{\boldsymbol{p}}h\right\rangle] \\
= & -2b\alpha\beta^{-1}\mathbb{E}_{\pi}[\left\langle \nabla_{\boldsymbol{p}}h,((\nabla_{\boldsymbol{q}})^{\star}\nabla_{\boldsymbol{q}}+\nabla_{\boldsymbol{q}}(\nabla_{\boldsymbol{q}})^{\star})\nabla_{\boldsymbol{q}}h\right\rangle] \\
= & -2b\alpha\beta^{-1}\mathbb{E}_{\pi}[\left\langle \nabla_{\boldsymbol{p}}h,(2(\nabla_{\boldsymbol{q}})^{\star}\nabla_{\boldsymbol{q}}+[\nabla_{\boldsymbol{q}},(\nabla_{\boldsymbol{q}})^{\star}])\nabla_{\boldsymbol{q}}h\right\rangle] \\
= & -4b\alpha\beta^{-1}\mathbb{E}_{\pi}[\left\langle \nabla_{\boldsymbol{q}}\nabla_{\boldsymbol{p}}h,\nabla_{\boldsymbol{q}}\nabla_{\boldsymbol{q}}h\right\rangle_{F}]-2b\alpha\mathbb{E}_{\pi}[\left\langle \nabla_{\boldsymbol{p}}h,\nabla^{2}f(\boldsymbol{q})\nabla_{\boldsymbol{q}}h\right\rangle]
\end{aligned}
$$

where we make use of the commutator $[\nabla_{\boldsymbol{q}},(\nabla_{\boldsymbol{q}})^{\star}]$ of $\nabla_{\boldsymbol{q}}$ and $(\nabla_{\boldsymbol{q}})^{\star}$

$$
[\nabla_{\boldsymbol{q}},(\nabla_{\boldsymbol{q}})^{\star}]=\nabla_{\boldsymbol{q}}(\nabla_{\boldsymbol{q}})^{\star}-(\nabla_{\boldsymbol{q}})^{\star}\nabla_{\boldsymbol{q}}=\beta\nabla^{2}f(\boldsymbol{q})+\nabla_{\boldsymbol{q}}^{T}\nabla_{\boldsymbol{q}}-\nabla_{\boldsymbol{q}}\nabla_{\boldsymbol{q}}^{T}.
$$

For the term in Equation (B.14), we have

$$
\begin{aligned}
& -2b\gamma\beta^{-1}\mathbb{E}_{\pi}[\left\langle \nabla_{\boldsymbol{p}}((\nabla_{\boldsymbol{q}})^{\star}\nabla_{\boldsymbol{p}}h+(\nabla_{\boldsymbol{p}})^{\star}\nabla_{\boldsymbol{q}}h),\nabla_{\boldsymbol{p}}h\right\rangle] \\
= & -2b\gamma\beta^{-1}\mathbb{E}_{\pi}[\left\langle \nabla_{\boldsymbol{p}}(\nabla_{\boldsymbol{q}})^{\star}\nabla_{\boldsymbol{p}}h,\nabla_{\boldsymbol{p}}h\right\rangle +\left\langle \nabla_{\boldsymbol{p}}(\nabla_{\boldsymbol{p}})^{\star}\nabla_{\boldsymbol{q}}h,\nabla_{\boldsymbol{p}}h\right\rangle] \\
= & -2b\gamma\beta^{-1}\mathbb{E}_{\pi}[\left\langle \nabla_{\boldsymbol{q}}h,(\nabla_{\boldsymbol{p}})^{\star}\nabla_{\boldsymbol{p}}\nabla_{\boldsymbol{p}}h\right\rangle +\left\langle \nabla_{\boldsymbol{q}}h,\nabla_{\boldsymbol{p}}(\nabla_{\boldsymbol{p}})^{\star}\nabla_{\boldsymbol{p}}h\right\rangle] \\
= & -2b\gamma\beta^{-1}\mathbb{E}_{\pi}[\left\langle \nabla_{\boldsymbol{q}}h,((\nabla_{\boldsymbol{p}})^{\star}\nabla_{\boldsymbol{p}}+\nabla_{\boldsymbol{p}}(\nabla_{\boldsymbol{p}})^{\star})\nabla_{\boldsymbol{p}}h\right\rangle] \\
= & -2b\gamma\beta^{-1}\mathbb{E}_{\pi}[\left\langle \nabla_{\boldsymbol{q}}h,(2(\nabla_{\boldsymbol{p}})^{\star}\nabla_{\boldsymbol{p}}+[\nabla_{\boldsymbol{p}},(\nabla_{\boldsymbol{p}})^{\star}])\nabla_{\boldsymbol{p}}h\right\rangle] \\
= & -4b\gamma\beta^{-1}\mathbb{E}_{\pi}[\left\langle \nabla_{\boldsymbol{p}}\nabla_{\boldsymbol{q}}h,\nabla_{\boldsymbol{p}}\nabla_{\boldsymbol{p}}h\right\rangle_{F}]-2b\gamma\mathbb{E}_{\pi}[\left\langle \nabla_{\boldsymbol{p}}h,\nabla_{\boldsymbol{q}}h\right\rangle]
\end{aligned}
$$

where we make use of the commutator $[\nabla_{\boldsymbol{p}},(\nabla_{\boldsymbol{p}})^{\star}]$ of $\nabla_{\boldsymbol{p}}$ and $(\nabla_{\boldsymbol{p}})^{\star}$

$$
[\nabla_{\boldsymbol{p}},(\nabla_{\boldsymbol{p}})^{\star}]=\nabla_{\boldsymbol{p}}(\nabla_{\boldsymbol{p}})^{\star}-(\nabla_{\boldsymbol{p}})^{\star}\nabla_{\boldsymbol{p}}=\beta I+\nabla_{\boldsymbol{p}}^{T}\nabla_{\boldsymbol{p}}-\nabla_{\boldsymbol{p}}\nabla_{\boldsymbol{p}}^{T}.
$$

For the term in Equation (B.15), we have

$$
\begin{aligned}
&- 2b\beta^{-1}\mathbb{E}_\pi[\left\langle \nabla_{\boldsymbol{x}}((\nabla_{\boldsymbol{q}})^\star \nabla_{\boldsymbol{p}} h + (\nabla_{\boldsymbol{p}})^\star \nabla_{\boldsymbol{q}} h), \begin{bmatrix} 0 & -I \\ I & 0 \end{bmatrix} \nabla_{\boldsymbol{x}} h \right\rangle] \\
=&- 2b\beta^{-1}\mathbb{E}_\pi[((\nabla_{\boldsymbol{q}})^\star \nabla_{\boldsymbol{p}} h + (\nabla_{\boldsymbol{p}})^\star \nabla_{\boldsymbol{q}} h) \cdot (\nabla_{\boldsymbol{x}} h)^\star (\begin{bmatrix} 0 & -I \\ I & 0 \end{bmatrix} \nabla_{\boldsymbol{x}} h)] \\
=&- 2b\beta^{-1}\mathbb{E}_\pi[((\nabla_{\boldsymbol{q}})^\star \nabla_{\boldsymbol{p}} h + (\nabla_{\boldsymbol{p}})^\star \nabla_{\boldsymbol{q}} h) \cdot (-(\nabla_{\boldsymbol{q}})^\star \nabla_{\boldsymbol{p}} h + (\nabla_{\boldsymbol{p}})^\star \nabla_{\boldsymbol{q}} h)] \\
=&- 2b\beta^{-1}\mathbb{E}_\pi[((\nabla_{\boldsymbol{p}})^\star \nabla_{\boldsymbol{q}} h)^2 - ((\nabla_{\boldsymbol{q}})^\star \nabla_{\boldsymbol{p}} h)^2] \\
=&- 2b\beta^{-1}\mathbb{E}_\pi[\left\langle \nabla_{\boldsymbol{q}} h, \nabla_{\boldsymbol{p}} (\nabla_{\boldsymbol{p}})^\star \nabla_{\boldsymbol{q}} h \right\rangle - \left\langle \nabla_{\boldsymbol{p}} h, \nabla_{\boldsymbol{q}} (\nabla_{\boldsymbol{q}})^\star \nabla_{\boldsymbol{p}} h \right\rangle] \\
=&- 2b\beta^{-1}\mathbb{E}_\pi[\left\langle \nabla_{\boldsymbol{q}} h, \nabla_{\boldsymbol{p}} (\nabla_{\boldsymbol{p}})^\star \nabla_{\boldsymbol{q}} h \right\rangle - \left\langle \nabla_{\boldsymbol{q}} h, (\nabla_{\boldsymbol{p}})^\star \nabla_{\boldsymbol{p}} \nabla_{\boldsymbol{q}} h \right\rangle] \\
&- 2b\beta^{-1}\mathbb{E}_\pi[\left\langle \nabla_{\boldsymbol{q}} h, (\nabla_{\boldsymbol{p}})^\star \nabla_{\boldsymbol{p}} \nabla_{\boldsymbol{q}} h \right\rangle - \left\langle \nabla_{\boldsymbol{p}} h, \nabla_{\boldsymbol{q}} (\nabla_{\boldsymbol{q}})^\star \nabla_{\boldsymbol{p}} h \right\rangle] \\
=&- 2b\beta^{-1}\mathbb{E}_\pi[\left\langle \nabla_{\boldsymbol{q}} h, [\nabla_{\boldsymbol{p}}, (\nabla_{\boldsymbol{p}})^\star] \nabla_{\boldsymbol{q}} h \right\rangle + \left\langle \nabla_{\boldsymbol{q}} h, (\nabla_{\boldsymbol{p}})^\star \nabla_{\boldsymbol{p}} \nabla_{\boldsymbol{q}} h \right\rangle - \left\langle \nabla_{\boldsymbol{p}} h, \nabla_{\boldsymbol{q}} (\nabla_{\boldsymbol{q}})^\star \nabla_{\boldsymbol{p}} h \right\rangle] \\
=&- 2b\beta^{-1}\mathbb{E}_\pi[\left\langle \nabla_{\boldsymbol{q}} h, [\nabla_{\boldsymbol{p}}, (\nabla_{\boldsymbol{p}})^\star] \nabla_{\boldsymbol{q}} h \right\rangle + \left\langle \nabla_{\boldsymbol{p}} h, (\nabla_{\boldsymbol{q}})^\star \nabla_{\boldsymbol{q}} \nabla_{\boldsymbol{p}} h \right\rangle - \left\langle \nabla_{\boldsymbol{p}} h, \nabla_{\boldsymbol{q}} (\nabla_{\boldsymbol{q}})^\star \nabla_{\boldsymbol{p}} h \right\rangle] \\
=&- 2b\beta^{-1}\mathbb{E}_\pi[\left\langle \nabla_{\boldsymbol{q}} h, [\nabla_{\boldsymbol{p}}, (\nabla_{\boldsymbol{p}})^\star] \nabla_{\boldsymbol{q}} h \right\rangle - \left\langle \nabla_{\boldsymbol{p}} h, [\nabla_{\boldsymbol{q}}, (\nabla_{\boldsymbol{q}})^\star] \nabla_{\boldsymbol{p}} h \right\rangle] \\
=&- 2b\mathbb{E}_\pi[\left\langle \nabla_{\boldsymbol{q}} h, \nabla_{\boldsymbol{q}} h \right\rangle - \left\langle \nabla_{\boldsymbol{p}} h, \nabla^2 f(\boldsymbol{q}) \nabla_{\boldsymbol{p}} h \right\rangle]
\end{aligned}
$$

For the quadratic term in Equation (B.10), we have

$$
\begin{aligned}
&- 2a\beta^{-1}\mathbb{E}_\pi[\left\langle \nabla_{\boldsymbol{x}}(\nabla_{\boldsymbol{q}})^\star \nabla_{\boldsymbol{q}} h, \begin{bmatrix} \alpha I & -I \\ I & \gamma I \end{bmatrix} \nabla_{\boldsymbol{x}} h \right\rangle] \\
=&- 2a\beta^{-1}\mathbb{E}_\pi[(\nabla_{\boldsymbol{q}})^\star \nabla_{\boldsymbol{q}} h \cdot (\nabla_{\boldsymbol{x}})^\star \begin{bmatrix} \alpha\nabla_{\boldsymbol{q}} h - \nabla_{\boldsymbol{p}} h \\ \nabla_{\boldsymbol{q}} h + \gamma \nabla_{\boldsymbol{p}} h \end{bmatrix}] \\
=&- 2a\beta^{-1}\mathbb{E}_\pi[(\nabla_{\boldsymbol{q}})^\star \nabla_{\boldsymbol{q}} h \cdot (\alpha(\nabla_{\boldsymbol{q}})^\star \nabla_{\boldsymbol{q}} h - (\nabla_{\boldsymbol{q}})^\star \nabla_{\boldsymbol{p}} h + (\nabla_{\boldsymbol{p}})^\star \nabla_{\boldsymbol{q}} h + \gamma(\nabla_{\boldsymbol{p}})^\star \nabla_{\boldsymbol{p}} h)] \\
=&- 2a\alpha\beta^{-1}\mathbb{E}_\pi[\left\langle \nabla_{\boldsymbol{q}} h, \nabla_{\boldsymbol{q}}(\nabla_{\boldsymbol{q}})^\star \nabla_{\boldsymbol{q}} h \right\rangle] - 2a\gamma\beta^{-1}\mathbb{E}_\pi[(\nabla_{\boldsymbol{q}})^\star \nabla_{\boldsymbol{q}}(\nabla_{\boldsymbol{p}})^\star \nabla_{\boldsymbol{p}} h] \\
&- 2a\beta^{-1}\mathbb{E}_\pi[-(\nabla_{\boldsymbol{q}})^\star \nabla_{\boldsymbol{q}} h(\nabla_{\boldsymbol{q}})^\star \nabla_{\boldsymbol{p}} h + (\nabla_{\boldsymbol{q}})^\star \nabla_{\boldsymbol{q}} h(\nabla_{\boldsymbol{p}})^\star \nabla_{\boldsymbol{q}} h] \\
=&- 2a\alpha\beta^{-1}\mathbb{E}_\pi[\left\langle \nabla_{\boldsymbol{q}} h, ((\nabla_{\boldsymbol{q}})^\star \nabla_{\boldsymbol{q}} + \beta\nabla^2 f(\boldsymbol{q}))\nabla_{\boldsymbol{q}} h \right\rangle] + 2a\beta^{-1}\mathbb{E}_\pi[\left\langle \nabla_{\boldsymbol{p}} h, [\nabla_{\boldsymbol{q}}, (\nabla_{\boldsymbol{q}})^\star]\nabla_{\boldsymbol{q}} h \right\rangle] \\
&- 2a\beta^{-1}\mathbb{E}_\pi[\left\langle \nabla_{\boldsymbol{p}}\nabla_{\boldsymbol{q}} h, \nabla_{\boldsymbol{q}}\nabla_{\boldsymbol{q}} h \right\rangle_F] \\
=&- 2a\alpha\beta^{-1}\mathbb{E}_\pi[\left\langle \nabla_{\boldsymbol{q}}\nabla_{\boldsymbol{q}} h, \nabla_{\boldsymbol{q}}\nabla_{\boldsymbol{q}} h \right\rangle_F] - 2a\alpha\mathbb{E}_\pi[\left\langle \nabla_{\boldsymbol{q}} h, \nabla^2 f(\boldsymbol{q})\nabla_{\boldsymbol{q}} h \right\rangle] \\
&+ 2a\mathbb{E}_\pi[\left\langle \nabla_{\boldsymbol{p}} h, \nabla^2 f(\boldsymbol{q})\nabla_{\boldsymbol{q}} h \right\rangle] - 2a\gamma\beta^{-1}\mathbb{E}_\pi[\left\langle \nabla_{\boldsymbol{q}}\nabla_{\boldsymbol{p}} h, \nabla_{\boldsymbol{q}}\nabla_{\boldsymbol{p}} h \right\rangle_F]
\end{aligned}
$$

Similarly, for the term in Equation (B.12), we have

$$
\begin{aligned}
&- 2d\beta^{-1}\mathbb{E}_\pi[\left\langle \nabla_{\boldsymbol{x}}(\nabla_{\boldsymbol{p}})^\star \nabla_{\boldsymbol{p}} h, \begin{bmatrix} \alpha I & -I \\ I & \gamma I \end{bmatrix} \nabla_{\boldsymbol{x}} h \right\rangle] \\
=&- 2d\beta^{-1}\mathbb{E}_\pi[(\nabla_{\boldsymbol{p}})^\star \nabla_{\boldsymbol{p}} h \cdot (\nabla_{\boldsymbol{x}})^\star \begin{bmatrix} \alpha\nabla_{\boldsymbol{q}} h - \nabla_{\boldsymbol{p}} h \\ \nabla_{\boldsymbol{q}} h + \gamma \nabla_{\boldsymbol{p}} h \end{bmatrix}] \\
=&- 2d\beta^{-1}\mathbb{E}_\pi[(\nabla_{\boldsymbol{p}})^\star \nabla_{\boldsymbol{p}} h \cdot (\alpha(\nabla_{\boldsymbol{q}})^\star \nabla_{\boldsymbol{q}} h - (\nabla_{\boldsymbol{q}})^\star \nabla_{\boldsymbol{p}} h + (\nabla_{\boldsymbol{p}})^\star \nabla_{\boldsymbol{q}} h + \gamma(\nabla_{\boldsymbol{p}})^\star \nabla_{\boldsymbol{p}} h)] \\
=&- 2d\alpha\beta^{-1}\mathbb{E}_\pi[\left\langle \nabla_{\boldsymbol{p}} h, \nabla_{\boldsymbol{p}}(\nabla_{\boldsymbol{q}})^\star \nabla_{\boldsymbol{q}} h \right\rangle] \\
&- 2d\beta^{-1}\mathbb{E}_\pi[-(\nabla_{\boldsymbol{p}})^\star \nabla_{\boldsymbol{p}} h (\nabla_{\boldsymbol{q}})^\star \nabla_{\boldsymbol{p}} h + (\nabla_{\boldsymbol{p}})^\star \nabla_{\boldsymbol{p}} h (\nabla_{\boldsymbol{p}})^\star \nabla_{\boldsymbol{q}} h] \\
&- 2d\gamma\beta^{-1}\mathbb{E}_\pi[\left\langle \nabla_{\boldsymbol{p}}, \nabla_{\boldsymbol{p}}(\nabla_{\boldsymbol{p}})^\star \nabla_{\boldsymbol{p}} h \right\rangle] \\
=&- 2d\alpha\beta^{-1}\mathbb{E}_\pi[\left\langle \nabla_{\boldsymbol{q}} \nabla_{\boldsymbol{p}} h, \nabla_{\boldsymbol{q}} \nabla_{\boldsymbol{p}} h \right\rangle_F] - 2d\beta^{-1}\mathbb{E}_\pi[\left\langle \nabla_{\boldsymbol{q}} h, [\nabla_{\boldsymbol{p}}, (\nabla_{\boldsymbol{p}})^\star] \nabla_{\boldsymbol{p}} h \right\rangle] \\
&- 2d\gamma\beta^{-1}\mathbb{E}_\pi[\left\langle \nabla_{\boldsymbol{p}}, ((\nabla_{\boldsymbol{p}})^\star \nabla_{\boldsymbol{p}} + \beta I) \nabla_{\boldsymbol{p}} h \right\rangle] \\
=&- 2d\alpha\beta^{-1}\mathbb{E}_\pi[\left\langle \nabla_{\boldsymbol{q}} \nabla_{\boldsymbol{p}} h, \nabla_{\boldsymbol{q}} \nabla_{\boldsymbol{p}} h \right\rangle_F] - 2d\mathbb{E}_\pi[\left\langle \nabla_{\boldsymbol{q}} h, \nabla_{\boldsymbol{p}} h \right\rangle] \\
&- 2d\gamma\beta^{-1}\mathbb{E}_\pi[\left\langle \nabla_{\boldsymbol{p}} \nabla_{\boldsymbol{p}} h, \nabla_{\boldsymbol{p}} \nabla_{\boldsymbol{p}} h \right\rangle_F] - 2d\gamma\mathbb{E}_\pi[\left\langle \nabla_{\boldsymbol{p}} h, \nabla_{\boldsymbol{p}} h \right\rangle]
\end{aligned}
$$

