Langevin Monte Carlo Beyond Lipschitz Gradient Continuity

Matej Benko[1], **Iwona Chlebicka**[2], **Jørgen Endal**[3], **Błażej Miasojedow**[2]

[1]Institute of Mathematics, Faculty of Mechanical Engineering, Brno University of Technology
Technická 2896/2, 616 69 Brno, Czech Republic
[2] Institute of Applied Mathematics and Mechanics, University of Warsaw
ul. Banacha 2, 02-097 Warsaw, Poland
[3]Department of Mathematical Sciences, Norwegian University of Science and Technology (NTNU)
N-7491 Trondheim, Norway
matej.benko@vutbr.cz, i.chlebicka@mimuw.edu.pl, jorgen.endal@ntnu.no, b.miasojedow@mimuw.edu.pl

Abstract

We present a significant advancement in the field of Langevin Monte Carlo (LMC) methods by introducing the Inexact Proximal Langevin Algorithm (IPLA). This novel algorithm broadens the scope of problems that LMC can effectively address while maintaining controlled computational costs. IPLA extends LMC's applicability to potentials that are convex, strongly convex in the tails, and exhibit polynomial growth, beyond the conventional L-smoothness assumption. Moreover, we extend LMC's applicability to super-quadratic potentials and offer improved convergence rates over existing algorithms. Additionally, we provide bounds on all moments of the Markov chain generated by IPLA, enhancing its analytical robustness.

Code — https://github.com/192459/lmc-beyond-lipschitz-gradient-continuity

Introduction

Langevin Monte Carlo methods are powerful tools for sampling and optimization in complex, high-dimensional problems across various fields, including machine learning, statistics, physics, and beyond. When the exact form of a distribution is unknown or difficult to compute, efficient sampling is essential for high-dimensional data, where traditional sampling methods often become computationally prohibitive. The ability of such algorithms to scale efficiently with dimensionality and manage non-smooth elements enhances their applicability in various machine learning tasks, including regression, classification, and clustering, providing a powerful tool for developing robust and accurate models. Given the potential $V \colon \mathbb{R}^d \to \mathbb{R}$ and associated density

$$\mu^*(x) = \frac{\exp(-V(x))}{\int_{\mathbb{R}^d} \exp(-V(y))\, \mathrm{d}y}$$

our goal is to generate samples $X_1, \ldots, X_n$ approximately from μ^*. The ability to perform effective sampling from a distribution with a density known up to a constant is crucial in Bayesian statistical inference (Gelman et al. 2014; Robert and Casella 2004), machine learning (Andrieu et al. 2003), ill-posed inverse problems (Stuart 2010), and computational physics (Krauth 2006).

One of the most efficient solutions to the problem of generating a sample from μ^* is Langevin Monte Carlo (LMC), which was designed via discretization of Overdamped Langevin Equation

$$\mathrm{d}Y_t = -\nabla V(Y_t)\mathrm{d}t + \sqrt{2}\, \mathrm{d}B_t \tag{1}$$

where B_t is a d-dimensional Brownian motion. Classically LMC is derived by Euler–Maruyama discretization of the above diffusion process. Namely, we iteratively define Markov Chains such that for a given X_0 and all $k \geq 0$ we have $X_{k+1} = X_k - \tau_k \nabla V(X_k) + \sqrt{2\tau_k}\, \overline{Z}_{k+1}$, where $\{\tau_k\}_{k\geq0}$ is sequence of stepsizes and $\{\overline{Z}_{k+1}\}_{k\geq0}$ are i.i.d standard d-dimensional Gaussian random variables which are independent of X_0. In the recent years the LMC algorithm has been intensively investigated see (Dalalyan and Tsybakov 2012; Chewi et al. 2022; Dalalyan 2017a,b; Durmus, Majewski, and Miasojedow 2019; Dalalyan, Karagulyan, and Riou-Durand 2022; Durmus and Moulines 2017, 2019; Erdogdu and Hosseinzadeh 2021; Erdogdu, Hosseinzadeh, and Zhang 2022; Mousavi-Hosseini et al. 2023; Vempala and Wibisono 2019). Efficiently scaling with dimension ensures that machine learning models remain computationally feasible and accurate as the complexity of the data increases. To the best of our knowledge, the overall complexity of generating one sample from μ^* with given precision ε scales linearly with dimension, see (Durmus, Majewski, and Miasojedow 2019). More precisely, it is proven therein that in the case of a convex potential V an error in the Kullback–Leibler divergence scales linearly with dimension. The same dependence on dimension is shown therein for error expressed in the Wasserstein distance when V is strongly convex.

The abovementioned studies and related research are typically conducted under the assumption that V is L-smooth, i.e., ∇V is L-Lipschitz. A particular consequence of such an assumption is that a quadratic function majorizes V. This narrows the scope of possible applications excluding many classical potentials behaving like polynomials with higher power than 2, such as the one generating the processes of Ginzburg–Landau, cf. (Brosse et al. 2019; Goldenfeld 1992). Henceforth, aiming at capturing the natural potentials we relax the L-smoothness condition trading it with λ-convexity outside a ball.

We stress that it is necessary to assume that ∇V is L-

Lipschitz while considering Euler–Maruyama scheme, because otherwise the generated chain is transient (Roberts and Tweedie 1996; Mattingly, Stuart, and Higham 2002). Therefore, to tackle non-L-smooth potentials one is forced to modify the scheme. The natural idea how to adjust the scheme comes from a variational formulation of (1) by Jordan, Kinderlehrer, and Otto (1998); Ambrosio, Gigli, and Savaré (2008):

$$\mu^* = \underset{\mu \in \mathscr{P}_2(\mathbb{R}^d)}{\arg\min} \ \mathcal{F}[\mu] \,, \tag{2}$$

where the functional $\mathcal{F}$ is defined for $\mu \in \mathscr{P}(\mathbb{R}^d)$ (probabilistic measures) via the following formula

$$\mathcal{F}[\mu] := \mathcal{F}_V[\mu] + \mathcal{F}_{\mathcal{E}}[\mu] \,,$$

where (denoting $\mathrm{d}\mu(x) = \mu(x)\,\mathrm{d}x$ when a density exists)

$$\mathcal{F}_V[\mu] := \int_{\mathbb{R}^d} V(x)\mu(\mathrm{d}x)\,,$$

$$\mathcal{F}_{\mathcal{E}}[\mu] := \begin{cases} \int_{\mathbb{R}^d} \mu(x)\log\mu(x)\,\mathrm{d}x\,, & \text{if } \mu \in \mathscr{P}_{\mathrm{ac}}\,, \\ +\infty\,, & \text{otherwise}\,. \end{cases}$$

Note that μ^* is a minimizer of $\mathcal{F}$. Indeed, due to (Durmus, Majewski, and Miasojedow 2019, Lemma 1) we know that if $\mu^* \in \mathscr{P}_2(\mathbb{R}^d)$, then $\mathcal{F}_V[\mu^*] < +\infty$ and $\mathcal{F}_{\mathcal{E}}[\mu^*] < +\infty$. Moreover, for $\mu \in \mathscr{P}_2(\mathbb{R}^d)$ satisfying $\mathcal{F}_{\mathcal{E}}[\mu] < +\infty$, it holds $\mathcal{F}[\mu] - \mathcal{F}[\mu^*] = \mathrm{KL}(\mu|\mu^*)$, where KL stands for the Kullback–Leibler divergence between measures (it is also called the relative entropy). Since $\mathrm{KL}(\mu|\mu^*) \geq 0$ and the equality holds only if $\mu = \mu^*$ we get that μ^* is a minimizer of $\mathcal{F}$. By this observation the LMC algorithm can be seen as an inexact optimization of $\mathcal{F}$, see also (Bernton 2018; Durmus, Majewski, and Miasojedow 2019).

