Stochastic Gradient Monomial Gamma Sampler

Yizhe Zhang [1] **Changyou Chen** [1] **Zhe Gan** [1] **Ricardo Henao** [1] **Lawrence Carin** [1]

Abstract

Recent advances in stochastic gradient techniques have made it possible to estimate posterior distributions from large datasets via Markov Chain Monte Carlo (MCMC). However, when the target posterior is multimodal, mixing performance is often poor. This results in inadequate exploration of the posterior distribution. A framework is proposed to improve the sampling efficiency of stochastic gradient MCMC, based on Hamiltonian Monte Carlo. A generalized kinetic function is leveraged, delivering superior stationary mixing, especially for multimodal distributions. Techniques are also discussed to overcome the practical issues introduced by this generalization. It is shown that the proposed approach is better at exploring complex multimodal posterior distributions, as demonstrated on multiple applications and in comparison with other stochastic gradient MCMC methods.

1. Introduction

The development of increasingly sophisticated Bayesian models in modern machine learning has accentuated the need for efficient generation of asymptotically exact samples from complex posterior distributions. Markov Chain Monte Carlo (MCMC) is an important framework for drawing samples from a target density function. MCMC sampling typically aims to estimate a desired expectation in terms of a collection of samples, avoiding the need to compute intractable integrals. The Metropolis algorithm (Metropolis et al., 1953) was originally proposed to tackle this task. Despite great success, this method is based on *random walk* exploration, which often leads to inefficient posterior sampling (with a finite number of samples). Alternatively, exploration of a target distribution can be guided using *proposals* inspired by *Hamiltonian dynamics*, leading to Hamiltonian Monte Carlo (HMC) (Duane et al., 1987). Aided by gradient information, HMC is able to move efficiently in parameter space, thus greatly improving exploration. However, the emergence of big datasets poses a new challenge for HMC, as evaluation of gradients on whole datasets becomes computationally demanding, if not prohibitive, in many cases.

To scale HMC methods to big data, recent advances in Stochastic Gradient MCMC (SG-MCMC) have subsampled the dataset into *minibatches* in each iteration, to decrease computational burden (Welling & Teh, 2011; Chen et al., 2014; Ding et al., 2014; Ma et al., 2015). Stochastic Gradient Langevin Dynamics (SGLD) (Welling & Teh, 2011) was first proposed to generate approximate samples from a posterior distribution using minibatches. Since then, research has focused on leveraging the minibatch idea while also providing theoretical guarantees. For instance, Teh et al. (2014) showed that by appropriately injecting noise while using a stepsize-decay scheme, SGLD is able to converge asymptotically to the desired posterior. Stochastic Gradient Hamiltonian Monte Carlo (SGHMC) (Chen et al., 2014) extended SGLD with auxiliary momentum variables, akin to HMC, and introduced a *friction* term to counteract the stochastic noise due to subsampling. However, exact estimation of such noise is needed to guarantee a correct SGHMC sampler. To alleviate this issue, the Stochastic Gradient Nosé-Hoover Thermostat (SGNHT) (Ding et al., 2014) algorithm introduced so-called *thermostat* variables to adaptively estimate stochastic noise via a thermal-equilibrium condition.

One standing challenge of SG-MCMC methods is inefficiency when exploring complex multimodal distributions. This limitation is commonly found in latent variable models with a multi-layer structure. Inefficiency is manifested because sampling algorithms have difficulties moving across modes, while traveling along the surface of the distribution. As a result, it may take a very large number of iterations (posterior samples) to cover more than one mode, greatly limiting scalability.

We investigate strategies for improving mixing in SG-MCMC. We propose the Stochastic Gradient Monomial Gamma Thermostat (SGMGT), building upon the Monomial Gamma Sampler (MGS) proposed by Zhang et al.

[1]Duke University, Durham, NC, 27708. Correspondence to: Yizhe Zhang <yizhe.zhang@duke.edu>.

Proceedings of the 34^{th} International Conference on Machine Learning, Sydney, Australia, PMLR 70, 2017.

(2016). They showed that a generalized kinetic function typically improves the stationary mixing efficiency of HMC, especially when the target distribution has multiple modes. However, this advantage comes with numerical difficulties, and convergence problems due to poor initialization. By defining a *smooth* version of this generalized kinetic function, we can leverage its mixing efficiency, while satisfying the required conditions for stationarity of the corresponding stochastic process, as well as alleviating numerical difficulties arising from differentiability issues. To ameliorate the convergence issues, we further introduce i) a sampler with an underlying elliptic stochastic differential equation system and ii) a resampling scheme for auxiliary variables (momentum and thermostats) with theoretical guarantees. The result is an elegant framework to improve stationary mixing performance on existing SG-MCMC algorithms augmented with auxiliary variables.

2. Preliminaries

Hamiltonian Monte Carlo Suppose we are interested in sampling from a posterior distribution represented as $\pi(\theta|X) \propto p(X|\theta)p(\theta) = \exp[-U(\theta|X)]$, where θ denotes model parameters and $X = \{x_1, \dots, x_N\}$ represents N data points. Assuming i.i.d. data, the *potential energy function* $U(\theta|X)$ denotes the negative log posterior density, up to a normalizing constant, *i.e.*, $U(\theta|X) = -\sum_{i=1}^{N} \log p(x_i|\theta) - \log p(\theta)$. For simplicity, in the following we omit the conditioning of X in $U(\theta|X)$, and write $U(\theta)$. In HMC, the posterior density is augmented with an auxiliary momentum random variable p; p is independent of θ, and typically has a marginal Gaussian distribution with zero-mean and covariance matrix M. The joint distribution is written as $p(\theta, p) \propto \exp[-H(\theta, p)] \triangleq \exp[-U(\theta) - K(p)]$, where $H(\theta, p)$ is the total energy (or *Hamiltonian*) and $K(p) = \frac{1}{2} p^T M^{-1} p$ is the standard (Gaussian) *kinetic energy function*, and M is the mass matrix. HMC leverages Hamiltonian dynamics, driven by the following differential equations

$$d\theta = M^{-1} p dt\,, \qquad dp = -\nabla U(\theta) dt\,, \tag{1}$$

where t is the system's time index.

