Proceedings of the 2022 Winter Simulation Conference
B. Feng, G. Pedrielli, Y. Peng, S. Shashaani, E. Song, C.G. Corlu, L.H. Lee, E.P. Chew, T. Roeder, and P. Lendermann, eds.

A PROXIMAL ALGORITHM FOR SAMPLING FROM NON-SMOOTH POTENTIALS

Jiaming Liang

Department of Computer Science
Yale University
New Haven, CT 06511, USA

Yongxin Chen

School of Aerospace Engineering
Georgia Institute of Technology
Atlanta, GA 30332, USA

ABSTRACT

In this work, we examine sampling problems with non-smooth potentials and propose a novel Markov chain Monte Carlo algorithm for it. We provide a non-asymptotical analysis of our algorithm and establish a polynomial-time complexity $\tilde{\mathcal{O}}(M^2 d \mathscr{M}_4^{1/2} \varepsilon^{-1})$ to achieve ε error in terms of total variation distance to a log-concave target density with 4th moment $\mathscr{M}_4$ and M-Lipschitz potential, better than most existing results under the same assumptions. Our method is based on the proximal bundle method and an alternating sampling framework. The latter framework requires the so-called restricted Gaussian oracle, which can be viewed as a sampling counterpart of the proximal mapping in convex optimization. One key contribution of this work is a fast algorithm that realizes the restricted Gaussian oracle for any convex non-smooth potential with bounded Lipschitz constant.

1 INTRODUCTION

Core to many scientific and engineering problems that face uncertainty is the task of drawing samples from a given, often unnormalized, probability density. Sampling plays a crucial role in many applications such as statistical inference/estimation, operations research, physics, biology, and machine learning, etc (Bertsimas and Vempala 2004; Durmus et al. 2018; Dyer et al. 1991; Gelman et al. 2013; Kannan et al. 1997; Krauth 2006; Sites Jr and Marshall 2003). For instance, in Bayesian statistics, we can sample from the posterior distribution to infer its mean, covariance, or other important statistics.

A very popular framework for sampling from high dimensional complex distributions is the Markov chain Monte Carlo (MCMC) method (Chen et al. 2018; Cheng and Bartlett 2018; Cheng et al. 2018; Durmus et al. 2019; Durmus et al. 2018). Several widely used MCMC methods include Langevin Monte Carlo (LMC) (Dalalyan 2017; Grenander and Miller 1994; Parisi 1981; Roberts and Tweedie 1996), Metropolis-adjusted Langevin algorithm (MALA) (Bou-Rabee and Hairer 2013; Roberts and Stramer 2002; Roberts and Tweedie 1996), and Hamiltonian Monte Carlo (HMC) (Neal 2011). These three methods use gradient information of the potential (log-density) to construct the Markov chain. They resemble the gradient-based algorithms in optimization and can be viewed as the sampling counterparts of them. Over the last few years, many theoretical results (Chen et al. 2020; Dalalyan 2017; Durmus et al. 2019; Dwivedi et al. 2019; Lee et al. 2021; Lee and Vempala 2017; Roberts and Rosenthal 1998; Roberts and Tweedie 1996) have been established to understand the computational complexities of these MCMC algorithms.

Most existing gradient-based MCMC methods are only applicable to settings with smooth potentials (Dalalyan 2017; Lee et al. 2021; Bernton 2018b) whose gradient is Lipschitz continuous. However, non-smooth sampling is also an important problem as many applications of sampling involve non-smooth potentials. For instance, in Bayesian image deconvolution with total-variation and ℓ_1 priors (Durmus et al. 2018), the posterior distribution $\pi^{X|Z}(x \mid z)$ given by

$$\pi^{X|Z}(x \mid z) \propto \exp\left(-\frac{\|z - Hx\|^2}{2\sigma^2} - \beta \|x\|_{\mathrm{TV}}\right)$$

has a non-smooth potential, where $\sigma > 0$, $\beta > 0$ and H is a convolution kernel. Many problems in deep learning are also non-smooth, not only due to non-smooth activation functions like ReLU used in the neural networks, but also due to intrinsic scaling symmetries. Nevertheless, the study of sampling without smoothness is nascent. This is in sharp contrast to optimization where a plethora of algorithms, e.g., subgradient method, proximal algorithm, bundle method have been developed for non-smooth optimization (Lemaréchal 1975; Liang and Monteiro 2021; Mifflin 1982; Rockafellar 1976; Wolfe 1975).

The goal of this work is to establish an efficient algorithm to draw samples from a distribution with non-smooth potential. We focus on the case where the potential is convex and Lipschitz continuous. Our algorithm is based on the alternative sampling framework (ASF) (Lee et al. 2021), which can be viewed as a sampling counterpart of the proximal point method in optimization (Rockafellar 1976). The key of the ASF is a step known as the restricted Gaussian oracle (RGO) (see Definition 1) to draw samples from a potential regularized by a large isotropic quadratic term. To utilize this framework to sample from general non-smooth potentials with bounded Lipschitz constants, we develop an efficient realization of the RGO through rejection sampling with a properly designed proposal. A non-smooth optimization technique known as the proximal bundle method (Lemaréchal 1975; Liang and Monteiro 2021) is used to compute the proposal. We establish a polynomial-time complexity $\tilde{\mathcal{O}}(M^2 d \mathscr{M}_4^{1/2} \varepsilon^{-1})$ to achieve ε total variation distance to a log-concave target density with 4th moment $\mathscr{M}_4$ and M-Lipschitz potential, better than most existing results under the same assumptions.

A key contribution of this paper is a fast algorithm for implementing RGO for convex non-smooth functions. When the potential g is decomposable, e.g., g is an ℓ_1 norm or an indicator function of an orthant, there exists simple sampling algorithms for RGO (Mou et al. 2022). In general, the implementation of RGO is a difficult algorithmic task, which makes ASF (Lee et al. 2021) a conceptual method without implementable algorithms in most cases. Our algorithm of implementing the RGO for any convex non-smooth function broads the applicability of the ASF significantly. In fact, our algorithm for RGO can be used for any framework, not only ASF, that requires sampling from $\exp(-g(x) - \frac{1}{2\eta}\|x-y\|^2)$ for any y and some proper $\eta > 0$. From an optimization point of view, our algorithm for RGO provides an efficient realization of the proximal oracle for a wide range of functions/potentials, solidifying the connections between the ongoing research at the interface of optimization and sampling (Bernton 2018b).

