A SCALED STOCHASTIC NEWTON ALGORITHM FOR MARKOV CHAIN MONTE CARLO SIMULATIONS

TAN BUI-THANH † AND OMAR GHATTAS ‡

Abstract. We propose a scaled stochastic Newton algorithm (sSN) for local Metropolis-Hastings Markov chain Monte Carlo (MCMC) simulations. The method can be considered as an Euler-Maruyama discretization of the Langevin diffusion on a Riemann manifold with piecewise constant Hessian of the negative logarithm of the target density as the metric tensor. The sSN proposal consists of deterministic and stochastic parts. The former corresponds to a Newton step that attempts to move the current state to a region of higher probability, hence potentially increasing the acceptance probability. The latter is distributed by a Gaussian tailored to the local Hessian as the inverse covariance matrix. The proposal step is then corrected by the standard Metropolization to guarantee that the target density is the stationary distribution. We study asymptotic convergence and geometric ergodicity of sSN chains. At the heart of the paper is the optimal scaling analysis, in which we show that, for inhomogeneous product target distribution at stationarity, the sSN proposal variance scales like $\mathcal{O}\left(n^{-1/3}\right)$ for the average acceptance rate to be bounded away from zero, as the dimension n approaches infinity. As a result, a sSN chain explores the stationary distribution in $\mathcal{O}\left(n^{1/3}\right)$ steps, regardless of the variance of the target density. The optimal scaling behavior of sSN chains in the transient phase is also discussed for Gaussian target densities, and an extension to inverse problems using the Bayesian formulation is presented. The theoretical optimal scaling result is verified for two i.i.d. targets in high dimensions. We also compare the sSN approach with other similar Hessian-aware methods on i.i.d. targets, Bayesian logistic regression, and log-Gaussian Cox process examples. Numerical results show that sSN outperforms the others by providing Markov chains with small burn-in and small correlation length. Finally, we apply the sSN method to a Bayesian inverse thermal fin problem to predict the posterior mean and its uncertainty.

Key words. Markov chain Monte Carlo; Metropolis-Hastings; Optimal scaling; Hessian; Langevin diffusion; Riemannian manifold; Euler-Maruyama discretization.

AMS subject classifications. 60J05, 60J20, 65C05, 60J22, 65C40

1. Introduction.

Perhaps the Metropolis-Hastings (MH) algorithm, first developed by Metropolis *et al.* [21] and then generalized by Hastings [15], is the most popular Markov chain Monte Carlo method. Its popularity and attractiveness come from the easiness in implementation and minimal requirements on the target density and the proposal density [27]. Indeed, a MH pseudo code for generating N samples from target density $\pi(\boldsymbol{x})$, $\boldsymbol{x} \in \mathbb{R}^n$, can be succinctly described in Algorithm 1. One of the simplest instances of Algorithm 1 is the random walk Metropolis-Hastings

Algorithm 1 Metropolis-Hastings MCMC Algorithm

Choose initial $\boldsymbol{x}_0$

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

1. Draw a proposal $\boldsymbol{y}$ from the proposal density $q(\boldsymbol{x}_k, \boldsymbol{y})$
2. Compute $\pi(\boldsymbol{y})$, $q(\boldsymbol{x}_k, \boldsymbol{y})$, and $q(\boldsymbol{y}, \boldsymbol{x}_k)$
3. Compute the acceptance probability $\alpha(\boldsymbol{x}_k, \boldsymbol{y}) = \min\left\{1, \frac{\pi(\boldsymbol{y})q(\boldsymbol{y},\boldsymbol{x}_k)}{\pi(\boldsymbol{x}_k)q(\boldsymbol{x}_k,\boldsymbol{y})}\right\}$
4. **Accept** and set $\boldsymbol{x}_{k+1} = \boldsymbol{y}$ with probability $\alpha(\boldsymbol{x}_k, \boldsymbol{y})$. Otherwise, **reject** and set $\boldsymbol{x}_{k+1} = \boldsymbol{x}_k$

end for

‡Institute for Computational Engineering & Sciences, The University of Texas at Austin, Austin, TX 78712, USA.

‡Institute for Computational Engineering & Sciences, Jackson School of Geosciences, and Department of Mechanical Engineering, The University of Texas at Austin, Austin, TX 78712, USA.

(RWMH) in which the proposal density $q(\boldsymbol{x}_k, \boldsymbol{y})$ is the isotropic Gaussian kernel

$$q(\boldsymbol{x}_k, \boldsymbol{y}) = \frac{1}{\sqrt{(2\pi)^n \sigma^{2n}}} \exp\left(-\frac{1}{2\sigma^2} \|\boldsymbol{y} - \boldsymbol{x}_k\|^2\right),$$

and hence the proposal $\boldsymbol{y}$ is given by

$$\boldsymbol{y} = \boldsymbol{x}_k + \sigma \mathcal{N}(0, \mathbf{I}_n), \tag{1.1}$$

where $\mathbf{I}_n$ is the $n \times n$ identity matrix. The above proposal can be considered as the Euler-Maruyama discretization, with step size $\Delta t = \sigma^2$, of the following stochastic differential equation

$$d\boldsymbol{x}(t) = d\boldsymbol{W}(t),$$

with $\boldsymbol{W}(t)$ as the standard n-dimensional Brownian motion. The RWMH method is simple except for a small detail: how to choose the optimal time step σ^2?