The total Hamiltonian is preserved under perfect simulation, *i.e.*, by solving (1) exactly. However, closed-form solutions for p and θ are often intractable, thus numerical integrators such as the *leap-frog* method are utilized to generate approximate samples of θ (Neal, 2011). This leads to the following update scheme:

$$\begin{aligned} p_{t+1/2} &= p_t - \tfrac{\epsilon}{2} \nabla U(\theta_t)\,, \\ \theta_{t+1} &= \theta_t + \epsilon M^{-1} p_{t+1/2}\,, \\ p_{t+1} &= p_{t+1/2} - \tfrac{\epsilon}{2} \nabla U(\theta_{t+1})\,, \end{aligned} \tag{2}$$

where ϵ is the *stepsize*.

Monomial Gamma HMC In the Monomial Gamma Hamiltonian Monte Carlo (MGHMC) (Zhang et al., 2016) algorithm, the following *generalized* kinetic function is employed as a substitute for the Gaussian kinetics of standard HMC:

$$K(p) = (|p|^{\frac{1}{2a}})^T M^{-1} |p|^{\frac{1}{2a}}\,, \tag{3}$$

where $|p|^{\frac{1}{2a}}$ denotes the element-wise power operation, a is the monomial parameter. Note that when $a = 1/2$, (3) recovers the standard (Gaussian) kinetics. For general a, the update equations are identical to (2), except for

$$\theta_{t+1} = \theta_t + \epsilon \nabla K(p_{t+1/2})\,. \tag{4}$$

Zhang et al. (2016) proved in the univariate case that MGHMC can yield better mixing performance when the sampler reaches its stationary distribution, under perfect dynamic simulation, *i.e.,* infinitesimal stepsize in the limit and adequate (finite) simulation stepsize. Additionally, it was shown that for multimodal distributions sampled via MGMHC, the probability of getting trapped in a single mode goes to zero, as $a \to \infty$.

However, these theoretical advantages are accompanied by two practical issues: i) the numerical difficulties accentuate dramatically as a increases, due to the lack of differentiability of $K(p)$ for $a \geq 1$, and ii) convergence is slow with poor initialization. For example, in (3) and (4), if θ_t is far away from the mode(s) of the distribution, $\nabla U(\theta_t)$ will be large, causing the updated momentum $p_{t+1/2}$ to blow up. This renders the change of θ, *i.e.,* $\nabla K(p_{t+1/2})$, to be arbitrarily small for large a, thus slowing convergence.

Stochastic Gradient MCMC SG-MCMC is desirable when the dataset, X, is too large to evaluate the potential $U(\theta)$ using all N samples. The idea behind SG-MCMC is to replace $U(\theta)$ with an unbiased *stochastic likelihood*, $\tilde{U}(\theta)$, evaluated from a subset of data (termed a minibatch)

$$\tilde{U}(\theta) = -\tfrac{N}{N'} \sum_{i=1}^{N'} \log p(x_{\tau_i}|\theta) - \log p(\theta)\,, \tag{5}$$

where $\{\tau_1, \cdots, \tau_{N'}\}$ is a random subset of $\{1, 2, \cdots, N\}$ of size $N' \ll N$. SG-MCMC algorithms are typically driven by a continuous-time Markov stochastic process of the form (Chen et al., 2015)

$$d\Gamma = V(\Gamma) dt + D(\Gamma) dW\,, \tag{6}$$

where Γ denotes the parameters of the *augmented* system, *e.g.*, p and θ, $V(\cdot)$ and $D(\cdot)$ are referred as *drift* and *diffusion* vectors, respectively, and W denotes a standard Wiener process.

In SGHMC (Chen et al., 2014), the resulting stochastic dynamic process is governed by the following Stochastic Dif-

ferential Equations (SDEs) (with $M = I$):

$$\begin{aligned} d\theta &= p dt\,, \\ dp &= -[\nabla \tilde{U}(\theta) + Ap]dt + \sqrt{2(AI - \hat{B}(\theta))}dW\,, \end{aligned} \tag{7}$$

where $\Gamma = \{\theta, p\}$, $V(\Gamma)$ is a function of $\{p, \nabla_\theta \tilde{U}, A\}$, and $D(\Gamma)$ is a function of $\{A, \hat{B}(\theta)\}$. $\nabla \tilde{U}(\theta)$ is modeled as $\nabla \tilde{U}(\theta) = \nabla U(\theta) + \sqrt{2B(\theta)}\nu$, where $\nu \sim \mathcal{N}(0, 1)$ and h is the discretization stepsize. $\hat{B}(\theta)$ is an estimator of $B(\theta)$, A is a user-specified *diffusion* factor and I is the identity matrix. Chen et al. (2014) set $\hat{B}(\theta) = 0$ for simplicity. The reasoning is that the injected noise $\mathcal{N}(0, 2Ah)$ will dominate as $h \to 0$ (A remains as a constant), whereas $B(\theta)$ goes to zero. Unfortunately, the covariance function, $B(\theta)$, of the stochastic noise, ν, is difficult to estimate in practice.

Recently, SGNHT (Ding et al., 2014) considered incorporating additional auxiliary variables (thermostats). The resulting SDEs correspond to

$$dp = -[\nabla \tilde{U}(\theta) + \xi \odot p]dt + \sqrt{2AI}dW\,, \tag{8}$$

$$d\theta = p dt\,,\ d\xi = (p \odot p - 1)dt\,, \tag{9}$$

where $\odot$ represents the Hadamard (element-wise) product and ξ are thermostat variables. Note that the diffusion factor, A, is decoupled in (8), thus ξ can adaptively fit to the unknown noise from the stochastic gradient $\nabla \tilde{U}(\theta)$.

3. Stochastic Gradient Monomial Gamma Sampler

We now consider i) a more efficient (generalized) kinetic function, ii) adapting the proposed kinetics to satisfy stationary requirements and alleviate numerical difficulties, iii) incorporating an additional first-order stochastic process to (8) and iv) stochastic resampling of the momentum and thermostats to lessen convergence issues.

Generalized kinetics The statistical physics literature traditionally considers a quadratic form of the kinetics function, and a Gaussian distribution for the thermostats in (8), when analyzing the dynamic system of a canonical ensemble (Tuckerman, 2010). Inspired by this, one typical assumption in previous SG-MCMC work is that the marginal distribution for the momentum and thermostat is Gaussian (Ding et al., 2014; Li et al., 2016). However, this assumption, while convenient, does not necessarily guarantee an optimal sampler.