(**?**) Over the last few years, several new algorithms and theoretical results in sampling with non-smooth potentials have been established. In (Mou et al. 2022), sampling for non-smooth composite potentials is considered. The algorithm needs the proximal sampling oracle that samples from the target potential regularized by a large isotropic quadratic term as well as computes the corresponding partition function, which is not realistic for general potentials. In (Lee et al. 2021), algorithms to sample from non-smooth composite potentials are developed; both are based on the RGO which is similar to the proximal sampling oracle but do not need to compute the partition function. In (Chatterji et al. 2020), the authors developed an algorithm to sample from non-smooth potentials by running LMC on the Gaussian smoothing of the potentials. In (Lehec 2021), the author developed the projected LMC algorithm and analyzed its complexity for non-smooth potentials. In (Freund et al. 2022), the authors developed a new analysis that leads to dimension-free complexity for sampling from a composite density which contains a non-smooth component. In (Durmus et al. 2019), the authors presented an optimization approach to analyze the complexity of sampling and established a complexity result for sampling with non-smooth composite potentials. In (Lee and Vempala 2017), the authors studied the complexity of the ball walk to sample from an isotropic logconcave density from a warm start. In (Bernton 2018a; Wibisono 2019), a proximal algorithm was proposed. This algorithm resembles the ASF for sampling with a major difference that the RGO is replaced by proximal point optimization step, which introduces bias for sampling.

To compare our results with (Bernton 2018a) and (Freund et al. 2022), consider sampling from $\exp(-f(x) - \mu\|x\|^2/2)$ where f is convex and M-Lipschitz continuous. Our complexity (see Theorem 5) is $\tilde{\mathcal{O}}(M^2 d/\mu)$, better than $\mathcal{O}(M^2/(\mu\varepsilon^2))$ in (Freund et al. 2022) and $\tilde{\mathcal{O}}(M^2 d/(\mu\varepsilon^4))$ (albeit in Wasserstein distance) (Bernton 2018a) when $\varepsilon < d^{-1/2}$. For sampling from non-smooth potentials, compared with

(Mou et al. 2022; Lee et al. 2021), our algorithm does not require any sampling oracle. Compared with (Lehec 2021), we consider sampling from a distribution supported on $\mathbb{R}^d$ instead of a convex compact set. Compared with (Chatterji et al. 2020; Durmus et al. 2019), our algorithm has better complexity in terms of total variation when the target error ε is small. Compared with (Lee and Vempala 2017), we consider a generic setting where the target distribution can be anisotropic, and our complexity is in general better in the low resolution region. See Table 1 for the detailed complexity bounds. Note that complexity results obtained in (Chatterji et al. 2020; Lee and Vempala 2017) and this paper are for the last iterate, while the bound established in (Durmus et al. 2019) (also (Freund et al. 2022)) is for the average of all iterates.

Table 1: Complexity bounds for sampling from non-smooth densities.

(Chatterji et al. 2020)	(Durmus et al. 2019)	(Lee and Vempala 2017)	this paper
$\tilde{\mathscr{O}}(M^6 d^5 \mathscr{M}_4^{3/2} \varepsilon^{-10})$	$\mathscr{O}(M^2 W_2^2 \varepsilon^{-4})$	$\mathscr{O}(d^{5/2}\log(\beta/\varepsilon))$	$\tilde{\mathscr{O}}(M^2 d \mathscr{M}_4^{1/2} \varepsilon^{-1})$

In Table 1, $\mathscr{M}_4$ denotes the finite 4th moment of the target distribution, W_2 denotes the Wasserstein distance between the initial and target distributions, and β denotes the warmness of the initial distribution. More specifically, $\mathscr{M}_4 \approx d^2$ in the isotropic case, $\log\beta \approx d$ if the initial distribution is not warm started, and $W_2 \approx \sqrt{d}$ in general. Under these simplifications, our bound is $\tilde{\mathscr{O}}(M^2 d^2 \varepsilon^{-1})$, better than $\tilde{\mathscr{O}}(M^6 d^8 \varepsilon^{-10})$ in (Chatterji et al. 2020), $\mathscr{O}(M^2 d \varepsilon^{-4})$ in (Durmus et al. 2019), and $\tilde{\mathscr{O}}(d^{7/2})$ in (Lee and Vempala 2017) when d is large and ε is relatively small. Note that, for typical problems, $M = \mathscr{O}(d^{1/2})$, but sparsity maybe exploited to improve this dependence. In any case, our method has better dependence on the dimensional d than (Lee and Vempala 2017) whose complexity scales as $d^{7/2}$ when warm start is not available.

For $\mu > 0$, $f : \mathbb{R}^d \to \mathbb{R}$ is said to be μ-strongly convex if f satisfies

$$f(u) - f(v) - \langle f'(v), u - v\rangle \geq \frac{\mu}{2}\|u - v\|^2, \quad \forall u, v \in \mathbb{R}^d,$$

where f' denotes a subgradient of f. For $M > 0$, $f : \mathbb{R}^d \to \mathbb{R}$ is said to be M-Lipschitz continuous if

$$\|f'(u) - f'(v)\| \leq M\|u - v\|, \quad \forall u, v \in \mathbb{R}^d.$$

The rest of this paper is structured as follows. In Section 2 we provide the problem formulation. We also briefly review ASF on which our algorithm is based. In Section 3 we present our key contribution, an efficient realization of the RGO for general convex potentials with bounded Lipschitz constants. This is then combined with the ASF in Section 4 to establish our results for sampling without smoothness. Finally, we present some concluding remarks and possible extensions in Section 5.

2 PROBLEM FORMULATION AND ALTERNATING SAMPLING FRAMEWORK

The problem of interest is to sample from a distribution on $\mathbb{R}^d$ proportional to $\exp(-f(x))$ where the potential f is **convex and M-Lipschitz continuous**. Note that the potential f does not need to be smooth. This violates the smoothness assumption for most existing gradient-based MCMC sampling methods (Lee et al. 2021; Bernton 2018b).

Our method is built on the ASF introduced in (Lee et al. 2021) (a similar method was developed in (Vono et al. 2022)), which is a generic framework for sampling from a distribution $\pi^X \propto \exp(-g(x))$. For a stepsize $\eta > 0$, starting from a given point $x \in \mathbb{R}^d$, the ASF repeats the two steps as in Algorithm 1.

Algorithm 1 Alternating Sampling Framework

1. Sample $y_k \sim \pi^{Y|X}(y \mid x_k) \propto \exp[-\frac{1}{2\eta}\|x_k - y\|^2]$
2. Sample $x_{k+1} \sim \pi^{X|Y}(x \mid y_k) \propto \exp[-g(x) - \frac{1}{2\eta}\|x - y_k\|^2]$

Clearly, the ASF is itself a special case of Gibbs sampling to sample from the joint distribution

$$\pi(x,y) \propto \exp\left[-g(x) - \frac{1}{2\eta}\|x-y\|^2\right].$$

In Algorithm 1, sampling y in step 1 can be easily done since $\pi^{Y|X}$ is a Gaussian distribution. Sampling x given y in step 2 corresponds to the restricted Gaussian oracle for g introduced in (Lee et al. 2021).