Choosing the time step, also known as the proposal variance, σ^2 optimally is vital since it determines the mixing time which is the number of steps to explore the stationary distribution. If σ^2 is too small, it is most likely that all the proposal moves are accepted, but the chain explores $\pi(\boldsymbol{x})$ very slowly since the proposed jump is small. On the other hand, if the proposal variance is large, it is most likely that the proposed move is in low probability regions, and hence rejected. This case also leads to slow mixing since the chain virtually does not move at all. As a result, the proposal variance should be in between these extremes, and this is known as the Goldilocks principle [35]. It turns out that the proposal variance for RWMH must scale like $\sigma^2 = \ell^2 n^{-1}$, with some constant ℓ, for the average acceptance rate to be bounded away from zero as the dimension n approaches infinity [13, 28]. In this case, the optimal average acceptance rate can be shown to be 0.234.

Meanwhile, the Langevin dynamics governed by the following stochastic differential equation

$$d\boldsymbol{x}(t) = \frac{1}{2}\nabla \log(\pi(\boldsymbol{x}))\, dt + d\boldsymbol{W}(t) \tag{1.2}$$

is well-known to admit $\pi(\boldsymbol{x})$ as its stationary distribution. A natural idea analogous to RWMH is to discretize the Langevin equation using the Euler-Maruyama scheme to construct the proposal as

$$\boldsymbol{y} = \boldsymbol{x}_k + \frac{\sigma^2}{2}\nabla \log(\pi(\boldsymbol{x}_k)) + \sigma \mathcal{N}(0, \mathbf{I}_n). \tag{1.3}$$

Although one can show that the diffusion process given by (1.2) converges to $\pi(\boldsymbol{x})$ as its unique stationary measure [32], the discretization (1.3) may not inherit this property as demonstrated in [33]. In order to avoid this undesired behavior, the Euler-Maruyama scheme (1.3) is typically equipped with the Metropolis mechanism outlined in Steps 2, 3, and 4 of Algorithm 1 so that the Markov chain converges to the desired distribution given by the target $\pi(\boldsymbol{x})$. This gives rise to the so-called Metropolis-adjusted Langevin algorithm (MALA) [33].

Compared to the random walk proposal (1.1), the Langevin proposal (1.3) has the additional deterministic term $\frac{\sigma^2}{2}\nabla \log(\pi(\boldsymbol{x}_k))$ (also known as the drift term). A careful look at this term reveals that it is the scaled negative gradient of $f(\boldsymbol{x}_k) =$

$-\log(\pi(\boldsymbol{x}_k))$, and hence $\boldsymbol{x}_k + \frac{\sigma^2}{2}\nabla\log(\pi(\boldsymbol{x}_k))$ is nothing more than a gradient descent step for minimizing $f(\boldsymbol{x})$ in the optimization literature [24]. Consequently, the drift term takes the current state $\boldsymbol{x}_k$ to a point with smaller value of $f(\boldsymbol{x})$, i.e. higher probability density $\pi(\boldsymbol{x})$, provided that σ^2 is sufficiently small. This explains why MALA chains in general mix better and explore the target faster than those generated by RWMH. Indeed, the work in [29] shows that the optimal proposal variance scales like $\sigma^2 = \ell^2 n^{-1/3}$, and the optimal acceptance rate is approximately 0.574, a substantial improvement over the RWMH algorithm.

The Langevin dynamics can be viewed as a particular instance of the following stochastic differential equation

$$d\boldsymbol{x}(t) = \mathbf{b}(\boldsymbol{x})\,dt + \boldsymbol{\beta}(\boldsymbol{x})\,d\boldsymbol{W}(t), \tag{1.4}$$

where

$$b_i(\boldsymbol{x}) = \frac{1}{2}\sum_{j=1}^{n} a_{ij}(\boldsymbol{x})\frac{\partial}{\partial x_j}\log(\pi(\boldsymbol{x})) + \sqrt{\delta(\boldsymbol{x})}\sum_{j=1}^{n}\frac{\partial}{\partial x_j}\left(a_{ij}(\boldsymbol{x})\sqrt{\delta(\boldsymbol{x})}\right),$$

$\mathbf{a}(\boldsymbol{x}) = \boldsymbol{\beta}(\boldsymbol{x})\boldsymbol{\beta}^T(\boldsymbol{x})$, and $\delta(\boldsymbol{x}) = \det(\mathbf{a}(\boldsymbol{x}))$. It can be shown that $\pi(\boldsymbol{x})$ is the unique stationary measure for (1.4) under mild conditions [32]. Equation (1.4) is also known as Langevin diffusion on Riemann manifold with metric tensor $\mathbf{a}^{-1}(\boldsymbol{x})$ [14]. Clearly, (1.4) collapses to the standard Langevin dynamics (1.2) if the metric tensor $\mathbf{a}^{-1}$ is the identity matrix.

