Applied Mathematics and Computation 464 (2024) 128377

Contents lists available at ScienceDirect

Applied Mathematics and Computation

journal homepage: www.elsevier.com/locate/amc

Smoothing unadjusted Langevin algorithms for nonsmooth composite potential functions

Susan Ghaderi [a,c,*], Masoud Ahookhosh [b], Adam Arany [a], Alexander Skupin [c,d,e], Panagiotis Patrinos [a], Yves Moreau [a]

[a] Department of Electrical Engineering ESAT-STADIUS, KU Leuven, Kasteelpark Arenberg 10, Leuven, 3001, Belgium
[b] Department of Mathematics, University of Antwerp, Antwerp, Belgium
[c] Luxembourg Centre for Systems Biomedicine, University of Luxembourg, Esch-sur-Alzette, Luxembourg
[d] Department of Physics and Material Science, University of Luxembourg, Esch-sur-Alzette, Luxembourg
[e] Department of Neuroscience, University of California San Diego, La Jolla, USA

ARTICLE INFO

Keywords:
Bayesian learning
Nonsmooth sampling
Convex optimization
MCMC methods
Langevin equation

ABSTRACT

This paper addresses a gradient-based Markov Chain Monte Carlo (MCMC) method to sample from the posterior distribution of problems with nonsmooth potential functions. Following the Bayesian paradigm, our potential function will be some of two convex functions, where one of which is smooth. We first approximate the potential function by the so-called forward-backward envelope function, which is a real-valued smooth function with the same critical points as the original one. Then, we incorporate this smoothing technique with the unadjusted Langevin algorithm (ULA), leading to smoothing ULA, called SULA. We next establish non-asymptotic convergence results of SULA under mild assumption on the original potential function. We finally report some numerical results to establish the promising performance of SULA on both synthetic and real chemoinformatics data.

1. Introduction

Sampling from high-dimensional distributions in statistical and machine learning, and specifically in Bayesian inference, is a challenging task due to the nature of the target distribution. Let us consider *sampling* from a canonical target distribution $\pi(x) : \mathbb{R}^p \to [0, \infty)$ given by

$$\pi(x) = \frac{e^{-\mathcal{U}(x)}}{\int_{\mathbb{R}^p} e^{-\mathcal{U}(y)} dy}, \tag{1}$$

where $\mathcal{U} : \mathbb{R}^p \to \mathbb{R}$ is a convex, nonsmooth and measurable (i.e., $\int_{\mathbb{R}^p} \exp\{-\mathcal{U}(y)\} dy < +\infty$) function, called the potential function. Methodologies based on Bayesian inference are among the most popular methods to design a proper target distribution, called posterior distribution, for the unknown parameters of a mathematical model by considering the prior beliefs on the parameters and

* Corresponding author at: Department of Electrical Engineering ESAT-STADIUS, KU Leuven, Kasteelpark Arenberg 10, Leuven, 3001, Belgium.
E-mail addresses: susan.ghaderi@kuleuven.be (S. Ghaderi), masoud.ahookhosh@uantwerpen.be (M. Ahookhosh), aarany@esat.kuleuven.be (A. Arany), alexander.skupin@uni.lu (A. Skupin), panos.patrinos@kuleuven.be (P. Patrinos), Yves.moreau@kuleuven.be (Y. Moreau).
URLs: https://scholar.google.at/citations?user=8RU96vMAAAAJ&hl=en (S. Ghaderi), https://www.esat.kuleuven.be/stadius/person.php?id=77 (Y. Moreau).

https://doi.org/10.1016/j.amc.2023.128377
Received 5 December 2022; Received in revised form 6 June 2023; Accepted 29 September 2023

Available online 14 October 2023
0096-3003/© 2023 Elsevier Inc. All rights reserved.

the likelihood function of the given data set. Imposing prior beliefs on the parameters allows modulating the uncertainty on those parameters by taking into account prior knowledge about the problem. This is accounted for by Bayes' rule via

$$\pi(x|D) \propto P(D|x)P(x),$$

where $D \in \mathbb{R}^{n\times p}$ is a given data set, $x \in \mathbb{R}^p$ is the parameter to be estimated, $\pi(x|D)$ is the posterior distribution, $P(D|x)$ is the likelihood function, and $P(x)$ is the prior distribution. As such, we come to $\log \pi(x|D) \propto \log P(D|x) + \log P(x) := \mathcal{U}(x)$ as the potential function of the problem (1). For the sake of simplicity, we set here $f(x) := \log P(D|x)$ and $g(x) := \log P(x)$, which leads to composite potential functions of the form $\mathcal{U}(x) = f(x) + g(x)$.

The goal of sampling methods is to generate a set of random vectors in $\mathbb{R}^p$ with a distribution close to the target distribution $\pi(x)$. However, in a high-dimensional space and without a closed analytical form for the target distribution (except for some special cases, in particular for conjugate prior–likelihood pairs in Bayesian inference), this is a challenging task. *Markov chain Monte Carlo* (MCMC) methods allow sampling from such a high-dimensional complex distribution [46], by providing ways to design a Markov chain whose *stationary equilibrium distribution* is $\pi(x)$. Under reasonable assumptions, the distribution of the successive samples of the Markov chain will converge to its equilibrium distribution. Metropolis-Hasting [34] and Gibbs sampling algorithms [25] are two famous examples of MCMC algorithms. The art of MCMC methods lies in the design of sampling schemes whose samples converge sufficiently rapidly toward the proper stationary distribution at a reasonable computational cost.

To improve the performance of MCMC methods in high-dimensional spaces, a general strategy is to combine stochastic differential equations (SDEs), such as *Langevin dynamics* and *Hamiltonian dynamics* with MCMC methods (where the equilibrium distribution of these SDEs is equivalent to the target distribution), which has shown a great theoretical and numerical improvement in the convergence of MCMC methods [11,20,37,43,49,56,58].

For instance, the Langevin stochastic differential equation [7] is defined as

$$dx_t = -\nabla\overline{\mathcal{U}}(x_t) + \sqrt{2p}\mathcal{B}_t, \tag{2}$$

where $\{\mathcal{B}_t\}_{t\geq 0}$ is a p-dimensional Brownian motion. It has $\pi(x)$ as its stationary distribution and its formulation gives the opportunity to sampling methods to use discretization techniques and gradient-based techniques.

Applying the Euler-Maruyama discretization scheme to SDE (2) leads to the discrete-time Markov chain $\{x_k\}$ given by

$$x_{k+1} = x_k - \gamma_{k+1}\nabla\overline{\mathcal{U}}(x_k) + \sqrt{2\gamma_{k+1}}\mathcal{Z}_k, \quad k \in \mathbb{N}, \tag{3}$$

where $\{\gamma_k\}_{k\geq 0}$ is a sequence of step sizes that can be a constant or a vanishing sequence, and $\{\mathcal{Z}_k\}_{k\geq 0}$ is a sequence of independent and identically distributed (i.i.d.) standard n-dimensional Gaussian random variables. This algorithm was introduced for the first time in [23] and [41] for molecular dynamics applications. Furthermore, the asymptotic and non-asymptotic convergence of the ULA algorithm have been studied under different conditions and with different metrics for constant or varying step sizes [14,15,19,27,33,47,51].

There are many practical problems in which the potential functions $\mathcal{U}$ are convex and nonsmooth (see [2,5,40] and references therein). Cases with smooth convex potential functions are well-studied [1,11,31,32,56]. By contrast, there is only a limited number of studies for nonsmooth potential functions $\mathcal{U}$ [10,20,43,49]. However, there are many examples of target distributions with nonsmooth potential functions, such as those commonly assuming that the prior distribution is the Laplace distribution ($\frac{1}{2}e^{-|x|}$) in [43] or the elastic net ($e^{-(\lambda_1\|x\|_2^2+\lambda_2\|x\|_1)}$) [28]. Moreover, in inverse problems, the mathematical models involve a data fidelity term (i.e., smooth loss function including an ill-conditioned matrix) and a regularization term, where the latter term is usually nonsmooth. The regularization term imposes some properties to the solution that can act as a prior in our Bayesian setting. These challenges clearly demonstrate the urgent need for studying sampling from target distributions arising from nonsmooth potential functions. This is the main objective of this paper.

In recent years, the idea of smoothing the nonsmooth function using advanced smoothing techniques from optimization [36,42,50] has received much attention from researchers tackling nonsmooth optimization problems. Inspired by this idea, one solution to handle nonsmooth potential functions would be to replace a nonsmooth potential function $\mathcal{U}$ by a smooth approximation of $\mathcal{U}$, see [16,18,20,43]. Hence, under some conditions, designing sampling schemes for such problems becomes possible.

To develop a gradient-based MCMC method in the nonsmooth setting, a generic framework is to first apply a smoothing strategy to the underlying nonsmooth potential function (i.e., leading to a smooth approximated function with Lipschitz gradients) and subsequently to discretize the resulting SDEs by some special techniques leading to gradient-based MCMC iterative schemes. It is notable that such methods guarantee the convergence of the distribution of the samples to the target distribution under some suitable assumptions. Hence, invoking this smoothing framework demands two main steps: (i) applying suitable smoothing techniques such as Moreau envelope [35,36], forward-backward envelope [42,50], and Gaussian smoothing [39]; (ii) applying an efficient discretization technique such as Euler-Maruyama and Runge-Kutta schemes. From smoothing point of view, most of the advancements have been developed based on Moreau envelope [8,9,17,20,30,43,54] leading to proximal-type MCMC methods; however, in 2021, a smoothing ULA algorithm was introduced in [24] by applying the forward-backward envelope, which further studied independently in the more recent work [22]. On the other hand, applying the Runge-Kutta discretization schemes may further accelerate these algorithms; see, e.g., [44,45]. In this paper, we introduce a gradient-based MCMC method by applying ULA to the forward-backward envelope of our nonsmooth composite potential function and study its convergence analysis, which uses different assumptions than [22].

1.1. Contribution and organization

This paper is organized as follows. Following the introduction, we will detail our problem setting and present a smoothing technique using the Forward-Backward Envelope (FBE) and detail its fundamental properties in Section 2. In particular, we show that the gradient of the FBE can be Lipschitz continuous under some suitable assumptions (see Theorem 1) and that FBE is strongly convex if f is a strongly convex quadratic function (see Lemma 2). These desirable properties of the FBE open the door to the design of gradient-based MCMC methods for composite nonsmooth potential functions, which frequently appear in Bayesian learning. In Section 3, we will present our Smoothing Unadjusted Langevin Algorithm (SULA) based on the gradient of the smoothing function FBE. We then investigate the nonasymptotic convergence of SULA for constant step size strategies (see Theorem 3), where we subsequently verify nonasymptotic convergence in terms of the 2-Wasserstein distance (i.e., $\mathcal{W}_2$) (see Theorem 5). Next, in Section 4, we apply SULA to a regression problem with both a synthetic data set and a chemoinformatics data set, which validates our theoretical foundations. Finally, we deliver some concluding remarks in Section 5.

2. Problem description and the forward-backward envelope

In this paper, we consider a nonsmooth composite potential function $\mathcal{U} : \mathbb{R}^p \to \mathbb{R}$ such that

$$\mathcal{U}(x) = f(x) + g(x), \tag{4}$$

where f is a smooth convex function, g is proper, convex, and lower semicontinuous (lsc), and $\mathcal{U}$ has a non-empty set of minimizers, i.e., $\arg\min \mathcal{U} \neq \emptyset$.

Throughout this paper, we suppose (4) satisfies the following assumptions:

(H1) $f : \mathbb{R}^p \to \mathbb{R}$ is an L_f-smooth function (smooth with Lipschitz continuous gradients):

$$\|\nabla f(x) - \nabla f(y)\| \leq L_f \|x - y\|; \tag{5}$$

(H2) $g : \mathbb{R}^p \to \overline{\mathbb{R}} = \mathbb{R} \cup \{\infty\}$ is an extended real-valued, proper, convex, lsc, and nonsmooth function.