Definition 1 Given a point $y \in \mathbb{R}^d$ and stepsize $\eta > 0$, a restricted Gaussian oracle (RGO) for convex $g : \mathbb{R}^d \to \mathbb{R}$ is a sampling oracle that returns a sample from a distribution proportional to $\exp(-g(\cdot) - \|\cdot - y\|^2/(2\eta))$.

RGO is an analogy of the proximal mapping in convex optimization, which is heavily used in proximal point methods. RGO is a key algorithmic ingredient used in (Lee et al. 2021) together with the alternating sampling framework to improve the iteration-complexity bounds for various sampling algorithms. Examples of a convex function g that admits an computationally efficient RGO have been presented in (Mou et al. 2022; Lee et al. 2021), including coordinate-separable regularizers, ℓ_1-norm, and group Lasso.

We recall the main result of (Lee et al. 2021), which gives the complexity of Algorithm 1 in terms of number of calls to RGO and is useful in this paper.

Theorem 1 (Theorem 1 of (Lee et al. 2021)) Let π^X be a distribution on $\mathbb{R}^d$ with $\pi^X(x) \propto \exp\left(-f_{\text{oracle}}\ (x)\right)$ such that f_{oracle} is μ-strongly convex, and let $\varepsilon \in (0,1)$. Let $\eta \le 1/\mu, T = \Theta\left(\frac{1}{\eta\mu}\log\frac{d}{\eta\mu\varepsilon}\right)$. Algorithm 1, initialized at the minimizer of f_{oracle}, runs in T iterations, each querying RGO for f_{oracle} with parameter η a constant number of times, and obtains ε total variation distance to π^X.

Note that the above minimizer can be an approximate solution, as long as $\|\hat{x} - x_{\text{opt}}\|^2 \le d/\mu$ where $\hat{x}$ and x_{opt} are the approximate and exact solutions, respectively.

We now describe our approach to sample from non-smooth potentials. We first consider a regularized density $\exp(-g(x))$ where $g(x) = f(x) + \mu\|x - x^0\|^2/2$ and $x^0 \in \mathbb{R}^d$ is an arbitrary point but preferred to be close to the minimum set of g. Since g is μ-strongly convex, Algorithm 1 and Theorem 1 are applicable. We then develop an efficient implementation of the RGO based on the proximal bundle method and rejection sampling for an arbitrary convex potential with bounded Lipschitz constant. This is the main contribution of this work. Finally, we justify the sample generated from $\exp(-g(x))$ by using Algorithm 1 and the implementable RGO with a proper choice of μ is a sample within ε total variation distance to the target density $\exp(-f(x))$.

3 KEY RESULT: AN IMPLEMENTABLE RESTRICTED GAUSSIAN ORACLE

The bottleneck of applying the alternating sampling framework (Algorithm 1) to sample from general log-concave distributions is the availability of RGO. In this section, we address this issue by designing a computationally efficient and implementable RGO for μ-strongly convex g of the form

$$g = f + \frac{\mu}{2}\|\cdot - x^0\|^2. \tag{1}$$

Our algorithm of RGO for g is based on rejection sampling but with a specially designed proposal. With this proposal and a sufficiently large regularization parameter, the expected number of rejection sampling steps to obtain one effective sample turns out to be bounded above by a dimension-free constant. To bound the complexity of the rejection sampling, we develop a novel technique to estimate a modified Gaussian integral (see Proposition 3).

Let

$$x^* = \operatorname*{argmin}_{x\in\mathbb{R}^d}\left\{g_y^\eta(x) := g(x) + \frac{1}{2\eta}\|x-y\|^2\right\}, \tag{2}$$

where y is the output of step 1 in Algorithm 1. Note that solving (2) is equivalent to invoking one proximal mapping of f since $g(x) = f(x) + \mu\|x - x^0\|^2/2$; the quadratic term can be combined with that in (2).

However, the proximal mapping of f is not available in general, for example, f is the maximum of finite number of affine functions. In this work we only assume f is convex and M-Lipschitz continuous but not the availability of the proximal mapping of f.

The RGO in each iteration requires sampling from $\exp(-g_y^\eta)$. Our strategy is rejection sampling with a proper Gaussian proposal centered at an approximate solution to (2) close to x^*. Since (2) is a non-smooth optimization, we use Algorithm 3 (see Section A for more details), i.e., the proximal bundle method, to generate two approximate solutions x_J and $\tilde{x}_J$, where J denotes the last iteration index of Algorithm 3. We define the following two functions that are crucial in the RGO sampling algorithm and its analysis, namely,

$$h_1 := \frac{1}{2\eta_\mu}\|\cdot - x_J\|^2 + g_y^\eta(\tilde{x}_J) - \delta, \tag{3}$$

$$h_2 := \frac{1}{2\eta_\mu}\|\cdot - \tilde{x}_J\|^2 + \left(2M + \frac{\sqrt{2\delta}}{\sqrt{\eta_\mu}}\right)\|\cdot - \tilde{x}_J\| + g_y^\eta(\tilde{x}_J), \tag{4}$$

where

$$\eta_\mu := \frac{\eta}{1+\eta\mu}.$$

The potential g_y^η in RGO is bounded below and above by these two functions as in Lemma 2 below. This naturally leads to the rejection sampling algorithm (Algorithm 2).

Algorithm 2 RGO Rejection Sampling

1. Compute x_J and $\tilde{x}_J$ as in Algorithm 3;
2. Generate $X \sim \exp(-h_1(x))$;
3. Generate $U \sim \mathscr{U}[0,1]$;
4. If
$$U \le \frac{\exp(-g_y^\eta(X))}{\exp(-h_1(X))},$$
then accept/return X; otherwise, reject X and go to step 2.

Lemma 2 Assume f is convex and M-Lipschitz continuous. Let g and g_y^η be as in (1) and (2), respectively. Then, for every $x \in \mathbb{R}^d$, we have

$$h_1(x) \le g_y^\eta(x) \le h_2(x) \tag{5}$$

where h_1 and h_2 are as in (3) and (4), respectively.