In recent work, Lu et al. (2016) extended the standard (Newtonian) kinetics to a more general form inspired by relativity theory. By bounding the momentum, their *relativistic Monte Carlo* can lessen the problem associated with large potential gradients, $\nabla U(\theta_t)$, thus resulting in a more robust alternative to standard HMC. Further, Zhang et al. (2016) demonstrated that adopting non-Gaussian kinetics delivers better mixing and reduces sampling autocorrelation, especially for cases where the posterior distribution has multiple modes.

These ideas motivate a more general framework to characterize SG-MCMC, with potentially non-Gaussian kinetics and thermostats. As a relaxation of SGNHT (Ding et al., 2014; Ma et al., 2015), we consider a Hamiltonian system defined in a more general form

$$H = K(p) + U(\theta) + F(\xi)\,, \tag{10}$$

where $K(\cdot)$ and $F(\cdot)$ are any valid potential functions, inherently implying that $\exp[-K(\cdot)]$ and $\exp[-F(\cdot)]$, define valid probability density functions.

We first consider the SDEs of SGNHT with generalized kinetics $K(p)$. The system can be obtained by generalizing $K(p) = p^T p/2$ (with identity mass matrix M for simplicity) in (8) with arbitrary $K(p)$, thus

$$\begin{aligned} d\theta &= \nabla K(p) dt\,, \\ dp &= -[\nabla \tilde{U}(\theta) + \xi \odot \nabla K(p)]dt + \sqrt{2AI}dW\,, \\ d\xi &= (\nabla K(p) \odot \nabla K(p) - \nabla^2 K(p))dt\,. \end{aligned} \tag{11}$$

However, if we set $K(p)$ as in (3) with $a \geq 1$, the dynamics governing the SDEs in (11) will often fail to converge. This is because the sufficient condition to guarantee that the Itô process governed by the SDEs in (11) converge to a stationary distribution generally requires the Fokker-Planck equation to hold (Risken, 1984). Further, the existence and uniqueness of the solutions to the Fokker-Planck equation require Lipschitz continuity of drift and diffusion vectors in (6) (Bris & Lions, 2008). Unfortunately, this is not the case for the drift vectors in (11) when $a \geq 1$, as $\nabla K(p)$ is non-differentiable at the origin, *i.e.*, $p = 0$.

Softened kinetics The above limitation can be avoided by using a *softened* kinetic function $K_c(p)$. However, to keep the performance benefits from the original *stiff* kinetics, we must ensure that $K_c(p)$ has the same tail behavior. We propose that for $a = \{1, 2\}$, the softened kinetics are (for clarity we consider 1D case, however higher dimensions still apply)

$$K_c(p) = \begin{cases} -p + 2/c \log(1 + e^{cp}), & a = 1 \\ |p|^{1/2} + \frac{4}{c(1+e^{c|p|^{1/2}})}, & a = 2 \end{cases}\,, \tag{12}$$

where $c > 0$ is a *softening* parameter. Note that $K_c(p)$ is (infinitely) differentiable for any c and asymptotically approaches the stiff kinetics as $c \to \infty$. A comparison between stiff kinetics, $K(p)$, and softened kinetics $K_c(p)$ is shown in Figure 1, for different values of c. Discussion and formulation of the softened kinetics for arbitrary a (and M) are provided in the Supplementary Material (SM).

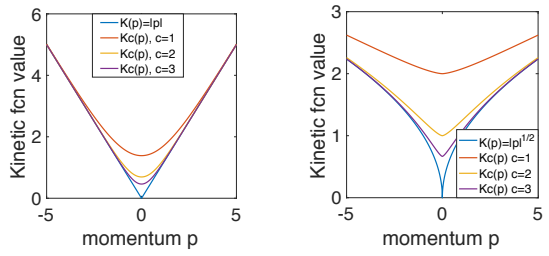

Figure 1. Softened *vs.* stiff kinetics (1D). Left: $a = 1$. Right: $a = 2$.

To generate samples of the momentum variable, p, from the density with softened kinetics, which is proportional to $\exp[-K_c(p)]$, we use a coordinate-wise rejection sampling, *i.e.*, the proposed p_d for the d-th dimension is rejected with probability $1 - \exp[K(p_d) - K_c(p_d)]$.

In practice, setting c to a relatively large value would still make the gradient $\nabla K_c(p)$ ill-posed close to $p = 0$, thus causing high *integration error* when simulating the Hamiltonian dynamics. Conversely, setting c to a small value will cause a *high approximation* error w.r.t. the original $K(p)$, thus resulting in a less efficient sampler. Consequently, c has to be determined empirically as a trade-off between integration and approximation errors.

Additional First Order Dynamics Inspired by Ma et al. (2015), we consider adding Brownian motion to θ and ξ in (8), with variances σ_θ and σ_ξ, respectively, while maintaining the stochastic process (asymptotically) converging to the correct marginal distribution of θ. Specifically, we consider the following SDEs:

$$\begin{aligned} d\theta =& -\sigma_\theta \nabla \tilde{U}(\theta)dt + \nabla K_c(p)dt + \sqrt{2\sigma_\theta}dW\,, \\ dp =& -(\sigma_p + \gamma \nabla F(\xi)) \odot \nabla K_c(p)dt \\ & - \nabla \tilde{U}(\theta)dt + \sqrt{2\sigma_p}dW\,, \\ d\xi =& \gamma \left[\nabla K_c(p) \odot \nabla K_c(p) - \nabla^2 K_c(p)\right] dt \\ & - \sigma_\xi \nabla F(\xi)dt + \sqrt{2\sigma_\xi}dW\,. \end{aligned} \tag{13}$$

The variances $\{\sigma_\theta, \sigma_p, \sigma_\xi\}$ control the Brownian motion for $\{\theta, p, \xi\}$, respectively, and $\gamma > 0$ denotes a rescaling factor for the friction term of momentum updates. The additional terms $-\sigma_\theta \nabla \tilde{U}(\theta)dt + \sqrt{2\sigma_\theta}dW$ and $-\sigma_\xi \nabla F(\xi)dt + \sqrt{2\sigma_\xi}dW$ can be understood as first-order Langevin dynamics (Welling & Teh, 2011). The variance term, σ_θ, controls the contribution of $\nabla \tilde{U}(\theta)$ to the update of θ w.r.t. $\nabla K_c(p)$. This is analogous to the hyperparameter balancing $\nabla \tilde{U}(\theta)$ and p in the SGD-with-momentum algorithm (Rumelhart et al., 1988). Derivation details for $\nabla K_c(p)$ and $\nabla^2 K_c(p)$ in (12), as well as other values of a, are provided in the SM.