The typical approach in order to find a minimizer of the variational functional $\mathcal{F}$ is going along the gradient flow in the Wasserstein space. We split the gradient flow of $\mathcal{F}$ into two gradient flows (of $\mathcal{F}_{\mathcal{E}}$ and of $\mathcal{F}_V$). By Ambrosio, Gigli, and Savaré (2008, Theorem 11.1.4) and Øksendal (2010, Theorem 2.1.5) the gradient flow of $\mathcal{F}_{\mathcal{E}}$ is the standard Brownian motion. To describe the gradient flow of $\mathcal{F}_V$ we introduce the *proximal operator* defined for the function V and the time step τ by

$$\mathrm{prox}_V^\tau(x) := \underset{y \in \mathbb{R}^d}{\arg\min} \left\{ V(y) + \tfrac{1}{2\tau}|y-x|^2 \right\}\,.$$

By Ambrosio, Gigli, and Savaré (2008, Proposition 10.4.2(ii)) the gradient flow of $\mathcal{F}_V$ starting from μ with time step τ is given by $(\mathrm{prox}_V^\tau)_{\#}\mu$. Therefore, the splitting algorithm for minimizing $\mathcal{F}$ is the Proximal Langevin Monte Carlo (PLMC) given by

$$X_{k+1} = \mathrm{prox}_V^\tau(X_k) + \sqrt{2\tau_k}\,\overline{Z}_{k+1}\,.$$

The definition of $\mathrm{prox}_V^\tau(x)$ implies $V(\mathrm{prox}_V^\tau(x)) \leq V(x)$, which prevents PLMC from exploding. PLMC was already studied for machine learning problems by Bernton (2018); Salim, Korba, and Luise (2020) and in the context of numerical solutions to aggregation-diffusion equations by Benko et al. (2024). The main drawback of PLMC is that it is typically not feasible to perform the proximal step exactly, so this step should be approximated. However, $\mathcal{F}_V$ and $\mathcal{F}_{\mathcal{E}}$ are contractions in the Wasserstein space and an error introduced by an inexact proximal step does not accumulate significantly. In addition, the function to be optimized in the proximal step is strongly convex and there are several accurate and fast computational methods to approximate prox_V^τ.

In the current paper, we propose and give theoretical guarantees that the Inexact Proximal Langevin Algorithm (IPLA for short) can be used successfully to generate from μ^* a sample beyond the assumption of global Lipschitz gradient continuity of V. The typical assumption of the strong convexity of V is also relaxed.

Our Contribution

We make significant progress in the field of Langevin Monte Carlo methods by introducing a novel algorithm handling essentially broader scope of problems, than treated up to now, and preserving controlled computational cost. Namely, we propose the Inexact Proximal Langevin Algorithm IPLA (Algorithm 1). We extend the current scope of LMC algorithms by effective handling a large and natural class of potentials beyond the typical assumption of L-smoothness.

We assume that the potential V is convex, strongly convex in tails, and with polynomial growth with power $q_V + 1 \geq 2$. Under such hypothesis, when an error of the inexact proximal step is of order τ^2, then IPLA generates one sample from μ^* with accuracy ε in the KL-divergence with total complexity $d^{\frac{q_V+1}{2}}\mathcal{O}(\varepsilon^{-2})$, see Theorem 5 and Corollary 6. Moreover, when V is strongly convex, we can allow an error of the inexact proximal step of order $\tau^{\frac{3}{2}}$. In this setting we obtain that the total number of iterations to get one sample with the error ε in the Wasserstein distance is of order $d^{\frac{q_V+1}{2}}\mathcal{O}(\varepsilon^{-2})$, see Theorem 7 and Corollary 8. Let us note that a lower accuracy of approximation of the proximal step in the strongly convex regime is expected. Since the proximal step is a gradient flow of $\mathcal{F}_{\mathcal{E}}$, which is contracting in the Wasserstein space, the introduced error accumulates linearly. However, for strongly convex V the gradient flow of $\mathcal{F}_{\mathcal{E}}$ is strongly contracting, and the error is progressively suppressed during the iterations.

Our results provide better convergence rates than bounds under similar assumptions for the Tamed Unadjusted Langevin Algorithm, cf. Brosse et al. (2019). Moreover, we get the same complexity as the Unadjusted Barker Algorithm (Livingstone et al. 2024), but under weaker assumptions – we do not assume one-sided Lipschitz conditions on ∇V. We note that when $q_V = 1$ we retrieve dependence on the dimension that is the best-known for LMC, see (Durmus, Majewski, and Miasojedow 2019).

Let us recall that the known results for LMC cover only potentials with growth between linear and quadratic. Our contribution is complementary and embrace super-quadratic potentials. Henceforth, we complete the scope of an application of LMC to all (sufficiently smooth) convex potentials of power growth. Besides the convergence rates of IPLA, we show a bound on all moments of the Markov chain $\{X_k\}_{k\geq 0}$ generated by IPLA, see Theorem 1. Such bounds are not only crucial for analyzing IPLA, but are also of separate in-

terest for analysis of Langevin Monte Carlo algorithms (Cattiaux, Guillin, and Malrieu 2008; Durmus and Moulines 2019; Livingstone et al. 2024).

Description of Our Setting

We consider an optimization problem on the space of probability measures over $\mathbb{R}^d$ for which we use the notation $\mathscr{P}(\mathbb{R}^d)$. We define $\mathscr{P}_{ac}(\mathbb{R}^d)$ – the subspace of $\mathscr{P}(\mathbb{R}^d)$ consisting of measures absolutely continuous with respect to the Lebesgue measure. By $\mathscr{P}_m(\mathbb{R}^d)$, for $m \geq 1$, we denote the subspace of $\mathscr{P}(\mathbb{R}^d)$ consisting of measures with finite m-th moment, i.e.,

$$\mathscr{P}_m(\mathbb{R}^d) := \{\mu \in \mathscr{P}(\mathbb{R}^d) \colon \int_{\mathbb{R}^d} |x|^m \mu(\mathrm{d}x) < \infty\}$$
$$= \{\mu \in \mathscr{P}(\mathbb{R}^d) \colon \mu(|\cdot|^m) < \infty\}.$$

We denote by $C^1(\mathbb{R}^d, \mathbb{R})$ the space of functions from $\mathbb{R}^d$ to $\mathbb{R}$ with continuous derivatives.