Following the spirit of RWMH and MALA, one can discretize (1.4) as

$$\begin{aligned}\boldsymbol{y} = \boldsymbol{x}_k + \frac{\sigma^2}{2}\mathbf{a}(\boldsymbol{x}_k)\nabla\log(\pi(\boldsymbol{x}_k)) + \sigma\boldsymbol{\beta}(\boldsymbol{x}_k)\mathcal{N}(0,\mathbf{I}_n)\\ + \sigma^2\sqrt{\delta(\boldsymbol{x}_k)}\sum_{j=1}^{n}\frac{\partial}{\partial x_j}\left(a_{ij}(\boldsymbol{x}_k)\sqrt{\delta(\boldsymbol{x}_k)}\right),\end{aligned} \tag{1.5}$$

and use this result as the proposal for the MH algorithm. However, each proposal step is now complicated and expensive since (1.5) involves the derivative of the metric tensor in the last term. Note that this last term can be viewed as the changes in local curvature of the manifold. This observation immediately invites us to use the Hessian $\nabla^2 f(\boldsymbol{x})$ as metric tensor, i.e., $\mathbf{a}^{-1}(\boldsymbol{x}) = \nabla^2 f(\boldsymbol{x})$. As a consequence, the drift term $\frac{\sigma^2}{2}\mathbf{a}(\boldsymbol{x}_k)\nabla\log(\pi(\boldsymbol{x}_k))$ becomes a scaled Newton step for minimizing $f(\boldsymbol{x})$ [24]. Similar to the above discussion, the drift term, assuming zero curvature changes, takes the current state $\boldsymbol{x}_k$ to a point with smaller value of $f(\boldsymbol{x})$, which is the same as higher probability density $\pi(\boldsymbol{x})$, provided that σ^2 is sufficiently small and the Hessian $\nabla^2 f(\boldsymbol{x})$ is positive definite. Since Newton method is in general more efficient than the gradient descent in finding smaller $f(\boldsymbol{x})$ [24] we expect that MCMC chains using proposal (1.5) mix better and explore the target faster than those using MALA.

In fact, using the Hessian to speed up MCMC simulations is not new. This can be traced back to [36] under the name of smart Monte Carlo simulation, and its connection to the Newton method in optimization is succinctly presented in [9, 20]. Since then, variants of the method have been rediscovered, including Hessian-based MCMC [16, 26] and Newton-like methods for MCMC [20, 40, 41]. On the other hand, it can be viewed as a preconditioned Langevin MCMC approach [37] or a manifold Langevin MCMC [14]. These Hessian-aware approaches have been observed, in many

cases, to outperform the other existing methods including MALA [20, 26]. Most of them, however, use either fixed or random proposal variance σ^2 without respect to the dimensions of the problem under consideration, especially large dimensions. Optimal scaling results for these Hessian-aware methods, similar to those of RWMH and MALA, are therefore desirable in order to use them efficiently in practice, but there has been no attempt in this direction.

We take up the idea of using Hessian to speed up MCMC simulations again in this paper. We shall show how the connection presented above for the proposal $\boldsymbol{y}$ with optimization and with the Euler-Maruyama discretization of stochastic differential equations leads us to a new method called scaled stochastic Newton algorithm (sSN) in Section 2. After that, we discuss asymptotic convergence of sSN chains and their geometric ergodicity in Section 3. The optimal scaling analysis for the sSN method is carried out at length in Section 4. In particular, we aim to answer the question on how the sSN proposal variance scales so that the asymptotic average acceptance rate is bounded away from zero, as the dimension approaches infinity. We also discuss the optimal scaling in transient phase of sSN chain for Gaussian targets and an extension of optimal scaling analysis to Bayesian posterior measure with Gaussian priors. Next, we verify the theoretical results and compare sSN with the simplified manifold MALA of [14], the Hessian-based MCMC of [26], the stochastic Newton of [20], and a variant of sSN in Section 5. As an application to uncertainty quantification, we apply the sSN approach to estimate the posterior mean of a Bayesian inverse thermal fin problem and discuss its uncertainty. Finally, Section 6 concludes the paper with discussions on future studies.

2. A scaled stochastic Newton algorithm (sSN). Motivated by the discussion in Section 1, we would like to use the Hessian as the metric tensor, i.e.,

$$\mathbf{a}^{-1}(\boldsymbol{x}) = \nabla^2 f(\boldsymbol{x}).$$

However, the last term of (1.5) is in general not zero. To avoid this expensive term, we make the following assumption. We assume that the local curvature of the manifold between $\boldsymbol{x}_k$ and $\boldsymbol{y}$ is constant, and drop the last term. In other words, we discretize the general Langevin diffusion (1.4) using an Euler-Maruyama scheme with piecewise constant metric tensor. If we do that, we end up with the new proposal