We now collect all terms in regular Euclidean inner product, i.e., $\left\langle , \right\rangle$, we have

$$
\begin{cases}
\left\langle \nabla_{\boldsymbol{q}} h, \nabla_{\boldsymbol{q}} h \right\rangle : -2a\alpha\nabla^2 f(\boldsymbol{q}) - 2b \\
\left\langle \nabla_{\boldsymbol{q}} h, \nabla_{\boldsymbol{p}} h \right\rangle : 2a\nabla^2 f(\boldsymbol{q}) - 2b\alpha\nabla^2 f(\boldsymbol{q}) - 2b\gamma - 2d \\
\left\langle \nabla_{\boldsymbol{p}} h, \nabla_{\boldsymbol{p}} h \right\rangle : 2b\nabla^2 f(\boldsymbol{q}) - 2d\gamma
\end{cases}
$$

Therefore, if we denote

$$
M_{\text{cross}} = \begin{bmatrix} 2a\alpha\nabla^2 f(\boldsymbol{q}) + 2bI & -a\nabla^2 f(\boldsymbol{q}) + b\alpha\nabla^2 f(\boldsymbol{q}) + b\gamma I + dI \\ -a\nabla^2 f(\boldsymbol{q}) + b\alpha\nabla^2 f(\boldsymbol{q}) + b\gamma I + dI & -2b\nabla^2 f(\boldsymbol{q}) + 2d\gamma I \end{bmatrix} \tag{B.16}
$$

then the component containing regular Euclidean inner product can be written in a compact

form

$$-\mathbb{E}_\pi[\langle \nabla_{\boldsymbol{x}} h, M_{\text{cross}} \nabla_{\boldsymbol{x}} h \rangle]$$

Next, we collect all terms in Frobenius inner product, i.e., $\langle , \rangle_F$, we have

$$\begin{aligned}
&-2\alpha\beta^{-1}\mathbb{E}_\pi \left[a \langle \nabla_{\boldsymbol{q}} \nabla_{\boldsymbol{q}} h, \nabla_{\boldsymbol{q}} \nabla_{\boldsymbol{q}} h \rangle_F + 2b \langle \nabla_{\boldsymbol{q}} \nabla_{\boldsymbol{p}} h, \nabla_{\boldsymbol{q}} \nabla_{\boldsymbol{q}} h \rangle_F + d \langle \nabla_{\boldsymbol{q}} \nabla_{\boldsymbol{p}} h, \nabla_{\boldsymbol{q}} \nabla_{\boldsymbol{p}} h \rangle_F \right] \\
&-2\gamma\beta^{-1}\mathbb{E}_\pi \left[a \langle \nabla_{\boldsymbol{q}} \nabla_{\boldsymbol{p}} h, \nabla_{\boldsymbol{q}} \nabla_{\boldsymbol{p}} h \rangle_F + 2b \langle \nabla_{\boldsymbol{p}} \nabla_{\boldsymbol{q}} h, \nabla_{\boldsymbol{p}} \nabla_{\boldsymbol{p}} h \rangle_F + d \langle \nabla_{\boldsymbol{p}} \nabla_{\boldsymbol{p}} h, \nabla_{\boldsymbol{p}} \nabla_{\boldsymbol{p}} h \rangle_F \right] \\
&= -2\alpha\beta^{-1}\mathbb{E}_\pi[\langle \nabla_{\boldsymbol{x}} \nabla_{\boldsymbol{q}} h, S \nabla_{\boldsymbol{x}} \nabla_{\boldsymbol{q}} h \rangle_F] - 2\gamma\beta^{-1}\mathbb{E}_\pi[\langle \nabla_{\boldsymbol{x}} \nabla_{\boldsymbol{p}} h, S \nabla_{\boldsymbol{x}} \nabla_{\boldsymbol{p}} h \rangle_F]
\end{aligned}$$

Now we sum up all terms and obtain

$$\begin{aligned}
&\frac{d}{dt}\mathcal{L}_{\text{cross}}(\rho_t)(\rho_t) \\
=& -\mathbb{E}_\pi[\langle \nabla_{\boldsymbol{x}} \frac{\rho_t}{\pi}, M_{\text{cross}} \nabla_{\boldsymbol{x}} \frac{\rho_t}{\pi} \rangle] \\
& -2\alpha\beta^{-1}\mathbb{E}_\pi[\langle \nabla_{\boldsymbol{x}} \nabla_{\boldsymbol{q}} h, S \nabla_{\boldsymbol{x}} \nabla_{\boldsymbol{q}} h \rangle_F] - 2\gamma\beta^{-1}\mathbb{E}_\pi[\langle \nabla_{\boldsymbol{x}} \nabla_{\boldsymbol{p}} h, S \nabla_{\boldsymbol{x}} \nabla_{\boldsymbol{p}} h \rangle_F] \\
\leq& -\mathbb{E}_\pi[\langle \nabla_{\boldsymbol{x}} \frac{\rho_t}{\pi}, M_{\text{cross}} \nabla_{\boldsymbol{x}} \frac{\rho_t}{\pi} \rangle]
\end{aligned}$$

■

B.4 Dependence of error of SDE on initial values

Lemma 30 *Consider the following two SDE with different initial condition*

$$\begin{cases} d\boldsymbol{x}_t = \boldsymbol{a}(\boldsymbol{x}_t)dt + \boldsymbol{\sigma} d\boldsymbol{W}_t, \\ \boldsymbol{x}(0) = \boldsymbol{x}_0 \end{cases} \qquad \begin{cases} d\boldsymbol{y}_t = \boldsymbol{a}(\boldsymbol{y}_t)dt + \boldsymbol{\sigma} d\boldsymbol{W}_t, \\ \boldsymbol{y}(0) = \boldsymbol{y}_0 \end{cases}$$

where $\boldsymbol{a}(\boldsymbol{u}) \in \mathbb{R}^d$ is L-Lipschitz, and $\boldsymbol{\sigma} \in \mathbb{R}^{n\times n}$ is a constant matrix. For $0 < h < \frac{1}{4L}$, we have the following representation

$$\boldsymbol{x}_h - \boldsymbol{y}_h = \boldsymbol{x}_0 - \boldsymbol{y}_0 + \boldsymbol{z}$$

with

$$E\|\boldsymbol{z}\|^2 \leq 2L^2\|\boldsymbol{x}_0 - \boldsymbol{y}_0\|^2 h^2$$

Proof: Let $\boldsymbol{z} = (\boldsymbol{x}_h - \boldsymbol{y}_h) - (\boldsymbol{x}_0 - \boldsymbol{y}_0) = \int_0^h \boldsymbol{a}(\boldsymbol{x}_s) - \boldsymbol{a}(\boldsymbol{y}_s)ds$. Ito's lemma readily implies that

$$\begin{aligned}\mathbb{E}\|\boldsymbol{x}_h - \boldsymbol{y}_h\|^2 =& \|\boldsymbol{x}_0 - \boldsymbol{y}_0\|^2 + 2\mathbb{E}\int_0^h \langle \boldsymbol{x}_s - \boldsymbol{y}_s, \boldsymbol{a}(\boldsymbol{x}_s) - \boldsymbol{a}(\boldsymbol{y}_s)\rangle ds \\ \leq& \|\boldsymbol{x}_0 - \boldsymbol{y}_0\|^2 + 2L\int_0^h \mathbb{E}\|\boldsymbol{x}_s - \boldsymbol{y}_s\|^2 ds\end{aligned}$$

By Gronwall's inequality, it follows that

$$\mathbb{E}\|\boldsymbol{x}_h - \boldsymbol{y}_h\|^2 \leq \|\boldsymbol{x}_0 - \boldsymbol{y}_0\|^2 e^{2Lh} \leq 2\|\boldsymbol{x}_0 - \boldsymbol{y}_0\|^2, \text{ for } 0 < h < \frac{1}{4L}$$

We have that

$$\begin{aligned}\mathbb{E}\|\boldsymbol{z}\|^2 = \left\|\mathbb{E}\left[\int_0^h \boldsymbol{a}(\boldsymbol{x}_s) - \boldsymbol{a}(\boldsymbol{y}_s)ds\right]\right\|^2 \leq& \left(\int_0^h \left\|\mathbb{E}\left[\boldsymbol{a}(\boldsymbol{x}_s) - \boldsymbol{a}(\boldsymbol{y}_s)\right]\right\| ds\right)^2 \\ \leq& \int_0^h 1^2 ds \int_0^h \left\|\mathbb{E}\left[\boldsymbol{a}(\boldsymbol{x}_s) - \boldsymbol{a}(\boldsymbol{y}_s)\right]\right\|^2 ds \\ \leq& h\int_0^h \mathbb{E}\left\|\boldsymbol{a}(\boldsymbol{x}_s) - \boldsymbol{a}(\boldsymbol{y}_s)\right\|^2 ds \\ \leq& L^2 h\int_0^h \mathbb{E}\|\boldsymbol{x}_s - \boldsymbol{y}_s\|^2 ds \\ \leq& 2L^2\|\boldsymbol{x}_0 - \boldsymbol{y}_0\|^2 h^2\end{aligned}$$

■

B.5 Growth bound of SDE with additive noise

Lemma 31 *Consider the following SDE with constant diffusion*

$$\begin{cases} d\boldsymbol{x}_t = \boldsymbol{a}(\boldsymbol{x}_t)dt + \boldsymbol{\sigma} d\boldsymbol{W}_t, \\ \boldsymbol{x}(0) = \boldsymbol{x}_0 \end{cases}$$

where $\boldsymbol{a}(\boldsymbol{x}) \in \mathbb{R}^d$ is L-smooth, i.e., $|\boldsymbol{a}(\boldsymbol{y}) - \boldsymbol{a}(\boldsymbol{x})| \leq L|\boldsymbol{y} - \boldsymbol{x}|$, $\boldsymbol{a}(\boldsymbol{0}) = \boldsymbol{0}$ and $\boldsymbol{\sigma} \in \mathbb{R}^{d\times d}$ is a constant matrix independent of time t and $\boldsymbol{x}_t$. Then for $0 < h < \frac{1}{4L}$, we have

$$\mathbb{E}\|\boldsymbol{x}_h - \boldsymbol{x}_0\|^2 \leq 2.57 \left(\|\boldsymbol{\sigma}\|_F^2 + 2hL^2\|\boldsymbol{x}_0\|^2 \right) h.$$

Proof: We have

$$\begin{aligned}
\mathbb{E}\|\boldsymbol{x}_h - \boldsymbol{x}_0\|^2 =& \mathbb{E}\left\| \int_0^h \boldsymbol{a}(\boldsymbol{x}_t)dt + \int_0^h \boldsymbol{\sigma} d\boldsymbol{W}_t \right\|^2 \\
\leq& 2\mathbb{E}\left\| \int_0^h \boldsymbol{a}(\boldsymbol{x}_t)dt \right\|^2 + 2\mathbb{E}\left\| \int_0^h \boldsymbol{\sigma} d\boldsymbol{W}_t \right\|^2 \\
\overset{(i)}{=}& 2\mathbb{E}\left\| \int_0^h \boldsymbol{a}(\boldsymbol{x}_t)dt \right\|^2 + 2\int_0^h \|\boldsymbol{\sigma}\|_F^2 dt \\
\leq& 2\mathbb{E}\left[\left(\int_0^h \left\|\boldsymbol{a}(\boldsymbol{x}_t)\right\| dt \right)^2 \right] + 2h\|\boldsymbol{\sigma}\|_F^2 \\
\leq& 2\mathbb{E}\left[\left(\int_0^h \left\|\boldsymbol{a}(\boldsymbol{x}_t) - \boldsymbol{a}(\boldsymbol{x}_0)\right\| dt + \int_0^h \left\|\boldsymbol{a}(\boldsymbol{x}_0)\right\| dt \right)^2 \right] + 2h\|\boldsymbol{\sigma}\|_F^2 \\
\leq& 2\mathbb{E}\left[\left(L\int_0^h \|\boldsymbol{x}_t - \boldsymbol{x}_0\| dt + h\left\|\boldsymbol{a}(\boldsymbol{x}_0)\right\| \right)^2 \right] + 2h\|\boldsymbol{\sigma}\|_F^2 \\
\leq& 4\mathbb{E}\left[L^2 \left(\int_0^h \|\boldsymbol{x}_t - \boldsymbol{x}_0\| dt \right)^2 + h^2\left\|\boldsymbol{a}(\boldsymbol{x}_0)\right\|^2 \right] + 2h\|\boldsymbol{\sigma}\|_F^2 \\
\overset{(ii)}{\leq}& 2h\|\boldsymbol{\sigma}\|_F^2 + 4h^2\left\|\boldsymbol{a}(\boldsymbol{x}_0)\right\|^2 + 4L^2 h\int_0^h \mathbb{E}\|\boldsymbol{x}_t - \boldsymbol{x}_0\|^2 dt
\end{aligned}$$

where (i) is due to Ito's isometry, (ii) is due to Cauchy-Schwarz inequality and $\|\boldsymbol{\sigma}\|_F$ is the Frobenius norm of $\boldsymbol{\sigma}$. By Gronwall's inequality, we obtain

$$\mathbb{E}\|\boldsymbol{x}_h - \boldsymbol{x}_0\|^2 \leq \left(2h\|\boldsymbol{\sigma}\|_F^2 + 4h^2\big\|\boldsymbol{a}(\boldsymbol{x}_0)\big\|^2\right)\exp\left\{4L^2h^2\right\}.$$

Since $\big\|\boldsymbol{a}(\boldsymbol{x}_0)\big\| = \big\|\boldsymbol{a}(\boldsymbol{x}_0) - \boldsymbol{a}(\boldsymbol{0})\big\| \leq L\|\boldsymbol{x}_0\|$, when $0 < h < \frac{1}{4L}$, we finally reach at

$$\mathbb{E}\|\boldsymbol{x}_h - \boldsymbol{x}_0\|^2 \leq 2\left(\|\boldsymbol{\sigma}\|_F^2 + 2hL^2\|\boldsymbol{x}_0\|^2\right)e^{\frac{1}{4}}h \leq 2.57\left(\|\boldsymbol{\sigma}\|_F^2 + 2hL^2\|\boldsymbol{x}_0\|^2\right)h$$

■

B.6 Lipschitz continuity of the drift of HFHR dynamics

Lemma 32 *Assume ∇f is L-Lipschitz, i.e. $\big\|\nabla f(\boldsymbol{x}) - \nabla f(\boldsymbol{y})\big\| \leq L\|\boldsymbol{x} - \boldsymbol{y}\|$, then the drift term of HFHR dynamics*

$$\begin{bmatrix} \boldsymbol{p} - \alpha\nabla f(\boldsymbol{q}) \\ -\gamma\boldsymbol{p} - \nabla f(\boldsymbol{q}) \end{bmatrix}$$

is L'-Lipschitz, where $L' \triangleq \sqrt{2}\max\{\sqrt{1+\alpha^2}\max\{\frac{1}{\sqrt{2}}, L\}, \sqrt{1+\gamma^2}\}$. Denote $P \triangleq \begin{bmatrix} \gamma I & I \\ 0 & \sqrt{1+\alpha\gamma}I \end{bmatrix}$ and $\begin{bmatrix} \boldsymbol{\phi} \\ \boldsymbol{\psi} \end{bmatrix} = P\begin{bmatrix} \boldsymbol{q} \\ \boldsymbol{p} \end{bmatrix}$, then $\begin{bmatrix} \boldsymbol{\phi} \\ \boldsymbol{\psi} \end{bmatrix}$ satisfies the following SDE

$$\begin{bmatrix} d\boldsymbol{\phi} \\ d\boldsymbol{\psi} \end{bmatrix} = P\begin{bmatrix} \boldsymbol{p}(\boldsymbol{\phi},\boldsymbol{\psi}) - \alpha\nabla f(\boldsymbol{q}(\boldsymbol{\phi},\boldsymbol{\psi})) \\ -\gamma\boldsymbol{p}(\boldsymbol{\phi},\boldsymbol{\psi}) - \nabla f(\boldsymbol{q}(\boldsymbol{\phi},\boldsymbol{\psi})) \end{bmatrix}dt + P\begin{bmatrix} \sqrt{2\alpha}I & 0 \\ 0 & \sqrt{2\gamma}I \end{bmatrix}\begin{bmatrix} d\boldsymbol{W} \\ d\boldsymbol{B} \end{bmatrix}$$

and the drift term

$$P\begin{bmatrix} \boldsymbol{p}(\boldsymbol{\phi},\boldsymbol{\psi}) - \alpha\nabla f(\boldsymbol{q}(\boldsymbol{\phi},\boldsymbol{\psi})) \\ -\gamma\boldsymbol{p}(\boldsymbol{\phi},\boldsymbol{\psi}) - \nabla f(\boldsymbol{q}(\boldsymbol{\phi},\boldsymbol{\psi})) \end{bmatrix}$$

is L''-Lipschitz, where $L'' = \kappa' L'$ and κ' is the condition number of P.