Proof. Using Lemma 8(a)-(b) and the definition of g_J^η in (18) in Appendix A , we have

$$\begin{aligned} g(\tilde{x}_J) - g(x) + \frac{1}{2\eta_\mu}\|x - x_J\|^2 &\le g(\tilde{x}_J) - g_J(x) + \frac{1}{2\eta_\mu}\|x - x_J\|^2 \\ \le g(\tilde{x}_J) - g_J^\eta(x_J) + \frac{1}{2\eta}\|x - y\|^2 &\le \delta - \frac{1}{2\eta}\|\tilde{x}_J - y\|^2 + \frac{1}{2\eta}\|x - y\|^2. \end{aligned}$$

The first inequality in (5) holds in view of the definition of h_1 in (3). Using the definition of g_y^η in (2) and the fact that f is M-Lipschtz, we have

$$\begin{aligned} g_y^\eta(x) - g_y^\eta(\tilde{x}_J) &= f(x) - f(\tilde{x}_J) + \frac{\mu}{2}\|x - x^0\|^2 - \frac{\mu}{2}\|\tilde{x}_J - x^0\|^2 + \frac{1}{2\eta}\|x - y\|^2 - \frac{1}{2\eta}\|\tilde{x}_J - y\|^2 \\ &\le M\|x - \tilde{x}_J\| + \frac{\mu}{2}\|x - \tilde{x}_J\|^2 + \mu\langle x - \tilde{x}_J, \tilde{x}_J - x^0\rangle + \frac{1}{2\eta}\|x - \tilde{x}_J\|^2 + \frac{1}{\eta}\langle x - \tilde{x}_J, \tilde{x}_J - y\rangle \\ &= M\|x - \tilde{x}_J\| + \frac{1}{2\eta_\mu}\|x - \tilde{x}_J\|^2 + \mu\langle x - \tilde{x}_J, \tilde{x}_J - x_J + x_J - x^0\rangle + \frac{1}{\eta}\langle x - \tilde{x}_J, \tilde{x}_J - x_J + x_J - y\rangle. \end{aligned}$$

The above inequality, the Cauchy-Schwarz inequality and Lemma 8(c)-(d) imply that

$$\begin{aligned} g_y^\eta(x) - g_y^\eta(\tilde{x}_J) &\le M\|x-\tilde{x}_J\| + \frac{1}{2\eta_\mu}\|x-\tilde{x}_J\|^2 + \frac{1}{\eta_\mu}\|x-\tilde{x}_J\|\|\tilde{x}_J - x_J\| + \|x-\tilde{x}_J\|\left\|\mu(x_J-x^0) + \frac{x_J-y}{\eta}\right\| \\ &\le M\|x-\tilde{x}_J\| + \frac{1}{2\eta_\mu}\|x-\tilde{x}_J\|^2 + \frac{\sqrt{2\delta}}{\sqrt{\eta_\mu}}\|x-\tilde{x}_J\| + M\|x-\tilde{x}_J\| \\ &= \left(2M + \frac{\sqrt{2\delta}}{\sqrt{\eta_\mu}}\right)\|x-\tilde{x}_J\| + \frac{1}{2\eta_\mu}\|x-\tilde{x}_J\|^2. \end{aligned}$$

It follows from the above inequality and the definition of h_2 in (4) that the second inequality in (5) holds. □

From the expression of h_1 in (3), it is clear the proposal distribution is a Gaussian centered at x_J. To achieve a tight bound on the expected runs of the rejection sampling, we use a function h_2 which is not purely quadratic; the standard choice of quadratic function does not give as tight results due to the lack of smoothness. To use this h_2 in the complexity analysis, we need to estimate the integral $\int \exp(-h_2)$, which turns out to be *a highly nontrivial task*. Below we establish a technical result on a modified Gaussian integral, which will be used later to bound the integral $\int \exp(-h_2)$ and hence the complexity of the RGO rejection sampling in Algorithm 2.

Proposition 3 For $\lambda > 0$, $a \ge 0$ and $d \ge 1$, if

$$\lambda \le \frac{1}{4a^2 d}, \tag{6}$$

then

$$\int_{\mathbb{R}^d} \exp\left(-\frac{1}{2\lambda}\|x\|^2 - a\|x\|\right) \mathrm{d}x \ge \frac{(2\pi\lambda)^{d/2}}{2}. \tag{7}$$

Proof. Denote $r = \|x\|$, then $\mathrm{d}x = r^{d-1}\mathrm{d}r\mathrm{d}S^{d-1}$, where $\mathrm{d}S^{d-1}$ is the surface area of the $(d-1)$-dimensional unit sphere. It follows that

$$\begin{aligned} \int_{\mathbb{R}^d} \exp\left(-\frac{1}{2\lambda}\|x\|^2 - a\|x\|\right) \mathrm{d}x &= \int_0^\infty \int \exp\left(-\frac{1}{2\lambda}r^2 - ar\right) r^{d-1}\mathrm{d}r\mathrm{d}S^{d-1} \\ &= \frac{2\pi^{d/2}}{\Gamma\left(\frac{d}{2}\right)} \int_0^\infty \exp\left(-\frac{1}{2\lambda}r^2 - ar\right) r^{d-1}\mathrm{d}r. \end{aligned} \tag{8}$$

In the above equation, we have used the fact that the total surface area of a $(d-1)$-dimensional unit sphere is $2\pi^{d/2}/\Gamma\left(\frac{d}{2}\right)$ where $\Gamma(\cdot)$ is the gamma function, i.e.,

$$\Gamma(z) = \int_0^\infty t^{z-1}e^{-t}\mathrm{d}t. \tag{9}$$

Defining

$$F_{d,\lambda}(a) := \int_0^\infty \exp\left(-\frac{1}{2\lambda}r^2 - ar\right) r^d \mathrm{d}r, \tag{10}$$

to establish (7), it suffices to bound $F_{d-1,\lambda}(a)$ from below.

It follows directly from the definition of $F_{d,\lambda}$ in (10) that

$$\frac{\mathrm{d}F_{d-1,\lambda}(a)}{\mathrm{d}a} = \int_0^\infty \exp\left(-\frac{1}{2\lambda}r^2 - ar\right)(-r)r^{d-1}\mathrm{d}r = -F_{d,\lambda}(a).$$

This implies $F_{d,\lambda}$ is monotonically decreasing and thus $F_{d,\lambda}(a) \le F_{d,\lambda}(0)$. As a result, $\frac{\mathrm{d}F_{d-1,\lambda}(a)}{\mathrm{d}a} \ge -F_{d,\lambda}(0)$, and therefore,

$$F_{d-1,\lambda}(a) \ge F_{d-1,\lambda}(0) - aF_{d,\lambda}(0). \tag{11}$$

Setting $t = r^2/(2\lambda)$, we can write

$$F_{d,\lambda}(0) = \int_0^\infty \exp\left(-\frac{1}{2\lambda}r^2\right) r^d \mathrm{d}r = \int_0^\infty e^{-t}(2\lambda t)^{\frac{d-1}{2}} \lambda \,\mathrm{d}t = 2^{\frac{d-1}{2}} \lambda^{\frac{d+1}{2}} \int_0^\infty e^{-t} t^{\frac{d-1}{2}} \mathrm{d}t.$$