The following theorem, proven in the SM, shows that under regularity conditions, the SDEs in (13) lead to posterior samples from the invariant joint distribution $p(\Gamma) \propto \exp[-H(\Gamma)]$, yielding the desired marginal distribution w.r.t. θ as $p(\theta) \propto \exp[-U(\theta)]$.

Theorem 1. *The stochastic process governed by* (13) *converges to a stationary distribution* $p(\Gamma) \propto \exp[-H(\Gamma)]$, *where* $H(\Gamma)$ *is as defined in* (10), *and* $\Gamma = \{\theta, p, \xi\}$.

The reasoning behind increasing *stochasticity* in the SDEs is two-fold. First, the additional Langevin dynamics are crucial to SG-MCMC with generalized kinetics for large a. For instance, for $\sigma_\theta = 0$, the update for θ from (11) is $\theta_{t+1} = \theta_t + \nabla K(p_t)h$. When $a > 1$ and $|p_t|$ is large, $\nabla K(p) = \frac{1}{a}|p|^{1/a-1}$ will be close to zero, thus θ_{t+1} (the next sample) will be close to θ_t, *i.e.*, the sampler moves arbitrarily slow. As discussed by Zhang et al. (2016), this can happen when θ moves to a region where the gradient $\nabla U(\theta)$ takes a large absolute value, *e.g.*, near the low-density regions in a light-tailed distribution. Fortunately, the additional Langevin dynamics in (13), $-\sigma_\theta \nabla \tilde{U}(\theta)dt + \sqrt{2\sigma_\theta}dW$, compensate for the weak updating signal from $\nabla K(p)$, by an immediate gradient signal $\nabla \tilde{U}(\theta)$. Additionally, when $\tilde{U}(\theta)$ becomes small, $\nabla K(p)$ will become large. As a result, these two updating signals $\nabla K(p)$ and $\nabla \tilde{U}(\theta)$ compensate each other, thereby delivering a stable updating scheme. Likewise, the immediate gradient $\nabla F(\xi)$ in (13) can provide complementary updating signal for the thermostat variables, ξ, to offset the weak deterministic update $\nabla K_c(p) \odot \nabla K_c(p) - \nabla^2 K_c(p)$, when p is large.

Second, (13) has noise components on all parameters, $\{\theta, p, \xi\}$, making the corresponding SDEs *elliptic*. From a theoretical perspective, ellipticity/hypoellipticity are necessary conditions to guarantee existence of bounded solutions for a particular partial differential equation related to the diffusion's *infinitesimal generator*, which lies in the core of most recent SG-MCMC theory (Teh et al., 2014; Vollmer et al., 2016; Chen et al., 2015). Ellipticity is characterized by a noise process (Brownian motion) covering all components of the system, via the diffusion, $D(\Gamma)$, in (6). This means $D(\Gamma)$ is block diagonal, thus a positive definite matrix (Mattingly et al., 2010). In a typical hypoelliptic case, the noise process is imposed on a subset of Γ. However, hypoellipticity also requires the noise to be able to spread through the system via the drift term, $V(\Gamma)$, which may not be true for general $V(\Gamma)$. For instance, in (8), $\Gamma = \{\theta, p, \xi\}$ and $D(\Gamma)$ is block diagonal with entries $\{0, \sqrt{2AI}, 0\}$, *i.e.*, θ and ξ are not explicitly influenced by the noise process, W, thus hypoellipticity cannot be guaranteed.

To the authors' knowledge, for existing SG-MCMC algorithms, only SGLD where $d\theta = -\nabla_\theta \tilde{U}(\theta)dt + \sqrt{2}dW$, satisfies the ellipticity property, while other algorithms such as SGHMC and SGNHT assume hypoellipticity, thus their corresponding $D(\Gamma)$ are not positive definite.

One caveat of (13) is that if σ_θ and σ_ξ are too large, the up-

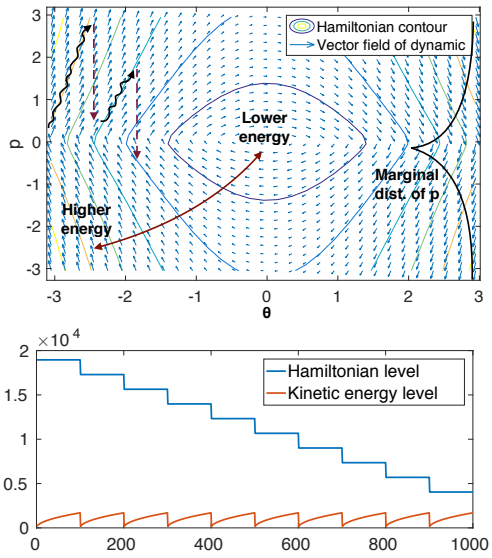

Figure 2. Momentum resampling. Top: stochastic process with resampling helps sampler move quickly to a lower Hamiltonian contour. Bottom: resampling decreases energy step-wise during burn-in stage. Resampling of p occurs every 100 iterations.

dates will be dominated by first-order dynamics, thus losing the convergence benefits from second-order dynamics (Chen et al., 2014). In practice, σ_θ and σ_ξ are problem-specific, thus need to be tuned, *e.g.* , by cross-validation.

Stochastic resampling When generating samples from the stochastic process in (13), we resample momentum and thermostats from their marginal distribution with a fixed frequency, instead of every iteration from their conditionals. Since the momentum and thermostats are drawn from the independent marginals of stationary distribution $p(\Gamma) \propto \exp[-H(\Gamma)]$, it can be shown that reconstructing the stochastic process with the solution of the SDEs will still leave the stochastic process invariant to the target stationary distribution (Brunick et al., 2013).

To simplify the discussion, consider a stochastic process of a particle $\{\theta, p\}$ as in (11) with fixed ξ. As show in Figure 2, suppose the initial value of θ is far from the maximum *a posteriori* value. The dynamics governed by (11) will stochastically move along the Hamiltonian contour. The total Hamiltonian energy level is affected by the joint effect of the stochastic diffusion and momentum refraction (*i.e.*, -ξpdt), which changes continuously over time.