A continuously differentiable function $f : \mathbb{R}^d \to \mathbb{R}$ is called

- λ-convex (strong convex) for $\lambda \in \mathbb{R}$ on $\mathbb{R}^d$, if for all $x, y \in \mathbb{R}^d$ it satisfies
$$f(x) \geq f(y) + \nabla f(y) \cdot (x - y) + \tfrac{\lambda}{2}|x - y|^2 \,;$$

- λ-convex for $\lambda \in \mathbb{R}$ outside of a ball $B \subset \mathbb{R}^d$, if for all $x, y \in \mathbb{R}^d$ it satisfies
$$f(x) \geq f(y) + \nabla f(y) \cdot (x - y) + \tfrac{\lambda}{2}\mathbb{1}_{\mathbb{R}^d \setminus B}(y)|x - y|^2 \,.$$

If f is λ-convex, the function $f - \frac{\lambda}{2}|\cdot|^2$ is 0-convex, so we immediately get that

$$\left(\nabla f(x) - \nabla f(y)\right) \cdot (x - y) \geq \lambda|x - y|^2 \,.$$

We denote by $\overline{Z}_k$ the standard Gaussian distribution on $\mathbb{R}^d$. For the sake of clarity of the exposition, throughout the paper we denote by Z_k the rescaled one, namely $Z_k \sim \mathcal{N}(0, 2\tau \,\mathsf{Id})$, where τ will be clear from the context.

We will analyze IPLA in terms of two distances between measures. The first of them is the Kullback-Leibler divergence, also known as relative entropy, given for any measure $\mu, \nu \in \mathscr{P}(\mathbb{R}^d)$ by

$$\mathrm{KL}(\mu|\nu) := \begin{cases} \int_{\mathbb{R}^d} \frac{d\mu}{d\nu}(x) \log\left(\frac{d\mu}{d\nu}(x)\right) \nu(\mathrm{d}x), & \text{if } \mu \ll \nu, \\ +\infty, & \text{otherwise}. \end{cases}$$

Let us note that due to the Jensen inequality for all measures, $\mu, \nu \in \mathscr{P}(\mathbb{R}^d)$, $\mathrm{KL}(\mu|\nu) \geq 0$ and equality holds only if $\mu = \nu$. The second distance used in our analysis is the 2-Wasserstein distance defined for all $\mu, \nu \in \mathscr{P}_2(\mathbb{R}^d)$ by

$$W_2^2(\mu, \nu) := \inf_{\gamma \in \Pi(\mu, \nu)} \int_{\mathbb{R}^d} |x - y|^2 \gamma(\mathrm{d}x, \mathrm{d}y), \quad (3)$$

where $\Pi(\mu, \nu)$ denotes the coupling between μ and ν, that is, $\Pi(\mu, \nu)$ is a set of probability measures $\gamma \in \mathscr{P}_2(\mathbb{R}^d \times \mathbb{R}^d)$ such that for every measurable set A it holds $\gamma(A, \mathbb{R}^d) = \mu(A)$ and $\gamma(\mathbb{R}^d, A) = \nu(A)$. We observe that by Villani

(2009, Theorem 4.1) there exists an optimal coupling γ^*, for which the infimum in (3) is obtained.

By the pushforward of the function $f : \mathbb{R}^d \to \mathbb{R}^d$, we mean the measure $f_\#$ defined as $f_\# \nu(U) := \nu(f^{-1}(U))$ for any measure $\nu \in \mathscr{P}(\mathbb{R}^d)$ and any measurable set $U \subseteq \mathbb{R}^d$.

Let us present our regime.

(V) For the potential V we assume what follows:

- $V \in C^1(\mathbb{R}^d, \mathbb{R})$ and is convex on $\mathbb{R}^d$.
- V is λ_V-convex for $\lambda_V > 0$ outside a given ball $B_V(0, R_V)$ with the center in 0 and a radius $R_V \geq 0$.
- The minimum of V is attained at $x^* = 0$.
- For $q_V \geq 1$, $C_V > 0$, and for all $x, y \in \mathbb{R}^d$ it holds
$$V(y) \leq V(x) + \nabla V(x) \cdot (y - x)$$
$$+ C_V \left(1 + |x|^{q_V - 1} + |y|^{q_V - 1}\right) |y - x|^2. \quad (4)$$

(ϱ_0) The initial distribution $\varrho_0 \in \mathscr{P}_{m_0}(\mathbb{R}^d)$ has a finite moment of order $m_0 = q_V + 1$.

Let us comment on the assumptions **(V)** and **(ϱ_0)**:

(i) Assumption **(V)** embraces convex potentials with tails of polynomial growth with power $q_V + 1 \geq 2$. The common assumption in the literature for LMC allows for convex functions with growth bounded by square function.

(ii) The assumption that $x^* = 0$ is imposed just to simplify the presentation. All results remain true when $x^* \neq 0$ is any vector. There is no need for x^* to be the unique minimizer of V.

(iii) In assumption **(V)** condition (4) is satisfied for $q_V = 1$ if ∇V is L-Lipschitz with $C_V = \frac{L}{2}$. Taking $q_V > 1$ allows for treating way broader class of potentials V.

(iv) If ∇V is locally Lipschitz with q_V-power growth, i.e., if there exists $L_q > 0$ such that for all $x, y \in \mathbb{R}^d$ it holds
$$|\nabla V(x) - \nabla V(y)|$$
$$\leq L_q \min\{|x - y|, 1\}(1 + |x|^{q_V - 1} + |y|^{q_V - 1}),$$
then (4) is satisfied for $C_V = 2^{2q_V - 2} L_q$, cf. (Benko et al. 2024, Lemma A.2).

(v) In assumption **(ϱ_0)** we require finite $(q_V + 1)$-th moment of the initial distribution. This is a very weak assumption, as typically the finiteness of higher moments is used to prove the existence and uniqueness of solutions to Langevin equation (1), see Cattiaux, Guillin, and Malrieu (2008). Moreover, such an assumption is not restrictive in the application, as the initial measure is chosen arbitrarily in practice.

Inexact Proximal Langevin Algorithm

Let us present the idea of our algorithm. By (2) we know that μ^* is a minimizer of $\mathcal{F}$ over a space of measures, so we design an algorithm optimizing this functional. The basic idea to reach the minimizer is the analog of the gradient descent algorithm. In our case the functional $\mathcal{F}$ lives on

Algorithm 1: Inexact Proximal Langevin Algorithm (IPLA)

Initialize: Sample initial distribution $X_0 \sim \varrho_0$

  for $k = 0, \ldots, n-1$ **do**

   Step 1: Run routine for computing $\mathrm{prox}_V^\tau(X_k)$

    with an output $X_{k+\frac{2}{3}} = \mathrm{prox}_V^\tau(X_k) + \Theta_{k+\frac{2}{3}}$.

   Step 2: Add Gaussian noise g_τ, i.e

    $X_{k+1} = X_{k+\frac{2}{3}} + Z_{k+1}, \ \ Z_{k+1} \sim \mathcal{N}(0, 2\tau \mathsf{Id})$.

  end for

---

the Wasserstein space, so – instead of the classical gradient – we need to employ the gradient flow of the functional. For definition and basic properties of the gradient flows on Wasserstein spaces we refer to Ambrosio, Gigli, and Savaré (2008); Santambrogio (2017). We note that the gradient flow of $\mathcal{F}$ is not given by explicit formula, but we know that $\mathcal{F} = \mathcal{F}_V + \mathcal{F}_\mathcal{E}$ where the gradient flows of both $\mathcal{F}_V$ and $\mathcal{F}_\mathcal{E}$ are well-understood. As mentioned in the introduction, the gradient flow of $\mathcal{F}_V$ is given by a pushforward of the proximal operator, i.e., $(\mathrm{prox}_V^\tau)_\#$. The gradient flow of $\mathcal{F}_\mathcal{E}$ is the standard Brownian motion. Therefore, we use the splitting algorithm of the form

$$X_{k+1} = \mathrm{prox}_V^{\tau_k}(X_k) + \sqrt{2\tau_k}\,\overline{Z}_{k+1}\,,$$

which can be called Proximal Langevin Algorithm. However, in practice, computing the exact proximal operator is challenging. Therefore, we perform its more feasible numerical approximation. Since under our assumptions, both flows are contracting the introduced error will not accumulate too much and can be controlled.