In view of the definition of the gamma function (9), we obtain

$$F_{d,\lambda}(0) = 2^{\frac{d-1}{2}} \lambda^{\frac{d+1}{2}} \Gamma\left(\frac{d+1}{2}\right). \tag{12}$$

Recall that the Wendel's inequality is $\frac{\Gamma(t+1)}{\Gamma(t+s)} \le (t+s)^{1-s}$ for $0 < s < 1$ and $t > 0$ (see (Wendel 1948)). Applying the Wendel's inequality with $t = \frac{d+\alpha-1}{2}$ and $s = \frac{1-\alpha}{2}$ yields $\frac{\Gamma\left(\frac{d+1}{2}\right)}{\Gamma\left(\frac{d}{2}\right)} \le \left(\frac{d}{2}\right)^{\frac{1}{2}}$. This inequality, (11), (12), and the assumption (6) imply that

$$\begin{aligned} F_{d-1,\lambda}(a) &\ge F_{d-1,\lambda}(0) - aF_{d,\lambda}(0) = 2^{\frac{d}{2}-1}\lambda^{\frac{d}{2}}\Gamma\left(\frac{d}{2}\right) - a2^{\frac{d-1}{2}}\lambda^{\frac{d+1}{2}}\Gamma\left(\frac{d+1}{2}\right) \\ &= 2^{\frac{d}{2}-1}\lambda^{\frac{d}{2}}\Gamma\left(\frac{d}{2}\right)\left(1 - a(2\lambda)^{\frac{1}{2}}\frac{\Gamma\left(\frac{d+1}{2}\right)}{\Gamma\left(\frac{d}{2}\right)}\right) \ge 2^{\frac{d}{2}-1}\lambda^{\frac{d}{2}}\Gamma\left(\frac{d}{2}\right)\left(1 - a(\lambda d)^{\frac{1}{2}}\right) \ge \frac{1}{4}(2\lambda)^{\frac{d}{2}}\Gamma\left(\frac{d}{2}\right). \end{aligned}$$

The result (7) then follows from the above inequality and (8). □

Based on the modified Gaussian integral in Proposition 3, we establish Proposition 4, the main result of this section that shows the number of rejections in Algorithm 2 is small in expectation. Hence, the implementation of RGO for g is computationally efficient. Its proof is postponed to Appendix B. We remark that even though rejection sampling is an elementary method, how to use it on non-smooth potentials g_y^η of the form (2) is highly nontrivial. To our best knowledge, our algorithm (Algorithm 2) is the first effective algorithm to sample from g_y^η with non-smooth g.

Proposition 4 Assume f is convex and M-Lipschitz continuous. Let g be as in (1) and $\pi^{X|Y}(x|y) \propto \exp\left(-g(x) - \frac{1}{2\eta}\|x-y\|^2\right)$ for a fixed y, then X generated by Algorithm 2 satisfies $X \sim \pi^{X|Y}(x|y)$. Moreover, if

$$\eta_\mu \le \frac{1}{64M^2 d}, \quad \delta \le \frac{1}{32d}, \tag{13}$$

then the expected number of iterations in the rejection sampling is at most 3.

4 SAMPLING FROM NON-SMOOTH POTENTIALS

We now combine our implementation of RGO (Algorithm 2) and the ASF (Algorithm 1) to sample from log-concave probability densities with non-smooth potentials. This section contains two subsections. Subsection 4.1 presents the iteration-complexity bound for sampling from $\exp(-g(x))$ where $g = f + \mu\|\cdot - x^0\|^2/2$. Based on this, Subsection 4.2 provides the iteration-complexity for sampling from $\exp(-f(x))$ where f is convex and Lipschitz continuous. Apart from being a transition step to our final result for sampling without smoothness, the results in Subsection 4.1 provide an efficient method to sample from composite potentials of the form $f + \mu\|\cdot - x^0\|^2/2$ and may be of independent interest.

4.1 Complexity of sampling for regularized convex Lipschitz continuous potentials

Using the efficient implementation of RGO for g developed in Section 3 and the alternating sampling framework Algorithm 1, we are now able to sample from $\exp(-g(x))$ and establish the complexity for this sampling task. The following theorem states the iteration-complexity bound for Algorithm 1 using Algorithm 2 as the RGO to sample from $\exp(-g(x))$. Note that this iteration-complexity bound is poly-logarithmic in the precision ε in terms of total variation.

Theorem 5 Let $x^0 \in \mathbb{R}^d$, $\varepsilon > 0$, $M > 0$, and $\mu > 0$ be given. Assume f is convex and M-Lipschitz continuous and let g be as in (1). Set

$$\delta = \frac{1}{64d}, \quad \eta = \frac{1}{64M^2 d}. \tag{14}$$

Then the ASF (Algorithm 1) with Algorithm 2 as an RGO achieves ε error in terms of KL divergence with respect to the target distribution $\pi^X \propto \exp(-g)$ in $\tilde{\mathscr{O}}\left(\frac{M^2 d}{\mu}\right)$ iterations, and each iteration queries $\mathscr{O}(1)$ subgradient oracles of f and $\mathscr{O}(1)$ Gaussian distribution sampling oracles.

Proof. With the parameter choice (14), by Proposition 9 and Proposition 4, Algorithm 2 queries the subgradient oracle of f and the Gaussian distribution sampling oracle $\mathscr{O}(1)$ times to terminate. The total complexity $\tilde{\mathscr{O}}\left(\frac{M^2 d}{\mu}\right)$ then follows by plugging (14) into the ASF complexity $\tilde{\Theta}(\frac{1}{\eta\mu})$ as in Theorem 1. □

4.2 Complexity of sampling for convex Lipschitz continuous potentials

This subsection studies the main problem of this paper, i.e., sampling from $\exp(-f(x))$. Building upon Theorem 5 for sampling from $\exp(-g(x)) = \exp(-f(x) - \mu\|x - x^0\|^2/2)$ and a proper choice of μ, the following theorem establishes the iteration-complexity bound for Algorithm 1 to sample from $\exp(-f(x))$. Its proof is postponed to Appendix B.