From previous discussions, moving on a high Hamiltonian contour when $a > 1$ is less efficient because the absolute value of the momentum, $|p|$, will get increasingly large, slowing down the movement of θ. Resampling of momentum according to its marginal will enable the sampler to immediately move to a lower Hamiltonian energy level.

At the burn-in stage, this *momentum-accumulation/energy-drop* cycle seen in Figure 2(bottom) via resampling momentum can happen several times, until equilibrium is found. In practice, the resulting energy level is often much lower than initially, thereby delivering a more efficient and accurate dynamic updating.

The frequency of resampling from the marginal of the stationary distribution can have a direct impact on the mixing performance. Setting the frequency too high will result in a random-walk behavior. Conversely, with a low frequency resampling, the random-walk behavior is suppressed at a cost of fewer jumps between trajectories associated with different energy levels. It is advisable to increase the resampling frequency if the sampler is initialized on low-density (*e.g.* light-tailed) region.

The resampling step on p and ξ plays a role that is similar to adding a Langevin component to θ, in the sense that both improve convergence for $a > 1$. However, these two strategies (resampling and Langevin) are fundamentally different. We empirically observe that resampling is most helpful during burn-in, while the additional Langevin-style updates are more helpful with mixing during stationary sampling.

SGMGT The specifications described above constitute an SG-MCMC method for the SDEs in (11), which we call Stochastic Gradient Monomial Gamma Thermostat (SGMGT). We denote the SG-MCMC method with additional Brownian motion on θ and ξ in (13) as SGMGT-D (Diagonal), *i.e.*, with $\sigma_\theta > 0$ and $\sigma_\xi > 0$. The complete update scheme, with Euler integrator, for SGMGT is presented in the SM. Note that with $a = 1/2, \sigma_\theta = 0, \sigma_\xi \to 0, c \to \infty$, SGMGT-D recovers SGHMC as in Chen et al. (2014). Moreover, when $a = 1/2, \sigma_\theta = 0, c \to \infty$, it becomes SGNHT as in Ding et al. (2014).

We note that SGMGT-D improves upon SGNHT in three respects: (i) we introduce generalized kinetics, which provably yield lower autocorrelations than standard HMC, especially in multimodal cases; (ii) the additional stochastic noise on thermostat variables yields more efficient mixing; (iii) we use stochastic resampling to allow for faster interchange between different energy levels, thus alleviating sampling *stickiness*.

To the authors' knowledge, despite existing analysis for Langevin Monte Carlo (Bubeck et al., 2015; Dalalyan, 2016), rigorous analysis and comparison of the mixing performance of general SG-MCMC is very difficult, thus not yet established. Toward understanding the mixing performance of SGMGT-D, we argue that as the minibatch size increases, and the contribution of the diffusion in (6) decreases, the SGMGT-D will approach MGHMC, in which case, a large a will result in high stationary mixing performance, especially when sampling multimodal distribution,

as theoretically shown by Zhang et al. (2016). Although our experiments support our intuition, a more formal theoretical justification is needed. We leave this as interesting future work.

We observe empirically that when increasing the value of a, SGMGT-D may not always achieve superior mixing performance. One possible reason for this is a larger value of a induces "stiffer" behavior of $\exp[-K(p)]$ at $p = 0$, which typically requires a higher level of softening, thus higher rejection rates during the rejection sampling step. Also, when the dimensionality of p is higher, the rejection rate of the rejection sampling step increases (proportional to p). In such cases, the efficiency of the sampler decreases with large a. For these reasons, we limit our experiments to $a = \{1, 2\}$.

We clearly have more hyperparameters than SGNHT. In practice, we fix $M = I$, $a = \{1, 2\}$, and set the resampling frequency $T_p = T_\xi = 100$, which provides robust performance. Thus, only two additional hyperparameters are employed (σ_θ and σ_ξ) compared to SGNHT, and these parameters require further tuning. We use either validation or a hold-out set in our experiments.

More accurate numerical integrators Using a first-order Euler integrator to approximate the solution of the continuous-time SDEs in (13), leads to $O(h)$ errors in the approximate samples (Chen et al., 2016). Alternatively, we can use the symmetric splitting scheme of Chen et al. (2016) to reduce the order of the approximate error to $O(h^2)$. Details of the splitting used in this work are provided in the SM.

Convergence properties The SGMGT framework, as an instance of SG-MCMC, enjoys the same convergence properties of general SG-MCMC algorithms studied in Chen et al. (2015). It's worth to mention that on challenging problems the posterior may not be densely sampled to yield ideal posterior computation, and the asymptotic theory is being used as a useful heuristic. Specifically, it is of interest to quantify how fast the sample average, $\hat{\phi}_T$, converges to the true posterior average, $\bar{\phi} \triangleq \int \phi(\theta)\pi(\theta|X)\mathrm{d}\theta$, for $\hat{\phi}_T \triangleq \frac{1}{T}\sum_{t=1}^{T}\phi(\theta_t)$, where T is number of iterations. Here we make the same assumptions of Chen et al. (2015), and further assume that a first-order Euler integrator and a fixed stepsize are used.

Proposition 2. *For the proposed SGMGT and SGMGT-D algorithms, if a fixed stepsize h is used, we have:*

$$\textit{Bias: } \left|\mathbb{E}\hat{\phi}_T - \bar{\phi}\right| = O\left(1/(Th) + h\right) ,$$

$$\textit{MSE: } \mathbb{E}\left(\hat{\phi} - \bar{\phi}\right)^2 = O\left(1/(Th) + h^2\right) .$$

This proposition indicates that with larger number of iterations and smaller step sizes, smaller bias and MSE bounds can be achieved. We note that these bounds have similar rates compared to other SG-MCMC algorithms such as SGLD, however, as we demonstrate below in the experiments, SGMGT and SGMGT-D usually converge faster than existing SG-MCMC methods.

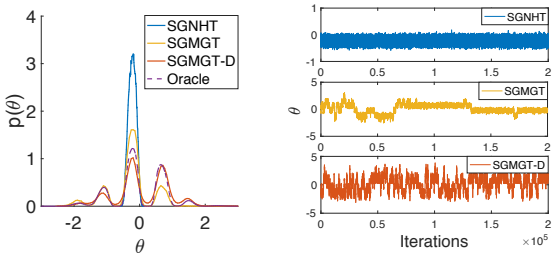

Figure 3. Synthetic multimodal distribution. Left: empirical distributions for different methods. Right: traceplot for each method.