The IPLA (Algorithm 1) is defined as follows: in each step we perform two half steps. Given the previous state X_k, first we move according to the inexact proximal step of a function V with stepsize τ, where the error $\Theta_{k+\frac{2}{3}}$ (which could be stochastic or deterministic) is bounded by δ. Next we add an independent of history and centered Gaussian random variable with covariance $2\tau \mathsf{Id}$.

For the theoretical analysis, we will split the inexact proximal step into two substeps: exact proximal step and additive error. Further, we will use the following notation.

  (i) $X_{k+\frac{1}{3}} := \mathrm{prox}_V^\tau(X_k)$,

  (ii) $X_{k+\frac{2}{3}} := X_{k+\frac{1}{3}} + \Theta_{k+\frac{2}{3}}; \ \Theta_{k+\frac{2}{3}} \sim \xi_\delta$,

  (iii) $X_{k+1} := X_{k+\frac{2}{3}} + Z_{k+1}; \ Z_{k+1} \sim g_\tau$.

In view of densities ϱ_k s.t. $X_k \sim \varrho_k$ it reads as follows

  (i) $\varrho_{k+\frac{1}{3}} := (\mathrm{prox}_V^\tau)_\# \varrho_k$,

  (ii) $\varrho_{k+\frac{2}{3}} := \varrho_{k+\frac{1}{3}} * \xi_\delta$,

  (iii) $\varrho_{k+1} := \varrho_{k+\frac{2}{3}} * g_\tau$,

where ξ_δ is a probability measure supported on $B(0, \delta)$ and g_τ denotes the density of Gaussian distribution given by $Z_k \sim \mathcal{N}(0, 2\tau \mathsf{Id})$. We will also assume that for some $\kappa > 0$ and $\alpha \geq 0$ the error bound δ and stepsize τ satisfy $\delta = \kappa \tau^{1+\alpha}$.

Theoretical Results

In this section, we provide theoretical guarantees for the accuracy of IPLA. We start with bounds for moments of the Markov chain $\{X_k\}_k$ generated by IPLA, see Theorem 1. Then we show error bounds of IPLA in KL-divergence and in Wasserstein distance, see Theorems 5 and 7, respectively.

Now we show that provided that for $m \geq 0$ the m-th moment of initial measure ϱ_0 is bounded, it holds $\sup_k \mathbb{E}|X_k|^m < \infty$. Such bounds are required since we allow super quadratic growth of V.

Theorem 1 (Moment bound). *Let $m \geq 0$. Suppose that V satisfies* **(V)** *and ϱ_0 satisfies* **(ϱ_0)**, *and, for $k = 1, \ldots, n$, let X_k be as in Algorithm 1. Then there exists a constant $\mathcal{C}_m > 0$ such that for all $0 < \tau < 1/\lambda_V$, we have*

$$\sup_k \mathbb{E}|X_k|^m \leq \mathcal{C}_m \min\{1, \lambda_V^{-m}\} d^{\frac{m}{2}}\,.$$

If the error and the stepsize satisfy $\delta = \kappa \tau^{1+\alpha}$ for some $\kappa > 0$ and $\alpha \geq 0$, then $\mathcal{C}_m = \mathcal{C}_m(m, \varrho_0, \kappa, \lambda_V)$. Moreover, $\mathcal{C}_m \ll +\infty$ when $\lambda_V \to 0$.

The proof of the above theorem and the exact value of $\mathcal{C}_m$ are given in Appendix B. Let us observe that since V has super-quadratic growth in tails, μ^* is sub-Gaussian distribution with all moments finite.

Remark 2. If the minimum of V is not at $x^* = 0$, then the result would have been

$$\sup_k \mathbb{E}|X_k - x^*|^m \leq \mathcal{C}_m \min\{1, \lambda_V^{-m}\} d^{\frac{m}{2}}\,.$$

In order to prove the main result on the error bound of IPLA in KL-divergence, we mimic the classical reasoning from the convex optimization based on the following observation. Let us consider $\{x_k\}_{k\geq 0}$ as an output of the first-order algorithm for minimizing convex function f over $\mathbb{R}^d$. It is possible to obtain an inequality of the following form

$$2\tau(f(x_{k+1}) - f(x^*)) \leq |x_k - x^*|^2 - |x_{k+1} - x^*|^2 + C\tau^2$$

where x^* is minimizer of f and $C > 0$ is some constant, see for example Beck (2017). Such an inequality is a key step in the proofs of convergence of certain algorithms. We provide the counterpart of such a result in the form relevant to analyze IPLA on the Wasserstein space. It employs the quantity

$$K(\tau) := 2^{4q_V - 3} L_{q_V} \left((1 + \mathcal{C}_{q_V - 1} d^{\frac{q_V - 1}{2}} \lambda_V^{-\frac{q_V - 1}{2}}) 2d\tau \right.$$
$$\left. + \mathcal{C}_{q_V + 1} d^{\frac{q_V + 1}{2}} \tau^{\frac{q_V + 1}{2}} \right)\,. \quad (5)$$

Remark 3. *There exists a constant $C_{q_V} < \infty$ such that for $\tau \leq 1$ it holds $K(\tau) \leq C_{q_V} \tau d^{\frac{q_V + 1}{2}}$.*

The mentioned auxiliary result reads as follows.

Proposition 4. *Assume that the function $V : \mathbb{R}^d \to \mathbb{R}$, $d \geq 1$, satisfies assumption* **(V)** *with some $q_V \geq 1$, the initial measure ϱ_0 satisfies* **(ϱ_0)**, *and $\nu \in \mathscr{P}_{q_V + 1}(\mathbb{R}^d)$ is arbitrary. Then there exist $C(\nu) > 0$, such that*

$$2\tau(\mathcal{F}[\varrho_{k+1}] - \mathcal{F}[\nu]) \leq W_2^2(\varrho_k, \nu) - W_2^2(\varrho_{k+1}, \nu)$$
$$+ C(\nu)\delta + K(\tau)\tau\,,$$

where $C(\nu)$ is given in Appendix B, while K in (5).

The proof is postponed to Appendix B.

We state our main results for a sequence of probability measures $\{\nu_n^N\}_{n\in\mathbb{N}}$, called average measures, defined for every $n, N \in \mathbb{N}$, $n \geq 1$ by $\nu_n^N := \frac{1}{n}\sum_{k=N+1}^{N+n}\varrho_k$, where N is a burn-in time. By the use of Proposition 4 we infer the following main theorem on convergence of IPLA.