The composite function $\mathcal{U}$ of the form (4) has a favorable structure and provides a broad enough framework to encompass many interesting nonsmooth potential functions (see [20,43] and references therein). A typical method for dealing with nonsmooth optimization problems is to replace the nonsmooth objective with a smooth approximation and optimize the approximated one. By contrast, for sampling problems with nonsmooth potential functions, we need a smooth approximation of such potential functions to apply the Langevin MCMC sampling algorithms. There are several smoothing approaches in the optimization context, see [3,4,6,12,13,29,38,50,52,53]. Here, we rely on the *forward-backward envelope* [42] technique as shown below.

2.1. Smoothing with the forward-backward envelope

The *forward-backward envelope* (FBE) technique is a smoothing technique for the sum of two functions, which has many desirable properties that are essential for sampling from a target distribution with a nonsmooth composite potential function.

We here assume that the function $\mathcal{U}$ is of the form (4), where $f : \mathbb{R}^p \to \mathbb{R}$ and $g : \mathbb{R}^p \to \overline{\mathbb{R}}$ satisfies the assumptions (H1) and (H2), respectively. Since f is smooth, we can approximate it by its linear form, while keeping the nonsmooth term g as it is (i.e., $\mathcal{U}(z) \approx f(x) + \langle \nabla f(x), z - x \rangle + g(z)$) if z is sufficiently close to x. Using this approximation and adding the regularization term $\frac{1}{2\lambda}\|z - x\|^2$ for $\lambda < 1/L_f$, we end up with an upper estimate of $\mathcal{U}$, where its minimizer leads to the *forward-backward mapping* as

$$\mathcal{T}_\lambda^{f,g}(x) = \operatorname{argmin}_{z \in \mathbb{R}^n} \left\{ f(x) + \langle \nabla f(x), z - x \rangle + g(z) + \tfrac{1}{2\lambda}\|z - x\|^2 \right\} \tag{6}$$

$$= \operatorname{prox}_{\lambda g}(x - \lambda \nabla f(x)), \tag{7}$$

where $\operatorname{prox}_{\lambda g}(x) = \operatorname{argmin}_{z \in \mathbb{R}^n} \left\{ g(z) + \frac{1}{2\lambda}\|z - x\|^2 \right\}$.

Motivated by the Moreau envelope definition [35,57], the *forward-backward envelope* (FBE) is defined as

$$\mathcal{U}_\lambda^{f,g}(x) = \inf_{z \in \mathbb{R}^n} \left\{ f(x) - \langle \nabla f(x), z - x \rangle + g(z) + \tfrac{1}{2\lambda}\|z - x\|^2 \right\} \tag{8}$$

$$= f(x) + \tfrac{\lambda}{2}\|\nabla f(x)\|^2 + g^\lambda(x - \lambda \nabla f(x)), \tag{9}$$

which was introduced in [42]. The FBE function $\mathcal{U}_\lambda^{f,g}(x)$ has many desirable properties including its smoothness and having the same minimizer as $\mathcal{U}(x)$. In the next result, we will describe some fundamental properties of the forward-backward mapping and its envelope, which are key steps toward its applications in Langevin MCMC sampling.

Theorem 1 *(Properties of the forward-backward mapping and envelope). Let $\mathcal{U} : \mathbb{R}^p \to \mathbb{R}$ be of the composite form $\mathcal{U}(x) = f(x) + g(x)$, where $f : \mathbb{R}^p \to \mathbb{R}$ and $g : \mathbb{R}^p \to \overline{\mathbb{R}}$ satisfies the assumptions (H1) and (H2), respectively. Then, for $\lambda \in (0, 1/L_f)$, the following statements hold true:*

(i) $\mathcal{T}_\lambda^{f,g}(x)$ *is nonempty, locally bounded and single-valued;*
(ii) $\mathcal{R}_\lambda^{f,g} = \frac{1}{\lambda}(x - \mathcal{T}_\lambda^{f,g}(x))$ *is Lipschitz continuous with the constant* $(2+\lambda L_f)/\lambda$*;*
(iii) $\mathcal{U}_\lambda^{f,g} \leq \mathcal{U}$ *and* $\lim_{\lambda\to 0} \mathcal{U}_\lambda^{f,g}(x) = \mathcal{U}(x)$*;*
(iv) $\inf_{x\in \mathrm{dom}\, g}\ \mathcal{U}(x) = \inf_{x\in \mathrm{dom}\, g}\ \mathcal{U}_\lambda^{f,g}(x)$ *and* $\operatorname{argmin}\ \mathcal{U}(x) = \operatorname{argmin}_x\ \mathcal{U}_\lambda^{f,g}(x)$ *for* $\lambda \in (0, 1/L_f)$*;*
(v) $\mathcal{U}_\lambda^{f,g}$ *is real-valued, locally Lipschitz, and smooth with*

$$\nabla \mathcal{U}_\lambda^{f,g}(x) = \left(I - \lambda \nabla^2 f(x)\right) \mathcal{R}_\lambda^{f,g}(x); \tag{10}$$

(vi) *if* $\nabla^2 f$ *and* $\mathcal{R}_\lambda^{f,g}(x)$ *are uniformly continuous (i.e.,* $\|\nabla^2 f(\cdot)\| \leq L_1, \|\mathcal{R}_\lambda^{f,g}(\cdot)\| \leq L_3$*, where* $L_1, L_3 > 0$*), and* $\nabla^2 f(x)$ *is* L_2*-smooth, then* $\mathcal{U}_\lambda^{f,g}(x)$ *is* L*-smooth with* $L := \frac{2+\lambda L_f}{\lambda} + L_1(2+\lambda L_f) + \lambda L_3 L_2$.

Proof. The proofs of items (i), (iii), (iv), and (v) are given in [53, Theorem 4.4 and Theorem 4.7] and [48, Theorem 1.25]. By non-expansiveness of the $\mathrm{prox}_{\lambda g}$ mapping and L_f-smoothness of f, we get

$$\begin{aligned} \|\mathcal{T}_\lambda^{f,g}(x) - \mathcal{T}_\lambda^{f,g}(y)\| &= \|\mathrm{prox}_{\lambda g}(x - \lambda\nabla f(x)) - \mathrm{prox}_{\lambda g}(y - \lambda \nabla f(y))\| \\ &\leq \|x - y - \lambda(\nabla f(x) - \nabla f(y))\| \\ &\leq \|x - y\| + \lambda \|\nabla f(x) - \nabla f(y)\| \\ &\leq (1 + \lambda L_f)\|x - y\|, \end{aligned} \tag{11}$$

which consequently implies that

$$\|\mathcal{R}_\lambda^{f,g}(x_k) - \mathcal{R}_\lambda^{f,g}(y)\| \leq \tfrac{1}{\lambda}\|x - y\| + \tfrac{1}{\lambda}\|\mathcal{T}_\lambda^{f,g}(x) - \mathcal{T}_\lambda^{f,g}(y)\| \leq \tfrac{2+\lambda L_f}{\lambda}\|x - y\| \tag{12}$$

giving Assertion (ii). It follows from Assertion (v), the uniform and Lipschitz continuity of $\nabla^2 f$, the uniform continuity of $\mathcal{R}_\lambda^{f,g}$, and Assertion (ii) that

$$\begin{aligned} &\|\nabla \mathcal{U}_\lambda^{f,g}(x) - \nabla \mathcal{U}_\lambda^{f,g}(y)\| = \|(I - \lambda\nabla^2 f(x))\mathcal{R}_\lambda^{f,g}(x) - (I - \lambda\nabla^2 f(y))\mathcal{R}_\lambda^{f,g}(y)\| \\ &\leq \|\mathcal{R}_\lambda^{f,g}(x) - \mathcal{R}_\lambda^{f,g}(y)\| + \lambda\|\nabla^2 f(x)\mathcal{R}_\lambda^{f,g}(x) - \nabla^2 f(y)\mathcal{R}_\lambda^{f,g}(y)\| \\ &\leq \|\mathcal{R}_\lambda^{f,g}(x) - \mathcal{R}_\lambda^{f,g}(y)\| + \lambda\|\nabla^2 f(x)\|\ \|\mathcal{R}_\lambda^{f,g}(x) - \mathcal{R}_\lambda^{f,g}(y)\| + \lambda\|\mathcal{R}_\lambda^{f,g}(y)\|\ \|\nabla^2 f(x) - \nabla^2 f(y)\| \\ &\leq \left(\tfrac{2+\lambda L_f}{\lambda} + L_1(2+\lambda L_f) + \lambda L_3 L_2\right)\|x - y\|, \end{aligned} \tag{13}$$

which justifies Assertion (vi). □

The next result shows that if f is a convex quadratic function, then the forward-backward envelope $\mathcal{U}_\lambda^{f,g}$ is convex as well.

Lemma 2 *(Strong convexity of FBE* $\mathcal{U}_\lambda^{f,g}$*). [26, Proposition 4.4] and [52, Theorem 15.4.6] Let* f *be convex and quadratic (i.e.,* $f(x) := \frac{1}{2}\langle Px, x\rangle + \langle t, x\rangle$ *for some* $P \in S^{p\times p}$ *and* $t \in \mathbb{R}^p$*). Then, for all* $\lambda \in (0, 1/L_f)$*, the forward-backward envelope* $\mathcal{U}_\lambda^{f,g}$ *is convex and smooth with the parameters*

$$\begin{aligned} &\mu = \min\{\mu_f(1 - \lambda\mu_f), L_f(1 - \lambda L_f)\}, \\ &L = \tfrac{1-\lambda\mu_f}{\lambda} \end{aligned}$$

in which $\mu_f = \lambda_{min}(P)$ *and* $L_f = \lambda_{max}(P)$*. If* f *is additionally* μ_f*-strongly convex, then* $\mathcal{U}_\lambda^{f,g}$ *is* $\frac{\mu_f L_f}{\mu_f + L_f}$*-strongly convex.*

3. The smoothing unadjusted Langevin algorithm

In this section, we apply the forward-backward envelope technique for smoothing $\mathcal{U}$ and substitute its gradient $\nabla \mathcal{U}_\lambda^{f,g}(\cdot)$ into the ULA, leading to

$$\begin{aligned} x_{k+1} &= x_k - \tfrac{\gamma_{k+1}}{\lambda} Q_k \left(x_k - \mathcal{T}_\lambda^{f,g}(x_k)\right) + \sqrt{2\gamma_{k+1}}\mathcal{Z}_k \\ &= x_k - \gamma_{k+1} Q_k \mathcal{R}_\lambda^{f,g}(x_k) + \sqrt{2\gamma_{k+1}}\mathcal{Z}_k, \end{aligned} \tag{14}$$

which is called Smoothing ULA. A formal presentation of this scheme is presented in Algorithm 1.

Let us now consider the convergence result for SULA to assess the guarantees of convergence to the true target distribution. For this purpose, we assume that the forward-backward envelope $\mathcal{U}_\lambda^{f,g}$ satisfies the following two conditions:

(H3) The FBE function $\mathcal{U}_\lambda^{f,g}$ is L-smooth; i.e., for all $x, y \in \mathbb{R}^p$,

$$\|\nabla \mathcal{U}_\lambda^{f,g}(y) - \nabla \mathcal{U}_\lambda^{f,g}(x)\|_2 \leq L\|y - x\|_2; \tag{15}$$

Input: $x_0 \in \mathbb{R}^p$, $\lambda \in (0, 1/L_f)$, $\gamma_0 \in (0, 1/L)$;

for $k = 1 : N$ **do**
  Compute $\mathcal{T}_\lambda^{f,g}(x_k)$ and $\mathcal{R}_\lambda^{f,g}(x_k)$;
  Set $Q_k = I - \lambda \nabla^2 f(x_k)$;
  $x_{k+1} = x_k - \gamma_{k+1} Q_k \mathcal{R}_\lambda^{f,g}(x_k) + \sqrt{2\gamma_{k+1}}\mathcal{Z}_k$
end

Algorithm 1: SULA (Smoothing Unadjusted Langevin Algorithm).

(H4) The FBE function $\mathcal{U}_\lambda^{f,g}$ is strongly convex with modulus $\mu > 0$; i.e., for all $x, y \in \mathbb{R}^p$,

$$\mathcal{U}_\lambda^{f,g}(y) \geq \mathcal{U}_\lambda^{f,g}(x) + \langle \nabla \mathcal{U}_\lambda^{f,g}(x), y - x\rangle + \frac{\mu}{2}\|y - x\|_2^2.$$

Note that the condition (H3) has been proved to satisfy FBE in Theorem 1(vi), while the condition (H4) has been shown when the function f in our composite setting is quadratic (Theorem 2).