In addition, for stochastic resampling, we can extend Proposition 2 to the following complementary results:

Lemma 1. *Let π_h be the stationary distribution of SGMGT-D. The stationary distribution of SGMGT-D with momentum resampling is the same as π_h.*

Lemma 2. *The optimal finite-time bias and MSE bounds for SGMGT-D with momentum replacement remain the same as SGMGT-D.*

Proofs of Lemma 1 and Lemma 2 are provided in the SM. The proposed SGMGT framework has a strong connection with second-order stochastic optimization methods, leading to a sampling scheme with minibatches with similar mixing performance as slice sampling (Neal, 2003). We discuss the details of this in the SM.

4. Experiments

4.1. Multiple-well Synthetic Potential

We first evaluate the mixing efficiency of SGMGT and SGMGT-D for a synthetic problem, to generate samples from a complex multimodal distribution. The distribution is shown in Figure 3(left). See SM for the definition of its potential energy. The modes are almost isolated with a low-density region connecting each other. Consequently, traversing between modes is difficult. In order to simulate the noise of the gradient estimates, we set $\nabla\tilde{U}(\theta) = \nabla U(\theta) + \mathcal{N}(0, 2B)$, similar to Ding et al. (2014), where $B = 1$.

We compare SGNHT with SGMGT and SGMGT-D with monomial parameter $a = 2$ and fix $\gamma = 1$. For all three algorithms, we try a number of hyperparameter settings, *e.g.*, stepsize h, $\{\sigma_\theta, \sigma_p, \sigma_\xi\}$, and the soft parameter c, and present the best results in Figure 3. Standard SGNHT fails to escape from one of the modes of the distribution. For

Table 1. Average AUROC and median ESS. Dataset dimensionality is indicated in parenthesis after the name of each dataset.

AUROC (D)	A (15)	G (25)	H (14)	P(8)	R (7)	C (87)
SGNHT	0.89	0.75	0.90	0.86	0.95	0.65
SGMGT(a=1)	0.92	0.78	0.91	0.86	0.87	0.70
SGMGT-D(a=1)	0.95	0.86	0.95	**0.93**	**0.98**	**0.73**
SGMGT(a=2)	0.93	0.79	0.93	0.88	0.86	0.62
SGMGT-D(a=2)	**0.95**	**0.90**	**0.95**	0.90	0.97	0.69
ESS (D)	A (15)	G (25)	H (14)	P(8)	R (7)	C (87)
SGNHT	869	941	1911	2077	1761	1873
SGMGT-D(a=1)	**3147**	**2131**	2448	**4244**	1494	**3605**
SGMGT-D(a=2)	2700	1989	**2768**	3430	**2265**	2969

SGMGT with $a = 2$, the generated samples reached 3 modes. For SGMGT-D with $a = 2$, the sampler identified all 5 modes. In Figure 3(right), SGMGT-D adequately moves across different modes and yields rapid mixing performance, unlike SGMGT which exhibits stickier behavior.

4.2. Bayesian Logistic Regression

We evaluated the mixing efficiency and accuracy of SGMGT and SGMGT-D using Bayesian logistic regression (BLR) on 6 real-world datasets from the UCI repository (Bache & Lichman, 2013): German credit (G), Australian credit (A), Pima Indian (P), Heart (H), Ripley (R) and Caravan (C). The data dimensionality ranges from 7 to 87, and total observations vary between 250 to 5822. Gaussian priors are imposed on the regression coefficients. We set the minibatch size to 16. Other hyperparameters are provided in the SM. For each experiment, we draw 5000 iterations with 1000 burn-in samples.

Results in terms of median Effective Sample Size (ESS) and prediction accuracies as Area Under Receiver Operating Characteristic (AUROC) are summarized in Table 1. All the results are averages over 5 independent runs with random initialization. In general, SGMGT-D performs better than SGMGT. For higher-dimensional dataset Cavaran, the performance of $a = 2$ decreases significantly, indicating numerical difficulties. The performance gap between SGMGT and SGMGT-D with $a = 1$ or $a = 2$ is usually larger than the gap between SGNHT ($a = 0.5$). Presumably when a is greater than 1, SGMGT-D has better convergence.

4.3. Latent Dirichlet Allocation

We also test our methods on Latent Dirichlet Allocation (LDA) (Blei et al., 2003). Details of LDA and our implementation are provided in the SM. We use the ICML dataset (Chen et al., 2013), which contains 765 documents corresponding to abstracts of ICML proceedings from 2007 to 2011. After stopword removal, we obtain a vocabulary size of 1918 and about 44K words. We use 80% of the documents for training and the remaining 20% for testing. The number of topics is set to 30, resulting in 57,540 parameters. We use a symmetric Dirichlet prior with concentration $\beta = 0.1$. All experiments are based on 5000 MCMC samples with 1000 burn-in rounds. We set the minibatch size to 16. Other hyperparameter settings are provided in the SM.

Table 2. The test perplexity with varying stepsize.

stepsize	0.04	0.05	0.06	0.07	0.08	0.09	0.1
SGLD	1058	1054	1058	1067	**1037**	1048	1057
SGNHT	1104	1144	1039	**1024**	1043	1067	1107
SGMGT(a=1)	996	988	990	**986**	996	998	997
SGMGT-D(a=1)	987	**983**	996	996	992	1013	1029
SGMGT(a=2)	1024	1029	1030	**1013**	1030	1022	1043
SGMGT-D(a=2)	968	994	973	957	961	**954**	970

Table 2 shows the test perplexities for SGMGT and SGMGT-D for different stepsizes. For each method we highlight the best perplexity. The SGMGT-D with $a = 2$ outperforms other methods, however SGMGT with $a = 2$ fails to achieve a comparable result with SGMGT with $a = 1$, probably because a good initialization is hard to achieve for a high-dimensional distribution.

4.4. Discriminative RBM

We applied our SGMGT to the Discriminative Restricted Boltzmann Machine (DRBM) (Larochelle & Bengio, 2008) on MNIST data. We choose DRBM instead of RBM because it provides explicit stochastic gradient formulas.