Proof: By direct computation and Cauchy-Schwarz inequality, we have

$$
\begin{aligned}
&\left\| \begin{bmatrix} \boldsymbol{p}_1 - \alpha \nabla f(\boldsymbol{q}_1) \\ -\gamma \boldsymbol{p}_1 - \nabla f(\boldsymbol{q}_1) \end{bmatrix} - \begin{bmatrix} \boldsymbol{p}_2 - \alpha \nabla f(\boldsymbol{q}_2) \\ -\gamma \boldsymbol{p}_2 - \nabla f(\boldsymbol{q}_2) \end{bmatrix} \right\| \\
=&\sqrt{\left\| -\alpha \left(\nabla f(\boldsymbol{q}_1) - \nabla f(\boldsymbol{q}_2) \right) + (\boldsymbol{p}_1 - \boldsymbol{p}_2) \right\|^2 + \left\| - \left(\nabla f(\boldsymbol{q}_1) - \nabla f(\boldsymbol{q}_2) \right) - \gamma (\boldsymbol{p}_1 - \boldsymbol{p}_2) \right\|^2} \\
\leq&\sqrt{2\alpha^2 \left\| \nabla f(\boldsymbol{q}_1) - \nabla f(\boldsymbol{q}_2) \right\| + 2 \|\boldsymbol{p}_1 - \boldsymbol{p}_2\|^2 + 2 \left\| \nabla f(\boldsymbol{q}_1) - \nabla f(\boldsymbol{q}_2) \right\| + 2\gamma^2 \|\boldsymbol{p}_1 - \boldsymbol{p}_2\|^2} \\
\leq&\sqrt{(2\alpha^2 L^2 + 2L^2) \|\boldsymbol{q}_1 - \boldsymbol{q}_2\| + (2 + 2\gamma^2) \|\boldsymbol{p}_1 - \boldsymbol{p}_2\|^2} \\
\leq&\sqrt{2} \max\{L\sqrt{1+\alpha^2}, \sqrt{1+\gamma^2}\} \left\| \begin{bmatrix} \boldsymbol{q}_1 - \boldsymbol{q}_2 \\ \boldsymbol{p}_1 - \boldsymbol{p}_2 \end{bmatrix} \right\| \\
\leq&\sqrt{2} \max\{\sqrt{1+\alpha^2} \max\{\frac{1}{\sqrt{2}}, L\}, \sqrt{1+\gamma^2}\} \left\| \begin{bmatrix} \boldsymbol{q}_1 - \boldsymbol{q}_2 \\ \boldsymbol{p}_1 - \boldsymbol{p}_2 \end{bmatrix} \right\| \\
\triangleq&L' \left\| \begin{bmatrix} \boldsymbol{q}_1 - \boldsymbol{q}_2 \\ \boldsymbol{p}_1 - \boldsymbol{p}_2 \end{bmatrix} \right\|
\end{aligned}
$$

By Ito's lemma, we have

$$
\begin{bmatrix} d\boldsymbol{\phi} \\ d\boldsymbol{\psi} \end{bmatrix} = P \begin{bmatrix} \boldsymbol{p}(\boldsymbol{\phi}, \boldsymbol{\psi}) - \alpha \nabla f(\boldsymbol{q}(\boldsymbol{\phi}, \boldsymbol{\psi})) \\ -\gamma \boldsymbol{p}(\boldsymbol{\phi}, \boldsymbol{\psi}) - \nabla f(\boldsymbol{q}(\boldsymbol{\phi}, \boldsymbol{\psi})) \end{bmatrix} dt + P \begin{bmatrix} \sqrt{2\alpha} I & 0 \\ 0 & \sqrt{2\gamma} I \end{bmatrix} \begin{bmatrix} d\boldsymbol{W} \\ d\boldsymbol{B} \end{bmatrix}
$$

Using the Lipschitz constant obtained for the drift of HFHR, we further have

$$
\begin{aligned}
&\left\| P \begin{bmatrix} \boldsymbol{p}(\boldsymbol{\phi}_1, \boldsymbol{\psi}_1) - \alpha \nabla f(\boldsymbol{q}(\boldsymbol{\phi}_1, \boldsymbol{\psi}_1)) \\ -\gamma \boldsymbol{p}(\boldsymbol{\phi}_1, \boldsymbol{\psi}_1) - \nabla f(\boldsymbol{q}(\boldsymbol{\phi}_1, \boldsymbol{\psi}_1)) \end{bmatrix} - P \begin{bmatrix} \boldsymbol{p}(\boldsymbol{\phi}_2, \boldsymbol{\psi}_2) - \alpha \nabla f(\boldsymbol{q}(\boldsymbol{\phi}_2, \boldsymbol{\psi}_2)) \\ -\gamma \boldsymbol{p}(\boldsymbol{\phi}_2, \boldsymbol{\psi}_2) - \nabla f(\boldsymbol{q}(\boldsymbol{\phi}_2, \boldsymbol{\psi}_2)) \end{bmatrix} \right\| \\
\leq & \sigma_{\max} \left\| \begin{bmatrix} \boldsymbol{p}_1 - \alpha \nabla f(\boldsymbol{q}_1) \\ -\gamma \boldsymbol{p}_1 - \nabla f(\boldsymbol{q}_1) \end{bmatrix} - \begin{bmatrix} \boldsymbol{p}_2 - \alpha \nabla f(\boldsymbol{q}_2) \\ -\gamma \boldsymbol{p}_2 - \nabla f(\boldsymbol{q}_2) \end{bmatrix} \right\| \\
\leq & \sigma_{\max} L' \left\| \begin{bmatrix} \boldsymbol{q}_1 - \boldsymbol{q}_2 \\ \boldsymbol{p}_1 - \boldsymbol{p}_2 \end{bmatrix} \right\| \\
\leq & \sigma_{\max} L' \left\| P^{-1} \begin{bmatrix} \boldsymbol{\phi}_1 - \boldsymbol{\phi}_2 \\ \boldsymbol{\psi}_1 - \boldsymbol{\psi}_2 \end{bmatrix} \right\| \\
\leq & \sigma_{\max} L' \frac{1}{\sigma_{\min}} \left\| \begin{bmatrix} \boldsymbol{\phi}_1 - \boldsymbol{\phi}_2 \\ \boldsymbol{\psi}_1 - \boldsymbol{\psi}_2 \end{bmatrix} \right\| \\
= & \kappa' L' \left\| \begin{bmatrix} \boldsymbol{\phi}_1 - \boldsymbol{\phi}_2 \\ \boldsymbol{\psi}_1 - \boldsymbol{\psi}_2 \end{bmatrix} \right\|
\end{aligned}
$$

where $\sigma_{\max}, \sigma_{\min}$ and κ' are the largest, smallest singular values and the condition number (w.r.t. 2-norm) of matrix P. ■

Remark: The following inequalities associated with L' will turn out to be useful in many proofs

$$L' \geq 1,\ L' \geq \sqrt{2}\gamma, L' \geq \sqrt{2}\alpha, L \geq \sqrt{2}L \text{ and } L' \geq \sqrt{2}\alpha L.$$

B.7 Contraction of (Transformed) HFHR Dynamics

Lemma 33 *Suppose f is L-smooth, m-strongly convex and $\gamma^2 > L$. Consider two copies of HFHR dynamics $\begin{bmatrix} \boldsymbol{q}_t \\ \boldsymbol{p}_t \end{bmatrix}$, $\begin{bmatrix} \tilde{\boldsymbol{q}}_t \\ \tilde{\boldsymbol{p}}_t \end{bmatrix}$ (driven by the same Brownian motion) with initialization*

$\begin{bmatrix} \boldsymbol{q}_0 \\ \boldsymbol{p}_0 \end{bmatrix}, \begin{bmatrix} \tilde{\boldsymbol{q}}_0 \\ \tilde{\boldsymbol{p}}_0 \end{bmatrix}$ *respectively, then we have*

$$\left\| P \begin{bmatrix} \boldsymbol{q}_t - \tilde{\boldsymbol{q}}_t \\ \boldsymbol{p}_t - \tilde{\boldsymbol{p}}_t \end{bmatrix} \right\| \leq e^{-\lambda' t} \left\| P \begin{bmatrix} \boldsymbol{q}_0 - \tilde{\boldsymbol{q}}_0 \\ \boldsymbol{p}_0 - \tilde{\boldsymbol{p}}_0 \end{bmatrix} \right\|$$

where $P \triangleq \begin{bmatrix} \gamma I & I \\ 0 & \sqrt{1+\alpha\gamma} I \end{bmatrix}$ *and* $\lambda' = \min\{\frac{m}{\gamma} + \alpha m, \frac{\gamma^2 - L}{\gamma}\}$.

Proof: Consider two copies of HFHR that are driven by the same Brownian motion

$$\begin{cases} d\boldsymbol{q}_t = (\boldsymbol{p}_t - \alpha \nabla f(\boldsymbol{q}_t))dt + \sqrt{2\alpha} d\boldsymbol{B}_t^1 \\ d\boldsymbol{p}_t = (-\gamma \boldsymbol{p}_t - \nabla f(\boldsymbol{q}_t))dt + \sqrt{2\gamma} d\boldsymbol{B}_t^2 \end{cases}, \quad \begin{cases} d\tilde{\boldsymbol{q}}_t = (\tilde{\boldsymbol{p}}_t - \alpha \nabla f(\tilde{\boldsymbol{q}}_t))dt + \sqrt{2\alpha} d\boldsymbol{B}_t^1 \\ d\tilde{\boldsymbol{p}}_t = (-\gamma \tilde{\boldsymbol{p}}_t - \nabla f(\tilde{\boldsymbol{q}}_t))dt + \sqrt{2\gamma} d\boldsymbol{B}_t^2 \end{cases}.$$

The difference of the two copies satisfies the following equation

$$\frac{d}{dt} \begin{bmatrix} \boldsymbol{q}_t - \tilde{\boldsymbol{q}}_t \\ \boldsymbol{p}_t - \tilde{\boldsymbol{p}}_t \end{bmatrix} = - \begin{bmatrix} \alpha H_t & -I \\ H_t & \gamma I \end{bmatrix} \begin{bmatrix} \boldsymbol{q}_t - \tilde{\boldsymbol{q}}_t \\ \boldsymbol{p}_t - \tilde{\boldsymbol{p}}_t \end{bmatrix} \triangleq -A \begin{bmatrix} \boldsymbol{q}_t - \tilde{\boldsymbol{q}}_t \\ \boldsymbol{p}_t - \tilde{\boldsymbol{p}}_t \end{bmatrix}$$

where $H_t = \int_0^1 \nabla^2 f(\tilde{\boldsymbol{q}}_t + s(\boldsymbol{q} - \tilde{\boldsymbol{q}}_t))ds$. Denote the eigenvalues of H_t by $\eta_i, 1 \leq i \leq d$, by strong convexity and smoothness assumption on f, we have $m \leq \eta_i \leq L, 1 \leq i \leq d$.

Denote $\begin{bmatrix}\boldsymbol{\phi}_t\\ \boldsymbol{\psi}_t\end{bmatrix} = P\begin{bmatrix}\boldsymbol{q}_t-\tilde{\boldsymbol{q}}_t\\ \boldsymbol{p}_t-\tilde{\boldsymbol{p}}_t\end{bmatrix}$ and consider $\mathcal{L}_t = \frac{1}{2}\left\|\begin{bmatrix}\boldsymbol{\phi}_t\\ \boldsymbol{\psi}_t\end{bmatrix}\right\|^2$, we have

$$\begin{aligned}
\frac{d}{dt}\mathcal{L}_t &= -\begin{bmatrix}\boldsymbol{\phi}_t\\ \boldsymbol{\psi}_t\end{bmatrix}^T PAP^{-1}\begin{bmatrix}\boldsymbol{\phi}_t\\ \boldsymbol{\psi}_t\end{bmatrix}\\
&= -\begin{bmatrix}\boldsymbol{\phi}_t\\ \boldsymbol{\psi}_t\end{bmatrix}^T \frac{1}{2}(PAP^{-1}+(P^{-1})^TA^TP^T)\begin{bmatrix}\boldsymbol{\phi}_t\\ \boldsymbol{\psi}_t\end{bmatrix}\\
&= -\begin{bmatrix}\boldsymbol{\phi}_t\\ \boldsymbol{\psi}_t\end{bmatrix}^T \frac{1}{\gamma}\begin{bmatrix}(1+\alpha\gamma)H_t & 0_{d\times d}\\ 0_{d\times d} & \gamma^2 I - H_t\end{bmatrix}\begin{bmatrix}\boldsymbol{\phi}_t\\ \boldsymbol{\psi}_t\end{bmatrix}\\
&\triangleq -\begin{bmatrix}\boldsymbol{\phi}_t\\ \boldsymbol{\psi}_t\end{bmatrix}^T B(\alpha)\begin{bmatrix}\boldsymbol{\phi}_t\\ \boldsymbol{\psi}_t\end{bmatrix}
\end{aligned}$$

It is easy to see that

$$\lambda_{\min}(B(\alpha)) = \min_{i=1,2,\cdots,d}\{\min\{\frac{\eta_i}{\gamma}+\alpha\eta_i, \gamma-\frac{\eta_i}{\gamma}\}\} \geq \min\{\frac{m}{\gamma}+\alpha m, \frac{\gamma^2-L}{\gamma}\} \triangleq \lambda'.$$

Therefore we have $\frac{d}{dt}\mathcal{L}_t \leq -2\lambda_{\min}B(\alpha)\mathcal{L}_t \leq -2\lambda'\mathcal{L}_t$. By Gronwall's inequality, we obtain

$$\left\|\begin{bmatrix}\boldsymbol{\phi}_t\\ \boldsymbol{\psi}_t\end{bmatrix}\right\|^2 \leq e^{-2\lambda' t}\left\|\begin{bmatrix}\boldsymbol{\phi}_0\\ \boldsymbol{\psi}_0\end{bmatrix}\right\|^2.$$

and the desired inequality follows by taking square root. ■

B.8 Local error between the exact Strang's splitting method and HFHR dynamics

Lemma 34 *Assume f is L-smooth and $\mathbf{0} \in \operatorname{argmin}_{\boldsymbol{x}\in\mathbb{R}^d} f(\boldsymbol{x})$, i.e. $\nabla f(\mathbf{0}) = \mathbf{0}$. If $0 < h \leq \frac{1}{4L'}$, then compared with the HFHR dynamics, the exact Strang's splitting method has local mathematical expectation of deviation of order $p_1 = 2$ and local mean-squared error*

of order $p_2 = 2$, i.e. there exist constants $\widehat{C}_1, \widehat{C}_2 > 0$ such that

$$\left\|\mathbb{E}\boldsymbol{x}(h) - \mathbb{E}\hat{\boldsymbol{x}}(h)\right\| \leq \widehat{C}_1 h^{p_1}$$

$$\left(\mathbb{E}\left[\left\|\boldsymbol{x}(h) - \hat{\boldsymbol{x}}(h)\right\|^2\right]\right)^{\frac{1}{2}} \leq \widehat{C}_2 h^{p_2}$$

where $\boldsymbol{x}(h) = \begin{bmatrix} \boldsymbol{q}(h) \\ \boldsymbol{p}(h) \end{bmatrix}$ *is the solution of the HFHR dynamics with initial value* $\boldsymbol{x}_0 = \begin{bmatrix} \boldsymbol{q}_0 \\ \boldsymbol{p}_0 \end{bmatrix}$ *and* $\hat{\boldsymbol{x}}(h) = \begin{bmatrix} \hat{\boldsymbol{q}}(h) \\ \hat{\boldsymbol{p}}(h) \end{bmatrix}$ *is the solution of the implementable Strang's splitting with initial value* $\boldsymbol{x}_0 = \begin{bmatrix} \boldsymbol{q}_0 \\ \boldsymbol{p}_0 \end{bmatrix}$, $p_1 = 2$ *and* $p_2 = 2$. *More concretely, we have*

$$\widehat{C}_1 = L\max\{\alpha + 1.25, \gamma + 1\}\left(1.74\|\boldsymbol{x}_0\| + (1.26\sqrt{\alpha} + 2.84\sqrt{\gamma})\sqrt{hd}\right),$$

$$\widehat{C}_2 = L\max\{\alpha + 1.25, \gamma + 1\}\left(1.92\|\boldsymbol{x}_0\| + (1.30\sqrt{\alpha} + 3.22\sqrt{\gamma})\sqrt{hd}\right).$$

Proof: The exact Strang's splitting integrator with step size h reads as $\phi^{\frac{h}{2}} \circ \psi^h \circ \phi^{\frac{h}{2}}$ where

$$\phi: \begin{cases} d\boldsymbol{q} = \boldsymbol{p}dt \\ d\boldsymbol{p} = -\gamma\boldsymbol{p}dt + \sqrt{2\gamma}d\boldsymbol{B} \end{cases} \qquad \psi: \begin{cases} d\boldsymbol{q} = -\alpha\nabla f(\boldsymbol{q})dt + \sqrt{2\alpha}d\boldsymbol{W} \\ d\boldsymbol{p} = -\nabla f(\boldsymbol{q})dt \end{cases}.$$

The ϕ flow can be explicitly solved and the solution is

$$\begin{cases} \boldsymbol{q}(t) = \boldsymbol{q}_0 + \frac{1-e^{-\gamma t}}{\gamma}\boldsymbol{p}_0 + \sqrt{2\gamma}\int_0^t \frac{1-e^{-\gamma(t-s)}}{\gamma}d\boldsymbol{B}(s) \\ \boldsymbol{p}(t) = e^{-\gamma t}\boldsymbol{p}_0 + \sqrt{2\gamma}\int_0^t e^{-\gamma(t-s)}d\boldsymbol{B}(s) \end{cases}.$$

The ψ flow can be written as

$$\begin{cases} \boldsymbol{q}(t) = \boldsymbol{q}_0 - \int_0^t \alpha \nabla f(\boldsymbol{q}(s)) ds + \sqrt{2\alpha} \int_0^t d\boldsymbol{W}(s) \\ \boldsymbol{p}(t) = \boldsymbol{p}_0 - \int_0^t \nabla f(\boldsymbol{q}(s)) ds \end{cases}.$$