Our first result studies the linear convergence for cases where the FBE function $\mathcal{U}_\lambda^{f,g}(\cdot)$ is L-smooth and μ-strongly convex.

Theorem 3 *(Convergence analysis of SULA with constant step size). Let $f : \mathbb{R}^p \to \mathbb{R}$ and $g : \mathbb{R}^p \to \mathbb{R}$ be two functions satisfying (H1) and (H2). If $\{x_k\}_{k\geq 0}$ is generated by SULA (Algorithm 1) with constant step size $\gamma_k = \gamma \in (0, 2/L)$ $\{x_k\}_{k\geq 0}$, then*

$$\mathbb{E}\left[\mathcal{U}_\lambda^{f,g}(x_{k+1}) - \mathcal{U}_\lambda^{f,g}(x^*)\right] \leq \left(1 - \frac{2\omega_k}{\mu}\right)^{k+1} \mathbb{E}\left[\mathcal{U}_\lambda^{f,g}(x_0) - \mathcal{U}_\lambda^{f,g}(x^*)\right] + \frac{\mu L \gamma p}{\omega_k} \tag{16}$$

and

$$\mathbb{E}(\|x_{k+1} - x^*\|^2) \leq \frac{L}{\mu}\left(1 - \frac{2\omega_k}{\mu}\right)^{k+1} \mathbb{E}\left(\|x_0 - x^*\|^2\right) + \frac{\mu L \gamma p}{\omega_k} \tag{17}$$

where $\omega_k := \frac{\gamma}{\lambda}\left(1 - \frac{L\gamma}{2}\right)\|Q_k\|^2(\lambda L_f(\lambda L_f + 2))$ and $Q_k = I - \lambda\nabla^2 f(x_k)$.

Proof. It follows from the Lipschitz smoothness of the forward-backward envelope $\mathcal{U}_\lambda^{f,g}$ (Theorem 1(vi)) and (14) that

$$\begin{aligned}
\mathcal{U}_\lambda^{f,g}(x_{k+1}) &\leq \mathcal{U}_\lambda^{f,g}(x_k) + \left\langle Q_k \mathcal{R}_\lambda^{f,g}(x_k), x_{k+1} - x_k\right\rangle + \frac{L}{2}\|x_{k+1} - x_k\|^2 \\
&\leq \mathcal{U}_\lambda^{f,g}(x_k) - \gamma\left\langle Q_k \mathcal{R}_\lambda^{f,g}(x_k), Q_k \mathcal{R}_\lambda^{f,g}(x_k)\right\rangle + \sqrt{2\gamma}\left\langle Q_k \mathcal{R}_\lambda^{f,g}(x_k), \mathcal{Z}_{k+1}\right\rangle + \frac{L}{2}\left\|-\gamma Q_k \mathcal{R}_\lambda^{f,g}(x_k) + \sqrt{2\gamma}\mathcal{Z}_{k+1}\right\|^2 \\
&\leq \mathcal{U}_\lambda^{f,g}(x_k) - \left(\gamma - \frac{L\gamma^2}{2}\right)\left\|Q_k \mathcal{R}_\lambda^{f,g}(x_k)\right\|^2 + \sqrt{2\gamma}(1 - L\gamma)\left\langle Q_k \mathcal{R}_\lambda^{f,g}(x_k), \mathcal{Z}_{k+1}\right\rangle + L\gamma\|\mathcal{Z}_{k+1}\|^2.
\end{aligned}$$

Taking expectation from both sides of the latter inequality and using $E\left[\mathcal{Z}_{k+1}\right] = 0$, $E\left[\|\mathcal{Z}_{k+1}\|^2\right] = p$, and the uniform boundedness of $\nabla^2 f(\cdot)$ (i.e., $\|\nabla^2 f(\cdot)\| \leq L_1$), we come to

$$\begin{aligned}
\mathbb{E}\left[\mathcal{U}_\lambda^{f,g}(x_{k+1}) - \mathcal{U}_\lambda^{f,g}(x_k)\right] &\leq -\left(\gamma - \frac{L\gamma^2}{2}\right)\|Q_k\|^2 \mathbb{E}\left[\left\|\mathcal{R}_\lambda^{f,g}(x_k)\right\|^2\right] + \sqrt{2\gamma}(1 - L\gamma)\left\langle Q_k \mathcal{R}_\lambda^{f,g}(x_k), \mathbb{E}[\mathcal{Z}_k]\right\rangle \\
&\quad + 2L\gamma\mathbb{E}\left[\|\mathcal{Z}_k\|^2\right] \\
&= -\left(\gamma - \frac{L\gamma^2}{2}\right)\|Q_k\|^2 \mathbb{E}\left[\left\|\mathcal{R}_\lambda^{f,g}(x_k)\right\|^2\right] + L\gamma p.
\end{aligned}$$

Setting $\alpha_k := \gamma\left(1 - \frac{L\gamma}{2}\right)\|Q_k\|^2$, the fact that $x^* = \mathcal{T}_\lambda^{f,g}(x^*)$, and using (11) and the definition $\mathcal{R}_\lambda^{f,g}(x_k) = \frac{1}{\lambda}(x_k - \mathcal{T}_\lambda^{f,g}(x_k))$, we deduce that

$$\begin{aligned}
\mathbb{E}\left[\mathcal{U}_\lambda^{f,g}(x_{k+1}) - \mathcal{U}_\lambda^{f,g}(x_k)\right] &\leq L\gamma p - \frac{\alpha_k}{\lambda}\mathbb{E}\left[\left\|x_k - x^* - \left(\mathcal{T}_\lambda^{f,g}(x_k) - \mathcal{T}_\lambda^{f,g}(x^*)\right)\right\|^2\right] \\
&\leq -\frac{\alpha_k}{\lambda}\mathbb{E}\left[\|\mathcal{T}_\lambda^{f,g}(x_k) - \mathcal{T}_\lambda^{f,g}(x^*)\|^2 - \|x_k - x^*\|^2\right] + L\gamma p \\
&= -\frac{\alpha_k}{\lambda}(\lambda L_f(\lambda L_f + 2))\mathbb{E}\left[\|x_k - x^*\|^2\right] + L\gamma p.
\end{aligned}$$

Let us set $\omega_k := \frac{\alpha_k}{\lambda}(\lambda L_f(\lambda L_f + 2))$. Subtracting $\mathcal{U}_\lambda^{f,g}(x^*)$ from both sides of the above inequality leads to

$$\mathbb{E}\left[\mathcal{U}_\lambda^{f,g}(x_{k+1}) - \mathcal{U}_\lambda^{f,g}(x^*)\right] \leq \mathbb{E}\left[\mathcal{U}_\lambda^{f,g}(x_k) - \mathcal{U}_\lambda^{f,g}(x^*)\right] - \omega_k \mathbb{E}\left[\|x_k - x^*\|^2\right] + L\gamma p.$$

Here, using strong convexity of $\mathcal{U}_\lambda^{f,g}$ and $\nabla \mathcal{U}_\lambda^{f,g}(x^*) = 0$, it holds that

$$\|x_k - x^*\|^2 \leq \frac{2}{\mu}\left(\mathcal{U}_\lambda^{f,g}(x_k) - \mathcal{U}_\lambda^{f,g}(x^*)\right),$$

and thus

$$\mathbb{E}\left[\|x_k - x^*\|^2\right] \le \frac{2}{\mu}\mathbb{E}\left[\mathcal{U}_\lambda^{f,g}(x_k) - \mathcal{U}_\lambda^{f,g}(x^*)\right].$$

Combining the last three inequalities yields

$$\mathbb{E}\left[\mathcal{U}_\lambda^{f,g}(x_{k+1}) - \mathcal{U}_\lambda^{f,g}(x^*)\right] \le \left(1 - \frac{2\omega_k}{\mu}\right)\mathbb{E}\left[\mathcal{U}_\lambda^{f,g}(x_k) - \mathcal{U}_\lambda^{f,g}(x^*)\right] + L\gamma p.$$

Now, by recursion, we get

$$\begin{aligned}
\mathbb{E}\left[\mathcal{U}_\lambda^{f,g}(x_{k+1}) - \mathcal{U}_\lambda^{f,g}(x^*)\right] &\le \left(1 - \frac{2\omega_k}{\mu}\right)^2 \mathbb{E}\left[\mathcal{U}_\lambda^{f,g}(x_{k-1}) - \mathcal{U}_\lambda^{f,g}(x^*)\right] + 2L\gamma p\left(1 + \left(1 - \frac{2\omega_k}{\mu}\right)\right) \\
&\le \dots \\
&\le \left(1 - \frac{2\omega_k}{\mu}\right)^{k+1} \mathbb{E}\left[\mathcal{U}_\lambda^{f,g}(x_0) - \mathcal{U}_\lambda^{f,g}(x^*)\right] + L\gamma p\left(1 + \dots + \left(1 - \frac{2\omega_k}{\mu}\right)^k\right) \\
&\le \left(1 - \frac{2\omega_k}{\mu}\right)^{k+1} \mathbb{E}\left[\mathcal{U}_\lambda^{f,g}(x_0) - \mathcal{U}_\lambda^{f,g}(x^*)\right] + \tfrac{\mu L\gamma p}{2\omega_k},
\end{aligned}$$

which gives (16).

From the μ-strong convexity of $\mathcal{U}_\lambda^{f,g}$, we obtain

$$\mathbb{E}\left[\|x_{k+1} - x^*\|^2\right] \le \frac{2}{\mu}\mathbb{E}\left[\mathcal{U}_\lambda^{f,g}(x_{k+1}) - \mathcal{U}_\lambda^{f,g}(x^*)\right].$$

Moreover, it follows from L-Lipschitz smoothness of $\mathcal{U}_\lambda^{f,g}$ that

$$\frac{2}{\mu}\mathbb{E}\left[\mathcal{U}_\lambda^{f,g}(x_0) - \mathcal{U}_\lambda^{f,g}(x^*)\right] \le \frac{L}{\mu}\mathbb{E}\left[\|x_0 - x^*\|^2\right].$$

Together with (16), these two inequalities imply

$$\begin{aligned}
\mathbb{E}\left[\|x_{k+1} - x^*\|^2\right] &\le \frac{2}{\mu}\mathbb{E}\left[\mathcal{U}_\lambda^{f,g}(x_{k+1}) - \mathcal{U}_\lambda^{f,g}(x^*)\right] \\
&\le \frac{2}{\mu}\left(1 - \frac{2\omega_k}{\mu}\right)^{k+1} \mathbb{E}\left[\mathcal{U}_\lambda^{f,g}(x_0) - \mathcal{U}_\lambda^{f,g}(x^*)\right] + \tfrac{L\gamma p}{\omega_k} \\
&\le \frac{L}{\mu}\left(1 - \frac{2\omega_k}{\mu}\right)^{k+1} \mathbb{E}\left[\|x_0 - x^*\|^2\right] + \tfrac{L\gamma p}{\omega_k},
\end{aligned}$$

adjudicating our claim. □

In the subsequent result, we measure the error of our sampling scheme in 2-*Wasserstein distance* (i.e., $\mathcal{W}_2$), which we define next. Let u and v be two probability distributions, then the 2-Wasserstein distance between these two probability distributions is given by

$$\mathcal{W}_2^2(u, v) = \inf_{X\sim u, Y\sim v} \mathbb{E}\left[\|X - Y\|_2^2\right].$$

Let $\mathcal{X}_k$ be the distribution of the kth iterate of our algorithm SULA (i.e., $x_k \sim \mathcal{X}_k$).

To prove Theorem 5, we need the following fact:

Fact 4. *[15, Lemma 2, Lemma 4] The following statements are true:*

(a) *Let $h : \mathbb{R}^p \to \mathbb{R}$ be L-smooth and μ-strongly convex. If $\gamma \in (0, 2/L)$, then the mapping $I - \gamma\nabla h$ satisfies*

$$\|x - y - \gamma(\nabla h(x) - \nabla h(y))\| \le \rho\|x - y\|,$$

for all $x, y \in \mathbb{R}^p$ with $\rho = \max\{1 - \gamma\mu, \gamma L - 1\}$. In particular, for $\Delta_k = L_0 - x_k$ and $A := \nabla\mathcal{U}_\lambda^{f,g}(x_k + \Delta_k) - \nabla\mathcal{U}_\lambda^{f,g}(x_k)$, we have $\|\Delta_k - \gamma A\| \le \rho\|\Delta_k\|$.