Theorem 5 (KL-error bound). *Suppose that V satisfies* **(V)** *and ϱ_0 satisfies* **(ϱ_0)**. *Let $\tau < 1/\lambda_V$, $\kappa, \alpha > 0$, and $\delta \leq \kappa\tau^{1+\alpha}$. Then it holds:*

$$\mathrm{KL}(\nu_n^N|\mu^*) \leq \frac{1}{2n\tau}W_2^2(\varrho_N,\mu^*) - \frac{1}{2n\tau}W_2^2(\varrho_{N+n},\mu^*) + C(\mu^*)\kappa\tau^\alpha + K(\tau),$$

where K is defined by (5) *and $C(\mu^*)$ is the same as in Proposition 4 evaluated in $\nu = \mu^*$.*

Proof. We use the convexity of KL-divergence (Cover and Thomas 2012, Theorem 2.7.2.) and get

$$\mathrm{KL}(\nu_n^N|\mu^*) \leq \frac{1}{n}\sum_{k=N+1}^{N+n}\mathrm{KL}(\varrho_k|\mu^*).$$

Applying Proposition 4 we are able to write

$$\mathrm{KL}(\nu_n^N|\mu^*) \leq \frac{1}{2n\tau}\sum_{k=N+1}^{N+n}\left(W_2^2(\varrho_k,\nu) - W_2^2(\varrho_{k+1},\nu)\right) + C(\mu^*)\kappa\tau^\alpha + K(\tau).$$
□

A direct consequence of Theorem 5 is the following result on the overall complexity of generating one sample from μ^* with accuracy ε.

Corollary 6. *Suppose that the assumptions of Theorem 5 are satisfied. Assume further that*

$$0 < \tau_\varepsilon \leq \min\left\{\left(\frac{\varepsilon}{3C(\mu^*)\kappa}\right)^{\frac{1}{\alpha}}, 1\right\}$$

and $K(\tau_\varepsilon) \leq \varepsilon/3$. Let the number of iterations n_ε be such that $n_\varepsilon \geq 3W_2^2(\varrho_0,\mu^*)/(2\varepsilon\tau_\varepsilon)$. Then

$$\mathrm{KL}(\nu_{n_\varepsilon}^0|\mu^*) \leq \varepsilon. \tag{6}$$

Moreover, for computing one sample in terms of KL *with precision ε, in a case of warm start ($W_2^2(\varrho_0,\mu^*) \leq C$ for some absolute constant C), we need*

(i) $d^{\frac{q_V+1}{2}}\mathcal{O}(\varepsilon^{-2})$ *iterations, if $\alpha \geq 1$;*
(ii) $d^{\frac{q_V+1}{2}}\mathcal{O}(\varepsilon^{-1-\alpha^{-1}})$ *iterations, if $\alpha < 1$.*

Proof. The inequality (6) is a straightforward consequence of Theorem 5. To get estimates of computational complexity it is enough to observe that by Remark 3 we have that condition $K(\tau_\varepsilon) \leq \varepsilon$ is satisfied if

$$0 < \tau_\varepsilon \leq \min\left\{\varepsilon C_{q_V}^{-1}d^{-\frac{q_V+1}{2}}, 1\right\}.$$
□

Finally, let us show that, when the potential V is globally λ_V-convex, we can refine our result by improving the convergence rates of IPLA. Based on results from Benko et al. (2024) we have the following estimate on precision of IPLA in terms of Wasserstein distance.

Theorem 7 (Wasserstein error bound). *Suppose that the potential V satisfies* **(V)** *for $R_V = 0$ and $\lambda_V > 0$, and that the initial measure ϱ_0 satisfies* **(ϱ_0)**. *Let $\tau < 1/\lambda_V$, $\alpha \geq 0$, and $\delta \leq \kappa\tau^{1+\alpha}$. Then for all $k \in \mathbb{N}$ it holds*

$$W_2^2(\varrho_k,\mu^*) \leq 2\left(1 - \frac{\tau\lambda_V}{2}\right)^k W_2^2(\varrho_0,\mu^*) + \frac{4}{\lambda_V}K(\tau) + 2\kappa^2\tau^{2+2\alpha}\left(\frac{1 - e^{-\lambda_V\tau(k-1)}}{1 - e^{-\lambda_V\tau}}\right)^2.$$

This leads to the following complexity bounds.

Corollary 8. *Suppose that the assumptions of Theorem 7 are satisfied. Assume further that*

$$\tau_\varepsilon^{2\alpha} \leq \frac{\lambda_V^2\varepsilon}{96\kappa^2\log^2(6W_2^2(\varrho_0,\mu^*)\varepsilon^{-1})}$$

and $K(\tau_\varepsilon) \leq \frac{1}{12}\lambda_V\varepsilon$. Let the number of iterations n_ε be such that

$$2\log(6W_2^2(\varrho_0,\mu^*)\varepsilon^{-1})\tau_\varepsilon^{-1}\lambda_V^{-1} \leq n_\varepsilon$$
$$\leq 4\log(6W_2^2(\varrho_0,\mu^*)\varepsilon^{-1})\tau_\varepsilon^{-1}\lambda_V^{-1}.$$

Then

$$W_2^2(\varrho_{n_\varepsilon},\mu^*) \leq \varepsilon.$$

Moreover, for computing one sample in terms of the Wasserstein distance with precision ε, in the case of warm start, up to logarithmic terms we need

(i) $d^{\frac{q_V+1}{2}}\mathcal{O}(\varepsilon^{-2})$ *iterations, if $\alpha \geq \frac{1}{2}$;*
(ii) $d^{\frac{q_V+1}{2}}\mathcal{O}(\varepsilon^{-\alpha^{-1}})$ *iterations, if $\alpha < \frac{1}{2}$.*

Proof. Let us note that

$$\left(1 - \frac{\tau\lambda_V}{2}\right)^{n_\varepsilon} \leq \exp\left(-n_\varepsilon\frac{\tau\lambda_V}{2}\right) \text{ and } \frac{1 - e^{-\lambda_V\tau(n_\varepsilon-1)}}{1 - e^{-\lambda_V\tau}} \leq n_\varepsilon.$$

With the above bounds, the result follows directly from Theorem 7.
□

Note that if in Corollary 8 the stepsize $\tau_\varepsilon = \tau$ is fixed, then $W_2^2(\varrho_n,\mu^*) \leq \varepsilon$ for all sufficiently large n.

Remark 9. *The assumption of global λ_V-convexity of V decreases the computational cost of a single iteration of IPLA. We can approximate the proximal step with smaller precision and keep the complexity of the whole algorithm of the order ε^{-2}. Indeed, as a consequence of λ_V-convexity of the functional $\mathcal{F}_\mathcal{E}$ we get*

(i) *in Theorem 5 – under assumption* **(V)** *with $R_V > 0$ – to achieve complexity of order ε^{-2} we need $\delta \leq \kappa\tau^2$;*
(ii) *in Theorem 7 – in the strongly convex case – the order ε^{-2} we have for $\delta \leq \kappa\tau^{3/2}$.*

Remark 10. *The complexity of IPLA, as shown in Corollaries 6 and 8, depends on the number of iterations requiring additional computations. Since the optimized function is strongly convex, the cost of one iteration with precision δ is $\mathcal{O}(d\log(\delta))$, as detailed in Appendix E.*

Experiments

In this section, we demonstrate the application of IPLA on 3 examples implemented in Python. We analyze the convergence rates and bias of the algorithm compared to the two known related LMC algorithms, namely TULA (Tamed Unadjusted Langevin Algorithm by Brosse et al. (2019)) and ULA (Unadjusted Langevin Algorithm by Durmus and Moulines (2019)). The codes of our simulations can be found at `https://github.com/192459/lmc-beyond--lipschitz-gradient-continuity`. See Appendix D for additional results.