Theorem 6 Let π^X be a distribution on $\mathbb{R}^d$ satisfying $\pi^X(x) \propto \exp(-f(x))$ where f is convex and M-Lipschitz continuous on $\mathbb{R}^d$. Let $x^0 \in \mathbb{R}^d$ and $\varepsilon > 0$ be given and

$$\mu = \frac{\varepsilon}{\sqrt{2}\left(\sqrt{\mathscr{M}_4} + \|x^0 - x_{\min}\|^2\right)} \tag{15}$$

where $\mathscr{M}_4 = \int_{x\in\mathbb{R}^d} \|x - x_{\min}\|^4 \mathrm{d}\pi^X(x)$ and $x_{\min} = \operatorname{argmin}\{f(x) : x \in \mathbb{R}^d\}$. Choose δ and η as in (14) and consider Algorithm 1 using Algorithm 2 as an RGO for step 1, applied to $g = f + \mu\|\cdot - x^0\|^2/2$. Then, the iteration-complexity bound to achieve ε error to π^X in terms of total variation is

$$\tilde{\mathscr{O}}\left(\frac{M^2 d\left(\sqrt{\mathscr{M}_4} + \|x^0 - x_{\min}\|^2\right)}{\varepsilon\delta}\right). \tag{16}$$

Remark 7 The strategy we use to sample from a non-smooth potential f by considering a regularized one $g = f + \mu\|\cdot - x^0\|^2/2$ first is not the only option. The reason we do so is that the complexity bound for ASF in (Lee et al. 2021) requires the potential to be strongly convex. This convergence result for ASF can, however, be extended. Following a similar argument as in (Lee et al. 2021), in particular Proposition 2 and Lemma 2, one can establish the complexity bound $\mathscr{O}(\frac{1}{\eta\psi^2}\log\frac{\beta}{\varepsilon})$ for ASF with convex potential where ψ is the isoperimetry constant of the target distribution and β is a warm start constant. Combining this with our RGO implementation (Algorithm 2) yields a method to sample from non-smooth potential f with complexity $\mathscr{O}(\frac{M^2 d}{\psi^2}\log\frac{\beta}{\varepsilon})$. This is better than $\mathscr{O}(d^{5/2}\log\frac{\beta}{\varepsilon})$ in (Lee and Vempala 2017), even in the high accuracy region, if M scales slower than $\sqrt{d}$, which is typical if sparsity exists.

5 CONCLUSION

This paper presents an algorithm based on the alternating sampling framework for sampling from non-smooth potentials and establishes a complexity bound $\tilde{\mathcal{O}}(M^2 d \mathscr{M}_4^{1/2} \varepsilon^{-1})$ to achieve ε accuracy in terms of total variation distance to the target density. The key contribution of this paper is a computationally efficient implementation of RGO for any convex non-smooth function. One direct extension of the paper is to apply the proposed algorithm to sample from semi-smooth densities, which include smooth and non-smooth densities as two extreme cases. Another possible extension of our analysis in this paper is to consider sampling from composite densities proportional to $\exp(-f(x)-h(x))$ where f is convex and smooth, and h is convex and semi-smooth.

ACKNOWLEDGMENTS

This work was supported by NSF under grant 1942523 and 2008513.

A REVIEW OF THE PROXIMAL BUNDLE METHOD

The proximal bundle method (Liang and Monteiro 2021) is an efficient algorithm for solving convex non-smooth optimization problems. In this section, we briefly review an approach to solve the subproblem considered in the proximal bundle method, the properties of the solution to the subproblem, and the iteration-complexity for solving the subproblem.

Consider the optimization subproblem (recall (2))

$$g_y^\eta(x^*) = \min\left\{ g_y^\eta(x) := g(x) + \frac{1}{2\eta}\|x-y\|^2 : x \in \mathbb{R}^d \right\}, \tag{17}$$

and we aim at obtaining a δ-solution (i.e., a point $\bar{x}$ such that $g_y^\eta(\bar{x}) - g_y^\eta(x^*) \le \delta$) to (17). The algorithm is summarized in Algorithm 3.

Algorithm 3 Proximal Bundle Method

1. Let y, $\eta > 0$ and $\delta > 0$ be given, and set $x_0 = \tilde{x}_0 = y$, $C_1 = \{y\}$ and $j = 1$
2. Update $f_j = \max\left\{ f(x) + \langle f'(x), \cdot - x\rangle : x \in C_j \right\}$;
3. Define $g_j := f_j + \mu\|\cdot - x^0\|^2/2$ and compute

$$x_j = \operatorname*{argmin}_{u\in\mathbb{R}^n}\left\{ g_j^\eta(u) := g_j(u) + \frac{1}{2\eta}\|u-y\|^2 \right\}, \tag{18}$$

$$\tilde{x}_j = \operatorname{argmin}\left\{ g_y^\eta(u) : u \in \{x_j, \tilde{x}_{j-1}\} \right\}$$

4. If $g_y^\eta(\tilde{x}_j) - g_j^\eta(x_j) \le \delta$, then **stop** and **return** $J = j, x_J, \tilde{x}_J$; else, go to step 5
5. Set $C_{j+1} = C_j \cup \{x_j\}$ and $j \leftarrow j+1$, and go to step 2

The basic idea of Algorithm 3 is to approximate the non-smooth part of the objective function g_y^η with piece-wise affine functions constructed by a collection of cutting-planes and solve the resulting simplified problem. Note that (18) can be reformulated into a convex quadratic programming with affine constraints and the number of constraints is equal to the cardinality of C_j.

The following lemma contains technical results about Algorithm 3 that are useful in the complexity analysis in Section 4.

Lemma 8 Assume f is convex and M-Lipschitz continuous and let g be as in (1). Let $J, x_J, \tilde{x}_J$ be the outputs of Algorithm 3, then the following statements hold:

a) $f_J \le f$, $g_J \le g$ and $g_J^\eta(x_J) + \|x - x_J\|^2/(2\eta_\mu) \le g_J^\eta(x)$ for every $x \in \mathbb{R}^d$;
b) $g_y^\eta(\tilde{x}_J) - g_J^\eta(x_J) \le \delta$;
c) $\left\|\mu(x_J - x^0) + (x_J - y)/\eta\right\| \le M$;
d) $\|x_J - \tilde{x}_J\|^2 \le 2\eta_\mu\delta$.

Proof. a) The first two inequalities directly follow from the convexity of f, and the definitions of g and g_J. The third inequality follows from (18).

b) This statement immediately follows from step 3 of Algorithm 3.

c) It follows from the optimality condition of (18) and the definition of g_J that

$$-\mu(x_J - x^0) - \frac{x_J - y}{\eta} \in \partial f_J(x_J).$$

This inclusion and the fact that $\|f'(x)\| \le M$ imply that c) holds.

d) The last inequality in (a) with $x = \tilde{x}_J$ and (b) imply this statement. □

The following result states the iteration-complexity bound for Algorithm 3 to obtain a δ-solution to the subproblem (17). We have omitted the proof since it is relatively technical and beyond the scope of this paper, however, a complete proof can be found in Section 4 of (Liang and Monteiro 2021).