(b) *Let $\mathcal{U}_\lambda^{f,g} : \mathbb{R}^p \to \mathbb{R}$ be L-smooth, and let $\mathcal{L}$ be the Langevin diffusion given by*

$$\mathcal{L}_t = \mathcal{L}_0 - \int_0^t \nabla\mathcal{U}_\lambda^{f,g}(\mathcal{L}_s)ds + \sqrt{2}W_t, \quad \forall t > 0, \tag{18}$$

and $B := \int_a^{a+\gamma}(\nabla\mathcal{U}_\lambda^{f,g}(\mathcal{L}_t) - \nabla\mathcal{U}_\lambda^{f,g}(\mathcal{L}_a))dt$ for $a \ge 0$, then

$$\|B\|_{L_2} \le \frac{1}{2}\sqrt{\gamma^4 L^3 p} + \frac{2}{3}L\sqrt{2\gamma^3 p}.$$

Theorem 5 *(Convergence analysis of SULA in 2-Wasserstein distance). Let $f:\mathbb{R}^p\to\mathbb{R}$ and $g:\mathbb{R}^p\to\mathbb{R}$ be two functions satisfying (H1) and (H2) such that (H3) and (H4) hold for the forward-backward envelope $U_\lambda^{f,g}:\mathbb{R}^p\to\mathbb{R}$. If $\{x_k\}_{k\geq 0}$ is generated by SULA (Algorithm 1) with constant step size $\gamma_k=\gamma\in(0,2/L)$ $\{x_k\}_{k\geq 0}$, then the following statements hold:*

(i) *If $\gamma\leq\frac{2}{\mu+L}$, then*

$$\mathcal{W}_2(\mathcal{X}_k,\pi)\leq(1-\gamma\mu)^k\mathcal{W}_2(\mathcal{X}_0,\pi)+1.65\left(\frac{L}{\mu}\right)\sqrt{\gamma p}. \tag{19}$$

(ii) *If $\gamma\geq\frac{2}{\mu+L}$, then*

$$\mathcal{W}_2(\mathcal{X}_k,\pi)\leq(L\gamma-1)^k\mathcal{W}_2(\mathcal{X}_0,\pi)+\left(\frac{1.65L\gamma}{2-L\gamma}\right)\sqrt{\gamma p}. \tag{20}$$

Proof. Let $\mathcal{L}_0$ be a random vector drawn from π such that the identity $\mathcal{W}_2(\mathcal{X}_k,\pi)=\|\mathcal{L}_0-x_k\|_{L_2}$, where we define the norm $\|X\|_{L_2}=(\mathbb{E}[\|X\|_2^2])^{1/2}$. Let W be a p-dimensional Brownian Motion independent of $(x_k,\mathcal{L}_0)$ such that $W_\gamma=\sqrt{\gamma}\mathcal{Z}_{k+1}$. Now, let us define the stochastic process $\mathcal{L}$ by (18), which leads to

$$\begin{aligned}\mathcal{L}_\gamma&=\mathcal{L}_0-\int_0^\gamma\nabla U_\lambda^{f,g}(\mathcal{L}_s)ds+\sqrt{2}W_\gamma\\&=\mathcal{L}_0-\int_0^\gamma\nabla U_\lambda^{f,g}(\mathcal{L}_s)ds+\sqrt{2\gamma}\mathcal{Z}_{k+1}.\end{aligned}$$

We note that $\{\mathcal{L}_t\mid t\geq 0\}$ is a diffusion process having π as the stationary distribution. Since the initial value $\mathcal{L}_0$ is drawn from π, we get $\mathcal{L}_t\sim\pi$ for every $t\geq 0$.

We here define $\Delta_k=\mathcal{L}_0-x_k$ and $\Delta_{k+1}=\mathcal{L}_\gamma-x_{k+1}$, which consequently leads to

$$\begin{aligned}\Delta_{k+1}&=\Delta_k+\gamma\nabla U_\lambda^{f,g}(x_k)-\int_0^\gamma\nabla U_\lambda^{f,g}(\mathcal{L}_t)dt\\&=\Delta_k-\gamma\left(\nabla U_\lambda^{f,g}(x_k+\Delta_k)-\nabla U_\lambda^{f,g}(x_k)\right)\\&\quad-\int_0^\gamma(\nabla U_\lambda^{f,g}(\mathcal{L}_t)-\nabla U_\lambda^{f,g}(\mathcal{L}_0))dt.\end{aligned}$$

Setting $A:=\nabla U_\lambda^{f,g}(x_k+\Delta_k)-\nabla U_\lambda^{f,g}(x_k)$ and $B:=\int_0^\gamma(\nabla U_\lambda^{f,g}(\mathcal{L}_t)-\nabla U_\lambda^{f,g}(\mathcal{L}_0))dt$ and using the latter identity, we come to

$$\|\Delta_{k+1}\|_{L_2}\leq\|\Delta_k-\gamma A\|_{L_2}+\|B\|_{L_2}.$$

Invoking Facts 4(a) and 4(b) and using $\gamma L<2$ and the fact that $\mathcal{W}_2(\mathcal{X}_k,\pi)^2\leq\mathbb{E}\left[\|\Delta_{k+1}\|^2\right]$ yields

$$\begin{aligned}\mathcal{W}_2(\mathcal{X}_{k+1},\pi)&\leq\rho\mathcal{W}_2(\mathcal{X}_k,\pi)+\tfrac{1}{2}\sqrt{\gamma^4L^3p}+\tfrac{2}{3}\sqrt{2\gamma^3p}L\\&\leq\rho\mathcal{W}_2(\mathcal{X}_k,\pi)+\tfrac{1}{2}\sqrt{2\gamma^3L^2p}+\tfrac{2}{3}\sqrt{2\gamma^3p}L\\&=\rho\mathcal{W}_2(\mathcal{X}_k,\pi)+\tfrac{7\sqrt{2}}{6}\sqrt{\gamma^3p}L\\&\leq\rho\mathcal{W}_2(\mathcal{X}_k,\pi)+1.65\sqrt{\gamma^3p}L.\end{aligned}$$

Now, by recursion, we get

$$\begin{aligned}\mathcal{W}_2(\mathcal{X}_{k+1},\pi)&\leq\rho\mathcal{W}_2(\mathcal{X}_k,\pi)+1.65\sqrt{\gamma^3p}L\leq\rho^2\mathcal{W}_2(\mathcal{X}_{k-1},\pi)+(1+\rho)1.65\sqrt{\gamma^3p}L\\&\leq\dots\\&\leq\rho^{k+1}\mathcal{W}_2(\mathcal{X}_0,\pi)+(1+\rho+\dots+\rho^k)1.65\sqrt{\gamma^3p}L\\&\leq\rho^{k+1}\mathcal{W}_2(\mathcal{X}_0,\pi)+\tfrac{(1-\rho^{k+1})1.65}{1-\rho}\sqrt{\gamma^3p}L\\&\leq\rho^{k+1}\mathcal{W}_2(\mathcal{X}_0,\pi)+\tfrac{1.65}{1-\rho}\sqrt{\gamma^3p}L,\end{aligned}$$

giving our desired result. □

We next consider the case where the SULA algorithm employs varying step sizes.

Theorem 6 *(Convergence analysis of SULA with varying step size). ([15], Theorem 2) Let $f : \mathbb{R}^p \to \mathbb{R}$ and $g : \mathbb{R}^p \to \bar{\mathbb{R}}$ be two functions satisfying (H1) and (H2). If $\{x_k\}_{k\geq 0}$ is generated by SULA (Algorithm 1) with varying step size defined by*

$$\gamma_{k+1} = \frac{2}{L + \mu + (2/3)\mu(k - K1)}, \ k = 1, 2, \cdots \tag{21}$$

where K_1 is the smallest non-negative integer satisfying

$$K_1 \geq \frac{\ln(W_2(\nu_0, \pi)/\sqrt{p} + \ln(\mu/L) + (1/2)\ln(L + \mu)}{\ln(1 + 2\mu/L - \mu)}. \tag{22}$$

If $\mathcal{U}$ is strongly convex, then for every $k \geq K_1$, we have

$$W_2(\nu_0, \pi) \leq \frac{3.5L\sqrt{p}}{\mu\sqrt{L + \mu + (2/3)\mu(k - K1)}}. \tag{23}$$

4. Numerical experiments

In this section, we report numerical results to illustrate the performance of the Smoothing Unajusted Langevin Algorithm (SULA) developed in the previous section. In our implementation, all the code was written in Python and tested on a MacBook Pro with 32 GB Ram. In these experiments, we apply SULA to Bayesian linear regression with Laplace prior with both synthetic and actual chemoinformatics data. We will show that this problem satisfies all required assumptions to apply SULA. Moreover, in our implementation, the initial parameters were generated randomly from a uniform distribution.

4.1. Bayesian linear regression

Let us consider the problem of estimating the parameter x in the linear regression problem; i.e.,

$$Y = D^T x + \epsilon, \tag{24}$$

where $(D, Y) \in \mathbb{R}^{n\times p} \times \mathbb{R}^n$ are the given data (i.e., $D = \{d_1, d_2, \cdots, d_p\}$ with $d_i, Y \in \mathbb{R}^n$) and $\epsilon \in \mathbb{R}^p$ is a p-variate standard normal vector. The vector $x \in \mathbb{R}^p$ is the regression coefficient parameter to be estimated. For this case, we consider the likelihood function

$$P(D|x) = \Pi_{i=0}^{n} \frac{1}{\sqrt{2\pi}} \exp - \frac{\|y_i - d_i^T x_i\|^2}{2},$$
$$\log P(D|x) = -\frac{n}{2}\log(2\pi) - \frac{1}{2}\|Y - D^T x\|^2.$$

Since the term $\frac{p}{2}\ln(2\pi)$ is a constant, without loss of generality, we remove it from the log-likelihood function leading to $f(x) := -\log P(D|x)$. It is clear that $\nabla^2 f(x) = DD^T$, which implies that

$$L_f = \|DD^T\|_F.$$

Moreover, we are interested in a sparse solution for this problem in which most of the x_i be zero or close to zero. To this end, we consider the Laplace distribution as the prior distribution, which is

$$\ln P(x) = -\beta\|x\|_1,$$

for the regularization parameter $\beta > 0$ (which controls sparsity) leading to $g(x) := -\log P(x) = \beta\|x\|_1$. We finally come to the composite function

$$\mathcal{U}(x) = f(x) + g(x) = \frac{1}{2}\|Y - D^T x\|^2 + \beta\|x\|_1, \tag{25}$$

where f is quadratic function and g is a nonsmooth function.