$$\boldsymbol{y} = \boldsymbol{x}_k + \frac{\sigma^2}{2}\boldsymbol{A}\nabla \log\left(\pi\left(\boldsymbol{x}_k\right)\right) + \sigma\mathcal{N}\left(0, \boldsymbol{A}\right), \tag{2.1}$$

where $\boldsymbol{A}$ is the inverse of the Hessian, i.e.,

$$\boldsymbol{A}^{-1} = \boldsymbol{H} = -\nabla^2 \log\left(\pi\left(\boldsymbol{x}_k\right)\right), \tag{2.2}$$

evaluated at $\boldsymbol{x}_k$ and assumed to be fixed when we compute the probability $q(\boldsymbol{x}_k, \boldsymbol{y})$ of moving from $\boldsymbol{x}_k$ to $\boldsymbol{y}$, and the the probability $q(\boldsymbol{y}, \boldsymbol{x}_k)$ of moving from $\boldsymbol{y}$ to $\boldsymbol{x}_k$:

$$q(\boldsymbol{x}_k, \boldsymbol{y}) = \frac{\sqrt{\det \boldsymbol{H}}}{\sqrt{(2\pi)^n}} \exp\left\{-\frac{1}{2\sigma^2}\left\|\boldsymbol{y} - \boldsymbol{x}_k - \frac{\sigma^2}{2}\boldsymbol{A}\nabla \log\left(\pi\left(\boldsymbol{x}\right)\right)\right\|^2_{\boldsymbol{A}^{-1}}\right\}, \tag{2.3}$$

$$q(\boldsymbol{y}, \boldsymbol{x}_k) = \frac{\sqrt{\det \boldsymbol{H}}}{\sqrt{(2\pi)^n}} \exp\left\{-\frac{1}{2\sigma^2}\left\|\boldsymbol{x}_k - \boldsymbol{y} - \frac{\sigma^2}{2}\boldsymbol{A}\nabla \log\left(\pi\left(\boldsymbol{y}\right)\right)\right\|^2_{\boldsymbol{A}^{-1}}\right\}, \tag{2.4}$$

where we have defined the weighted norm $\|\cdot\|_{\boldsymbol{A}^{-1}} = \left(\cdot, \boldsymbol{A}^{-1}\cdot\right)$ for any symmetric positive definite matrix $\boldsymbol{A}$.

The deterministic part of (2.1) is therefore a (half) Newton step at $\boldsymbol{x}$ if $\sigma = 1$. For this reason, we shall call our method as scaled stochastic Newton algorithm (sSN), and its complete description is given in Algorithm 2.

Algorithm 2 Scaled stochastic Newton Algorithm

Choose initial $\boldsymbol{x}_0$
for $k = 0, \ldots, N-1$ **do**
1. Compute $\boldsymbol{\nabla} \log(\pi(\boldsymbol{x}_k)), \boldsymbol{H}(\boldsymbol{x}_k) = -\nabla^2 \log(\pi(\boldsymbol{x}_k))$
2. Draw sample $\boldsymbol{y}$ from the proposal density $q(\boldsymbol{x}_k, \cdot)$ defined in equation (2.3)
3. Compute $\pi(\boldsymbol{y})$, $q(\boldsymbol{x}_k, \boldsymbol{y})$ as in (2.3), and $q(\boldsymbol{y}, \boldsymbol{x}_k)$ as in (2.4)
4. Compute the acceptance probability $\alpha(\boldsymbol{x}_k, \boldsymbol{y}) = \min\left(1, \frac{\pi(\boldsymbol{y}) q(\boldsymbol{y}, \boldsymbol{x}_k)}{\pi(\boldsymbol{x}_k) q(\boldsymbol{x}_k, \boldsymbol{y})}\right)$
5. **Accept** and set $\boldsymbol{x}_{k+1} = \boldsymbol{y}$ with probability $\alpha(\boldsymbol{x}_k, \boldsymbol{y})$. Otherwise, **reject** and set $\boldsymbol{x}_{k+1} = \boldsymbol{x}_k$

end for

3. Asymptotic convergence and geometric ergodicity of sSN. Since the sSN method is an instance of the Metropolis-Hasting Algorithm 1 with a special proposal density, the standard result on asymptotic convergence [22, 31, 39] is straightforward. In particular, the fact that a sSN chain is Markov is clear. Moreover, since $\boldsymbol{A}$ is constant during the Metroplization, the detailed balance is satisfied.

As for the geometric ergodicity, instead of striking for total generality, which may not be possible [31, 34], we restrict the discussion for a class of super-exponential targets [1, 17]. In that case, if we assume that $\boldsymbol{g}(\boldsymbol{x})$ and spectrum of $\boldsymbol{H}(\boldsymbol{x})$ are bounded, then we can show that sSN chains are geometrically ergodic by adapting the proof of [1, Proposition 2.1] in a straightforward manner.