The solution of one-step exact Strang's splitting integrator with step size h can be written as

$$\begin{cases} \boldsymbol{q}_3 = \boldsymbol{q}_2(h) + \frac{1-e^{-\gamma\frac{h}{2}}}{\gamma} \boldsymbol{p}_2(h) + \sqrt{2\gamma} \int_{\frac{h}{2}}^h \frac{1-e^{-\gamma(h-s)}}{\gamma} d\boldsymbol{B}(s) \\ \boldsymbol{p}_3 = e^{-\gamma\frac{h}{2}} \boldsymbol{p}_2(h) + \sqrt{2\gamma} \int_{\frac{h}{2}}^h e^{-\gamma(h-s)} d\boldsymbol{B}(s) \\ \boldsymbol{q}_2(r) = \boldsymbol{q}_1 - \int_0^r \alpha \nabla f(\boldsymbol{q}_2(s)) ds + \sqrt{2\alpha} \int_0^r d\boldsymbol{W}(s) \qquad (0 \le r \le h) \\ \boldsymbol{p}_2(r) = \boldsymbol{p}_1 - \int_0^r \nabla f(\boldsymbol{q}_2(s)) ds \\ \boldsymbol{q}_1 = \boldsymbol{q}_0 + \frac{1-e^{-\gamma\frac{h}{2}}}{\gamma} \boldsymbol{p}_0 + \sqrt{2\gamma} \int_0^{\frac{h}{2}} \frac{1-e^{-\gamma(\frac{h}{2}-s)}}{\gamma} d\boldsymbol{B}(s) \\ \boldsymbol{p}_1 = e^{-\gamma\frac{h}{2}} \boldsymbol{p}_0 + \sqrt{2\gamma} \int_0^{\frac{h}{2}} e^{-\gamma(\frac{h}{2}-s)} d\boldsymbol{B}(s) \end{cases}$$

Therefore, we have $\hat{\boldsymbol{q}}(h) = \boldsymbol{q}_3, \hat{\boldsymbol{p}}(h) = \boldsymbol{p}_3$ and

$$
\begin{aligned}
&\hat{\boldsymbol{q}}(h) \\
=&\sqrt{2\gamma}\int_{\frac{h}{2}}^{h}\frac{1-e^{-\gamma(h-s)}}{\gamma}d\boldsymbol{B}(s) + \underbrace{\boldsymbol{q}_1 - \int_0^h \alpha\nabla f(\boldsymbol{q}_2(s))ds + \sqrt{2\alpha}\int_0^h d\boldsymbol{W}(s)}_{\boldsymbol{q}_2(h)} \\
&+ \frac{1-e^{-\gamma\frac{h}{2}}}{\gamma}\left[\underbrace{\boldsymbol{p}_1 - \int_0^h \nabla f(\boldsymbol{q}_2(s))ds}_{\boldsymbol{p}_2(h)}\right] \\
=&\sqrt{2\gamma}\int_{\frac{h}{2}}^{h}\frac{1-e^{-\gamma(h-s)}}{\gamma}d\boldsymbol{B}(s) - \int_0^h \alpha\nabla f(\boldsymbol{q}_2(s))ds + \sqrt{2\alpha}\int_0^h d\boldsymbol{W}(s) \\
&- \frac{1-e^{-\gamma\frac{h}{2}}}{\gamma}\int_0^h \nabla f(\boldsymbol{q}_2(s))ds + \underbrace{\boldsymbol{q}_0 + \frac{1-e^{-\gamma\frac{h}{2}}}{\gamma}\boldsymbol{p}_0 + \sqrt{2\gamma}\int_0^{\frac{h}{2}}\frac{1-e^{-\gamma(\frac{h}{2}-s)}}{\gamma}d\boldsymbol{B}(s)}_{\boldsymbol{q}_1} \\
&+ \frac{1-e^{-\gamma\frac{h}{2}}}{\gamma}\left[\underbrace{e^{-\gamma\frac{h}{2}}\boldsymbol{p}_0 + \sqrt{2\gamma}\int_0^{\frac{h}{2}} e^{-\gamma(\frac{h}{2}-s)}d\boldsymbol{B}(s)}_{\boldsymbol{p}_1}\right] \\
=&\boldsymbol{q}_0 + \frac{1-e^{-\gamma h}}{\gamma}\boldsymbol{p}_0 - \left(\alpha + \frac{1-e^{-\gamma\frac{h}{2}}}{\gamma}\right)\int_0^h \nabla f(\boldsymbol{q}_2(s))ds \\
&+ \sqrt{2\alpha}\int_0^h d\boldsymbol{W}(s) + \sqrt{2\gamma}\int_{\frac{h}{2}}^{h}\frac{1-e^{-\gamma(h-s)}}{\gamma}d\boldsymbol{B}(s) + \sqrt{2\gamma}\int_0^{\frac{h}{2}}\frac{1-e^{-\gamma(\frac{h}{2}-s)}}{\gamma}d\boldsymbol{B}(s) \\
&+ \frac{1-e^{-\gamma\frac{h}{2}}}{\gamma}\sqrt{2\gamma}\int_0^{\frac{h}{2}} e^{-\gamma(\frac{h}{2}-s)}d\boldsymbol{B}(s)
\end{aligned}
$$

$$
\begin{aligned}
\hat{\boldsymbol{p}}(h) =& e^{-\gamma\frac{h}{2}}\left[\underbrace{\boldsymbol{p}_1 - \int_0^h \nabla f(\boldsymbol{q}_2(s))ds}_{\boldsymbol{p}_2(h)}\right] + \sqrt{2\gamma}\int_{\frac{h}{2}}^h e^{-\gamma(h-s)}d\boldsymbol{B}(s) \\
=& e^{-\gamma\frac{h}{2}}\left[\underbrace{e^{-\gamma\frac{h}{2}}\boldsymbol{p}_0 + \sqrt{2\gamma}\int_0^{\frac{h}{2}} e^{-\gamma(\frac{h}{2}-s)}d\boldsymbol{B}(s)}_{\boldsymbol{p}_1}\right] - e^{-\gamma\frac{h}{2}}\int_0^h \nabla f(\boldsymbol{q}_2(s))ds \\
&+ \sqrt{2\gamma}\int_{\frac{h}{2}}^h e^{-\gamma(h-s)}d\boldsymbol{B}(s) \\
=& e^{-\gamma h}\boldsymbol{p}_0 - e^{-\gamma\frac{h}{2}}\int_0^h \nabla f(\boldsymbol{q}_2(s))ds + e^{-\gamma\frac{h}{2}}\sqrt{2\gamma}\int_0^{\frac{h}{2}} e^{-\gamma(\frac{h}{2}-s)}d\boldsymbol{B}(s) \\
&+ \sqrt{2\gamma}\int_{\frac{h}{2}}^h e^{-\gamma(h-s)}d\boldsymbol{B}(s)
\end{aligned}
$$

It is clear that $\hat{\boldsymbol{q}}(h), \hat{\boldsymbol{p}}(h)$ should be compared with the exact solution of HFHR at time h, which can be written as

$$
\begin{aligned}
\boldsymbol{q}(h) =& \boldsymbol{q}_0 + \frac{1-e^{-\gamma h}}{\gamma}\boldsymbol{p}_0 - \int_0^h \left(\frac{1-e^{-\gamma(h-s)}}{\gamma} + \alpha\right)\nabla f(\boldsymbol{q}(s))ds + \sqrt{2\alpha}\int_0^h d\boldsymbol{W}_s \\
&+ \sqrt{2\gamma}\int_0^h \frac{1-e^{-\gamma(h-s)}}{\gamma}d\boldsymbol{B}_s \\
\boldsymbol{p}(h) =& e^{-\gamma h}\boldsymbol{p}_0 - \int_0^h e^{-\gamma(h-s)}\nabla f(\boldsymbol{q}(s))ds + \sqrt{2\gamma}\int_0^h e^{-\gamma(h-s)}d\boldsymbol{B}(s)
\end{aligned}
$$

Subtracting $\boldsymbol{q}(h), \boldsymbol{p}(h)$ from $\hat{\boldsymbol{q}}(h), \hat{\boldsymbol{p}}(h)$ respectively, we obtain

$$
\begin{aligned}
\hat{\boldsymbol{q}}(h) - \boldsymbol{q}(h) =& -\left(\alpha + \frac{1-e^{-\gamma\frac{h}{2}}}{\gamma}\right)\int_0^h \nabla f(\boldsymbol{q}_2(s)) - \nabla f(\boldsymbol{q}(s))ds \\
&+ \int_0^h \left(\frac{1-e^{-\gamma(h-s)}}{\gamma} - \frac{1-e^{-\gamma\frac{h}{2}}}{\gamma}\right)\nabla f(\boldsymbol{q}(s))ds \\
\hat{\boldsymbol{p}}(h) - \boldsymbol{p}(h) =& -e^{-\gamma\frac{h}{2}}\int_0^h \nabla f(\boldsymbol{q}_2(s)) - \nabla f(\boldsymbol{q}(s))ds + \int_0^h \left(e^{-\gamma(h-s)} - e^{-\gamma\frac{h}{2}}\right)\nabla f(\boldsymbol{q}(s))ds
\end{aligned}
$$

It should be clear now that we will need to bound the term $\nabla f(\boldsymbol{q}_2) - \nabla f(\boldsymbol{q})$ and $\nabla f(\boldsymbol{q})$.

Since

$$\begin{aligned}\boldsymbol{q}_2(r) =&\boldsymbol{q}_0 + \frac{1-e^{-\gamma\frac{h}{2}}}{\gamma}\boldsymbol{p}_0 + \sqrt{2\gamma}\int_0^{\frac{h}{2}} \frac{1-e^{-\gamma(\frac{h}{2}-s)}}{\gamma} d\boldsymbol{B}(s) - \alpha\int_0^r \nabla f(\boldsymbol{q}_2(s))ds \\ &+ \sqrt{2\alpha}\int_0^r d\boldsymbol{W}(s) \\ \boldsymbol{q}(r) =&\boldsymbol{q}_0 + \frac{1-e^{-\gamma r}}{\gamma}\boldsymbol{p}_0 - \int_0^r \left(\frac{1-e^{-\gamma(r-s)}}{\gamma} + \alpha\right)\nabla f(\boldsymbol{q}(s))ds + \sqrt{2\alpha}\int_0^r d\boldsymbol{W}(s) \\ &+ \sqrt{2\gamma}\int_0^r \frac{1-e^{-\gamma(r-s)}}{\gamma} d\boldsymbol{B}(s),\end{aligned}$$

we then have

$$\begin{aligned}&\boldsymbol{q}_2(r) - \boldsymbol{q}(r) \\ =&\frac{e^{-\gamma r} - e^{-\gamma\frac{h}{2}}}{\gamma}\boldsymbol{p}_0 - \alpha\int_0^r \nabla f(\boldsymbol{q}_2(s)) - \nabla f(\boldsymbol{q}(s))ds + \int_0^r \frac{1-e^{-\gamma(r-s)}}{\gamma}\nabla f(\boldsymbol{q}(s))ds \\ +&\sqrt{2\gamma}\int_0^{\frac{h}{2}} \frac{1-e^{-\gamma(\frac{h}{2}-s)}}{\gamma} d\boldsymbol{B}(s) - \sqrt{2\gamma}\int_0^r \frac{1-e^{-\gamma(r-s)}}{\gamma} d\boldsymbol{B}(s)\end{aligned}$$

By Lemma 32 and 31, when $0 < h < \frac{1}{4L'}$, we have the following for the solution of HFHR dynamics

$$\mathbb{E}[\left\|\boldsymbol{x}_{0,\boldsymbol{x}_0}(h) - \boldsymbol{x}_0\right\|^2] \leq \widehat{C}_0 h$$

where $\widehat{C}_0 = 5.14\left\{(\alpha+\gamma)d + h\left(L'\right)^2\|\boldsymbol{x}_0\|^2\right\}$ and hence

$$
\begin{aligned}
&\mathbb{E}\left[\int_0^r \left\|\nabla f(\boldsymbol{q}(s))\right\|^2 ds\right] \\
\leq &\mathbb{E}\left[2\int_0^r \left\|\nabla f(\boldsymbol{q}(0))\right\|^2 ds + 2\int_0^r \left\|\nabla f(\boldsymbol{q}(s)) - \nabla f(\boldsymbol{q}(0))\right\|^2 ds\right] \\
\leq &\mathbb{E}\left[2L^2 r\left\|\boldsymbol{q}(0)\right\|^2 + 2L^2\int_0^r \left\|\boldsymbol{q}(s) - \boldsymbol{q}(0)\right\|^2 ds\right] \\
\leq &2L^2 r\|\boldsymbol{x}_0\|^2 + 2L^2\mathbb{E}\left[\int_0^r \left\|\boldsymbol{q}(s) - \boldsymbol{q}(0)\right\|^2 ds\right] \\
\leq &2L^2 r\|\boldsymbol{x}_0\|^2 + 2L^2\widehat{C}_0\int_0^r s ds \\
\leq &L^2 r\left(2\|\boldsymbol{x}_0\|^2 + h\widehat{C}_0\right) \\
\leq &L^2 r\left(2.33\|\boldsymbol{x}_0\|^2 + 5.14(\alpha+\gamma)dh\right)
\end{aligned}
\tag{B.17}
$$

Now $\mathbb{E}\left[\|\boldsymbol{q}_2-\boldsymbol{q}\|^2\right]$ can be bounded as follow

$$
\begin{aligned}
&\mathbb{E}\left[\left\|\boldsymbol{q}_2(r)-\boldsymbol{q}(r)\right\|^2\right]\\
\leq& 5\left\{\left(\frac{e^{-\gamma r}-e^{-\gamma\frac{h}{2}}}{\gamma}\right)^2\|\boldsymbol{p}_0\|^2+\alpha^2\mathbb{E}\left\|\int_0^r\nabla f(\boldsymbol{q}_2(s))-\nabla f(\boldsymbol{q}(s))ds\right\|^2\right\}\\
&+5\mathbb{E}\left\|\int_0^r\frac{1-e^{-\gamma(r-s)}}{\gamma}\nabla f(\boldsymbol{q}(s))ds\right\|^2\\
&+5\left\{2\gamma\mathbb{E}\left\|\int_0^{\frac{h}{2}}\frac{1-e^{-\gamma(\frac{h}{2}-s)}}{\gamma}d\boldsymbol{B}(s)\right\|^2+2\gamma\mathbb{E}\left\|\int_0^r\frac{1-e^{-\gamma(r-s)}}{\gamma}d\boldsymbol{B}(s)\right\|^2\right\}\\
\leq& 5\left\{\frac{h^2}{4}\|\boldsymbol{x}_0\|^2+\alpha^2L^2r\int_0^r\mathbb{E}\left\|\boldsymbol{q}_2(s)-\boldsymbol{q}(s)\right\|^2ds\right\}\\
&+5\int_0^r\left(\frac{1-e^{-\gamma(r-s)}}{\gamma}\right)^2ds\int_0^r\mathbb{E}\left\|\nabla f(\boldsymbol{q}(s))\right\|^2ds+5\left\{\frac{\gamma dh^3}{12}+\frac{2\gamma d}{3}r^3\right\}\\
\leq& 5\left\{\frac{h^2}{4}\|\boldsymbol{x}_0\|^2+\alpha^2L^2r\int_0^r\mathbb{E}\left\|\boldsymbol{q}_2(s)-\boldsymbol{q}(s)\right\|^2ds+\frac{h^3}{3}\mathbb{E}\left[\int_0^r\left\|\nabla f(\boldsymbol{q}(s))\right\|^2\right]+\frac{3\gamma d}{4}h^3\right\}\\
\leq& 5\left\{\frac{h^2}{4}\|\boldsymbol{x}_0\|^2+\frac{3\gamma d}{4}h^3+\frac{h^3}{3}L^2\left(2.33\|\boldsymbol{x}_0\|^2+5.14(\alpha+\gamma)dh\right)r\right\}\\
&+5\alpha^2L^2r\int_0^r\mathbb{E}\left\|\boldsymbol{q}_2(s)-\boldsymbol{q}(s)\right\|^2ds\\
\leq& 5h^2\left\{\frac{1}{4}\|\boldsymbol{x}_0\|^2+\frac{3\gamma d}{4}h+\frac{h^2}{3}L^2\left(2.33\|\boldsymbol{x}_0\|^2+5.14(\alpha+\gamma)dh\right)\right\}\\
&+5\alpha^2L^2h\int_0^r\mathbb{E}\left\|\boldsymbol{q}_2(s)-\boldsymbol{q}(s)\right\|^2ds
\end{aligned}
$$

By Gronwall's inequality and $0 < h \leq \frac{1}{4L'}$, we have

$$\begin{aligned}
&\mathbb{E}\left[\left\|\boldsymbol{q}_2(r) - \boldsymbol{q}(r)\right\|^2\right] && \text{(B.18)}\\
\leq& 5h^2\left\{\frac{1}{4}\|\boldsymbol{x}_0\|^2 + \frac{3\gamma d}{4}h + \frac{h^2}{3}L^2\left(2.33\|\boldsymbol{x}_0\|^2 + 5.14(\alpha+\gamma)dh\right)\right\}\exp\{5\alpha^2L^2h^2\}\\
\leq& 5h^2\left\{\frac{1}{4}\|\boldsymbol{x}_0\|^2 + \frac{3\gamma d}{4}h + \frac{h^2}{3}L^2\left(2.33\|\boldsymbol{x}_0\|^2 + 5.14(\alpha+\gamma)dh\right)\right\}e^{\frac{5}{32}}\\
\leq& 5.85h^2\left\{0.28\|\boldsymbol{x}_0\|^2 + (0.06\alpha + 0.81\gamma)hd\right\}\\
\leq& h^2\left\{1.64\|\boldsymbol{x}_0\|^2 + (0.36\alpha + 4.74\gamma)hd\right\}. && \text{(B.19)}
\end{aligned}$$