We shall compare the relative error (RE) and the coefficient of variance (CV) of various methods. In this context, the RE of the estimator $\widehat{\theta}$ of the true value θ is $\left|\frac{\widehat{\theta}-\theta}{\theta}\right|$, while CV is $\frac{\mathrm{sd}(\widehat{\theta})}{\theta}$, where sd stands for the standard deviation.

Example 1: Distribution With Light Tails

Let us start with a simple and natural case where the potential has non-Lipschitz gradient. Our goal is to sample from the density
$$\mu^*(x) \propto \exp\left(-\frac{|x|^4}{4}\right),$$
which is a stationary distribution of the process
$$\mathrm{d}Y_t = -Y_t^3\,\mathrm{d}t + \sqrt{2}\,\mathrm{d}B_t.$$
We estimate the moments $\mathbb{E}|Y|^2$, $\mathbb{E}|Y|^4$, and $\mathbb{E}|Y|^6$ in dimension $d = 10^3$. For error analysis as true values of the estimated quantities, we use the Metropolis–Hastings algorithm with 10^7 iterations. Note that ∇V is not globally Lipschitz and V is 1-convex outside a ball, so the assumptions on the convergence of ULA are not satisfied, but the convergence of TULA and IPLA is ensured.

We have run 10^5 samples with burn-in time 10^4. We considered two scenarios. An initial value in the first of them is $x_0 = 7 \times \mathbf{1}_d$ (start in tail), while in the second one considered we start at $x_0 = 0$ (start in the minimizer of V). Each experiment has been repeated 100 times. In Table 1 we compare the relative error and the coefficient of variance between the algorithms. We can see that IPLA has the smallest RE and its CV is comparable to ULA's.

In Figure 1 we compare the trajectories of algorithms for the selected coordinate ($i = 1$). We see that ULA blows up in a few iterations, which results from the fact that ∇V is not globally Lipschitz. To suppress this phenomenon, TULA adjusts stepsizes according to normalized ∇V, namely to $\frac{\nabla V}{1+|\nabla V|}$. This, on the other hand, results in unnecessarily small stepsizes in the tails. This can be seen in Figure 1. As shown on Figure 2 IPLA is less sensitive to the choice of stepsizes τ than TULA.

Example 2: Ginzburg–Landau Model

We pass to a more complicated, but physically relevant example. Ginzburg–Landau model was introduced to describe phase transitions in physics, i.e., superconductivity, see Goldenfeld (1992, Chapter 5). It involves the potential
$$V(x) = \sum_{i,j,k=1}^{q} \frac{1-\upsilon}{2}x_{ijk}^2 + \frac{\upsilon\varkappa}{2}|\widetilde{\nabla}x_{ijk}|^2 + \frac{\upsilon\varsigma}{4}x_{ijk}^4,$$

Mom.	Method	Start in tail		Start in $x_0 = 0$	
		RE	CV	RE	CV
$\mathbb{E}|Y|^2$	IPLA	0.0027	0.0019	0.0006	0.0018
	TULA	0.0047	0.0016	0.0030	0.0019
	ULA	NaN	NaN	0.0020	0.0018
$\mathbb{E}|Y|^4$	IPLA	0.0054	0.0039	0.0025	0.0036
	TULA	0.0095	0.0032	0.0073	0.0039
	ULA	NaN	NaN	0.0028	0.0035
$\mathbb{E}|Y|^6$	IPLA	0.0081	0.0058	0.0047	0.0054
	TULA	0.0144	0.0047	0.0120	0.0058
	ULA	NaN	NaN	0.0032	0.0053

Table 1: Estimation of the moments of light tails distribution from Example 1.

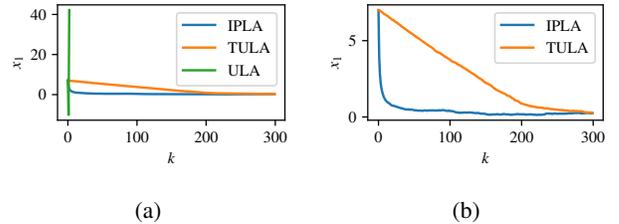(a)   (b)

Figure 1: Trajectory of the 1st coordinate from Example 1 starting in a tail. Both plots are based on the same data.
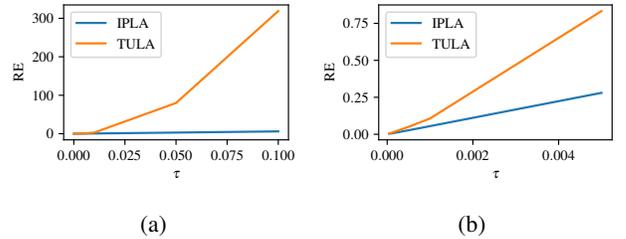

(a)   (b)

Figure 2: Dependence RE of IPLA and TULA (for Example 1) on a stepsize τ starting in a tail.

where $\widetilde{\nabla}x_{ijk} = \left(x_{i+jk}-x_{ijk}, x_{ij+k}-x_{ijk}, x_{ijk+}-x_{ijk}\right)$, with the notation such that $i_\pm := i \pm 1 \bmod q$ and similarly for j, k. We consider $\varkappa = 0.1$, $\varsigma = 0.5$, $\upsilon = 2$ and $q = 5$. The dimension in this example is $d = q^3 = 125$. Note that ∇V is not globally Lipschitz and V is λ_V-convex for $\lambda_V > 0$, (Brosse et al. 2019). We have run 2×10^4 samples with burn-in time 10^4. We consider two scenarios as in Example 1. As starting in tail, we consider the case $x_0 = (100, 0, \ldots, 0)$ and starting in the minimizer we take $x_0 = 0$. Each experiment has been repeated 100 times. From Table 2 we see that the results are analogous to those of Example 1. In the scenario starting in the minimizer, we have results similar to ULA's and better than TULA's. In the case of the start in tail, ULA blows up.
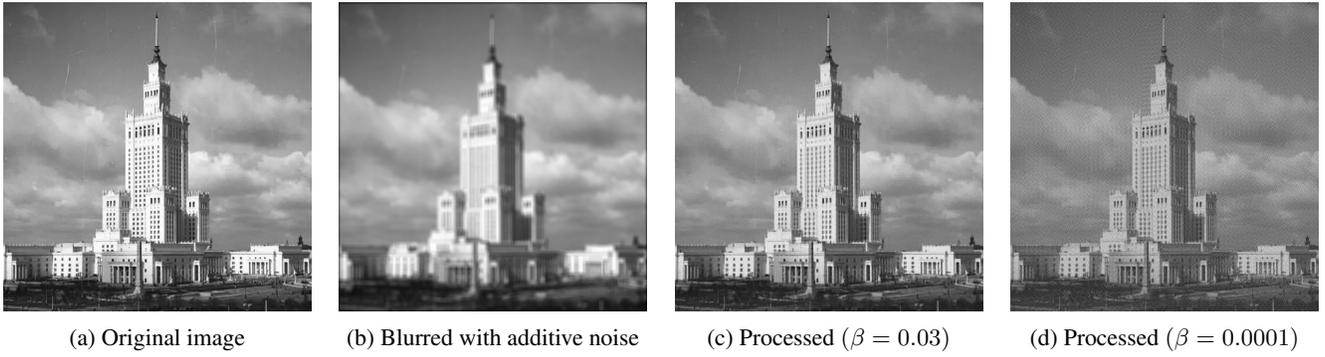
| (a) Original image | (b) Blurred with additive noise | (c) Processed ($\beta = 0.03$) | (d) Processed ($\beta = 0.0001$) |

Figure 3: Result of the Bayesian Image Denoising from Example 3. The original photo by Zbyszko Siemaszko 1955-56.