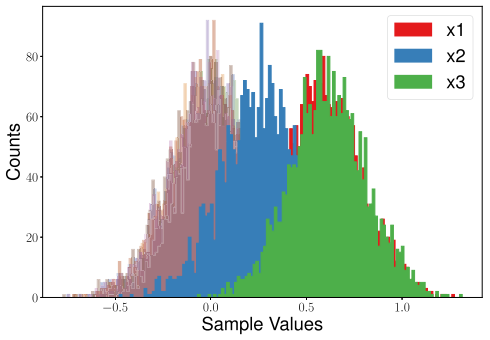
Applying SULA to the described Bayesian linear regression with Laplace prior demands the computation of the matrices P and Q and the mapping $\mathcal{T}_\lambda^{f,g}$, which are given by

$$P = \frac{\lambda - \gamma_{k+1}}{\lambda} I - \gamma_{k+1} DD^T, \quad Q = I - \lambda DD^T,$$
$$\mathcal{T}_\lambda^{f,g}(x) = \text{prox}_{\lambda\|\cdot\|_1}(\alpha),$$

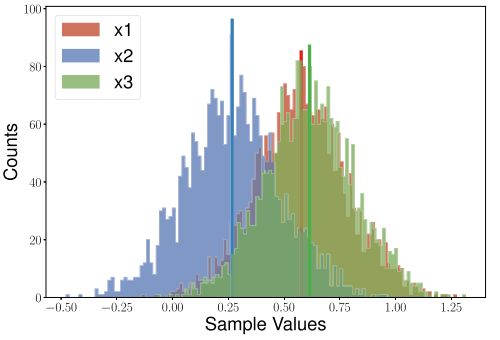
where $\text{prox}_{\lambda\|\cdot\|_1}(\cdot)$ is the soft thresholding operator given by

$$\text{prox}_{\lambda\beta\|\cdot\|_1}(\alpha) = \text{sign}(\alpha) \cdot \max\{|\alpha| - \lambda\beta, 0\},$$

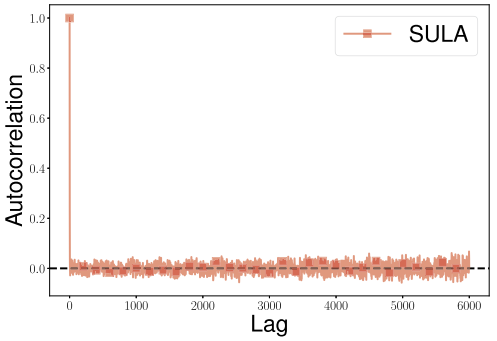
with $\alpha = x - \lambda D(Y - D^T x)$. Hence, all elements of SULA are effectively computed, thereby leading to its efficient implementation. Let us emphasize that the forward-backward envelope $\mathcal{U}_\lambda^{f,g}$ is L-smooth (see Theorem 1(vi)), and since f is quadratic, $\mathcal{U}_\lambda^{f,g}$ is μ-strongly convex (see Theorem 2).



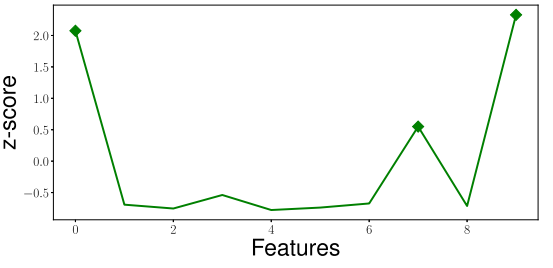
(a) The histogram of the posterior probability of nonzero coefficients with their mean.



(b) The histogram of the posterior distribution of x_is.



(c) SULA autocorrelation function plot of the first coordinate for 300 iterations.



(d) The z-score plot of all features.

Fig. 1. The results of SULA algorithm with constant step size on the synthetic data. We see that the mean value for the linear feature is apart from zero and the z-score of them is higher than others, while the samples are independent.

Let us clarify here that the sparse solution does not mean that the components of the solution are exactly zero because the noise term $\sqrt{2\gamma_{k+1}}\mathcal{Z}_k$ presents a convergence of the parameters to a fully sparse solution. However, we can see that the means of the features are fluctuating around zero, which can be considered as zero solutions. In all the implementations, we sample $2,000$ chains in parallel with $5,000$ samples per chain. Furthermore, we set $\lambda < \frac{1}{Lf}$.

4.2. Implementation of synthetic data

The goal of this section is to apply SULA to a simple synthetic data set to visualize the performance of the algorithm in a low-dimensional setting. More specifically, we apply SULA to the Bayesian linear regression model for randomly generated synthetic data with ten features, where we employ both the constant and dynamic step-size variants of SULA.

In the experiments depicted in Fig. 1 and 2, the size of our synthetic data is 100×10, where we randomly selected 20% of the data as test data. In this data, three of the features have a linear relationship with y, namely, x_1, x_8, and x_{10}, and seven features are irrelevant.

For these experiments, let us begin with the constant step-size variant of SULA. The following parameters are considered:

$$\beta = 2,\ \gamma = \frac{2}{\mu + L},\ \text{iteration} = 4000,\ \text{samples} = 2500,\ \lambda = \frac{0.8}{L_f}.$$

We plotted the histogram of the posterior distributions for all 10 coefficients of the features in Fig. 1(a). To have a clear representation, the histogram of the features whose mean value is different from zero is plotted in Fig. 1(a). In Fig. 1(c), we plot the autocorrelation of samples for 10,000 samples, as used in the code of [21]. Furthermore, we plotted the z-score of features as a measure of sparseness in Fig. 1(d). The results of Fig. 1(a) and (b) show that for the non-relevant features the mean is around zero and for the features with linear coefficients (i.e., x_1, x_8, and x_{10}) the mean is far from zero. Fig. 1(c) shows that the autocorrelation between samples is small. In addition, we computed the z-score of all features as

$$z = \frac{x - \mu}{\sigma},$$

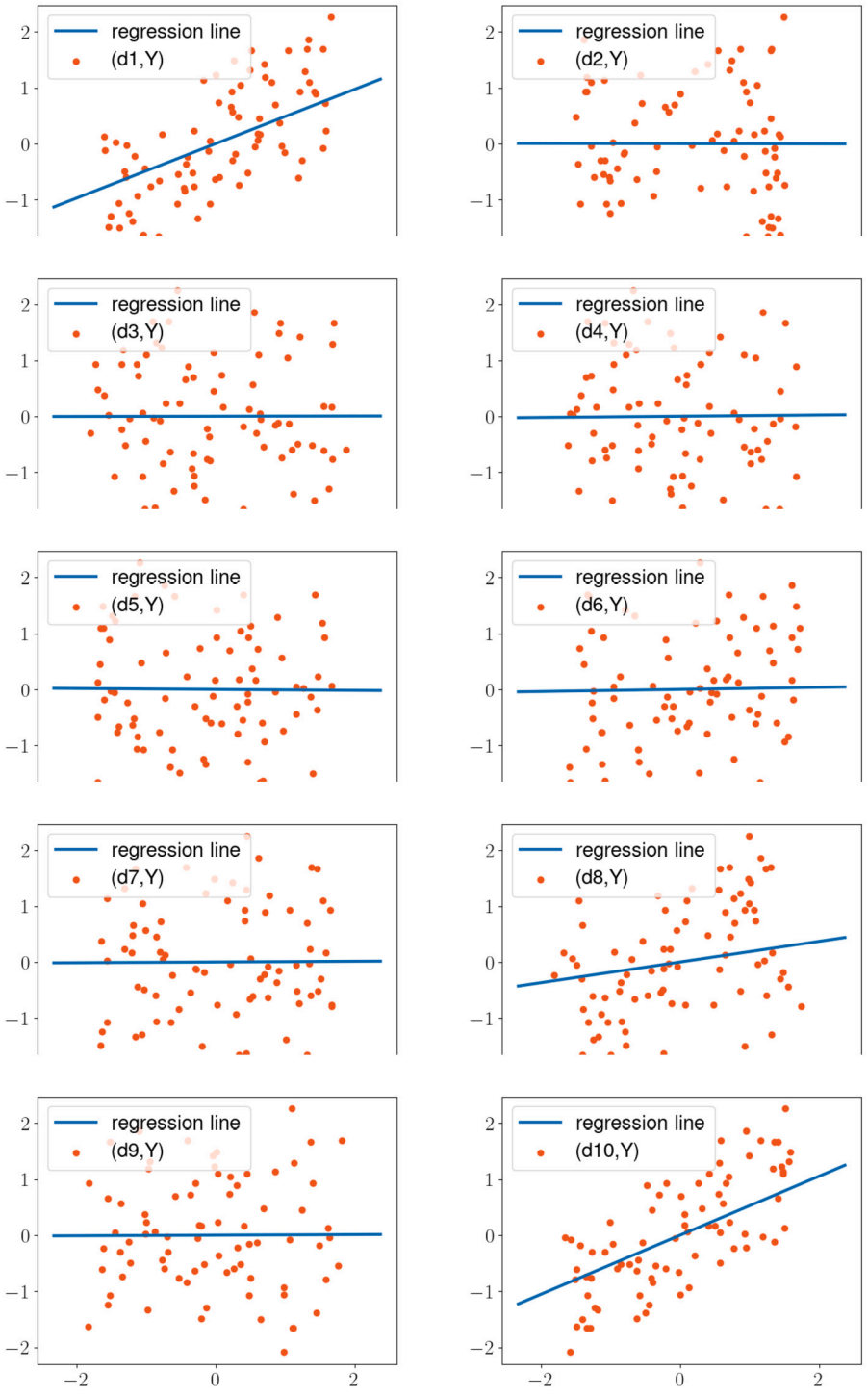
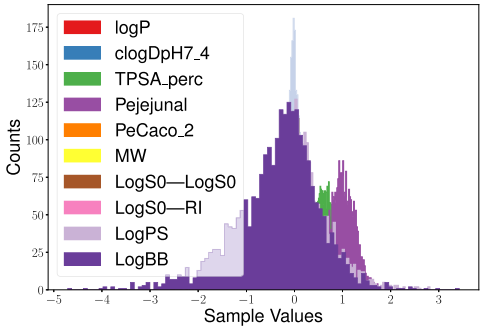
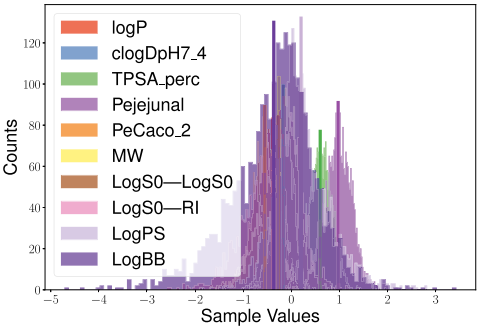


Fig. 2. The predicted regression line and the data points for each feature of synthetic data using SULA with constant step size on the synthetic data set. As it is shown, the slope of the regression line for features x_1, x_8, and x_{10} is not zero, while the slope of other features is zero.

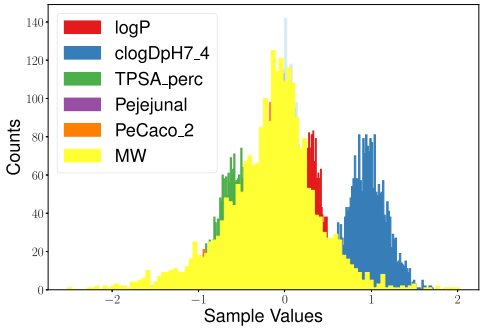
where μ is the mean of each feature and σ is the standard deviation of them. The results are plotted in Fig. 1(d), which shows the z-score of the posterior distribution of features. This score might be considered as a measure to decide about the zero features. For example, the z-scores of $\{x_1, x_8, x_9\}$ are significantly higher than the other features. Therefore, we can consider a threshold, κ, (i.e., in this example $\kappa = 0.5$), and if the z-score is less than κ consider those features as having a zero coefficient.



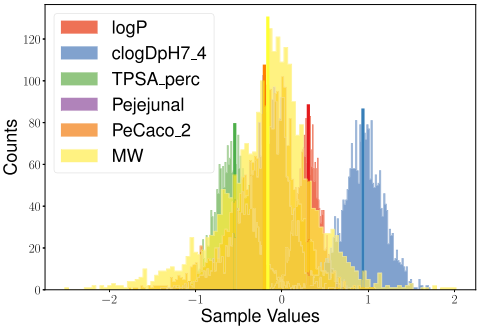
(a) The histogram of the posterior distributions of all features for $\beta = 1$.



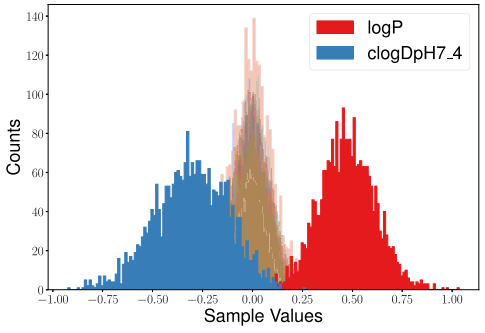
(b) The histogram of the posterior distributions of all features for $\beta = 2$.



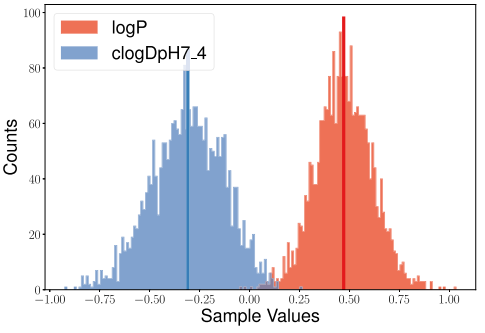
(c) The histogram of the posterior distributions of all features for $\beta = 10$.