With bounds in Equation (B.17) and (B.19), we are now ready to show p_1 and p_2. For p_1,

i.e. the order of the mathematical expectation of deviation, we have

$$
\begin{aligned}
&\left\|\mathbb{E}\left[\begin{bmatrix}\hat{\boldsymbol{q}}(h)\\ \hat{\boldsymbol{p}}(h)\end{bmatrix}-\begin{bmatrix}\boldsymbol{q}(h)\\ \boldsymbol{p}(h)\end{bmatrix}\right]\right\| \\
\leq&\left\|\mathbb{E}\left[\hat{\boldsymbol{q}}(h)-\boldsymbol{q}(h)\right]\right\|+\left\|\mathbb{E}\left[\hat{\boldsymbol{p}}(h)-\boldsymbol{p}(h)\right]\right\| \\
\leq&\left(\alpha+\frac{1-e^{-\gamma\frac{h}{2}}}{\gamma}\right)\left\|\int_0^h\mathbb{E}\left[\nabla f(\boldsymbol{q}_2(s))-\nabla f(\boldsymbol{q}(s))\right]ds\right\| \\
&+\left\|\int_0^h\left(\frac{1-e^{-\gamma(h-s)}}{\gamma}-\frac{1-e^{-\gamma\frac{h}{2}}}{\gamma}\right)\mathbb{E}\left[\nabla f(\boldsymbol{q}(s))\right]ds\right\| \\
&+e^{-\gamma\frac{h}{2}}\left\|\int_0^{\frac{h}{2}}\mathbb{E}\left[\nabla f(\boldsymbol{q}_2(s))-\nabla f(\boldsymbol{q}(s))\right]ds\right\| \\
&+\left\|\int_0^h\left(e^{-\gamma(h-s)}-e^{-\gamma\frac{h}{2}}\right)\mathbb{E}\left[\nabla f(\boldsymbol{q}(s))\right]ds\right\| \\
\leq&\left(\alpha+1+\frac{h}{2}\right)L\int_0^h\mathbb{E}\left\|\boldsymbol{q}_2(s)-\boldsymbol{q}(s)\right\|ds \\
&+\int_0^h\left(\left|\frac{1-e^{-\gamma(h-s)}}{\gamma}-\frac{1-e^{-\gamma\frac{h}{2}}}{\gamma}\right|+\left|e^{-\gamma(h-s)}-e^{-\gamma\frac{h}{2}}\right|\right)\left\|\mathbb{E}\left[\nabla f(\boldsymbol{q}(s))\right]\right\|ds \\
\leq&L\left(\alpha+1+\frac{h}{2}\right)\int_0^h\mathbb{E}\left\|\boldsymbol{q}_2(s)-\boldsymbol{q}(s)\right\|ds \\
&+\left\{\left(\int_0^h\left|\frac{1-e^{-\gamma(h-s)}}{\gamma}-\frac{1-e^{-\gamma\frac{h}{2}}}{\gamma}\right|^2ds\right)^{\frac{1}{2}}+\left(\int_0^h\left|e^{-\gamma(h-s)}-e^{-\gamma\frac{h}{2}}\right|^2ds\right)^{\frac{1}{2}}\right\} \\
&\times\left(\int_0^h\left\|\mathbb{E}\left[\nabla f(\boldsymbol{q}(s))\right]\right\|^2ds\right)^{\frac{1}{2}} \\
\leq&L\left(\alpha+1+\frac{h}{2}\right)\int_0^h\left(\mathbb{E}\left\|\boldsymbol{q}_2(s)-\boldsymbol{q}(s)\right\|^2\right)^{\frac{1}{2}}ds+\frac{1+\gamma}{2\sqrt{3}}h^{\frac{3}{2}}\left(\mathbb{E}\int_0^h\left\|\left[\nabla f(\boldsymbol{q}(s))\right]\right\|^2ds\right)^{\frac{1}{2}} \\
\leq&L\left(\alpha+1+\frac{h}{2}\right)h^2\left\{1.64\|\boldsymbol{x}_0\|^2+(0.36\alpha+4.74\gamma)hd\right\}^{\frac{1}{2}} \\
&+\frac{1+\gamma}{2\sqrt{3}}h^2L\left(2.33\|\boldsymbol{x}_0\|^2+5.14(\alpha+\gamma)dh\right)^{\frac{1}{2}} \\
\leq&Lh^2\max\{\alpha+1.25,\gamma+1\}\left(1.74\|\boldsymbol{x}_0\|+(1.26\sqrt{\alpha}+2.84\sqrt{\gamma})\sqrt{hd}\right)
\end{aligned}
$$

The above derivation proves $p_1 = 2$ with

$$\widehat{C}_1 = L\max\{\alpha + 1.25, \gamma + 1\}\left(1.74\|\boldsymbol{x}_0\| + (1.26\sqrt{\alpha} + 2.84\sqrt{\gamma})\sqrt{hd}\right).$$

We now proceed with p_2, i.e. mean-square error

$$\begin{aligned}
&\mathbb{E}\left\|\begin{bmatrix}\hat{\boldsymbol{q}}(h)\\ \hat{\boldsymbol{p}}(h)\end{bmatrix} - \begin{bmatrix}\boldsymbol{q}(h)\\ \boldsymbol{p}(h)\end{bmatrix}\right\|^2 \\
\leq& 2\left(\alpha + \frac{h}{2}\right)^2 \mathbb{E}\left\|\int_0^h \nabla f(\boldsymbol{q}_2(s)) - \nabla f(\boldsymbol{q}(s))ds\right\|^2 \\
&+ 2\mathbb{E}\left\|\int_0^h \left(\frac{1-e^{-\gamma(h-s)}}{\gamma} - \frac{1-e^{-\gamma\frac{h}{2}}}{\gamma}\right)\nabla f(\boldsymbol{q}(s))ds\right\|^2 \\
&+ 2\mathbb{E}\left\|\int_0^h \nabla f(\boldsymbol{q}_2(s)) - \nabla f(\boldsymbol{q}(s))ds\right\|^2 + 2\mathbb{E}\left\|\int_0^h \left(e^{-\gamma(h-s)} - e^{-\gamma\frac{h}{2}}\right)\nabla f(\boldsymbol{q}(s))ds\right\|^2 \\
\leq& 2\left((\alpha + \frac{h}{2})^2 + 1\right)L^2\mathbb{E}\left(\int_0^h |\boldsymbol{q}_2(s) - \boldsymbol{q}(s)|ds\right)^2 \\
&+ 2\int_0^h \left|\frac{1-e^{-\gamma(h-s)}}{\gamma} - \frac{1-e^{-\gamma\frac{h}{2}}}{\gamma}\right|^2 ds \int_0^h \mathbb{E}\left\|\nabla f(\boldsymbol{q}(s))\right\|^2 ds \\
&+ 2\int_0^h \left|e^{-\gamma(h-s)} - e^{-\gamma\frac{h}{2}}\right|^2 ds \int_0^h \mathbb{E}\left\|\nabla f(\boldsymbol{q}(s))\right\|^2 ds \\
\leq& 2\left((\alpha + \frac{h}{2})^2 + 1\right)L^2 h\int_0^h \mathbb{E}|\boldsymbol{q}_2(s) - \boldsymbol{q}(s)|^2 ds + \frac{1+\gamma^2}{6}h^3\int_0^h \mathbb{E}|\nabla f(\boldsymbol{q}(s))|^2 ds \\
\leq& 2\left((\alpha + \frac{h}{2})^2 + 1\right)L^2\left\{1.64\|\boldsymbol{x}_0\|^2 + (0.36\alpha + 4.74\gamma)hd\right\}h^4 \\
&+ \frac{1+\gamma^2}{6}L^2\left\{2.33\|\boldsymbol{x}_0\|^2 + 5.14(\alpha+\gamma)hd\right\}h^4 \\
\leq& L^2\max\{(\alpha+1.25)^2, 1+\gamma^2\}\left(3.67\|\boldsymbol{x}_0\|^2 + (1.68\alpha + 10.34\gamma)hd\right)h^4
\end{aligned}$$

The above derivation implies $p_2 = 2$ with

$$\widehat{C}_2 = L\max\{\alpha + 1.25, 1 + \gamma\}\left(1.92\|\boldsymbol{x}_0\| + (1.30\sqrt{\alpha} + 3.22\sqrt{\gamma})\sqrt{hd}\right).$$

■

B.9 Local error between HFHR algorithm and the exact Strang's splitting method

Lemma 35 *Assume f is L-smooth, $\mathbf{0} \in \operatorname{argmin}_{\boldsymbol{x}\in\mathbb{R}^d} f(\boldsymbol{x})$, i.e. $\nabla f(\mathbf{0}) = \mathbf{0}$ and the operator $\nabla\Delta f$ grows at most linearly, i.e. $\left\|\nabla\Delta f(\boldsymbol{q})\right\| \leq G\sqrt{1+\|\boldsymbol{q}\|^2}$. If $0 < h \leq \frac{1}{4L'}$, then compared with the exact Strang's splitting method of HFHR dynamics, the implementable Strang's splitting method has local mathematical expectation of deviation of order $p_1 = 2$ and local mean-squared error of order $p_2 = 1.5$, i.e. there exist constants $\bar{C}_1, \bar{C}_2 > 0$ such that*

$$\left\|\mathbb{E}\hat{\boldsymbol{x}}(h) - \mathbb{E}\bar{\boldsymbol{x}}(h)\right\| \leq \bar{C}_1 h^{p_1}$$

$$\left(\mathbb{E}\left[\left\|\hat{\boldsymbol{x}}(h) - \bar{\boldsymbol{x}}(h)\right\|^2\right]\right)^{\frac{1}{2}} \leq \bar{C}_2 h^{p_2}$$

where $\hat{\boldsymbol{x}}(h) = \begin{bmatrix}\hat{\boldsymbol{q}}(h)\\ \hat{\boldsymbol{p}}(h)\end{bmatrix}$ is the solution of the exact Strang's splitting method for HFHR with initial value $\boldsymbol{x}_0 = \begin{bmatrix}\boldsymbol{q}_0\\ \boldsymbol{p}_0\end{bmatrix}$ and $\bar{\boldsymbol{x}}(h) = \begin{bmatrix}\bar{\boldsymbol{q}}(h)\\ \bar{\boldsymbol{p}}(h)\end{bmatrix}$ is the one-step result of Algorithm 2 with initial value $\boldsymbol{x}_0 = \begin{bmatrix}\boldsymbol{q}_0\\ \boldsymbol{p}_0\end{bmatrix}$, $p_1 = 2$ and $p_2 = 1.5$. More concretely, we have

$$\bar{C}_1 = \alpha(\alpha + 1.125)(L+G)\left[0.5 + 0.71\|\boldsymbol{x}_0\| + (1.14\sqrt{\alpha} + 0.21\sqrt{\gamma}h)\sqrt{hd}\right]$$

and

$$\bar{C}_2 = L(\alpha + 0.73)\left(2.30\sqrt{h}\alpha L\|\boldsymbol{x}_0\| + (2.27\sqrt{\alpha} + 0.12\sqrt{\gamma}h)\sqrt{d}\right).$$

Proof: The solution of one-step exact Strang's splitting integrator with step size h can

be written as

$$\begin{cases} \boldsymbol{q}_3 = \boldsymbol{q}_2(h) + \frac{1-e^{-\gamma\frac{h}{2}}}{\gamma}\boldsymbol{p}_2(h) + \sqrt{2\gamma}\int_{\frac{h}{2}}^{h}\frac{1-e^{-\gamma(h-s)}}{\gamma}d\boldsymbol{B}(s) \\ \boldsymbol{p}_3 = e^{-\gamma\frac{h}{2}}\boldsymbol{p}_2(h) + \sqrt{2\gamma}\int_{\frac{h}{2}}^{h}e^{-\gamma(h-s)}d\boldsymbol{B}(s) \\ \boldsymbol{q}_2(r) = \boldsymbol{q}_1 - \int_0^r \alpha\nabla f(\boldsymbol{q}_2(s))ds + \sqrt{2\alpha}\int_0^r d\boldsymbol{W}(s) \qquad (0 \le r \le h) \\ \boldsymbol{p}_2(r) = \boldsymbol{p}_1 - \int_0^r \nabla f(\boldsymbol{q}_2(s))ds \\ \boldsymbol{q}_1 = \boldsymbol{q}_0 + \frac{1-e^{-\gamma\frac{h}{2}}}{\gamma}\boldsymbol{p}_0 + \sqrt{2\gamma}\int_0^{\frac{h}{2}}\frac{1-e^{-\gamma(\frac{h}{2}-s)}}{\gamma}d\boldsymbol{B}(s) \\ \boldsymbol{p}_1 = e^{-\gamma\frac{h}{2}}\boldsymbol{p}_0 + \sqrt{2\gamma}\int_0^{\frac{h}{2}}e^{-\gamma(\frac{h}{2}-s)}d\boldsymbol{B}(s) \end{cases}$$

and the solution of one-step implementable Strang's splitting integrator with step size h can be written as

$$\begin{cases} \bar{\boldsymbol{q}}_3 = \bar{\boldsymbol{q}}_2(h) + \frac{1-e^{-\gamma\frac{h}{2}}}{\gamma}\bar{\boldsymbol{p}}_2(h) + \sqrt{2\gamma}\int_0^{\frac{h}{2}}\frac{1-e^{-\gamma(\frac{h}{2}-s)}}{\gamma}d\boldsymbol{B}(\frac{h}{2}+s) \\ \bar{\boldsymbol{p}}_3 = e^{-\gamma\frac{h}{2}}\bar{\boldsymbol{p}}_2(h) + \sqrt{2\gamma}\int_0^{\frac{h}{2}}e^{-\gamma(\frac{h}{2}-s)}d\boldsymbol{B}(\frac{h}{2}+s) \\ \bar{\boldsymbol{q}}_2(r) = \boldsymbol{q}_1 - \int_0^r \alpha\nabla f(\boldsymbol{q}_1)ds + \sqrt{2\alpha}\int_0^r d\boldsymbol{W}(s) \qquad (0 \le r \le h) \\ \bar{\boldsymbol{p}}_2(r) = \boldsymbol{p}_1 - \int_0^r \nabla f(\boldsymbol{q}_1)ds \\ \boldsymbol{q}_1 = \boldsymbol{q}_0 + \frac{1-e^{-\gamma\frac{h}{2}}}{\gamma}\boldsymbol{p}_0 + \sqrt{2\gamma}\int_0^{\frac{h}{2}}\frac{1-e^{-\gamma(\frac{h}{2}-s)}}{\gamma}d\boldsymbol{B}(s) \\ \boldsymbol{p}_1 = e^{-\gamma\frac{h}{2}}\boldsymbol{p}_0 + \sqrt{2\gamma}\int_0^{\frac{h}{2}}e^{-\gamma(\frac{h}{2}-s)}d\boldsymbol{B}(s) \end{cases}$$

Note that in the implementable Strang's splitting method, ϕ flow can be explicitly integrated and hence $\boldsymbol{q}_1, \boldsymbol{p}_1$ are the same as that in the exact Strang's splitting method.

First, we will bound the deviation of mathematical expectation and mean squared error of $\boldsymbol{q}_2(h) - \bar{\boldsymbol{q}}_2(h)$ and $\boldsymbol{p}_2(h) - \bar{\boldsymbol{p}}_2(h)$. We have

$$\begin{cases} \boldsymbol{q}_2(h) - \bar{\boldsymbol{q}}_2(h) = & -\alpha\int_0^h \nabla f(\boldsymbol{q}_2(s)) - \nabla f(\boldsymbol{q}_1)ds \\ \boldsymbol{p}_2(h) - \bar{\boldsymbol{p}}_2(h) = & -\int_0^h \nabla f(\boldsymbol{q}_2(s)) - \nabla f(\boldsymbol{q}_1)ds \end{cases} \tag{B.20}$$

Square both sides of the first equation in (B.20) and take expectation, we obtain

$$\begin{aligned}
&\mathbb{E}\left\|\boldsymbol{q}_2(h)-\bar{\boldsymbol{q}}_2(h)\right\|^2\\
=&\alpha^2\mathbb{E}\left\|\int_0^h \nabla f(\boldsymbol{q}_2(s))-\nabla f(\boldsymbol{q}_1)ds\right\|^2\\
\leq&\alpha^2\mathbb{E}\left(\int_0^h \left\|\nabla f(\boldsymbol{q}_2(s))-\nabla f(\boldsymbol{q}_1)\right\| ds\right)^2\\
\leq&\alpha^2L^2\mathbb{E}\left(\int_0^h \left\|\boldsymbol{q}_2(s)-\boldsymbol{q}_1\right\| ds\right)^2\\
\leq&\alpha^2L^2h\int_0^h \mathbb{E}\left\|\boldsymbol{q}_2(s)-\boldsymbol{q}_1\right\|^2 ds
\end{aligned}$$

Note that $\boldsymbol{q}_2$ is the solution of a rescaled overdamped Langevin dynamics whose drift vector field is αL-Lipschitz, by conditional expectation version of Lemma 31, for $0<h<\frac{1}{4L'}<\frac{1}{4\alpha L}$, we have $\mathbb{E}\left\|\boldsymbol{q}_2(h)-\boldsymbol{q}_1\right\|^2 \leq \bar{C}_0 h$ with $\bar{C}_0 = 5.14\left\{\alpha d+h(\alpha L)^2\mathbb{E}\|\boldsymbol{q}_1\|^2\right\}$ and it follows that

$$\begin{cases}
\mathbb{E}\left\|\boldsymbol{q}_2(h)-\bar{\boldsymbol{q}}_2(h)\right\|^2 \leq & \alpha^2L^2\bar{C}_0h^3\\
\mathbb{E}\left\|\boldsymbol{p}_2(h)-\bar{\boldsymbol{p}}_2(h)\right\|^2 \leq & L^2\bar{C}_0h^3.
\end{cases}$$