Mom.	Method	Start in tail		Start in $x_0 = 0$	
		RE	CV	RE	CV
$\mathbb{E}|Y|^2$	IPLA	0.0025	0.0244	0.0748	0.0786
	TULA	0.0067	0.0213	0.0859	0.0739
	ULA	NaN	NaN	0.0727	0.0697
$\mathbb{E}|Y|^4$	IPLA	0.0053	0.0491	0.1425	0.1558
	TULA	0.0134	0.0425	0.1635	0.1473
	ULA	NaN	NaN	0.1385	0.1399
$\mathbb{E}|Y|^6$	IPLA	0.0083	0.0742	0.2040	0.2323
	TULA	0.0199	0.0638	0.2337	0.2208
	ULA	NaN	NaN	0.1980	0.2117

Table 2: Estimation of moments of Ginzburg–Landau model from Example 2.

Example 3: Bayesian Image Deconvolution

We show now an example that IPLA is effective in complex high-dimensional scenarios. We consider the Bayesian Image Deconvolution problem inspired by Experiment 1 in Durmus, Moulines, and Pereyra (2018, Subsection 4.1.2). Given a high-resolution grayscale image $x \in \mathbb{R}^d$ such that $d = n \times n = 600 \times 600$ pixels from Figure 3 (a), the goal of the problem is to recover image x from a blurred image. Namely, we are to sample from posterior density $p(x|y)$ of blurred image with additive noise (observation) such that

$$y = Hx + W \text{ for } W \sim \mathcal{N}(0, \sigma^2 \mathsf{Id}),$$

where H is a circulant blur matrix representing 2D discrete convolution and W is an additive noise.

We need to define 2D discrete isotropic total variation via

$$TV(x) := \sum_{i,j=1}^{n-1} \sqrt{|x_{in,j+1} - x_{in,j}|^2 + |x_{(i+1)n,j} - x_{in,j}|^2}$$
$$+ \sum_{j=1}^{n-1} |x_{nd,j+1} - x_{nd,j}| + \sum_{i=1}^{n-1} |x_{(i+1)n,n} - x_{in,n}|,$$

which was proposed in the context of image processing by Rudin, Osher, and Fatemi (1992). Note that TV is a discretization of the standard total variation norm, cf. Chambolle (2004, Section 2).

We choose prior distribution $p(x) \propto \exp(-\beta\, TV(x))$, where $\beta > 0$ is a regularizing factor. Henceforth, we provide sampling from the posterior distribution $\mu^* := p(x|y)$ satisfying

$$\mu^* \propto \exp\left(-\frac{1}{2\sigma^2}|y - Hx|^2 - \beta\, TV(x)\right).$$

Even if TV is not smooth, and consequently the theoretical convergence rates are not known for IPLA, TULA, or ULA, we show the performance of our algorithm. It might be expected that the inexact proximal step could be non-trivial in this case. Nonetheless, it can be effectively solved with the use of the Adaptive Primal–Dual Hybrid Algorithm by Goldstein, Li, and Yuan (2015, Algorithm 1).

Our results are presented on Figure 3. On (b) we show blurred and noisy image y, where uniform circular blur matrix has depth 9 and additive noise is of the standard deviation $\sigma = 0.5$. On (c) and (d) we present posterior mean obtained by IPLA for 10^5 iterations and with the desired precision of the inexact proximal step $\delta = 10^{-1}$.

Effective Implementation of IPLA

Despite involving the inexact proximal step, the computational complexity of IPLA is controlled. In Examples 1 and 2, we used the Newton conjugate gradient method with the provided analytically computed gradient and the Hessian matrix, making use of the sparsity of the Hessian matrix. Using the standard Python library `SciPy` (Virtanen et al. 2020), IPLA has comparable performance to ULA. We write the time of computations on standard Macbook Air M1 2020. In Example 1 a single run of 10^6 iterations took from 5 to 20 s for ULA or TULA and from 15 to 60 s, while in Example 2, all algorithms took from 3 to 5 min to run 10^4 iterations. In Example 3, the potential is not differentiable and we modify the existing implementation in the `PyProximal` library (Ravasi et al. 2024). Setting the initial value in the minimizer of V reduces the cost of the inexact proximal step. Let us stress that in our simulation of Example 3, 10^5 samples in the dimension $d = 360\,000$ are obtained in less than 1 min.

Conclusion

IPLA broadens the scope of problems that known LMC algorithms can address keeping low computational cost.

Acknowledgements

References

Ambrosio, L.; Gigli, N.; and Savaré, G. 2008. *Gradient flows in metric spaces and in the space of probability measures*. Lectures in Mathematics ETH Zürich. Birkhäuser Verlag, Basel, second edition. ISBN 978-3-7643-8721-1.

Andrieu, C.; de Freitas, N.; Doucet, A.; and Jordan, M. I. 2003. An Introduction to MCMC for Machine Learning. *Machine Learning*, 50(1): 5–43.

Beck, A. 2017. *First-order methods in optimization*, volume 25 of *MOS-SIAM Series on Optimization*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA; Mathematical Optimization Society, Philadelphia, PA. ISBN 978-1-611974-98-0.

Benko, M.; Chlebicka, I.; Endal, J.; and Miasojedow, B. 2024. Convergence rates of particle approximation of forward-backward splitting algorithm for granular medium equations. arXiv:2405.18034.

Bernton, E. 2018. Langevin Monte Carlo and JKO splitting. In Bubeck, S.; Perchet, V.; and Rigollet, P., eds., *Proceedings of the 31st Conference On Learning Theory*, volume 75 of *Proceedings of Machine Learning Research*, 1777–1798. PMLR.

Brosse, N.; Durmus, A.; Moulines, E.; and Sabanis, S. 2019. The tamed unadjusted Langevin algorithm. *Stochastic Process. Appl.*, 129(10): 3638–3663.

Cattiaux, P.; Guillin, A.; and Malrieu, F. 2008. Probabilistic approach for granular media equations in the non-uniformly convex case. *Probab. Theory Related Fields*, 140(1-2): 19–40.

Chambolle, A. 2004. An Algorithm for Total Variation Minimization and Applications. *Journal of Mathematical Imaging and Vision*, 20(1): 89–97.

Chewi, S.; Erdogdu, M. A.; Li, M.; Shen, R.; and Zhang, S. 2022. Analysis of Langevin Monte Carlo from Poincare to Log-Sobolev. In Loh, P.-L.; and Raginsky, M., eds., *Proceedings of Thirty Fifth Conference on Learning Theory*, volume 178 of *Proceedings of Machine Learning Research*, 1–2. PMLR.

Cover, T.; and Thomas, J. 2012. *Elements of Information Theory*. Wiley. ISBN 9781118585771.

Dalalyan, A. 2017a. Further and stronger analogy between sampling and optimization: Langevin Monte Carlo and gradient descent. In Kale, S.; and Shamir, O., eds., *Proceedings of the 2017 Conference on Learning Theory*, volume 65 of *Proceedings of Machine Learning Research*, 678–689. PMLR.

Dalalyan, A. S. 2017b. Theoretical guarantees for approximate sampling from smooth and log-concave densities. *J. R. Stat. Soc. Ser. B. Stat. Methodol.*, 79(3): 651–676.