(d) The histogram of the posterior distributions of non-zero coefficients $\beta = 1$, with their means as vertical lines.



(e) The histogram of the posterior distributions of non-zero coefficients $\beta = 2$, with their means as vertical lines.



(f) The histogram of the posterior distributions of non-zero coefficients $\beta = 10$, with their means as vertical lines.

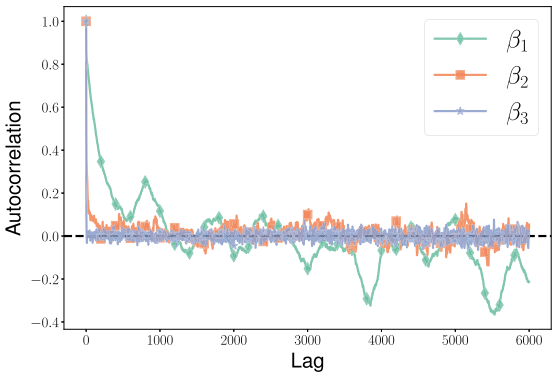
Fig. 3. The histogram of the posterior distributions of coefficient of all features (a), (b), and (c) and the non-zero coefficients (d), (e) and (f), using SULA with constant step size on the chemoinformatics data set.

In addition, we plotted the scatter plot of the data with a predicted regression line in which the slopes of the predicted lines are the mean of the posterior distribution of each coefficient, see Fig. 2. The results show that SULA predicts the non-zero features correctly in this example.

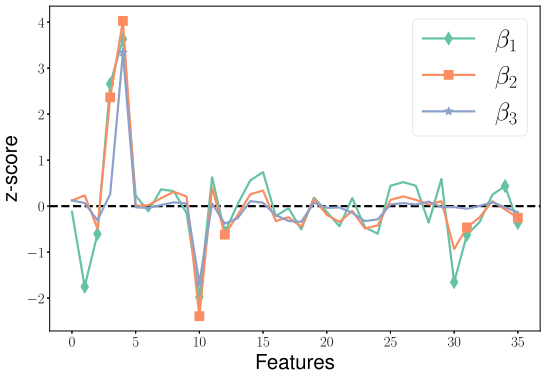
Since the step size varies slowly, it is close to the constant step size. The reason is that in practice since K_1 in (22) is going to be a large number and therefore $(2/3)\mu(k - K_1)_+$ has a very low impact on the step size, in practice the step size is close to $h_k = \frac{2}{\mu+L}$. Therefore, we moved the results of the varying step size scheme to Appendix A, see Fig. 9.

4.3. Experiments on actual data

In this section, we use our model on an actual data set from a chemoinformatics study. The dimension of this data set is 38×121. The features correspond to the physicochemical properties of chemical compounds, and the prediction target is the measured ability



(a) Autocorrelation function for SULA for β_1, β_2 and β_3 on actual data.



(b) The z-score of posterior distributions on actual data.

Fig. 4. Effect of the scale of sparsity parameter on the performance of SULA with constant step size on chemoinformatics data set.

of the compound to penetrate a lipid membrane. The data set is published by [55]. As in the previous experiment, we randomly selected 20 percent of the data as a test set. Since the scaling of the variables is very different (the range of eigenvalues is in the interval $[10^{-13}, 10^{13}]$), we first normalize the features and standardize the Y values.

We considered the same parameters as in the experiment on the synthetic data. However, here we considered three β as $\{1, 2, 10\}$ to test the ability of SULA in producing sparse solutions with different ranges of step sizes.

We plot the histogram of the posterior distributions of coefficients of all features in Fig. 3(a), (c), and (e) and the posterior histogram of all non-zero features in Fig. 3(b), (d), and (f). Fig. 3(a), (c), and (e) show that when $\beta = 1$, $\beta = 2$, and $\beta = 10$, the ratio of producing non-sparse solutions is $9/37$, $5/37$, and $2/37$, respectively. Moreover, since the mean of other features is zero we consider them as the zeros of a sparse solution. From Fig. 3 (c), we observe that the two features logP and clogDpH7 $-$ 4 have higher linear coefficients and that their z-value appears as significant in all three plots. These properties were also found relevant by the original publication [55] and have a reasonable mechanistic influence on the measured penetration.

In Fig. 4(a), we plotted the autocorrelation function for $\{\beta_1, \beta_2, \beta_3\}$ and we plotted the z-score of each solution in Fig. 4(b). The $R2$ score for $\{\beta_1, \beta_2, \beta_3\}$ are $\{0.63, 0.57, 0.28\}$. One can see that the autocorrelation between samples for β_1 is larger than for β_2 and β_3. From Fig. 4(b), we see that the higher z-scores are related to the features with non-zero mean in the histogram of Fig. 3. Considering $\kappa = 0.3$, and removing features with $|\kappa| < 0.3$ will give us the nonzero parameters and consider the rest as zero parameters.

We plotted the same results for the dynamic step size, which confirms the results for constant step size. The corresponding plots are shown in the Appendix A in Figs. 10 and 11.

4.4. Application to image deblurring

We here consider a deblurring problem using the "cameraman" image with dimensions of 256×256 pixels, and apply SULA and MYULA [20] to this problem, which allows us to further assess the efficacy of SULA compared to MYULA. To this end, a uniform blur operator H of size 5×5 and a Gaussian noise η with a sigma-to-noise level of $40dB$ added to the original image $x \in \mathbb{R}^d$ as $y = Hx + \eta$, resulting in the blurred image, which will be an ill-conditioned problem, due to the near-singularity of the blur operator H. In order to mitigate this issue, we incorporate regularization by adopting the isotropic total variation (TV) pseudonorm as a prior. This regularization approach, denoted by

$$TV(x) = \sum_i \sqrt{(\Delta_i^h x)^2 + (\Delta_i^v x)^2}, \tag{26}$$

where Δ_i^h and Δ_i^v represent horizontal and vertical first-order local difference operators. Consequently, the posterior distributions for this deblurring problem can be expressed as

$$p(x|y, \theta) \propto \exp\left[-\frac{1}{2}\|y - Hx\|^2\sigma^2 - \theta TV(x)\right], \tag{27}$$

where $\theta > 0$ is the regularization parameter. Hence, the potential function is of the composite form $\mathcal{U}(x) = f(x) + g(x)$ with $f(x) = \frac{1}{2}\|y - Hx\|^2$ and $g(x) = \theta TV(x)$.

We apply SULA and MYULA to the problem (27) and report the results in Figs. 5 and 6. In particular, the subfigures (a) and (b) of Fig. (5) stand for the original and blurred images, where the recovered image by MYULA and SULA is depicted in subfigures (c) and (d) of this figure, respectively. It is clear that both algorithm recovered the image with acceptable quality; however, SULA exhibits superior performance in terms of image resolution and lower Mean Squared Error (MSE) values, as is clear from subfigures (a) and (b) of Fig. 6.



(a) True image x



(b) Noisy and blurred observation y

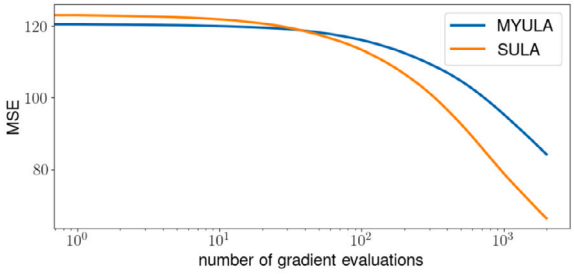


(c) MYULA: posterior mean

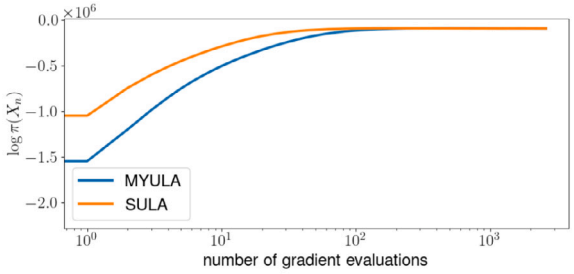


(d) SULA: posterior mean

Fig. 5. Image deblurring experiments: (a) Original image of dimension 256×256 pixels; (b) Blurred observation with $SNR = 40$. (c) Mean of 2000 samples generated by MYULA in stationary. (d) Mean of 4000 samples generated by SULA in stationary.



(a) MSE.

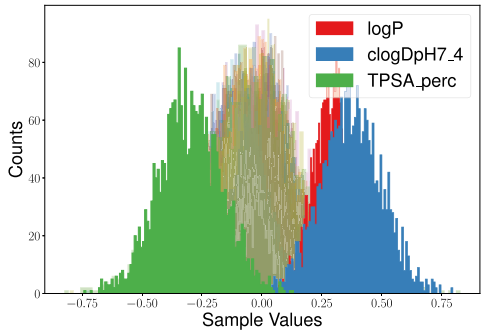


(b) $\log p(X_n|y, \theta)$.

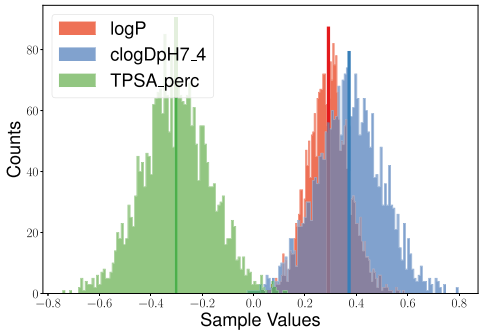
Fig. 6. Image deblurring experiments. MSE between the mean of the algorithms and the true image and convergence to the typical set of the posterior distribution (27) measured using 2000 samples of MYULA and SULA in stationarity.

5. Conclusion

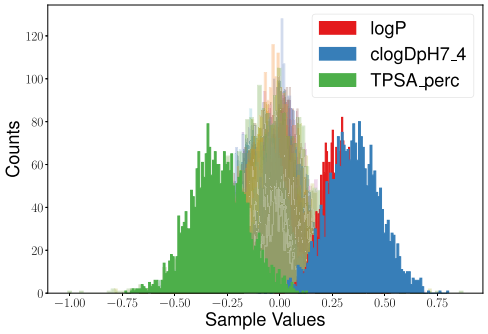
We introduced SULA (smoothing unadjusted Langevin algorithm) as a novel method to sample from a probability distribution with a nonsmooth composite potential function studied. This was achieved by using the *forward-back envelope* (FBE) smoothing technique in combination with the Langevin dynamics, which is a gradient-based MCMC algorithm. In particular, we showed that under some suitable assumptions the gradient of FBE is Lipschitz continuous and the FBE function is strongly convex, which opened the way toward the development of SULA. The non-asymptotic convergence of SULA was shown for both constant and varying step sizes. To



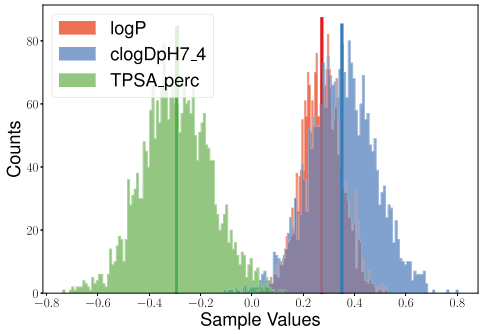
(a) The histogram of the posterior distributions of all features for $\beta = 1$.



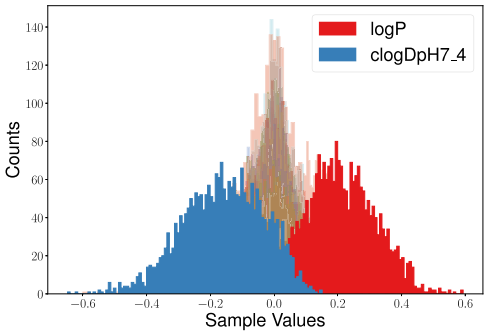
(b) The histogram of the posterior distributions of all features for $\beta = 2$.