Proposition 9 Algorithm 3 takes $\tilde{\mathcal{O}}(\eta_\mu M^2/\delta + 1)$ iterations to terminate, and each iteration solves an affinely constrained convex quadratic programming problem.

B MISSING PROOFS

Proof of Proposition 4: It is a well-known result for rejection sampling that $X \sim \pi^{X|Y}(x|y)$ and the probability that X is accepted is

$$\mathbb{P}\left(U \le \frac{\exp(-g_y^\eta(X))}{\exp(-h_1(X))}\right) = \frac{\int \exp(-g_y^\eta(x))\mathrm{d}x}{\int \exp(-h_1(x))\mathrm{d}x}.$$

We observe that the assumption (13) implies that $2\sqrt{\eta_\mu}M + \sqrt{2\delta} \le \frac{1}{2\sqrt{d}}$, which satisfies the assumption in Proposition 3 with $\lambda = \eta_\mu, a = 2M + \frac{\sqrt{2\delta}}{\sqrt{\eta_\mu}}$. Using the definition of h_2 in (4) and Proposition 3, we have

$$\int_{\mathbb{R}^d} \exp(-h_2(x))\mathrm{d}x \ge \frac{1}{2}\exp(-g_y^\eta(\tilde{x}_j))(2\pi\eta_\mu)^{d/2}.$$

It follows from the definition of h_1 in (3) and the Gaussian integral that

$$\int \exp(-h_1(x))\mathrm{d}x = \exp\left(-g_y^\eta(\tilde{x}_j) + \delta\right)(2\pi\eta_\mu)^{d/2}.$$

We conclude that

$$\mathbb{P}\left(U \le \frac{\exp(-g_y^\eta(X))}{\exp(-h_1(X))}\right) = \frac{\int \exp(-g_y^\eta(x))\mathrm{d}x}{\int \exp(-h_1(x))\mathrm{d}x} \ge \frac{\int \exp(-h_2(x))\mathrm{d}x}{\exp\left(-g_y^\eta(\tilde{x}_j) + \delta\right)(2\pi\eta_\mu)^{d/2}} \ge \frac{1}{2}\exp(-\delta),$$

and the expected number of the iterations is

$$\frac{1}{\mathbb{P}\left(U \le \frac{\exp(-g_y^\eta(X))}{\exp(-f(X))}\right)} \le 2\exp(\delta) \le 2(1 + 2\delta) \le 2\left(1 + \frac{1}{16d}\right) \le 3$$

where the last two inequalities are due to the second inequality in (13). ■

Proof of Theorem 6: Let ρ denote the distribution of the points generated by Algorithm 1 using Algorithm 2 as an RGO, and let $\hat{\pi}^X$ denote the distribution proportional to $\exp(-g(x))$. Following the proof of Corollary 4.1 of (Chatterji, Diakonikolas, Jordan, and Bartlett 2020), we similarly have $\|\rho - \pi^X\|_{\mathrm{TV}} \leq \|\rho - \hat{\pi}^X\|_{\mathrm{TV}} + \|\hat{\pi}^X - \pi^X\|_{\mathrm{TV}}$ and

$$\begin{aligned}
\|\hat{\pi}^X - \pi^X\|_{\mathrm{TV}} &\leq \frac{1}{2}\left(\int_{\mathbb{R}^d} [f(x) - g(x)]^2 \mathrm{d}\pi^X(x)\right)^{1/2} = \frac{1}{2}\left(\int_{\mathbb{R}^d} \left(\frac{\mu}{2}\|x - x^0\|^2\right)^2 \mathrm{d}\pi^X(x)\right)^{1/2} \\
&\leq \frac{\mu}{2}\left(\int_{\mathbb{R}^d} \left(\|x - x_{\min}\|^2 + \|x_{\min} - x^0\|^2\right)^2 \mathrm{d}\pi^X(x)\right)^{1/2} \\
&\leq \frac{\mu}{2}\left(\int_{\mathbb{R}^d} \left(2\|x - x_{\min}\|^4 + 2\|x_{\min} - x^0\|^4\right) \mathrm{d}\pi^X(x)\right)^{1/2} \\
&= \frac{\sqrt{2}\mu}{2}\left(\mathscr{M}_4 + \|x_{\min} - x^0\|^4\right)^{1/2} \leq \frac{\sqrt{2}\mu}{2}\left(\sqrt{\mathscr{M}_4} + \|x_0 - x_{\min}\|^2\right) = \frac{\varepsilon}{2}
\end{aligned}$$

where the last identity is due to the definition of μ in (15). Hence, it suffices to derive the iteration-complexity bound for Algorithm 1 to obtain $\|\rho - \hat{\pi}^X\|_{\mathrm{TV}} \leq \frac{\varepsilon}{2}$, which is (16) in view of Theorem 5 with μ as in (15). ■

REFERENCES

Bernton, E. 2018a. "Langevin Monte Carlo and JKO Splitting". In *Proceedings of the 31st Conference On Learning Theory*, edited by S. Bubeck, V. Perchet, and P. Rigollet, 1777–1798. Proceedings of Machine Learning Research.

Bernton, E. 2018b. "Sampling as Optimization in the Space of Measures: The Langevin Dynamics as A Composite Optimization Problem". In *Proceedings of the 31st Conference On Learning Theory*, edited by S. Bubeck, V. Perchet, and P. Rigollet, 2093–3027. Proceedings of Machine Learning Research.

Bertsimas, D., and S. Vempala. 2004. "Solving Convex Programs by Random Walks". *Journal of the ACM* 51(4):540–556.

Bou-Rabee, N., and M. Hairer. 2013. "Nonasymptotic Mixing of the MALA Algorithm". *IMA Journal of Numerical Analysis* 33(1):80–110.

Chatterji, N., J. Diakonikolas, M. I. Jordan, and P. Bartlett. 2020. "Langevin Monte Carlo without Smoothness". In *Proceedings of the 23rd International Conference on Artificial Intelligence and Statistics*, edited by S. Bubeck, V. Perchet, and P. Rigollet, 1716–1726. Proceedings of Machine Learning Research.

Chen, Y., R. Dwivedi, M. J. Wainwright, and B. Yu. 2018. "Fast MCMC Sampling Algorithms on Polytopes". *The Journal of Machine Learning Research* 19(1):2146–2231.

Chen, Y., R. Dwivedi, M. J. Wainwright, and B. Yu. 2020. "Fast Mixing of Metropolized Hamiltonian Monte Carlo: Benefits of multi-step gradients.". *Journal of Machine Learning Research* 21:92–1.

Cheng, X., and P. Bartlett. 2018. "Convergence of Langevin MCMC in KL-divergence". In *Proceedings of Algorithmic Learning Theory*, edited by F. Janoos, M. Mohri, and K. Sridharan, 186–211. Proceedings of Machine Learning Research.