Dalalyan, A. S.; Karagulyan, A.; and Riou-Durand, L. 2022. Bounding the error of discretized Langevin algorithms for non-strongly log-concave targets. *J. Mach. Learn. Res.*, 23: Paper No. [235], 38.

Dalalyan, A. S.; and Tsybakov, A. B. 2012. Sparse regression learning by aggregation and Langevin Monte-Carlo. *J. Comput. System Sci.*, 78(5): 1423–1443.

Durmus, A.; Majewski, S.; and Miasojedow, B. 2019. Analysis of Langevin Monte Carlo via convex optimization. *J. Mach. Learn. Res.*, 20: Paper No. 73, 46.

Durmus, A.; and Moulines, E. 2017. Nonasymptotic convergence analysis for the unadjusted Langevin algorithm. *Ann. Appl. Probab.*, 27(3): 1551–1587.

Durmus, A.; and Moulines, E. 2019. High-dimensional Bayesian inference via the unadjusted Langevin algorithm. *Bernoulli*, 25(4A): 2854–2882.

Durmus, A.; Moulines, E.; and Pereyra, M. 2018. Efficient Bayesian Computation by Proximal Markov Chain Monte Carlo: When Langevin Meets Moreau. *SIAM Journal on Imaging Sciences*, 11(1): 473–506.

Erdogdu, M. A.; and Hosseinzadeh, R. 2021. On the Convergence of Langevin Monte Carlo: The Interplay between Tail Growth and Smoothness. In Belkin, M.; and Kpotufe, S., eds., *Proceedings of Thirty Fourth Conference on Learning Theory*, volume 134 of *Proceedings of Machine Learning Research*, 1776–1822. PMLR.

Erdogdu, M. A.; Hosseinzadeh, R.; and Zhang, S. 2022. Convergence of Langevin Monte Carlo in Chi-Squared and Rényi Divergence. In Camps-Valls, G.; Ruiz, F. J. R.; and Valera, I., eds., *Proceedings of The 25th International Conference on Artificial Intelligence and Statistics*, volume 151 of *Proceedings of Machine Learning Research*, 8151–8175. PMLR.

Gelman, A.; Carlin, J. B.; Stern, H. S.; Dunson, D. B.; Vehtari, A.; and Rubin, D. B. 2014. *Bayesian data analysis*. Texts in Statistical Science Series. CRC Press, Boca Raton, FL, third edition. ISBN 978-1-4398-4095-5.

Goldenfeld, N. 1992. *Lectures On Phase Transitions And The Renormalization Group (1st ed.)*. CRC Press.

Goldstein, T.; Li, M.; and Yuan, X. 2015. Adaptive Primal-Dual Splitting Methods for Statistical Learning and Image Processing. In Cortes, C.; Lawrence, N.; Lee, D.; Sugiyama, M.; and Garnett, R., eds., *Advances in Neural Information Processing Systems*, volume 28. Curran Associates, Inc.

Jordan, R.; Kinderlehrer, D.; and Otto, F. 1998. The variational formulation of the Fokker-Planck equation. *SIAM J. Math. Anal.*, 29(1): 1–17.

Krauth, W. 2006. *Statistical mechanics*, volume 13 of *Oxford Master Series in Physics*. Oxford University Press, Oxford. ISBN 978-0-19-851536-4; 0-19-851536-7. Algorithms and computations, Oxford Master Series in Statistical Computational, and Theoretical Physics.

Livingstone, S.; Nüsken, N.; Vasdekis, G.; and Zhang, R.-Y. 2024. Skew-symmetric schemes for stochastic differential equations with non-Lipschitz drift: an unadjusted Barker algorithm. arXiv:2405.14373.

Mattingly, J. C.; Stuart, A. M.; and Higham, D. J. 2002. Ergodicity for SDEs and approximations: locally Lipschitz vector fields and degenerate noise. *Stochastic Process. Appl.*, 101(2): 185–232.

Mousavi-Hosseini, A.; Farghly, T. K.; He, Y.; Balasubramanian, K.; and Erdogdu, M. A. 2023. Towards a Complete Analysis of Langevin Monte Carlo: Beyond Poincaré Inequality. In Neu, G.; and Rosasco, L., eds., *Proceedings of Thirty Sixth Conference on Learning Theory*, volume 195 of *Proceedings of Machine Learning Research*, 1–35. PMLR.

Øksendal, B. 2010. *Stochastic Differential Equations: An Introduction with Applications*. Universitext. Springer Berlin Heidelberg. ISBN 9783642143946.

Ravasi, M.; Örnhag, M. V.; Luiken, N.; Leblanc, O.; and Uruñuela, E. 2024. PyProximal - scalable convex optimization in Python. *Journal of Open Source Software*, 9(95): 6326.

Robert, C. P.; and Casella, G. 2004. *Monte Carlo statistical methods*. Springer Texts in Statistics. Springer-Verlag, New York, second edition. ISBN 0-387-21239-6.

Roberts, G. O.; and Tweedie, R. L. 1996. Exponential convergence of Langevin distributions and their discrete approximations. *Bernoulli*, 2(4): 341–363.

Rudin, L. I.; Osher, S.; and Fatemi, E. 1992. Nonlinear total variation based noise removal algorithms. *Physica D: Nonlinear Phenomena*, 60(1): 259–268.

Salim, A.; Korba, A.; and Luise, G. 2020. The Wasserstein Proximal Gradient Algorithm. In Larochelle, H.; Ranzato, M.; Hadsell, R.; Balcan, M.; and Lin, H., eds., *Advances in Neural Information Processing Systems*, volume 33, 12356–12366. Curran Associates, Inc.

Santambrogio, F. 2017. Euclidean, metric, and Wasserstein gradient flows: an overview. *Bull. Math. Sci.*, 7(1): 87–154.

Stuart, A. M. 2010. Inverse problems: a Bayesian perspective. *Acta Numer.*, 19: 451–559.

Vempala, S.; and Wibisono, A. 2019. Rapid Convergence of the Unadjusted Langevin Algorithm: Isoperimetry Suffices. In Wallach, H.; Larochelle, H.; Beygelzimer, A.; d'Alché-Buc, F.; Fox, E.; and Garnett, R., eds., *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.

Villani, C. 2009. *Optimal transport. Old and new*, volume 338 of *Grundlehren der mathematischen Wissenschaften [Fundamental Principles of Mathematical Sciences]*. Springer-Verlag, Berlin. ISBN 978-3-540-71049-3.

Virtanen, P.; Gommers, R.; Oliphant, T. E.; Haberland, M.; Reddy, T.; Cournapeau, D.; Burovski, E.; Peterson, P.; Weckesser, W.; Bright, J.; van der Walt, S. J.; Brett, M.; Wilson, J.; Millman, K. J.; Mayorov, N.; Nelson, A. R. J.; Jones, E.; Kern, R.; Larson, E.; Carey, C. J.; Polat, İ.; Feng, Y.; Moore, E. W.; VanderPlas, J.; Laxalde, D.; Perktold, J.; Cimrman, R.; Henriksen, I.; Quintero, E. A.; Harris, C. R.; Archibald, A. M.; Ribeiro, A. H.; Pedregosa, F.; van Mulbregt, P.; and SciPy 1.0 Contributors. 2020. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, 17: 261–272.