We evaluated our methods empirically and compare them with SGNHT. We use one hidden layer with 500 units. For each method we performed 1500 iterations with 200 burn-in samples. The minibatch size is set to 100. Details of the hyperparameter settings for SGMGT and SGMGT-D are provided in the SM. As shown in Figure 4(right-most 3 panels), we observe that SGMGT-D with $a = 2$ yields a superior mixing performance. For SGMGT-D with $a = 2$, the posterior samples demonstrated both rapid local mixing, and long-range movement. In contrast, SGLD seems trapped into a local mode after around 500 iterations.

Figure 4(left) shows that SGMGT-D with $a = 2$ delivers the fastest convergence with the highest test accuracy, 0.976. The SGMGT-D improves over SGMGT, while performance of SGMGT-D seems to increase with a large value of a. We observed that the stochastic resampling played a crucial role for SGMGT, as removing the resampling step resulted in a large drop in testing performance and mixing efficiency.

4.5. Recurrent Neural Network

We test our framework on Recurrent Neural Networks (RNNs) for sequence modeling (Gan et al., 2017). We consider two tasks: (*i*) polyphonic music prediction; and (*ii*) word-level language modeling, detailed below. Additional details of the experiment are provided in the SM.

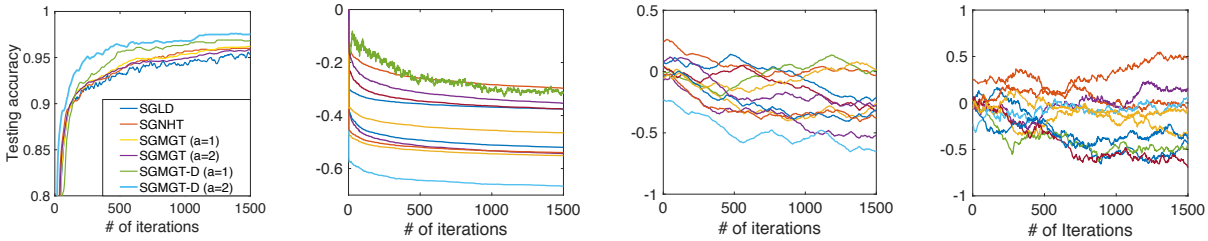

Figure 4. Experimental results for DRBM. Left: testing accuracies for SGLD, SGNHT, SGMGT and SGMGT-D. Middle-left through right: traceplots for SGLD, SGNHT and SGMGT-D with $a = 2$, respectively.

Polyphonic music prediction We use four datasets: Piano-midi.de (Piano), Nottingham (Nott), MuseData (Muse) and JSB chorales (JSB) (Boulanger-Lewandowski et al., 2012). Each of these are represented as a collection of 88-dimensional binary sequences, that span the whole range of piano from A0 to C8.

We use a one-layer LSTM (Hochreiter & Schmidhuber, 1997) model, and set the number of hidden units to 200. The total number of parameters is around 200K. Each model is trained for 100 epochs. We perform early stopping, while selecting the stepsize and other hyperparameters by monitoring the performance on validation sets. Updates are performed using minibatches from one sequence.

Language modeling The Penn Treebank (PTB) corpus (Marcus et al., 1993) is used for word-level language modeling. We adopt the standard split (929K training words, 73K validation words, and 82K test words). The vocabulary size is 10K. We train a two-layer LSTM model on this dataset. The total number of parameters is approximately 6M. Each LSTM layer contains 200 units.

Table 3. Test negative log-likelihood results on polyphonic music datasets and test perplexities on PTB using RNN.

Algorithms	Piano	Nott	Muse	JSB	PTB
SGLD	11.37	6.07	10.83	11.25	127.47
SGNHT	9.00	4.24	7.85	9.27	131.3
SGMGT (a=1)	7.90	4.35	8.42	8.67	120.6
SGMGT (a=2)	10.17	4.64	8.51	8.84	250.5
SGMGT-D (a=1)	**7.51**	**3.33**	7.11	8.46	113.8
SGMGT-D (a=2)	7.53	3.35	**7.09**	**8.43**	**109.0**
SGD	11.13	5.26	10.08	10.81	120.44
RMSprop	7.70	3.48	7.22	8.52	120.45

Results are shown in Table 3. The best log-likelihood results on the test set are achieved by using SGMGT-D with either $a = 1$ or $a = 2$ (depending on the dataset). To compare with optimization-based methods, we also include results for SGD (Bottou, 2010) and RMSprop (Tieleman & Hinton, 2012). A more comprehensive comparison is provided in the SM.

Learning curves for Nott and PTB datasets are shown in Figure 5. We omit the SGLD results since they are not comparable with other methods. For both datasets, we observe that SGMGT-D delivers fastest convergence. The best negative log-likelihood is achieved by SGMGT-D $a = 1$. The difference between $a = 1$ and $a = 2$ is small, though SGMGT-D with $a = 2$ seems to decrease slightly faster after 20 epochs for PTB data.

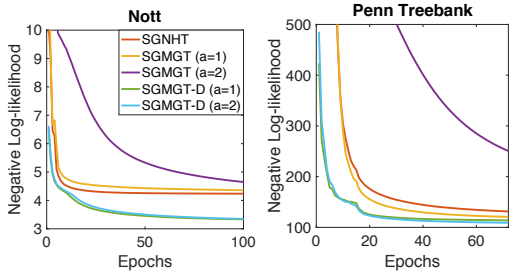

Figure 5. Learning curves of different SG-MCMC methods on sequence modeling using RNN. Left: Nott. Right: Penn Treebank.

We also observe that the SGMGT with $a = 2$ seems suboptimal compared with SGMGT with $a = 1$ and SGNHT. We hypothesize that numerical difficulties hinder the success of SGMGT with $a = 2$, especially in higher-dimensional cases, and without the additional Langevin components of SGMGT-D.

5. Conclusions

We improve upon existing SG-MCMC methods with several generalizations. We employed a more-general kinetic function, which we have shown to have better mixing efficiency, especially for multimodal distributions. Since practical use of the generalized kinetics is limited by convergence issues during burn-in, we injected additional Langevin dynamics and incorporated a stochastic resampling step to obtain generalized SDEs that alleviate the convergence issues. Possible areas of future research include designing an algorithm in a slice-sampling fashion, which maintains the invariant distribution by leveraging the connections between HMC and slice sampling (Zhang et al., 2016). In addition, it is desirable to design an algorithm that can adaptively choose the monomial parameter a, thereby achieving better mixing while automatically avoiding numerical difficulties.