Now consider p_1, i.e., the deviation of mathematical expectation. By Ito's lemma, we have

$$\begin{aligned}
&\boldsymbol{q}_2(h)-\bar{\boldsymbol{q}}_2(h)\\
=&-\alpha\int_0^h \nabla f(\boldsymbol{q}_2(s))-\nabla f(\boldsymbol{q}_1)ds\\
=&-\alpha\int_0^h\left[\int_0^s -\alpha\nabla^2 f(\boldsymbol{q}_2(r))\nabla f(\boldsymbol{q}_2(r))dr+\alpha\int_0^s \nabla\Delta f(\boldsymbol{q}_2(r))dr+\rho\right]ds \qquad \text{(B.21)}
\end{aligned}$$

where ρ is a stochastic integral term. Take expectation and norm for Equation (B.21), we

have

$$
\begin{aligned}
&\left\|\mathbb{E}\left[\boldsymbol{q}_2(h)-\bar{\boldsymbol{q}}_2(h)\right]\right\| \\
=&\alpha^2\left\|\int_0^h \mathbb{E}\left[\int_0^s \nabla^2 f(\boldsymbol{q}_2(r))\nabla f(\boldsymbol{q}_2(r))dr-\int_0^s \nabla\Delta f(\boldsymbol{q}_2(r))dr\right]\,ds\right\| \\
\leq&\alpha^2\int_0^h \mathbb{E}\left[\int_0^s \|\nabla^2 f(\boldsymbol{q}_2(r))\|_2\left\|\nabla f(\boldsymbol{q}_2(r))\right\|\,dr+\int_0^s \left\|\nabla\Delta f(\boldsymbol{q}_2(r))\right\|\,dr\right]\,ds \\
\leq&\alpha^2\int_0^h \mathbb{E}\left[L\int_0^s \left\|\boldsymbol{q}_2(r)\right\|\,dr+\int_0^s G(1+\left\|\boldsymbol{q}_2(r)\right\|)dr\right]\,ds \\
=&\alpha^2(L+G)\int_0^h\int_0^s \mathbb{E}\left\|\boldsymbol{q}_2(r)\right\|\,dr+\alpha^2G\frac{h^2}{2} \\
\leq&\alpha^2(L+G)\int_0^h\int_0^s \mathbb{E}\left\|\boldsymbol{q}_2(r)-\boldsymbol{q}_1\right\|+\mathbb{E}\|\boldsymbol{q}_1\|\,dr+\alpha^2G\frac{h^2}{2} \\
\leq&\alpha^2(L+G)\int_0^h\int_0^s \sqrt{\mathbb{E}\left\|\boldsymbol{q}_2(r)-\boldsymbol{q}_1\right\|^2}+\mathbb{E}\|\boldsymbol{q}_1\|\,dr+\alpha^2G\frac{h^2}{2} \\
\leq&\alpha^2(L+G)\sqrt{\bar{C}_0}h\frac{h^2}{2}+\alpha^2(L+G)\frac{h^2}{2}\mathbb{E}\|\boldsymbol{q}_1\|+\alpha^2G\frac{h^2}{2} \\
\leq&\alpha^2\left\{\frac{\sqrt{\bar{C}_0}h+\mathbb{E}\|\boldsymbol{q}_1\|}{2}(L+G)+\frac{G}{2}\right\}h^2 \\
\leq&\frac{1}{2}\alpha^2(L+G)\left\{\sqrt{\bar{C}_0}h+\mathbb{E}\|\boldsymbol{q}_1\|+1\right\}h^2
\end{aligned}
$$

Similarly, we have $\left\|\mathbb{E}\left[\boldsymbol{p}_2(h)-\bar{\boldsymbol{p}}_2(h)\right]\right\| \leq \frac{1}{2}\alpha(L+G)\left\{\sqrt{\bar{C}_0}h+\mathbb{E}\|\boldsymbol{q}_1\|+1\right\}h^2$.

For p_2, i.e., mean-square error, we have

$$
\begin{aligned}
\mathbb{E}\left\|\boldsymbol{q}_2(h)-\bar{\boldsymbol{q}}_2(h)\right\|^2 \leq&\alpha^2\mathbb{E}\left\{\int_0^h \left\|\nabla f(\boldsymbol{q}_2(s))-\nabla f(\boldsymbol{q}_1)\right\|\,ds\right\}^2 \\
\leq&\alpha^2\mathbb{E}\left\{\int_0^h 1ds\int_0^h \left\|\nabla f(\boldsymbol{q}_2(s))-\nabla f(\boldsymbol{q}_1)\right\|^2\,ds\right\} \\
\leq&\alpha^2L^2h\int_0^h \mathbb{E}\left\|\boldsymbol{q}_2(s)-\boldsymbol{q}_1\right\|^2\,ds \\
\leq&\frac{\alpha^2L^2\bar{C}_0}{2}h^3
\end{aligned}
$$

Similarly we obtain $\mathbb{E}\left\|\boldsymbol{p}_2(h)-\bar{\boldsymbol{p}}_2(h)\right\|^2 \leq \frac{L^2\bar{C}_0}{2}h^3$. Recall

$$\begin{cases}\boldsymbol{q}_3-\bar{\boldsymbol{q}}_3=\boldsymbol{q}_2(h)-\bar{\boldsymbol{q}}_2(h)+\frac{1-e^{-\gamma\frac{h}{2}}}{\gamma}(\boldsymbol{p}_2(h)-\bar{\boldsymbol{p}}_2(h))\\ \boldsymbol{p}_3-\bar{\boldsymbol{p}}_3=e^{-\gamma\frac{h}{2}}(\boldsymbol{p}_2(h)-\bar{\boldsymbol{p}}_2(h))\end{cases}.$$

and it follows that when $0<h\leq\frac{1}{4L'}<1$

$$\left\|\mathbb{E}\begin{bmatrix}\boldsymbol{q}_3-\bar{\boldsymbol{q}}_3\\ \boldsymbol{p}_3-\bar{\boldsymbol{p}}_3\end{bmatrix}\right\| \leq \alpha(\alpha+1+\frac{h}{2})(L+G)\frac{\sqrt{\bar{C}_0h}+\mathbb{E}\|\boldsymbol{q}_1\|+1}{2}h^2 \tag{B.22}$$

$$\mathbb{E}\left\|\begin{bmatrix}\boldsymbol{q}_3-\bar{\boldsymbol{q}}_3\\ \boldsymbol{p}_3-\bar{\boldsymbol{p}}_3\end{bmatrix}\right\|^2 \leq L^2\bar{C}_0\left(\alpha^2+\frac{1}{2}+\frac{h^2}{4}\right)h^3. \tag{B.23}$$

Finally we need to bound $\mathbb{E}\|\boldsymbol{q}_1\|^2$ by $\mathbb{E}\|\boldsymbol{x}_0\|^2$, to this end, we have

$$\begin{aligned}\mathbb{E}\|\boldsymbol{q}_1\|^2 =&\mathbb{E}\left\|\boldsymbol{q}_0+\frac{1-e^{-\gamma\frac{h}{2}}}{\gamma}\boldsymbol{p}_0+\sqrt{2\gamma}\int_0^{\frac{h}{2}}\frac{1-e^{-\gamma(\frac{h}{2}-s)}}{\gamma}d\boldsymbol{B}(s)\right\|^2\\ \leq&(1+\frac{h^2}{4})\mathbb{E}\|\boldsymbol{q}_0\|^2+(1+\frac{h^2}{4})\mathbb{E}\|\boldsymbol{p}_0\|^2+2\gamma d\int_0^{\frac{h}{2}}\left(\frac{1-e^{-\gamma(\frac{h}{2}-s)}}{\gamma}\right)^2ds\\ \leq&(1+\frac{h^2}{4})\mathbb{E}\|\boldsymbol{x}_0\|^2+\frac{\gamma d}{12}h^3 \qquad \text{(B.24)}\\ =&(1+\frac{h^2}{4})\|\boldsymbol{x}_0\|^2+\frac{\gamma d}{12}h^3 \qquad \text{(B.25)}\end{aligned}$$

Collecting all pieces together, including (B.22), (B.23), (B.25), the definition of $\bar{C}_0$ and

$0 < h < \frac{1}{4L'}$, it is not difficult to obtain the following

$$\begin{aligned}\left\|\mathbb{E}\begin{bmatrix}\boldsymbol{q}_3-\bar{\boldsymbol{q}}_3\\ \boldsymbol{p}_3-\bar{\boldsymbol{p}}_3\end{bmatrix}\right\| \leq & \bar{C}_1 h^2\\ \left(\mathbb{E}\left\|\begin{bmatrix}\boldsymbol{q}_3-\bar{\boldsymbol{q}}_3\\ \boldsymbol{p}_3-\bar{\boldsymbol{p}}_3\end{bmatrix}\right\|^2\right)^{\frac{1}{2}} \leq & \bar{C}_2 h^{\frac{3}{2}}\end{aligned}$$

with

$$\bar{C}_1 = \alpha(\alpha+1.125)(L+G)\left[0.5+0.71\|\boldsymbol{x}_0\|+(1.14\sqrt{\alpha}+0.21\sqrt{\gamma}h)\sqrt{hd}\right]$$

and

$$\bar{C}_2 = L(\alpha+0.73)\left(2.30\sqrt{h}\alpha L\|\boldsymbol{x}_0\|+(2.27\sqrt{\alpha}+0.12\sqrt{\gamma}h)\sqrt{d}\right)$$

■

B.10 Local error between HFHR algorithm and HFHR dynamics

Lemma 36 *Assume f is L-smooth, $\mathbf{0} \in \operatorname{argmin}_{\boldsymbol{x}\in\mathbb{R}^d} f(\boldsymbol{x})$, i.e. $\nabla f(\mathbf{0}) = \mathbf{0}$ and the operator $\nabla\Delta f$ grows at most linearly, i.e. $\left\|\nabla\Delta f(\boldsymbol{q})\right\| \leq G\sqrt{1+\|\boldsymbol{q}\|^2}$. If $0 < h \leq \frac{1}{4L'}$, then compared with the HFHR dynamics, the implementable Strang's splitting method has local weak error of order $p_1 = 2$ and local mean-squared error of order $p_2 = 1.5$, i.e. there exist constants $C_1, C_2 > 0$ such that*

$$\left\|\mathbb{E}\boldsymbol{x}(h) - \mathbb{E}\bar{\boldsymbol{x}}(h)\right\| \leq C_1 h^{p_1}$$

$$\left(\mathbb{E}\left[\left\|\boldsymbol{x}(h)-\bar{\boldsymbol{x}}(h)\right\|^2\right]\right)^{\frac{1}{2}} \leq C_2 h^{p_2}$$

where $\boldsymbol{x}(h) = \begin{bmatrix} \boldsymbol{q}(h) \\ \boldsymbol{p}(h) \end{bmatrix}$ *is the solution of HFHR with initial value* $\boldsymbol{x}_0 = \begin{bmatrix} \boldsymbol{q}_0 \\ \boldsymbol{p}_0 \end{bmatrix}$ *and* $\bar{\boldsymbol{x}}(h) = \begin{bmatrix} \bar{\boldsymbol{q}}(h) \\ \bar{\boldsymbol{p}}(h) \end{bmatrix}$ *is the solution of the implementable Strang's splitting with initial value* $\boldsymbol{x}_0 = \begin{bmatrix} \boldsymbol{q}_0 \\ \boldsymbol{p}_0 \end{bmatrix}$, $p_1 = 2$ *and* $p_2 = 1.5$. *More concretely, we have*

$$\begin{aligned} C_1 =& (L+G)\max\{\alpha+1.25, \gamma+1\}\left[0.5\alpha + (1.74+0.71\alpha)\|\boldsymbol{x}_0\|\right] \\ &+ (L+G)\max\{\alpha+1.25, \gamma+1\}\left[\left(1.26\sqrt{\alpha} + 1.14\alpha\sqrt{\alpha} + 2.32\sqrt{\gamma}\right)\sqrt{hd}\right] \end{aligned}$$

and

$$C_2 = L\max\{\alpha+1.25, \gamma+1\}\left[(1.92+2.30\alpha L)\sqrt{h}\|\boldsymbol{x}_0\| + (2.60\sqrt{\alpha} + 3.34\sqrt{\gamma}h)\sqrt{d}\right]$$

Proof: Denote by $\hat{\boldsymbol{x}}(h) = \begin{bmatrix} \hat{\boldsymbol{q}}(h) \\ \hat{\boldsymbol{p}}(h) \end{bmatrix}$ the solution of the exact Strang's splitting method with initial value $\boldsymbol{x}_0 = \begin{bmatrix} \boldsymbol{q}_0 \\ \boldsymbol{p}_0 \end{bmatrix}$. By triangle inequality and Minkowski's inequality, we have

$$\begin{aligned} \left\|\mathbb{E}\boldsymbol{x}(h) - \mathbb{E}\bar{\boldsymbol{x}}(h)\right\| \leq& \left\|\mathbb{E}\boldsymbol{x}(h) - \mathbb{E}\hat{\boldsymbol{x}}(h)\right\| + \left\|\mathbb{E}\hat{\boldsymbol{x}}(h) - \mathbb{E}\bar{\boldsymbol{x}}(h)\right\|, \\ \left(\mathbb{E}\left\|\boldsymbol{x}(h) - \bar{\boldsymbol{x}}(h)\right\|^2\right)^{\frac{1}{2}} \leq& \left(\mathbb{E}\left\|\boldsymbol{x}(h) - \hat{\boldsymbol{x}}(h)\right\|^2\right)^{\frac{1}{2}} + \left(\mathbb{E}\left\|\hat{\boldsymbol{x}}(h) - \bar{\boldsymbol{x}}(h)\right\|^2\right)^{\frac{1}{2}}. \end{aligned}$$

By Lemma 34 and 35, we have

$$\begin{aligned} \left\|\mathbb{E}\boldsymbol{x}(h) - \mathbb{E}\hat{\boldsymbol{x}}(h)\right\| \leq \widehat{C}_1 h^2, \quad \left\|\mathbb{E}\hat{\boldsymbol{x}}(h) - \mathbb{E}\bar{\boldsymbol{x}}(h)\right\| \leq \bar{C}_1 h^2 \\ \left(\mathbb{E}\left\|\boldsymbol{x}(h) - \hat{\boldsymbol{x}}(h)\right\|^2\right)^{\frac{1}{2}} \leq \widehat{C}_2 h^{\frac{3}{2}}, \quad \left(\mathbb{E}\left\|\hat{\boldsymbol{x}}(h) - \bar{\boldsymbol{x}}(h)\right\|^2\right)^{\frac{1}{2}} \leq \bar{C}_2 h^{\frac{3}{2}} \end{aligned}$$

and hence

$$\begin{aligned}\left\|\mathbb{E}\boldsymbol{x}(h)-\mathbb{E}\bar{\boldsymbol{x}}(h)\right\| \leq & (\widehat{C}_1+\bar{C}_1)h^2\\ \left(\mathbb{E}\left\|\boldsymbol{x}(h)-\bar{\boldsymbol{x}}(h)\right\|^2\right)^{\frac{1}{2}} \leq & (\widehat{C}_2+\bar{C}_2)h^{\frac{3}{2}}\end{aligned}$$

with

$$\begin{aligned}&\widehat{C}_1+\bar{C}_1\\ \leq & (L+G)\max\{\alpha+1.25,\gamma+1\}\left[0.5\alpha+(1.74+0.71\alpha)\|\boldsymbol{x}_0\|\right]\\ &+(L+G)\max\{\alpha+1.25,\gamma+1\}\left(1.26\sqrt{\alpha}+1.14\alpha\sqrt{\alpha}+2.32\sqrt{\gamma}\right)\sqrt{hd}\\ \triangleq & C_1\\ &\widehat{C}_2+\bar{C}_2\\ \leq & L\max\{\alpha+1.25,\gamma+1\}\left[(1.92+2.30\alpha L)\sqrt{h}\|\boldsymbol{x}_0\|+(2.60\sqrt{\alpha}+3.34\sqrt{\gamma}h)\sqrt{d}\right]\\ \triangleq & C_2\end{aligned}$$

■

B.11 α does create acceleration even after discretization

If $\alpha \to \infty$ while γ remains fixed, then $dq = -\alpha\nabla f(q) + \sqrt{2\alpha}dW$ is the dominant part of the dynamics, and in this case the role of α could be intuitively understood as to simply rescale the time of gradient flow, which does not create any algorithmic advantage, as the timestep of discretization has to scale like $1/\alpha$ in this case. However, finite α no longer corresponds to solely a time-scaling, but closely couples with the dynamics and creates acceleration. This is true even after the continuous dynamics is discretized by an algorithm

We will analytically illustrate this point by considering quadratic f. In this case, the diffusion process remains Gaussian, and it suffices to quantify the convergence of its mean and covariance. In fact, it can be shown that both have the same speed of convergence, and

therefore for simplicity we will only consider the mean process. Two demonstrations (with different focuses) will be provided.

Demonstration 1 (1D, γ given; infinite acceleration). Consider $f(x) = x^2/2$, γ fixed. The mean process is

$$\begin{cases} \dot{q} &= p - \alpha q \\ \dot{p} &= -q - \gamma p \end{cases}$$

Consider, for simplicity, an Euler-Maruyama discretization of the HFHR dynamics, which coressponds to a Forward Euler discretization of the mean process (other numerical methods can be analyzed analogously):

$$\begin{bmatrix} q_{k+1} \\ p_{k+1} \end{bmatrix} = A \begin{bmatrix} q_k \\ p_k \end{bmatrix}, \qquad A = \begin{bmatrix} 1 - \alpha h & h \\ -h & 1 - \gamma h \end{bmatrix}.$$

We will show that, unless $\gamma = 2$, an appropriately chosen α will converge infinitely faster than the case with $\alpha = 0$, if both cases use the optimal h.

To do so, let us compute A's eigenvalues, which are

$$\frac{1}{2}\left(2 - (\alpha + \gamma)h \pm h\sqrt{-4 + (\alpha - \gamma)^2}\right)$$

Consider the case where $|\alpha - \gamma| \leq 2$, then the eigenvalues are a pair of complex conjugates. Their modulus determines the speed of convergence, and it can be computed to be

$$\frac{1}{2}\sqrt{(2 - (\alpha + \gamma)h)^2 + h^2(4 - (\alpha - \gamma)^2)} = \sqrt{1 - (\alpha + \gamma)h + (1 + \alpha\gamma)h^2}$$

Minimizing the quadratic function gives the optimal h that ensures the fastest speed of

convergence, and the optimal h is

$$h = \frac{\alpha + \gamma}{2(1 + \alpha\gamma)}$$

and the optimal spectral radius is

$$\sqrt{1 - \frac{(\alpha + \gamma)^2}{4(1 + \alpha\gamma)}}.$$

When one uses low-resolution ODE, in which $\alpha = 0$, the optimal rate is $1 - \gamma^2/4$ (note it is not surprising that the critically damped case, i.e., $\gamma = 2$, will give the fastest convergence).

If $\gamma \neq 2$, the additional introduction of α can accelerate the convergence by reducing the spectral radius. For instance, if $\alpha = \gamma + 2$, upon choosing the optimal $h = \frac{1}{1+\gamma}$, the optimal spectral radius is 0 (note in this case A actually has Jordan canonical form of $\begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}$ and thus the discretization converges in 2 steps instead of 1, irrespective of the initial condition).