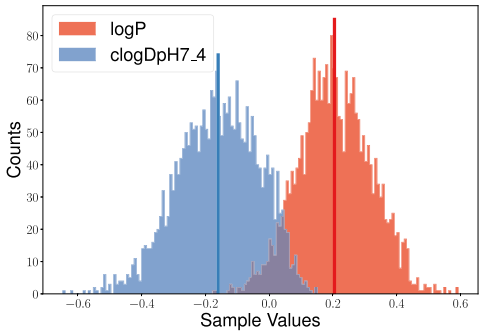
(c) The histogram of the posterior distributions of all features for $\beta = 10$



(d) The histogram of the posterior distributions of non-zero coefficients $\beta = 1$, with their means as vertical lines.



(e) The histogram of the posterior distributions of non-zero coefficients $\beta = 2$, with their means as vertical lines.



(f) The histogram of the posterior distributions of non-zero coefficients $\beta = 10$, with their means as vertical lines.

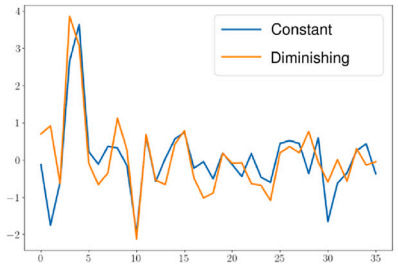
Fig. 7. The histogram of the posterior distributions of coefficient of all features (a), (c), and (e) and the non-zero coefficients (b), (d), and (f) using SULA with diminishing step size on the chemoinformatics data set.

validate our theoretical foundations, we applied SULA to a linear regression problem on synthetic and actual chemoinformatics data sets and demonstrated that the method accurately carried out Bayesian inference on a nonsmooth potential.

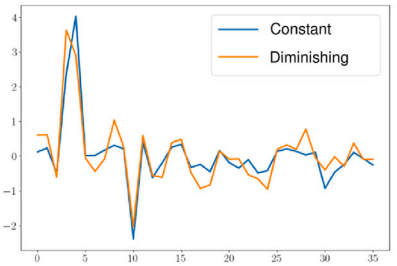
We believe that our SULA method can be further accelerated by employing higher-quality discretization techniques than Euler-Maruyama scheme (e.g., Runge-Kutta schemes as inspired by techniques introduced in [44,45]). This will eventually lead to faster algorithms; however, such algorithmic, theoretical, and numerical developments are out of scope of the current work and will be studied separately.

Data availability

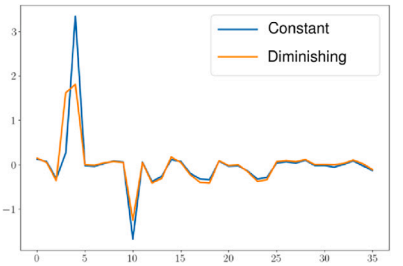
Data will be made available on request.



(a) The z-score plot of constant and diminishing step size for $\beta = 1$.

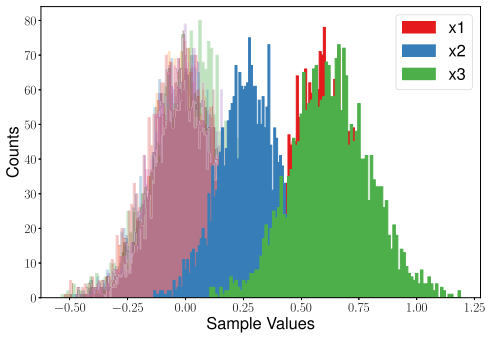


(b) The z-score plot of constant and diminishing step size for $\beta = 2$.

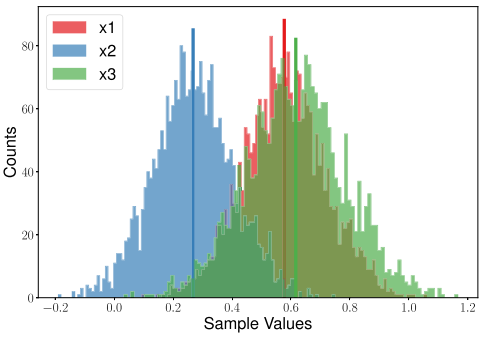


(c) The z-score plot of constant and diminishing step size for $\beta = 10$

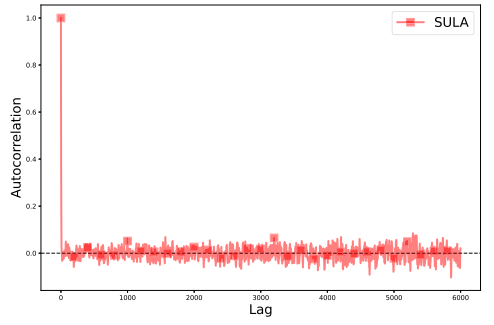
Fig. 8. Comparison between mean values for SULA with constant step size and diminishing step size on the chemoinformatics data set.



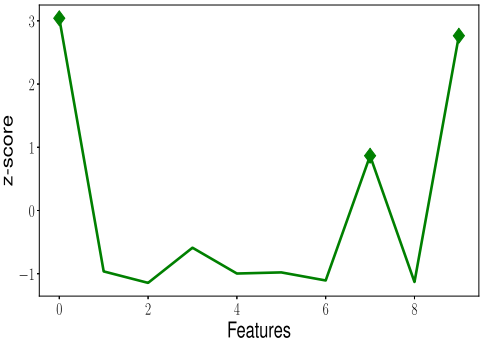
(a) The histogram of the posterior probability of nonzero coefficients with their mean.



(b) The histogram of the posterior distribution of x_is.



(c) SULA autocorrelation function plot of the first coordinate for 300 iterations.

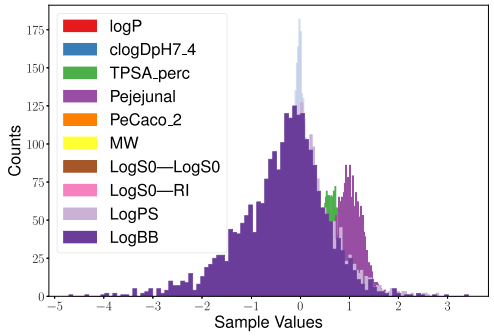


(d) The z-score plot of all features.

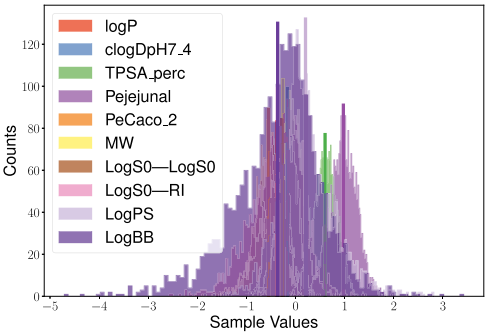
Fig. 9. The results of SULA algorithm with varying step size on the synthetic data set. We see that the mean value for the linear feature is apart from zero and the z-score of them is higher than others, while the samples are independent.

Acknowledgements

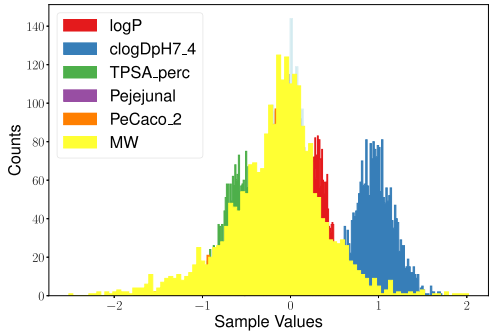
This project has received partially funded by Research Council, KU Leuven (Symbiosis3 (C14/18/092); Federated cloud-based Artificial Intelligence-driven platform for liquid biopsy analyses (C3/20/100); CELSA - Active Learning (CELSA/21/019), CELSA-HIDUCTION (CELSA/17/032)); Flemish Government (FWO: SBO (S003422N), Elixir Belgium (I002819N); SB and Postdoctoral grants; This research received funding from the Flemish Government (AI Research Program). Yves Moreau and Susan Ghaderi are affiliated to Leuven.AI - KU Leuven institute for AI, B-3000, Leuven, Belgium; VLAIO PM: Augmanting Therapeutic Effectiveness through Novel Analytics (HBC.2019.2528)); and EU ("MELLODDY" This project has received funding from the Innovative Medicines Initiative 2 Joint Undertaking under grant agreement No. 831472. This Joint Undertaking receives support from the European Union's Horizon 2020 research and innovation programme and EFPIA). The first author was supported by an FWO junior postdoctoral fellowship [12AK924N]. The second author was partially supported by the Research Foundation Flanders (FWO) grant G081222N and by the



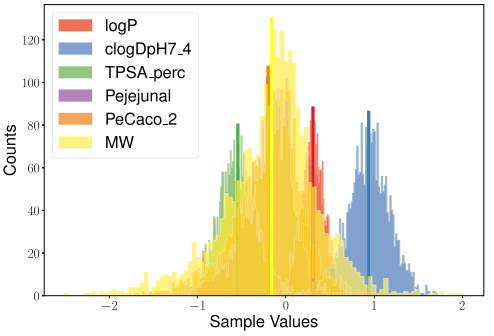
(a) The histogram of the posterior distributions of all features for $\beta = 1$.



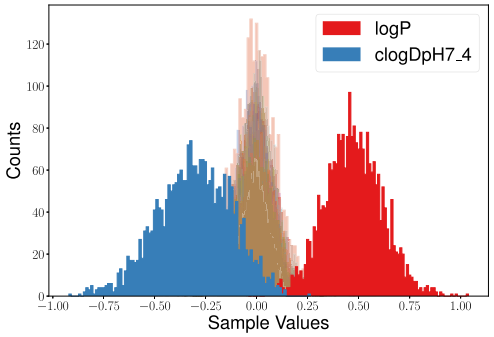
(b) The histogram of the posterior distributions of all features for $\beta = 2$.



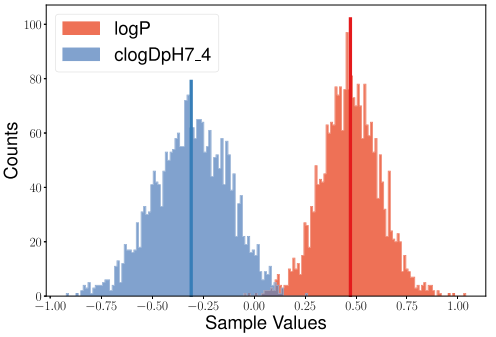
(c) The histogram of the posterior distributions of all features for $\beta = 10$



(d) The histogram of the posterior distributions of non-zero coefficients $\beta = 1$, with their means as vertical lines.



(e) The histogram of the posterior distributions of non-zero coefficients $\beta = 2$, with their means as vertical lines.



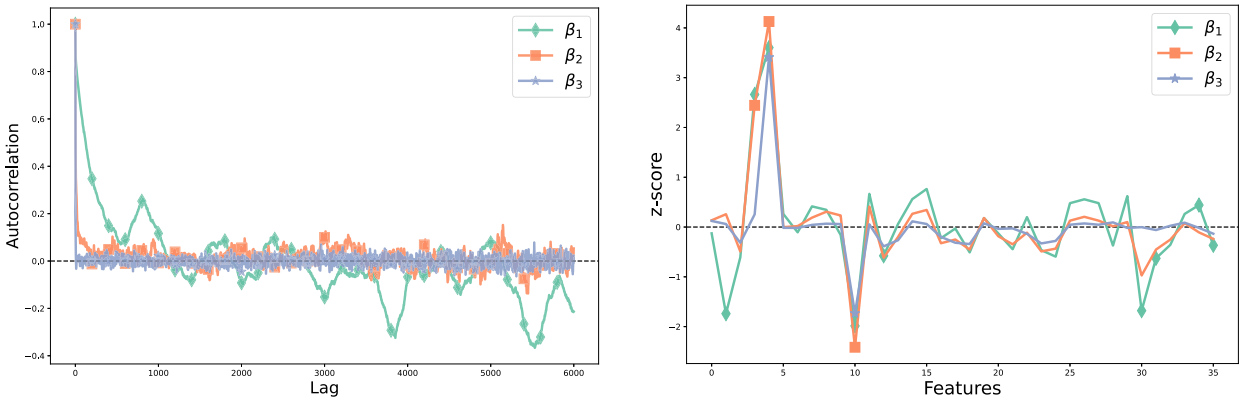
(f) The histogram of the posterior distributions of non-zero coefficients $\beta = 10$, with their means as vertical lines.