Acknowledgments

This research was supported by ARO, DARPA, DOE, NGA, ONR and NSF.

References

Bache, Kevin and Lichman, Moshe. UCI machine learning repository, 2013.

Betancourt, MJ. The fundamental incompatibility of Hamiltonian Monte Carlo and data subsampling. *ArXiv*, 2015.

Blei, David M., Ng, Andrew Y., and Jordan, Michael I. Latent Dirichlet allocation. *JMLR*, 3, 2003.

Bottou, Léon. Large-scale machine learning with stochastic gradient descent. In *COMPSTAT*, 2010.

Boulanger-Lewandowski, Nicolas, Bengio, Yoshua, and Vincent, Pascal. Modeling temporal dependencies in high-dimensional sequences: Application to polyphonic music generation and transcription. In *ICML*, 2012.

Bris, C Le and Lions, P-L. Existence and uniqueness of solutions to fokker–planck type equations with irregular coefficients. *Communications in Partial Differential Equations*, 33(7):1272–1317, 2008.

Brunick, Gerard, Shreve, Steven, et al. Mimicking an itô process by a solution of a stochastic differential equation. *The Annals of Applied Probability*, 23(4):1584–1628, 2013.

Bubeck, Sebastien, Eldan, Ronen, and Lehec, Joseph. Finite-time analysis of projected langevin monte carlo. In *NIPS*, 2015.

Chen, Changyou, Rao, Vinayak, Buntine, Wray, and Whye Teh, Yee. Dependent normalized random measures. In *ICML*, 2013.

Chen, Changyou, Ding, Nan, and Carin, Lawrence. On the convergence of stochastic gradient mcmc algorithms with high-order integrators. In *NIPS*, pp. 2278–2286, 2015.

Chen, Changyou, Carlson, David, Gan, Zhe, Li, Chunyuan, and Carin, Lawrence. Bridging the gap between stochastic gradient mcmc and stochastic optimization. In *AISTATS*, 2016.

Chen, Tianqi, Fox, Emily B, and Guestrin, Carlos. Stochastic gradient hamiltonian monte carlo. *ArXiv*, 2014.

Dalalyan, Arnak S. Theoretical guarantees for approximate sampling from smooth and log-concave densities. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 2016.

Ding, Nan, Fang, Y, Babbush, R, Chen, C, Skeel, R. D, and Neven, H. Bayesian sampling using stochastic gradient thermostats. In *NIPS*, 2014.

Duane, Simon, Kennedy, Anthony D, Pendleton, Brian J, and Roweth, Duncan. Hybrid Monte Carlo. *Physics letters B*, 195(2), 1987.

DuBois, Christopher, Balan, Anoop Korattikara, Welling, Max, and Smyth, Padhraic. Approximate slice sampling for bayesian posterior inference. In *AISTATS*, pp. 185–193, 2014.

Gan, Zhe, Li, Chunyuan, Chen, Changyou, Pu, Yunchen, Su, Qinliang, and Carin, Lawrence. Scalable bayesian learning of recurrent neural networks for language modeling. In *ACL*, 2017.

Geyer, C. J. Markov chain monte carlo lecture notes, 2005.

Hochreiter, Sepp and Schmidhuber, Jürgen. Long short-term memory. *Neural computation*, 1997.

Hwang, Chii-Ruey, Hwang-Ma, Shu-Yin, Sheu, Shuenn-Jyi, et al. Accelerating diffusions. *The Annals of Applied Probability*, 2005.

Larochelle, H. and Bengio, Y. Classification using discriminative restricted boltzmann machines. In *ICML*, 2008.

Li, Chunyuan, Chen, Changyou, Fan, Kai, and Carin, Lawrence. High-order stochastic gradient thermostats for bayesian learning of deep models. In *AAAI*, 2016.

Lu, Xiaoyu, Perrone, Valerio, Hasenclever, Leonard, Teh, Yee Whye, and Vollmer, Sebastian J. Relativistic monte carlo. *arXiv*, 2016.

Ma, Yi-An, Chen, Tianqi, and Fox, Emily. A complete recipe for stochastic gradient mcmc. In *NIPS*, pp. 2917–2925, 2015.

Marcus, Mitchell P, Marcinkiewicz, Mary Ann, and Santorini, Beatrice. Building a large annotated corpus of english: The penn treebank. *Computational linguistics*, 1993.

Mattingly, J. C., Stuart, A. M., and Tretyakov, M. V. Construction of numerical time-average and stationary measures via Poisson equations. *SIAM J. NUMER. ANAL.*, 48(2):552–577, 2010.

Metropolis, Nicholas, Rosenbluth, Arianna W, Rosenbluth, Marshall N, Teller, Augusta H, and Teller, Edward. Equation of state calculations by fast computing machines. *The journal of chemical physics*, 21(6), 1953.

Neal, Radford M. Slice sampling. *Annals of statistics*, 2003.

Neal, Radford M. MCMC using Hamiltonian dynamics. *Handbook of Markov Chain Monte Carlo*, 2, 2011.

Risken, Hannes. Fokker-planck equation. In *The Fokker-Planck Equation*, pp. 63–95. Springer, 1984.

Rumelhart, David E, Hinton, Geoffrey E, and Williams, Ronald J. Learning representations by back-propagating errors. *Cognitive modeling*, 5(3):1, 1988.

Teh, Yee Whye, Thiéry, Alexandre, and Vollmer, Sebastian. Consistency and fluctuations for stochastic gradient langevin dynamics. *ArXiv*, 2014.

Tieleman, Tijmen and Hinton, Geoffrey. Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude. *COURSERA: Neural networks for machine learning*, 2012.

Tuckerman, Mark. *Statistical mechanics: theory and molecular simulation*. Oxford University Press, 2010.

Vollmer, Sebastian J, Zygalakis, Konstantinos C, and Teh, Yee Whye. Exploration of the (non-) asymptotic bias and variance of stochastic gradient langevin dynamics. *Journal of Machine Learning Research*, 17(159):1–48, 2016.

Welling, Max and Teh, Yee W. Bayesian learning via stochastic gradient langevin dynamics. In *ICML*, 2011.

Zhang, Yizhe, Wang, Xiangyu, Chen, Changyou, Henao, Ricardo, Fan, Kai, and Carin, Lawrence. Towards unifying hamiltonian monte carlo and slice sampling. In *NIPS*, 2016.