Demonstration 2 (multi-dim, γ, α and h all to be chosen; acceleration quantified in terms of condition number). Consider quadratic f with positive definite Hessian, whose eigenvalues are $1 = \lambda_1 < \cdots < \lambda_n = \epsilon^{-1}$ for some $0 < \epsilon \ll 1$. Assume without loss of generality that $f = q_1^2/2 + \epsilon^{-1} q_2^2/2$. Similar to Demonstration 1, the forward Euler discretization of the mean process is

$$\begin{bmatrix} q_{1,k+1} \\ p_{1,k+1} \\ q_{2,k+1} \\ p_{2,k+1} \end{bmatrix} = \begin{bmatrix} A_1 & 0 \\ 0 & A_2 \end{bmatrix} \begin{bmatrix} q_{1,k} \\ p_{1,k} \\ q_{2,k} \\ p_{2,k} \end{bmatrix}, \quad A_1 = \begin{bmatrix} 1 - \alpha h & h \\ -h & 1 - \gamma h \end{bmatrix}, \quad A_2 = \begin{bmatrix} 1 - \alpha\epsilon^{-1} h & h \\ -\epsilon^{-1} h & 1 - \gamma h \end{bmatrix} \tag{B.26}$$

We will (i) find h and γ that lead to fastest convergence of the ULD discretization, i.e. the above iteration with $\alpha = 0$, and then (ii) constructively show the existence of h, γ and α that lead to faster convergence than the optimal one in (i) — note these may not even be the optimal choices for HFHR, but they already lead to significant acceleration. More specifically,

(i) In a ULD setup, $\alpha = 0$. It can be computed that the eigenvalues of A_1 and A_2 are respectively

$$\frac{1}{2}\left(2 - h\gamma \pm h\sqrt{-4+\gamma^2}\right) \qquad \text{and} \qquad \frac{1}{2}\left(2 - h\gamma \pm h\sqrt{-4\epsilon^{-1}+\gamma^2}\right)$$

We now seek $\gamma > 0, h > 0$ to minimize the maximum of their norms for obtaining the optimal convergence rate. This is done in cases.

Case (i1) When $\gamma \leq 2$, both A_1 and A_2 eigenvalues are complex conjugate pairs. To minimize the maximum of their norms, let's first see if their norms could be made equal.

A_1 eigenvalue's norm squared *4 is

$$(2 - h\gamma)^2 - h^2(-4+\gamma^2) = 4(h - \gamma/2)^2 + 4 - \gamma^2 \tag{B.27}$$

A_2 eigenvalue's norm squared *4 is

$$(2 - h\gamma)^2 - h^2(-4\epsilon^{-1}+\gamma^2) = 4\epsilon^{-1}(h - \epsilon\gamma/2)^2 + 4 - \epsilon\gamma^2 \tag{B.28}$$

It can be seen that for (B.27) is always strictly smaller than (B.28) for any $h > 0$. Therefore, the max of the two is minimized when $h = \epsilon\gamma/2$, and the corresponding max value is $4 - \epsilon\gamma^2$. γ that minimizes this max value is $\gamma = 2$. Corresponding rate of convergence is

$$\sqrt{1-\epsilon}.$$

Case (i2) When $\gamma \geq 2\epsilon^{-1/2}$, both A_1 and A_2 eigenvalues are real. Since $\epsilon \ll 1$, we can order them*2 as

$$\begin{aligned} 2 - h\gamma - h\sqrt{-4+\gamma^2} <& 2 - h\gamma - h\sqrt{-4\epsilon^{-1}+\gamma^2} < 2 - h\gamma + h\sqrt{-4\epsilon^{-1}+\gamma^2} \\ <& 2 - h\gamma + h\sqrt{-4+\gamma^2} < 2. \end{aligned}$$

To minimize the max of their norms, consider cases in which the smallest of four is negative, in which case at optimum one should have

$$-(2 - h\gamma - h\sqrt{-4+\gamma^2}) = 2 - h\gamma + h\sqrt{-4+\gamma^2}.$$

This gives $h = 2/\gamma$ (which does verify the assumption that the smallest of four is negative). Corresponding max of their norms is thus $\sqrt{1-4/\gamma^2}$. γ that minimizes this max value is $\gamma = 2\epsilon^{-1/2}$, which gives rate of convergence of

$$\sqrt{1-\epsilon}.$$

Case (i3) When $2 \leq \gamma \leq 2\epsilon^{-1/2}$, A_1 eigenvalues are real and A_2 eigenvalues are complex conjugates. Again, the max of their norms is minimized if the norms can be made all equal.

Note A_1 eigenvalues cannot be of the same sign, because otherwise $2-h\gamma-h\sqrt{-4+\gamma^2} = 2 - h\gamma + h\sqrt{-4+\gamma^2}$, which means either $h = 0$ or $\gamma = 2$, but if $\gamma = 2$ then $2 - h\gamma + h\sqrt{-4+\gamma^2}$ being equal to 2*norm of A_2 eigenvalue, which is $\sqrt{4\epsilon^{-1}(h-\epsilon\gamma/2)^2 + 4 - \epsilon\gamma^2}$, leads to $h = 0$ again.

Therefore, the equality of norms of A_1, A_2 eigenvalues means

$$-(2 - h\gamma - h\sqrt{-4+\gamma^2}) = 2 - h\gamma + h\sqrt{-4+\gamma^2} = \sqrt{4\epsilon^{-1}(h-\epsilon\gamma/2)^2 + 4 - \epsilon\gamma^2}.$$

The first equality gives $h\gamma = 2$, which, together with the second equality, gives $h =$

$\pm\sqrt{\frac{2\epsilon}{1+\epsilon}}$. Selecting the positive value of optimal h, we also obtain optimal $\gamma = \sqrt{2(1+\epsilon)}\epsilon^{-1/2}$, which is $\leq 2\epsilon^{-1/2}$ and thus satisfying our assumption ($2 \leq \gamma \leq 2\epsilon^{-1/2}$). The corresponding rate of convergence is thus

$$\frac{1}{2}\left(2 - h\gamma + h\sqrt{-4+\gamma^2}\right) = \sqrt{\frac{1-\epsilon}{1+\epsilon}}.$$

Summary of (i) Since $\sqrt{\frac{1-\epsilon}{1+\epsilon}} < \sqrt{1-\epsilon}$, the ULD Euler-Maruyama discretization converges the fastest when

$$h = \sqrt{\frac{2\epsilon}{1+\epsilon}}, \qquad \gamma = \sqrt{2(1+\epsilon)}\epsilon^{-1/2},$$

and the corresponding discount factor of convergence is

$$\sqrt{\frac{1-\epsilon}{1+\epsilon}}, \qquad \text{where } \epsilon = 1/\kappa \text{ with } \kappa \text{ being Hessian's condition number.} \tag{B.29}$$

(ii) Now consider the HFHR setup. Let's first state a result: when

$$\gamma = \frac{\sqrt{4c^2\epsilon^4 + 8c^2\epsilon^3 + 4c^2\epsilon^2 + \epsilon^2 - 2\epsilon + 1} + \epsilon + 3}{2c\epsilon^2 + 2c\epsilon} > 0, \tag{B.30}$$

$$\alpha = \frac{-\sqrt{4c^2\epsilon^4 + 8c^2\epsilon^3 + 4c^2\epsilon^2 + \epsilon^2 - 2\epsilon + 1} + 3\epsilon + 1}{2c\epsilon^2 + 2c\epsilon} > 0, \qquad h = c\epsilon \tag{B.31}$$

for any $c > 0$ independent of ϵ, the iteration (B.26) converges with discount factor

$$\frac{1}{\sqrt{2}(1+\epsilon)}\sqrt{(1-\epsilon)\left(1 - \epsilon + \sqrt{4c^2\epsilon^4 + 8c^2\epsilon^3 + (4c^2+1)\epsilon^2 - 2\epsilon + 1}\right)}. \tag{B.32}$$

While the exact expression is lengthy, it can proved that the HFHR non-optimal discount factor (B.32) is strictly smaller than the ULD optimal discount factor B.29 for not only small but also large ϵ's.

For some quantitative intuition, discount factors Taylor expanded in ϵ are respectively

$$\text{HFHR non-optimal:} \qquad 1 - 2\epsilon + \left(\frac{c^2}{2} + 2\right)\epsilon^2 + \mathcal{O}\left(\epsilon^3\right) \tag{B.33}$$

$$\text{ULD optimal:} \qquad 1 - \epsilon + \frac{\epsilon^2}{2} + \mathcal{O}\left(\epsilon^3\right) \tag{B.34}$$

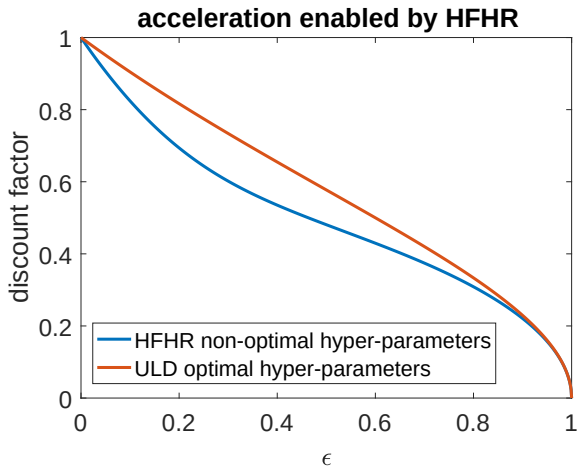


Figure B.1: Acceleration of HFHR algorithm over ULD algorithm (despite of an additional constraint α may place on h) for multi-dimensional quadratic objectives. $1/\epsilon$ is the condition number.

The exact expressions of discount factors are also plotted in Fig.B.1 ($c = 1$ was arbitrarily chosen) and one can see acceleration for any (not necessarily small) ϵ.

(ii details) How were values in (B.31) chosen? Following the idea detailed in (i), we consider a case where A_1 eigenvalues are both real, A_2 eigenvalues are complex conjugates, and all their norms are equal. Note there are 3 more cases, namely real/real, complex/real, and complex/complex, but we do not optimize over all cases for simplicity — the real/complex case is enough for outperforming the optimal ULD.

This case leads to at least the following equations

$$\begin{cases} \text{tr}A_1 & = 0 \\ \det A_1 + \det A_2 & = 0 \end{cases} \tag{B.35}$$

One can solve this system of equations to obtain α and γ as functions of h. Following the idea of choosing h small enough to resolve the stiffness of the ODE

$$\begin{cases} \dot{q}_2 & = p_2 - \alpha\epsilon^{-1}q_2 \\ \dot{p}_2 & = -\epsilon^{-1}q_2 - \gamma p_2 \end{cases},$$

pick $h = c\epsilon$. Then (B.35) gives

$$\gamma = \frac{\sqrt{4c^2\epsilon^4 + 8c^2\epsilon^3 + 4c^2\epsilon^2 + \epsilon^2 - 2\epsilon + 1} + \epsilon + 3}{2c\epsilon^2 + 2c\epsilon}$$
$$\alpha = \frac{-\sqrt{4c^2\epsilon^4 + 8c^2\epsilon^3 + 4c^2\epsilon^2 + \epsilon^2 - 2\epsilon + 1} + 3\epsilon + 1}{2c\epsilon^2 + 2c\epsilon}$$

or

$$\gamma = \frac{-\sqrt{4c^2\epsilon^4 + 8c^2\epsilon^3 + 4c^2\epsilon^2 + \epsilon^2 - 2\epsilon + 1} + \epsilon + 3}{2c\epsilon^2 + 2c\epsilon}$$
$$\alpha = \frac{\sqrt{4c^2\epsilon^4 + 8c^2\epsilon^3 + 4c^2\epsilon^2 + \epsilon^2 - 2\epsilon + 1} + 3\epsilon + 1}{2c\epsilon^2 + 2c\epsilon}$$

The former is our choice (B.31) because it can be checked that the latter leads to $\det A_1 > 0$ which violates the assumption of a pair of plus and minus real eigenvalues.

It is possible to find optimal α, γ, h for HFHR. One has to minimize $\det A_2$ under the constraint $\det A_2 > 0$ in addition to (B.35). And then do similar calculations for the other 3 cases, and then finally the best among the 4 cases. We chose not to carry out all the details in this paper.

REFERENCES

[1] P. M. Lee, *Bayesian statistics*. Oxford University Press London: 1989.

[2] J.-M. Marin and C. Robert, *Bayesian core: a practical approach to computational Bayesian statistics*. Springer Science & Business Media, 2007.

[3] A. Gelman, J. B. Carlin, H. S. Stern, D. B. Dunson, A. Vehtari, and D. B. Rubin, *Bayesian data analysis*. CRC press, 2013.

[4] M. Kardar, *Statistical physics of particles*. Cambridge University Press, 2007.

[5] J. M. Haile, I. Johnston, A. J. Mallinckrodt, and S. McKay, "Molecular dynamics simulation: Elementary methods," *Computers in Physics*, vol. 7, no. 6, pp. 625–625, 1993.

[6] C. Andrieu, N. De Freitas, A. Doucet, and M. I. Jordan, "An introduction to mcmc for machine learning," *Machine learning*, vol. 50, no. 1, pp. 5–43, 2003.

[7] C. M. Bishop, *Pattern recognition and machine learning*. springer, 2006.

[8] B. Ballnus, S. Hug, K. Hatz, L. Görlitz, J. Hasenauer, and F. J. Theis, "Comprehensive benchmarking of markov chain monte carlo methods for dynamical systems," *BMC systems biology*, vol. 11, no. 1, pp. 1–18, 2017.

[9] S. Brooks, A. Gelman, G. Jones, and X.-L. Meng, *Handbook of markov chain monte carlo*. CRC press, 2011.

[10] G. A. Pavliotis, *Stochastic processes and applications: diffusion processes, the Fokker-Planck and Langevin equations*. Springer, 2014, vol. 60.

[11] E. Nelson, "Dynamical theories of brownian motion," 1967.

[12] G. O. Roberts, R. L. Tweedie, *et al.*, "Exponential convergence of langevin distributions and their discrete approximations," *Bernoulli*, vol. 2, no. 4, pp. 341–363, 1996.

[13] S. F. Jarner and E. Hansen, "Geometric ergodicity of metropolis algorithms," *Stochastic processes and their applications*, vol. 85, no. 2, pp. 341–361, 2000.

[14] G. O. Roberts, J. S. Rosenthal, *et al.*, "General state space markov chains and mcmc algorithms," *Probability surveys*, vol. 1, pp. 20–71, 2004.

[15] G. O. Roberts and J. S. Rosenthal, "Optimal scaling of discrete approximations to langevin diffusions," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 60, no. 1, pp. 255–268, 1998.

[16] O. Stramer and R. Tweedie, "Langevin-type models i: Diffusions with given stationary distributions and their discretizations," *Methodology and Computing in Applied Probability*, vol. 1, no. 3, pp. 283–306, 1999.

[17] M. Welling and Y. W. Teh, "Bayesian learning via stochastic gradient langevin dynamics," *International Conference on Machine Learning*, pp. 681–688, 2011.

[18] T. Chen, E. B. Fox, and C. Guestrin, "Stochastic Gradient Hamiltonian Monte Carlo," *International Conference on Machine Learning*, pp. 1683–1691, 2014.

[19] S. Ahn, A. Korattikara, and M. Welling, "Bayesian posterior sampling via stochastic gradient fisher scoring," in *29th International Conference on Machine Learning, ICML 2012*, 2012, pp. 1591–1598.

[20] S. Patterson and Y. W. Teh, "Stochastic gradient Riemannian Langevin dynamics on the probability simplex," *Advances in Neural Information Processing Systems*, pp. 3102–3110, 2013.

[21] Y.-A. Ma, T. Chen, and E. Fox, "A complete recipe for stochastic gradient mcmc," in *Advances in Neural Information Processing Systems*, 2015, pp. 2917–2925.

[22] C. Chen, N. Ding, and L. Carin, "On the convergence of stochastic gradient mcmc algorithms with high-order integrators," in *Advances in Neural Information Processing Systems*, 2015, pp. 2278–2286.

[23] Y. W. Teh, A. H. Thiery, and S. J. Vollmer, "Consistency and fluctuations for stochastic gradient langevin dynamics," *The Journal of Machine Learning Research*, vol. 17, no. 1, pp. 193–225, 2016.

[24] C. Li, C. Chen, D. Carlson, and L. Carin, "Preconditioned stochastic gradient langevin dynamics for deep neural networks," in *Thirtieth AAAI Conference on Artificial Intelligence*, 2016.

[25] S. J. Vollmer, K. C. Zygalakis, and Y. W. Teh, "Exploration of the (non-) asymptotic bias and variance of stochastic gradient langevin dynamics," *The Journal of Machine Learning Research*, vol. 17, no. 1, pp. 5504–5548, 2016.

[26] A. S. Dalalyan, "Theoretical guarantees for approximate sampling from smooth and log-concave densities," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 79, no. 3, pp. 651–676, 2017.

[27] ——, "Further and stronger analogy between sampling and optimization: Langevin monte carlo and gradient descent," *conference on learning theory*, pp. 678–689, 2017.

[28] X. Cheng, N. S. Chatterji, P. L. Bartlett, and M. I. Jordan, "Underdamped langevin mcmc: A non-asymptotic analysis," *Proceedings of the 31st Conference On Learning Theory, PMLR*, 2018.

[29] X. Cheng and P. L. Bartlett, "Convergence of langevin mcmc in kl-divergence," *PMLR 83*, no. 83, pp. 186–211, 2018.

[30] 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," *arXiv preprint arXiv:1805.01648*, 2018.

[31] A. S. Dalalyan and L. Riou-Durand, "On sampling from a log-concave density using kinetic Langevin diffusions," *Bernoulli*, vol. 26, no. 3, pp. 1956–1988, 2020.

[32] A. Durmus and E. Moulines, "Sampling from strongly log-concave distributions with the unadjusted langevin algorithm," *arXiv preprint arXiv:1605.01559*, vol. 5, 2016.

[33] A. Durmus, E. Moulines, *et al.*, "High-dimensional bayesian inference via the unadjusted langevin algorithm," *Bernoulli*, vol. 25, no. 4A, pp. 2854–2882, 2019.

[34] A. Eberle, A. Guillin, R. Zimmer, *et al.*, "Couplings and quantitative contraction rates for langevin dynamics," *The Annals of Probability*, vol. 47, no. 4, pp. 1982–2010, 2019.