4. Complexity and optimal scaling analysis for sSN in high dimensions. In this section, we discuss the complexity and optimal scaling of the sSN for high dimensional problems, particularly when the dimension n approaches infinity. We shall address of the question how to choose the proposal variance σ^2 in (2.1) optimally and the corresponding optimal acceptance rate. The target of interest here has the following product form

$$\pi(\boldsymbol{x}) = \Pi_{i=1}^{n} \frac{1}{\lambda_i} \exp\left(g\left(\frac{x_i}{\lambda_i}\right)\right), \tag{4.1}$$

where $g : \mathbb{R} \to \mathbb{R}_- = \mathbb{R} \setminus (0, +\infty)$, and $\lim_{x_i \to \pm\infty} g(x_i/\lambda_i) = -\infty$ for $i = 1, \ldots, n$. The variance of each component λ_i^2 is allowed to depend on the dimension of the problem. Note that no explicit dependence of λ_i on n is required in this paper while a special form of λ_i^2 is assumed in [2, 3] to make analytical computations and proofs possible. Following [2, 3, 29], we assume that all moments of f are finite and g is infinitely differentiable with polynomially bounded derivatives.

As discussed above, the proposal variance σ^2 should be neither small nor large for a MCMC chain to mix well. A question immediately arises is what optimal criteria is sensible. A criteria that leads to explicit expressions for the optimal proposal variance and optimal acceptance rate is to maximize the square-jump distance while making the acceptance probability bounded away from zero [2]. Other criteria have been discussed and studied in the literature, e.g., see [28–30] and the references therein. Following the program in [2], we first find the condition for the proposal variance σ^2 of the sSN algorithm, under which the acceptance probability is bounded away

from zero when the dimension approaches infinity. Then we determine the maximal square-jump distance and finally deduce the corresponding optimal acceptance rate.

4.1. Asymptotic average acceptance rate. Let us define

$$h^{(j)}(x_i) = \frac{1}{\lambda_i^j} g^{(j)}(x_i/\lambda_i), \tag{4.2}$$

where the superscript (j) denotes the jth derivative. A simple computation gives

$$\nabla \log(\pi(\boldsymbol{x})) = \left[h^{(1)}(x_1), \ldots, h^{(1)}(x_n)\right]^T,$$
$$\boldsymbol{H}(\boldsymbol{x}) = -\nabla^2 \log(\pi(\boldsymbol{x})) = -\operatorname{diag}\left[h^{(2)}(x_1), \ldots, h^{(2)}(x_n)\right]^T,$$

where the superscript T denotes matrix or vector tranposition and diag operator maps a column vector to the corresponding diagonal matrix. The positiveness assumption on $\mathbf{H}$ translates into the positiveness condition on each component, i.e., $-h^{(2)}(x_i) > 0$. We further assume that $-g^{(2)}(x_i/\lambda_i)$ is bounded below away from zero and above away from infinity uniformly in x_i, namely, $0 < \ell < -g^{(2)}(x_i/\lambda_i) < L < \infty$. Consequently, the ith component of the proposal move in (2.1) can be written as

$$y_i = x_i - \frac{\sigma^2}{2}\frac{h^{(1)}(x_i)}{h^{(2)}(x_i)} + \frac{\sigma}{\sqrt{-h^{(2)}(x_i)}} z_i,$$

where $z_i, i = 1, \ldots, n$ are independent and identically distributed (i.i.d.) standard normal random variables. If $h^{(j)} = 0$ our method simplifies to a preconditioned RWMH which we shall exclude in the following analysis. Throughout this section, we assume that our Markov chain is started at stationarity, i.e., $\boldsymbol{x}_0 \sim \pi(\boldsymbol{x})$.

The proposal densities in (2.3) and (2.4) now read

$$q(\boldsymbol{x}, \boldsymbol{y}) = \frac{\sqrt{(-1)^n \Pi_{i=1}^n h^{(2)}(x_i)}}{\sqrt{(2\pi)^n}} \exp\left\{\frac{1}{2\sigma^2}\sum_{i=1}^n \left[y_i - x_i - \frac{\sigma^2}{2}\frac{h^{(1)}(x_i)}{h^{(2)}(x_i)}\right]^2 h^{(2)}(x_i)\right\},$$
$$q(\boldsymbol{y}, \boldsymbol{x}) = \frac{\sqrt{(-1)^n \Pi_{i=1}^n h^{(2)}(x_i)}}{\sqrt{(2\pi)^n}} \exp\left\{\frac{1}{2\sigma^2}\sum_{i=1}^n \left[x_i - y_i - \frac{\sigma^2}{2}\frac{h^{(1)}(y_i)}{h^{(2)}(x_i)}\right]^2 h^{(2)}(x_i)\right\}.$$

In this case, the acceptance probability becomes

$$\alpha(\boldsymbol{x}, \boldsymbol{y}) = 1 \wedge \exp(T(\sigma)),$$

where operator $\wedge$ takes minimum value of its left and right arguments, and $T(\sigma)$ is defined by

$$T(\sigma) = \sum_{i=1}^n \left[h(y_i) - h(x_i)\right] + \frac{(x_i - y_i)}{2}\left(h^{(1)}(y_i) + h^{(1)}(x_i)\right) + \frac{\sigma^2}{8h^{(2)}(x_i)}\left[\left(h^{(1)}(y_i)\right)^2 - \left(h^{(1)}(x_i)\right)^2\right].$$

In the sequel, we need the following technical Lemma.