Fig. 10. The histogram of the posterior distributions of coefficient of all features (a), (c), and (e) and the non-zero coefficients (b), (d) and (f), using SULA with varying step size on the chemoinformatics data set.

UA BOF DocPRO4 project with ID 46929. The authors are also grateful to the associate editor and to the anonymous referees for their helpful comments and suggestions that improved the quality of the paper.

Appendix A

In this section, we provide the numerical results of using varying step sizes on both the synthetic and chemoinformatics data sets.



(a) Autocorrelation function for SULA for β_1, β_2 and β_3 on actual data.

(b) The z-score of posterior distributions on actual data.

Fig. 11. The effect of the scale of the sparsity parameter on the performance of SULA with varying step size on the chemoinformatics data set.

A.1. Varying and diminishing step sizes

During our experiments, we also considered a diminishing step size $\gamma_k = \frac{10}{(i+L_f/10)^{2.2}}$, for which numerical results are presented in Fig. 7. We show the results for $\{\beta_1, \beta_2, \beta_3\}$. The first row presents the results for the histogram of the posterior distributions of all parameters and the second row displays the histogram of the posterior distributions of zero parameters. The experiments show that compared to constant step size, the results are sparser even for $\beta = 1$.

Furthermore, in Fig. 8 we compared the z-score and standard deviation of SULA with constant and diminishing step size for each parameter.

References

[1] S. Ahn, B. Shahbaba, M. Welling, Distributed stochastic gradient mcmc, in: International Conference on Machine Learning, PMLR, 2014, pp. 1044–1052.
[2] M. Ahookhosh, Optimal subgradient methods: computational properties for large-scale linear inverse problems, Optim. Eng. 19 (2018) 815–844.
[3] M. Ahookhosh, A. Themelis, P. Patrinos, A Bregman forward-backward linesearch algorithm for nonconvex composite optimization: superlinear convergence to nonisolated local minima, SIAM J. Optim. 31 (2021) 653–685.
[4] H.H. Bauschke, P.L. Combettes, Convex Analysis and Monotone Operator Theory in Hilbert Spaces, CMS Books in Mathematics, Springer, 2017.
[5] A. Beck, First-Order Methods in Optimization, Society for Industrial and Applied Mathematics, Philadelphia, PA, 2017.
[6] A. Beck, M. Teboulle, Smoothing and first order methods: a unified framework, SIAM J. Optim. 22 (2012) 557–580.
[7] J. Besag, P. Green, D. Higdon, K. Mengersen, Bayesian computation and stochastic systems, Stat. Sci. (1995) 3–41.
[8] N. Brosse, A. Durmus, É. Moulines, M. Pereyra, Sampling from a log-concave distribution with compact support with proximal Langevin Monte Carlo, in: Conference on Learning Theory, PMLR, 2017, pp. 319–342.
[9] X. Cai, J.D. McEwen, M. Pereyra, Proximal nested sampling for high-dimensional Bayesian model selection, Stat. Comput. 32 (2022) 87.
[10] N. Chatterji, J. Diakonikolas, M.I. Jordan, P. Bartlett, Langevin Monte Carlo without smoothness, in: International Conference on Artificial Intelligence and Statistics, PMLR, 2020, pp. 1716–1726.
[11] C. Chen, N. Ding, L. Carin, On the convergence of stochastic gradient mcmc algorithms with high-order integrators, arXiv:1610.06665, 2016.
[12] P.L. Combettes, J.C. Pesquet, Proximal splitting methods in signal processing, in: Fixed-Point Algorithms for Inverse Problems in Science and Engineering, Springer New York, New York, NY, 2011, pp. 185–212.
[13] P.L. Combettes, V.R. Wajs, Signal recovery by proximal forward-backward splitting, Multiscale Model. Simul. 4 (2005) 1168–1200.
[14] A. Dalalyan, Further and stronger analogy between sampling and optimization: Langevin Monte Carlo and gradient descent, in: Conference on Learning Theory, PMLR, 2017, pp. 678–689.
[15] A.S. Dalalyan, A. Karagulyan, User-friendly guarantees for the Langevin Monte Carlo with inaccurate gradient, Stoch. Process. Appl. 129 (2019) 5278–5311.
[16] A.S. Dalalyan, L. Riou-Durand, A. Karagulyan, Bounding the error of discretized Langevin algorithms for non-strongly log-concave targets, arXiv preprint, arXiv:1906.08530, 2019.
[17] V. De Bortoli, A. Durmus, M. Pereyra, A.F. Vidal, Maximum likelihood estimation of regularization parameters in high-dimensional inverse problems: an empirical Bayesian approach. Part II: theoretical analysis, SIAM J. Imaging Sci. 13 (2020) 1990–2028.
[18] A. Durmus, S. Majewski, B. Miasojedow, Analysis of Langevin Monte Carlo via convex optimization, J. Mach. Learn. Res. 20 (2019) 2666–2711.
[19] A. Durmus, E. Moulines, High-dimensional Bayesian inference via the unadjusted Langevin algorithm, Bernoulli 25 (2019) 2854–2882.
[20] A. Durmus, E. Moulines, M. Pereyra, Efficient Bayesian computation by proximal Markov chain Monte Carlo: when Langevin meets Moreau, SIAM J. Imaging Sci. 11 (2018) 473–506.
[21] R. Dwivedi, Y. Chen, M.J. Wainwright, B. Yu, Log-concave sampling: Metropolis-Hastings algorithms are fast!, in: Conference on Learning Theory, PMLR, 2018, pp. 793–797.
[22] A. Eftekhari, L. Vargas, K.C. Zygalakis, The forward–backward envelope for sampling with the overdamped Langevin algorithm, Stat. Comput. 33 (2023) 85.
[23] D.L. Ermak, A computer simulation of charged particles in solution. I. Technique and equilibrium properties, J. Chem. Phys. 62 (1975) 4189–4196.
[24] S. Ghaderi, Mathematical Approaches to Biological Complexity in Systems Biomedicine, Ph.D. thesis, University of Luxembourg, Esch-sur-Alzette, Luxembourg, 2021.
[25] W.R. Gilks, P. Wild, Adaptive rejection sampling for Gibbs sampling, J. R. Stat. Soc., Ser. C, Appl. Stat. 41 (1992) 337–348.
[26] P. Giselsson, M. Fält, Envelope functions: unifications and further properties, J. Optim. Theory Appl. (2018), https://doi.org/10.1007/s10957-018-1328-z.
[27] D. Lamberton, G. Pages, Recursive computation of the invariant distribution of a diffusion, Bernoulli 8 (3) (2002) 367–405.
[28] Q. Li, N. Lin, et al., The Bayesian elastic net, Bayesian Anal. 5 (2010) 151–170.

[29] P.L. Lions, B. Mercier, Splitting algorithms for the sum of two nonlinear operators, SIAM J. Numer. Anal. 16 (1979) 964–979, https://doi.org/10.1137/0716071.
[30] T.D. Luu, J. Fadili, C. Chesneau, Sampling from non-smooth distributions through Langevin diffusion, Methodol. Comput. Appl. Probab. 23 (2021) 1173–1201.
[31] Y.A. Ma, T. Chen, E.B. Fox, A complete recipe for stochastic gradient mcmc, arXiv:1506.04696, 2015.
[32] J. Martin, L.C. Wilcox, C. Burstedde, O. Ghattas, A stochastic Newton mcmc method for large-scale statistical inverse problems with application to seismic inversion, SIAM J. Sci. Comput. 34 (2012) A1460–A1487.
[33] J.C. Mattingly, A.M. Stuart, D.J. Higham, Ergodicity for sdes and approximations: locally Lipschitz vector fields and degenerate noise, Stoch. Process. Appl. 101 (2002) 185–232.
[34] N. Metropolis, A.W. Rosenbluth, M.N. Rosenbluth, A.H. Teller, E. Teller, Equation of state calculations by fast computing machines, J. Chem. Phys. 21 (1953) 1087–1092.
[35] J.J. Moreau, Propriétés des applications «prox», C. R. Hebd. Séances Acad. Sci. 256 (1963) 1069–1071.
[36] J.J. Moreau, Proximité et dualité dans un espace hilbertien, Bull. Soc. Math. Fr. 93 (1965) 273–299.
[37] R.M. Neal, et al., Mcmc using Hamiltonian dynamics, in: Handbook of Markov Chain Monte Carlo, vol. 2, 2011, p. 2.
[38] Y. Nesterov, Smooth minimization of non-smooth functions, Math. Program. 103 (2005) 127–152.
[39] Y. Nesterov, V. Spokoiny, Random gradient-free minimization of convex functions, Found. Comput. Math. 17 (2017) 527–566.
[40] Y. Nesterov, et al., Lectures on Convex Optimization, vol. 137, Springer, 2018.
[41] G. Parisi, Correlation functions and computer simulations, Nucl. Phys. B 180 (1981) 378–384.
[42] P. Patrinos, A. Bemporad, Proximal Newton methods for convex composite optimization, in: 52nd IEEE Conference on Decision and Control, 2013, pp. 2358–2363.
[43] M. Pereyra, Proximal Markov chain Monte Carlo algorithms, Stat. Comput. 26 (2016) 745–760.
[44] M. Pereyra, L.V. Mieles, K.C. Zygalakis, Accelerating proximal Markov chain Monte Carlo by using an explicit stabilized method, SIAM J. Imaging Sci. 13 (2020) 905–935.
[45] M. Pereyra, L.A. Vargas-Mieles, K.C. Zygalakis, The split Gibbs sampler revisited: improvements to its algorithmic structure and augmented target distribution, arXiv preprint, arXiv:2206.13894, 2022.
[46] C.P. Robert, V. Elvira, N. Tawn, C. Wu, Accelerating Mcmc Algorithms, Wiley Interdisciplinary Reviews: Computational Statistics, vol. 10, 2018, p. e1435.
[47] G.O. Roberts, R.L. Tweedie, Exponential convergence of Langevin distributions and their discrete approximations, Bernoulli 2 (4) (1996) 341–363.
[48] R.T. Rockafellar, R.J.B. Wets, Variational Analysis, vol. 317, Springer Science & Business Media, 2011.
[49] A. Salim, D. Kovalev, P. Richtárik, Stochastic proximal Langevin algorithm: potential splitting and nonasymptotic rates, arXiv:1905.11768, 2019.
[50] L. Stella, A. Themelis, P. Patrinos, Forward-backward quasi-Newton methods for nonsmooth optimization problems, Comput. Optim. Appl. 67 (2017) 443–487, https://doi.org/10.1007/s10589-017-9912-y.
[51] D. Talay, L. Tubaro, Expansion of the global error for numerical schemes solving stochastic differential equations, Stoch. Anal. Appl. 8 (1990) 483–509.
[52] A. Themelis, M. Ahookhosh, P. Patrinos, On the acceleration of forward-backward splitting via an inexact Newton method, in: Splitting Algorithms, Modern Operator Theory, and Applications, Springer, 2019, pp. 363–412.
[53] A. Themelis, L. Stella, P. Patrinos, Forward-backward envelope for the sum of two nonconvex functions: further properties and nonmonotone linesearch algorithms, SIAM J. Optim. 28 (2018) 2274–2303.
[54] A.F. Vidal, V. De Bortoli, M. Pereyra, A. Durmus, Maximum likelihood estimation of regularization parameters in high-dimensional inverse problems: an empirical Bayesian approach part I: methodology and experiments, SIAM J. Imaging Sci. 13 (2020) 1945–1989.
[55] A. Vincze, G. Dargó, A. Rácz, G.T. Balogh, A corneal-pampa-based in silico model for predicting corneal permeability, J. Pharm. Biomed. Anal. 203 (2021) 114218.
[56] M. Welling, Y.W. Teh, Bayesian learning via stochastic gradient Langevin dynamics, in: Proceedings of the 28th International Conference on Machine Learning (ICML-11), Citeseer, 2011, pp. 681–688.
[57] K. Yosida, Functional Analysis, vol. 123, Springer, 1988.
[58] Y. Zhang, Ö.D. Akyildiz, T. Damoulas, S. Sabanis, Nonasymptotic estimates for stochastic gradient Langevin dynamics under local conditions in nonconvex optimization, arXiv preprint, arXiv:1910.02008, 2019.