[35] Y.-A. Ma, N. Chatterji, X. Cheng, N. Flammarion, P. Bartlett, and M. I. Jordan, "Is there an analog of nesterov acceleration for mcmc?" *arXiv preprint arXiv:1902.00996*, 2019.

[36] R. Shen and Y. T. Lee, "The randomized midpoint method for log-concave sampling," in *Advances in Neural Information Processing Systems*, 2019, pp. 2098–2109.

[37] S. Vempala and A. Wibisono, "Rapid convergence of the unadjusted langevin algorithm: Isoperimetry suffices," in *Advances in Neural Information Processing Systems*, 2019, pp. 8092–8104.

[38] H. Robbins and S. Monro, "A stochastic approximation method," *The annals of mathematical statistics*, pp. 400–407, 1951.

[39] J. C. Spall, *Introduction to stochastic search and optimization: estimation, simulation, and control*. John Wiley & Sons, 2005, vol. 65.

[40] L. Bottou, "Stochastic learning," in *Advanced Lectures on Machine Learning*, ser. Lecture Notes in Artificial Intelligence, LNAI 3176, O. Bousquet and U. von Luxburg, Eds., Berlin: Springer Verlag, 2004, pp. 146–168.

[41] D. Maclaurin and R. P. Adams, "Firefly monte carlo: Exact mcmc with subsets of data," in *Twenty-Fourth International Joint Conference on Artificial Intelligence*, 2015.

[42] T. Fu and Z. Zhang, "Cpsg-mcmc: Clustering-based preprocessing method for stochastic gradient mcmc," in *Artificial Intelligence and Statistics*, 2017, pp. 841–850.

[43] N. Bou-Rabee and J. M. Sanz-Serna, "Geometric integrators and the Hamiltonian Monte Carlo method," *Acta Numerica*, vol. 27, pp. 113–206, 2018.

[44] N. Bou-Rabee, A. Eberle, and R. Zimmer, "Coupling and convergence for Hamiltonian Monte Carlo," *arXiv preprint arXiv:1805.00452*, 2018.

[45] F. Bach, "Stochastic gradient methods for machine learning," Tech. Rep., 2013.

[46] R. Johnson and T. Zhang, "Accelerating stochastic gradient descent using predictive variance reduction," in *Advances in neural information processing systems*, 2013, pp. 315–323.

[47] M. Schmidt, N. Le Roux, and F. Bach, "Minimizing finite sums with the stochastic average gradient," *Mathematical Programming*, vol. 162, no. 1-2, pp. 83–112, 2017.

[48] A. Defazio, F. Bach, and S. Lacoste-Julien, "Saga: A fast incremental gradient method with support for non-strongly convex composite objectives," in *Advances in neural information processing systems*, 2014, pp. 1646–1654.

[49] K. A. Dubey, S. J. Reddi, S. A. Williamson, B. Poczos, A. J. Smola, and E. P. Xing, "Variance reduction in stochastic gradient langevin dynamics," in *Advances in neural information processing systems*, 2016, pp. 1154–1162.

[50] J. Baker, P. Fearnhead, E. B. Fox, and C. Nemeth, "Control variates for stochastic gradient mcmc," *Statistics and Computing*, vol. 29, no. 3, pp. 599–615, 2019.

[51] N. S. Chatterji, N. Flammarion, Y.-A. Ma, P. L. Bartlett, and M. I. Jordan, "On the theory of variance reduction for stochastic gradient monte carlo," *ICML*, 2018.

[52] D. Needell, R. Ward, and N. Srebro, "Stochastic gradient descent, weighted sampling, and the randomized kaczmarz algorithm," in *Advances in Neural Information Processing Systems*, 2014, pp. 1017–1025.

[53] M. Schmidt, R. Babanezhad, M. Ahmed, A. Defazio, A. Clifton, and A. Sarkar, "Non-uniform stochastic average gradient method for training conditional random fields," in *artificial intelligence and statistics*, 2015, pp. 819–828.

[54] D. Csiba and P. Richtárik, "Importance sampling for minibatches," *The Journal of Machine Learning Research*, vol. 19, no. 1, pp. 962–982, 2018.

[55] P. Zhao and T. Zhang, "Stochastic optimization with importance sampling for regularized loss minimization," in *International Conference on Machine Learning*, 2015, pp. 1–9.

[56] R. Zhu, "Gradient-based sampling: An adaptive importance sampling for least-squares," in *Advances in Neural Information Processing Systems*, 2016, pp. 406–414.

[57] A. Korattikara, Y. Chen, and M. Welling, "Austerity in mcmc land: Cutting the metropolis-hastings budget," in *International Conference on Machine Learning*, PMLR, 2014, pp. 181–189.

[58] R. Bardenet, A. Doucet, and C. C. Holmes, "On markov chain monte carlo methods for tall data," *Journal of Machine Learning Research*, vol. 18, no. 47, 2017.

[59] R. Zhang and C. De Sa, "Poisson-minibatching for gibbs sampling with convergence rate guarantees," *arXiv preprint arXiv:1911.09771*, 2019.

[60] R. Zhang, A. F. Cooper, and C. De Sa, "Asymptotically optimal exact minibatch metropolis-hastings," *arXiv preprint arXiv:2006.11677*, 2020.

[61] R. Kubo, "The fluctuation-dissipation theorem," *Reports on progress in physics*, vol. 29, no. 1, p. 255, 1966.

[62] M. Tao and T. Ohsawa, "Variational optimization on lie groups, with examples of leading (generalized) eigenvalue problems," *AISTATS*, 2020.

[63] N. Bou-Rabee and H. Owhadi, "Long-run accuracy of variational integrators in the stochastic context," *SIAM Journal on Numerical Analysis*, vol. 48, no. 1, pp. 278–297, 2010.

[64] J. C. Mattingly, A. M. Stuart, and M. V. Tretyakov, "Convergence of numerical time-averaging and stationary measures via poisson equations," *SIAM Journal on Numerical Analysis*, vol. 48, no. 2, pp. 552–577, 2010.

[65] A. S. Dalalyan and A. G. Karagulyan, "User-friendly guarantees for the langevin monte carlo with inaccurate gradient," *arXiv preprint arXiv:1710.00095*, 2017.

[66] A. Defazio and L. Bottou, "On the ineffectiveness of variance reduced optimization for deep learning," in *Advances in Neural Information Processing Systems*, 2019, pp. 1755–1765.

[67] A. G. Wilson, "The case for bayesian deep learning," *arXiv preprint arXiv:2001.10995*, 2020.

[68] K. Jarrett, K. Kavukcuoglu, M. Ranzato, and Y. LeCun, "What is the best multi-stage architecture for object recognition?" In *2009 IEEE 12th international conference on computer vision*, IEEE, 2009, pp. 2146–2153.

[69] Y. Nesterov, "A method for unconstrained convex minimization problem with the rate of convergence o (1/k$\hat{2}$)," in *Doklady AN USSR*, vol. 269, 1983, pp. 543–547.

[70] ——, *Introductory lectures on convex optimization: A basic course*. Springer Science & Business Media, 2013, vol. 87.

[71] W. Su, S. Boyd, and E. Candes, "A differential equation for modeling nesterov's accelerated gradient method: Theory and insights," in *Advances in Neural Information Processing Systems*, 2014, pp. 2510–2518.

[72] A. Wibisono, A. C. Wilson, and M. I. Jordan, "A variational perspective on accelerated methods in optimization," *proceedings of the National Academy of Sciences*, vol. 113, no. 47, E7351–E7358, 2016.

[73] A. C. Wilson, B. Recht, and M. I. Jordan, "A lyapunov analysis of momentum methods in optimization," *arXiv preprint arXiv:1611.02635*, 2016.

[74] B. Hu and L. Lessard, "Dissipativity theory for nesterov's accelerated method," in *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, JMLR. org, 2017, pp. 1549–1557.

[75] H. Attouch, Z. Chbani, J. Peypouquet, and P. Redont, "Fast convergence of inertial dynamics and algorithms with asymptotic vanishing viscosity," *Mathematical Programming*, vol. 168, no. 1-2, pp. 123–175, 2018.

[76] B. Shi, S. S. Du, M. I. Jordan, and W. J. Su, "Understanding the acceleration phenomenon via high-resolution differential equations," *arXiv preprint arXiv:1810.08907*, 2018.

[77] R. Jordan, D. Kinderlehrer, and F. Otto, "The variational formulation of the fokker–planck equation," *SIAM journal on mathematical analysis*, vol. 29, no. 1, pp. 1–17, 1998.

[78] Q. Liu and D. Wang, "Stein variational gradient descent: A general purpose bayesian inference algorithm," in *Advances in neural information processing systems*, 2016, pp. 2378–2386.

[79] A. Wibisono, "Sampling as optimization in the space of measures: The langevin dynamics as a composite optimization problem," in *Conference On Learning Theory*, 2018, pp. 2093–3027.

[80] R. Zhang, C. Chen, C. Li, and L. Carin, "Policy optimization as wasserstein gradient flows," in *International Conference on Machine Learning*, 2018, pp. 5737–5746.

[81] C. Frogner and T. Poggio, "Approximate inference with wasserstein gradient flows," in *International Conference on Artificial Intelligence and Statistics*, 2020.

[82] L. Chizat and F. Bach, "On the global convergence of gradient descent for over-parameterized models using optimal transport," in *Advances in neural information processing systems*, 2018, pp. 3036–3046.

[83] C. Chen, R. Zhang, W. Wang, B. Li, and L. Chen, "A unified particle-optimization framework for scalable bayesian sampling," in *The Conference on Uncertainty in Artificial Intelligence*, 2018.

[84] C. Villani, *Optimal transport: old and new*. Springer Science & Business Media, 2008, vol. 338.

[85] R. Dwivedi, Y. Chen, M. J. Wainwright, and B. Yu, "Log-concave sampling: Metropolis-hastings algorithms are fast," *Journal of Machine Learning Research*, vol. 20, no. 183, pp. 1–42, 2019.

[86] S. Chewi, C. Lu, K. Ahn, X. Cheng, T. L. Gouic, and P. Rigollet, "Optimal dimension dependence of the metropolis-adjusted langevin algorithm," *arXiv preprint arXiv:2012.12810*, 2020.

[87] J. C. Mattingly, A. M. Stuart, and D. J. Higham, "Ergodicity for sdes and approximations: Locally lipschitz vector fields and degenerate noise," *Stochastic processes and their applications*, vol. 101, no. 2, pp. 185–232, 2002.

[88] Y. Cao, J. Lu, and L. Wang, "On explicit l_2-convergence rate estimate for underdamped langevin dynamics," *arXiv preprint arXiv:1908.04746*, 2019.

[89] J. Dolbeault, C. Mouhot, and C. Schmeiser, "Hypocoercivity for kinetic equations with linear relaxation terms," *Comptes Rendus Mathematique*, vol. 347, no. 9-10, pp. 511–516, 2009.

[90] ——, "Hypocoercivity for linear kinetic equations conserving mass," *Transactions of the American Mathematical Society*, vol. 367, no. 6, pp. 3807–3828, 2015.

[91] C. Villani, "Hypocoercivity," *Memoirs of the American Mathematical Society*, vol. 202, no. 950, 2009.

[92] J.-P. Eckmann and M. Hairer, "Spectral properties of hypoelliptic operators," *Communications in mathematical physics*, vol. 235, no. 2, pp. 233–253, 2003.

[93] F. Baudoin, "Bakry-emery meet villani," *Journal of Functional Analysis*, 2017.

[94] C. Liu, J. Zhuo, P. Cheng, R. Zhang, and J. Zhu, "Understanding and accelerating particle-based variational inference," in *International Conference on Machine Learning*, 2019, pp. 4082–4092.

[95] A. Taghvaei and P. Mehta, "Accelerated flow for probability distributions," in *Proceedings of the 36th International Conference on Machine Learning*, K. Chaudhuri and R. Salakhutdinov, Eds., ser. Proceedings of Machine Learning Research, vol. 97, Long Beach, California, USA: PMLR, Sep. 2019, pp. 6076–6085.

[96] Y. Wang and W. Li, "Accelerated information gradient flow," *arXiv preprint arXiv:1909.02102*, 2019.

[97] B. Leimkuhler, C. Matthews, and J. Weare, "Ensemble preconditioning for markov chain monte carlo simulation," *Statistics and Computing*, vol. 28, no. 2, pp. 277–290, 2018.

[98] G. Bierkens, P. Fearnhead, and G. Roberts, "The zig-zag process and super-efficient sampling for bayesian analysis of big data," *Annals of Statistics*, vol. 47, no. 3, 2019.

[99] C.-R. Hwang, S.-Y. Hwang-Ma, S.-J. Sheu, *et al.*, "Accelerating diffusions," *Annals of Applied Probability*, vol. 15, no. 2, pp. 1433–1444, 2005.

[100] T. Lelievre, F. Nier, and G. A. Pavliotis, "Optimal non-reversible linear drift for the convergence to equilibrium of a diffusion," *Journal of Statistical Physics*, vol. 152, no. 2, pp. 237–274, 2013.

[101] M. Ohzeki and A. Ichiki, "Langevin dynamics neglecting detailed balance condition," *Physical Review E*, vol. 92, no. 1, p. 012 105, 2015.

[102] L. Rey-Bellet and K. Spiliopoulos, "Irreversible langevin samplers and variance reduction: A large deviations approach," *Nonlinearity*, vol. 28, no. 7, p. 2081, 2015.

[103] A. B. Duncan, T. Lelievre, and G. Pavliotis, "Variance reduction using nonreversible langevin samplers," *Journal of statistical physics*, vol. 163, no. 3, pp. 457–491, 2016.

[104] B. T. Polyak, "Some methods of speeding up the convergence of iteration methods," *USSR Computational Mathematics and Mathematical Physics*, vol. 4, no. 5, pp. 1–17, 1964.

[105] F. Alvarez, H. Attouch, J. Bolte, and P. Redont, "A second-order gradient-like dissipative dynamical system with hessian-driven damping.: Application to optimization and mechanics," *Journal de mathématiques pures et appliquées*, vol. 81, no. 8, pp. 747–779, 2002.

[106] H. Attouch, Z. Chbani, J. Fadili, and H. Riahi, "First-order optimization algorithms via inertial systems with hessian driven damping," *Mathematical Programming*, pp. 1–43, 2020.

[107] X. Li, D. Wu, L. Mackey, and M. A. Erdogdu, "Stochastic Runge-Kutta accelerates Langevin Monte Carlo and beyond," *NeurIPS*, 2019.

[108] J. Zhang, A. Mokhtari, S. Sra, and A. Jadbabaie, "Direct runge-kutta discretization achieves acceleration," in *NeurIPS*, 2018.

[109] A. K. Kim, R. J. Samworth, *et al.*, "Global rates of convergence in log-concave density estimation," *The Annals of Statistics*, vol. 44, no. 6, pp. 2756–2779, 2016.

[110] S. Bubeck, R. Eldan, and J. Lehec, "Sampling from a log-concave distribution with projected langevin monte carlo," *Discrete & Computational Geometry*, vol. 59, no. 4, pp. 757–783, 2018.

[111] J. Roussel and G. Stoltz, "Spectral methods for langevin dynamics and associated error estimates," *ESAIM: Mathematical Modelling and Numerical Analysis*, vol. 52, no. 3, pp. 1051–1083, 2018.

[112] S. M. Kozlov, "Effective diffusion in the fokker-planck equation," *Mathematical notes of the Academy of Sciences of the USSR*, vol. 45, pp. 360–368, 1989.

[113] S. P. Meyn, R. L. Tweedie, *et al.*, "Computable bounds for geometric convergence rates of markov chains," *The Annals of Applied Probability*, vol. 4, no. 4, pp. 981–1011, 1994.

[114] Y. Nesterov, "Introductory lectures on convex programming volume i: Basic course," *Lecture notes*, vol. 3, no. 4, p. 5, 1998.

[115] S. Boucheron, G. Lugosi, and P. Massart, *Concentration inequalities: A nonasymptotic theory of independence*. Oxford university press, 2013.

[116] J. Cheeger, "A lower bound for the smallest eigenvalue of the laplacian," in *Proceedings of the Princeton conference in honor of Professor S. Bochner*, 1969, pp. 195–199.

[117] D. Bakry and M. Émery, "Diffusions hypercontractives," in *Séminaire de Probabilités XIX 1983/84*, Springer, 1985, pp. 177–206.

[118] J.-D. Deuschel and D. W. Stroock, *Large deviations*. American Mathematical Soc., 2001, vol. 342.

[119] F.-Y. Wang, "A generalization of poincaré and log-sobolev inequalities," *Potential Analysis*, vol. 22, no. 1, pp. 1–15, 2005.

[120] S. G. Bobkov *et al.*, "Isoperimetric and analytic inequalities for log-concave probability measures," *The Annals of Probability*, vol. 27, no. 4, pp. 1903–1921, 1999.

[121] G. Strang, "On the construction and comparison of difference schemes," *SIAM journal on numerical analysis*, vol. 5, no. 3, pp. 506–517, 1968.

[122] R. I. McLachlan and G. R. W. Quispel, "Splitting methods," *Acta Numerica*, vol. 11, p. 341, 2002.

[123] G. N. Milstein and M. V. Tretyakov, *Stochastic numerics for mathematical physics*. Springer Science & Business Media, 2013.

[124] H. Rosenbrock, "An automatic method for finding the greatest or least value of a function," *The Computer Journal*, vol. 3, no. 3, pp. 175–184, 1960.

[125] D. Dua and C. Graff, *UCI machine learning repository*, 2017.

[126] A. Durmus, S. Majewski, and B. Miasojedow, "Analysis of langevin monte carlo via convex optimization.," *J. Mach. Learn. Res.*, vol. 20, pp. 73–1, 2019.

[127] V. S. Borkar and S. K. Mitter, "A strong approximation theorem for stochastic recursive algorithms," *Journal of optimization theory and applications*, vol. 100, no. 3, pp. 499–513, 1999.

[128] S. Mandt, M. D. Hoffman, and D. M. Blei, "Stochastic gradient descent as approximate bayesian inference," *The Journal of Machine Learning Research*, vol. 18, no. 1, pp. 4873–4907, 2017.

[129] Q. Li, C. Tai, and E. Weinan, "Stochastic modified equations and adaptive stochastic gradient algorithms," in *International Conference on Machine Learning*, 2017, pp. 2101–2110.

[130] M. Lichman *et al.*, *UCI machine learning repository*, 2013.