LEMMA 4.1. *Suppose T is a real-valued random variable.*

i) For any $c > 0$, there holds $\mathbb{E}\left[1 \wedge \exp\left(T\right)\right] \geq \exp\left(-c\right)\left(1 - \frac{\mathbb{E}[|T|]}{c}\right)$.

ii) If $T \sim \mathcal{N}\left(\mu, \delta^2\right)$, *then the following holds*

$$\mathbb{E}\left[1 \wedge \exp\left(T\right)\right] = \Phi\left(\mu/\delta\right) + \exp\left(\mu + \delta^2/2\right)\Phi\left(-\delta - \mu/\delta\right),$$

where $\Phi\left(\cdot\right)$ is the cumulative distribution function of the standard normal distribution.

Proof. See [3] for a proof. □

Next performing Taylor expansion for $T\left(\sigma\right)$ around $\sigma = 0$ up to the sixth order term gives

$$T\left(\sigma\right) = \underbrace{\sum_{j=0}^{6} \sigma^j \sum_{i=1}^{n} C_{i,j}}_{C_j} + \underbrace{\sigma^7 \sum_{i=1}^{n} C_{i,7}\left(\sigma_i^*\right)}_{C_7}. \tag{4.3}$$

Recall that our goad is to find conditions for the proposal variance σ^2 so that the average acceptance probability $\mathbb{E}\left[\alpha\left(\boldsymbol{x}, \boldsymbol{y}\right)\right]$ is bounded away from zero. To this end, we observe that

$$\mathbb{E}\left[\alpha\left(\boldsymbol{x}, \boldsymbol{y}\right)\right] = \mathbb{E}\left[1 \wedge \exp\left(T\left(\sigma\right)\right)\right] \geq \exp\left(-c\right)\left[1 - \frac{\mathbb{E}\left[|T\left(\sigma\right)|\right]}{c}\right], \quad \forall c > 0,$$

where the last inequality is from Lemma 4.1. Consequently, our goal is fulfilled if $\mathbb{E}\left[|T\left(\sigma\right)|\right]$ is bounded from above. From (4.3), it is sufficient to seek the dependence of the proposal variance σ on the dimension n for which the expectation of the absolute value of each term in (4.3) is bounded from above as n approaches infinity.

Observe that $C_{i,j}$, $i = 1, \ldots, n$, are i.i.d. with respect to i, and hence the expectation of any power of $C_{i,j}$ does not depend on i. For example, we simply write $\mathbb{E}\left[C_{\cdot,j}\right]$ as the expectation of any $C_{i,j}$, for $i = 1, \ldots, n$. Moreover, $C_{i,j}$ and R_i do not depend explicitly on λ_i, and hence on the dimension n, but only implicitly through g and its derivatives. This is an important feature of sSN that makes it more efficient, as shall be shown, compared to MALA or RWMH. For example, $C_{i,3}$ is given by

$$C_{i,3} = -\frac{3z_i h^{(1)}(x_i)h^{(2)}(x_i) + z^3 h^{(3)}(x_i)}{12\left(-h^{(2)}(x_i)\right)^{3/2}} = -\frac{3z_i g^{(1)}(x_i/\lambda_i)g^{(2)}(x_i/\lambda_i) + z^3 g^{(3)}(x_i/\lambda_i)}{12\left(-g^{(2)}(x_i/\lambda_i)\right)^{3/2}}.$$

It should be pointed out that the Taylor expansion (4.3) of $T\left(\sigma\right)$ is almost identical to that of the MALA method [2, 3]. However, the coefficients $C_{i,j}$ corresponding to (2.2) are those corresponding to the MALA scaled by a factor $1/\left(\sqrt{-h^{(2)}\left(x_i\right)}\right)^j$. It is this extra factor that makes the analytical computation difficult. In particular, it is not clear to us whether the assertions $\mathbb{E}\left[C_{i,4}\right] = 0$ and $2\mathbb{E}\left[C_{i,6}\right] + \mathbb{E}\left[C_{i,3}^2\right] = 0$ still hold true in general. Therefore, let us study C_3, C_5, C_6 and C_7 first, then C_4.

Similar to the analyis of the MALA method [29], we observe that $C_0 = C_1 = C_2 = 0$, and $\mathbb{E}\left[C_3\right] = \mathbb{E}\left[C_5\right] = 0$ since they involve only odd orders of z_i. Here $\mathbb{E}$ denotes the expectation with respect to both $\boldsymbol{x}$ and $\mathbf{z}$.