Cheng, X., N. S. Chatterji, P. L. Bartlett, and M. I. Jordan. 2018. "Underdamped Langevin MCMC: A Non-asymptotic Analysis". In *Proceedings of the 31st Conference On Learning Theory*, edited by S. Bubeck, V. Perchet, and P. Rigollet, 300–323. Proceedings of Machine Learning Research.

Dalalyan, A. S. 2017. "Theoretical Guarantees for Approximate Sampling from Smooth and Log-concave Densities". *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 79(3):651–676.

Durmus, A., S. Majewski, and B. Miasojedow. 2019. "Analysis of Langevin Monte Carlo via Convex Optimization". *The Journal of Machine Learning Research* 20(1):2666–2711.

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

Dwivedi, R., Y. Chen, M. J. Wainwright, and B. Yu. 2019. "Log-concave sampling: Metropolis-Hastings algorithms are fast". *Journal of Machine Learning Research* 20(183):1–42.

Dyer, M., A. Frieze, and R. Kannan. 1991. "A Random Polynomial-time Algorithm for Approximating the Volume of Convex Bodies". *Journal of the ACM* 38(1):1–17.

Freund, Y., Y.-A. Ma, and T. Zhang. 2022. "When is the Convergence Time of Langevin Algorithms Dimension Independent? A Composite Optimization Viewpoint". *Journal of Machine Learning Research* 23(214):1–32.

Gelman, A., J. B. Carlin, H. S. Stern, D. B. Dunson, A. Vehtari, and D. B. Rubin. 2013. *Bayesian Data Analysis*. New York: CRC press.

Grenander, U., and M. I. Miller. 1994. "Representations of Knowledge in Complex Systems". *Journal of the Royal Statistical Society: Series B (Methodological)* 56(4):549–581.

Kannan, R., L. Lovász, and M. Simonovits. 1997. "Random Walks and An $O^*(n^5)$ Volume Algorithm for Convex Bodies". *Random Structures & Algorithms* 11(1):1–50.

Krauth, W. 2006. *Statistical Mechanics: Algorithms and Computations*, Volume 13. Oxford University Press, Oxford.

Lee, Y. T., R. Shen, and K. Tian. 2021. "Structured Logconcave Sampling with A Restricted Gaussian Oracle". In *Proceedings of 34th Conference on Learning Theory*, edited by M. Belkin and S. Kpotufe, 2993–3050. Proceedings of Machine Learning Research.

Lee, Y. T., and S. S. Vempala. 2017. "Eldan's Stochastic Localization And the KLS Hyperplane Conjecture: An Improved Lower Bound for Expansion". In *IEEE 58th Annual Symposium on Foundations of Computer Science*. October 15$^{\text{th}}$-17$^{\text{th}}$, Berkeley, California, 998–1007.

Lehec, J. 2021. *"The Langevin Monte Carlo Algorithm in the Non-smooth Log-concave Case"*. http://arxiv.org/abs/2101.10695 accessed January 7th, 2022.

Lemaréchal, C. 1975. "An Extension of Davidon Methods to Non-differentiable Problems". In *Nondifferentiable Optimization*, edited by M. L. Balinski and P. Wolfe, Volume 3, 95–109. Berlin, Heidelberg: Springer.

Liang, J., and R. D. C. Monteiro. 2021. "A Proximal Bundle Variant with Optimal Iteration-complexity for A Large Range of Prox Stepsizes". *SIAM Journal on Optimization* 31(4):2955–2986.

Mifflin, R. 1982. "A Modification and an Extension of Lemaréchal's Algorithm for Nonsmooth Minimization". In *Nondifferential and Variational Techniques in Optimization*, edited by D. C. Sorensen and R. J.-B. Wets, Volume 17, 77–90. Amsterdam: North-Holland Publishing Company.

Mou, W., N. Flammarion, M. J. Wainwright, and P. L. Bartlett. 2022. "An Efficient Sampling Algorithm for Non-smooth Composite Potentials". *Journal of Machine Learning Research* 23(233):1–50.

Neal, R. M. 2011. "MCMC Using Hamiltonian Dynamics". In *Handbook of Markov Chain Monte Carlo*, edited by S. Brooks, A. Gelman, G. Jones, and X.-L. Meng, 113–162. New York: Chapman and Hall/CRC.

Parisi, G. 1981. "Correlation Functions And Computer Simulations". *Nuclear Physics B* 180(3):378–384.

Roberts, G. O., and J. S. Rosenthal. 1998. "Optimal Scaling of Discrete Approximations to Langevin Diffusions". *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 60(1):255–268.

Roberts, G. O., and O. Stramer. 2002. "Langevin Diffusions And Metropolis-Hastings Algorithms". *Methodology and Computing in Applied Probability* 4(4):337–357.

Roberts, G. O., and R. L. Tweedie. 1996. "Exponential Convergence of Langevin Distributions And Their Discrete Approximations". *Bernoulli* 2(4):341–363.

Rockafellar, R. T. 1976. "Monotone Operators And the Proximal Point Algorithm". *SIAM Journal on Control and Optimization* 14(5):877–898.

Sites Jr, J. W., and J. C. Marshall. 2003. "Delimiting Species: A Renaissance Issue in Systematic Biology". *Trends in Ecology & Evolution* 18(9):462–470.

Vono, M., D. Paulin, and A. Doucet. 2022. "Efficient MCMC Sampling with Dimension-free Convergence Rate Using ADMM-type Splitting". *Journal of Machine Learning Research* 23(25):1–69.

Wendel, J. 1948. "Note on the Gamma Function". *The American Mathematical Monthly* 55(9):563–564.

Wibisono, A. 2019. *"Proximal Langevin Algorithm: Rapid Convergence under Isoperimetry"*. http://arxiv.org/abs/1911.01469 accessed November 4th, 2019.

Wolfe, P. 1975. "A Method of Conjugate Subgradients for Minimizing Nondifferentiable Functions". In *Nondifferentiable Optimization*, edited by M. L. Balinski and P. Wolfe, Volume 3, 145–173. Berlin, Heidelberg: Springer.

AUTHOR BIOGRAPHIES

JIAMING LIANG is a Postdoctoral Associate in the Department of Computer Science at Yale University and an Assistant Professor (on leave) in the School of Data Science at The Chinese University of Hong Kong, Shenzhen. His research interests include optimization and sampling algorithms. His email addresses are jiaming.liang@yale.edu ,liangjiaming@cuhk.edu.cn.

YONGXIN CHEN is an Assistant Professor in the School of Aerospace Engineering at Georgia Institute of Technology. His research interests include control, machine learning, optimization, and robotics. He is a senior member of IEEE. His email address is yongchen@gatech.edu.