We begin by bounding $\mathbb{E}\left[|C_j|\right]$ for $j = 3, 5$. Since $|\cdot|^2$ is convex, an application of Jensen's inequality gives

$$\mathbb{E}\left[|C_3|\right] \leq n^{1/2}\sigma^j\sqrt{\mathbb{E}\left[C_{\cdot,j}^2\right]}, \quad j = 3, 5. \tag{4.4}$$

Owing to the assumption that g has polynomially bounded derivatives and f has finite moments, $\mathbb{E}\left[C_{\cdot,j}^2\right]$ is bounded uniformly in n. As a result, the right side of (4.4) is bounded as $n \to \infty$ if $\sigma^2 = \mathcal{O}\left(n^{-1/3}\right)$.

For C_6, a simple application of the triangle inequality gives

$$\mathbb{E}\left[|C_6|\right] \leq n\sigma^6 \mathbb{E}\left[|C_{\cdot,6}|\right],$$

which is clearly bounded as $n \to \infty$ if $\sigma^2 = \mathcal{O}\left(n^{-1/3}\right)$.

In order to bound C_7 one realizes that $C_{i,7}$ is a ratio of product of polynomials of derivatives of $g\left(x_i/\lambda_i\right)$. This, together with the assumption on g, implies that

$$|C_{i,7}| \leq P_1\left(x_i/\lambda_i\right) P_2\left(z_i\right) P_3\left(\sigma_i^*\right),$$

where P_1, P_2 and P_3 are positive polynomials. We have the following two observations. First, $\mathbb{E}_{x_i/\lambda_i}\left[P_1\left(x_i/\lambda_i\right)\right]$ and $\mathbb{E}_{\mathbf{z_i}}\left[P_2\left(z_i\right)\right]$ are bounded independent of i due to the finiteness of moments of f and the Gaussianity of z_i, respectively. Second, $P_3\left(\sigma_i^*\right)$ is bounded above by some constant independent of n if $\sigma \to 0$, since $0 \leq \sigma_i^* \leq \sigma$. Therefore, we can bound $\mathbb{E}\left[|C_7|\right]$ as

$$\mathbb{E}\left[|C_7|\right] \leq cn\sigma^7 \mathbb{E}_{x_i/\lambda_i}\left[P_1\left(x_i/\lambda_i\right)\right] \mathbb{E}_{z_i}\left[P_2\left(z_i\right)\right] P_3\left(\sigma_i^*\right),$$

for some constant c independent of n and sufficiently large. It is readily to see that $\sigma^2 = \mathcal{O}\left(n^{-1/3}\right)$ is sufficient for $\mathbb{E}\left[|C_7|\right]$ to be bounded from above.

For C_4, we have not yet been able to bound it when the proposal variance scales as $\sigma^2 = \mathcal{O}\left(n^{-1/3}\right)$, so let us assume that C_4 converges to $\ell^4 K_3$ in $L^2\left(\pi\left(\boldsymbol{x}\right) \otimes \mathcal{N}\left(0, I\right)\right)$ as $n \to \infty$. By the triangle and Jensen's inequalities one has

$$\mathbb{E}\left[|C_4|\right] \leq \mathbb{E}\left[\left|C_4 - \ell^4 K_3\right|\right] + \mathbb{E}\left[|K_3|\right] \leq \sqrt{\mathbb{E}\left[\left(C_4 - \ell^4 K_3\right)^2\right]} + \mathbb{E}\left[\left|\ell^4 K_3\right|\right],$$

from which it follows that $\mathbb{E}\left[|C_4|\right]$ is bounded from above.

Remark 4.2. *The above assumption on C_4 is clearly satisfied if f is a Gaussian, for which $K_3 = 0$. However, whether this holds for larger class of densities f is an open question.*

Combining all the estimates we conclude that $\mathbb{E}\left[|T\left(\sigma\right)|\right]$ is bounded from above if $\sigma^2 = \mathcal{O}\left(n^{-1/3}\right)$. This motivates us to define $\sigma^2 = \ell^2 n^{\varepsilon - 1/3}$, where ℓ is independent of n. We have showed that the average acceptance rate $\mathbb{E}\left[\alpha\left(\mathbf{x}, \mathbf{y}\right)\right]$ is bounded away from zero if $\varepsilon = 0$. Furthermore, the above analysis also shows that $\mathbb{E}\left[|T\left(\sigma\right)|\right]$ converges to zero, and hence $\mathbb{E}\left[\alpha\left(\mathbf{x}, \mathbf{y}\right)\right]$ approaches 1, as $n \to \infty$ if $\varepsilon < 0$.

What remains to be investigated is the case $\varepsilon \in (0, 1/3)$. If $\mathbb{E}\left[C_{i,4}\right] = 0$ and $2\mathbb{E}\left[C_{i,6}\right] + \mathbb{E}\left[C_{i,3}^2\right] = 0$ hold, an argument similar to that in [2, 3] would yield zero average acceptance rate as $n \to \infty$. However, the appearance of $\left(\sqrt{-h^{(2)}\left(x_i\right)}\right)^j$ in the denominator of $C_{i,j}$ renders analytical computation intractable, and we have not yet been able to obtain such a result. Instead of making further assumptions, we leave the analysis of the case $\varepsilon \in (0, 1/3)$ as a subject for future work, and carry on our program with $\varepsilon \leq 0$.

As argued above, $\varepsilon < 0$ leads to unity average acceptance rate, which corresponds to the case of small proposal variance σ^2. That is, the Markov chain evolves very slowly in exploring the stationary distribution, and hence is not of our interest. From now on to the end of the paper, the proposal variance is specified as $\sigma^2 = \ell^2 n^{-1/3}$.

We have provided sufficient conditions for the average acceptance rate to be bounded away from zero, but we have not yet derived its explicit expression. This is our next goal. We first show that C_3 converges weakly using Lindeberg-Feller central limit theorem [12]. To that end, we define

$$X_{n,i} = \sigma^3 C_{i,3},$$

then it is obvious that $X_{n,i}$, $i = 1, \ldots, n$, are independent random numbers and $\mathbb{E}[X_{n,i}] = 0$. Next, simple applications of the integration by parts give

$$\sum_{i=1}^{n} \mathbb{E}\left[X_{n,i}^2\right] = \sigma^6 \sum_{i=1}^{n} \underbrace{\frac{1}{48}\mathbb{E}_{x_i}\left[9\frac{\left(h^{(1)}(x_i)\right)^2}{-h^{(2)}(x_i)} + 5\frac{\left(h^{(3)}(x_i)\right)^2}{\left(-h^{(2)}(x_i)\right)^3} - 6\right]}_{K_1} = \ell^6 K_1,$$

where K_1 is independent of x_i, since x_i, $i = 1, \ldots, n$, are i.i.d. random variables. Furthermore, since $\mathbb{E}\left[C_{\cdot,3}^2\right] < \infty$, an application of Lebesgue dominated convergence theorem yields, for any $\gamma > 0$,

$$\lim_{n\to\infty} \sum_{i=1}^{n} \mathbb{E}\left[X_{n,i}^2 1_{\{|X_{n,i}|>\gamma\}}\right] = \lim_{n\to\infty} \ell^2 \mathbb{E}\left[C_{\cdot,3}^2 1_{\left\{|C_{\cdot,3}|>\gamma\ell^{-2}\sqrt{n}\right\}}\right] = 0,$$

with $1_{\{\cdot\}}$ denoting the indicator function. At this point, we have demonstrated that all the conditions of Lindeberg-Feller theorem are satisfied, whence

$$C_3 = \sum_{i=1}^{n} \sigma^3 C_{i,3} = \sum_{i=1}^{n} X_{n,i} \Rightarrow \ell^3 \sqrt{K_1},$$

in the sense of weak convergence (also known as convergence in distribution [12]) as $n \to \infty$.

Now, combing $\sigma^2 = \ell^2 n^{-1/3}$, $\mathbb{E}[C_5] = 0$, and straightforward algebra shows that C_5 converges in $L^2(\pi(\boldsymbol{x}) \otimes \mathcal{N}(0, I))$ to zero. Similarly, one can show that C_6 converges to $\ell^6 K_2$, where $K_2 = \mathbb{E}[C_{\cdot,6}]$, in $L^2(\pi(\boldsymbol{x}) \otimes \mathcal{N}(0, I))$. The convergence of C_7 to zero in $L^2(\pi(\boldsymbol{x}) \otimes \mathcal{N}(0, I))$ is clear as shown above.

Since the covergence in $L^p(\pi(\boldsymbol{x}) \otimes \mathcal{N}(0, I))$, for any $1 \leq p < \infty$, implies convergence in distribution, we obtain the following weak convergence of $T(\sigma)$

$$T(\sigma) \Rightarrow \mathcal{N}\left(\ell^6 K_2 + \ell^4 K_3, \ell^6 K_1\right).$$

Finally, applying the second assertion in Lemma 4.1 gives the following expression for the asymptotic average acceptance rate

$$a(\ell) = \lim_{n\to\infty} \mathbb{E}[\alpha(\mathbf{x}, \mathbf{y})] = \Phi\left(\frac{\ell^3 K_2 + \ell K_3}{\sqrt{K_1}}\right) + \exp\left(\ell^6 (K_2 + K_1/2) + \ell^4 K_3\right) \Phi\left(-\frac{\ell^3 (K_1 + K_2) + \ell K_3}{\sqrt{K_1}}\right). \quad (4.5)$$

4.2. Optimal scaling. Following [3] we use the maximum square-jump distance S, also known as the first-order efficiency [29], as the criterion for efficiency of Markov chains. By definition, the square-jump distance for a component x_i of the current state $\mathbf{x}$ is specified by

$$S = \mathbb{E}\left[(x_i - x_i^*)^2\right],$$

where $\mathbf{x}^* = \mathbf{y}$ with probability $\alpha(\mathbf{x},\mathbf{y})$. We can rewrite S as

$$S = \mathbb{E}\left[(x_i - y_i)^2 \alpha(\mathbf{x},\mathbf{y})\right] = \mathbb{E}\left[(x_i - y_i)^2 1 \wedge \exp(T(\sigma))\right].$$

Let us define $T^i(\sigma)$ to be equal to $T(\sigma)$ subtracting the ith summands from C_j for $j = 1,\ldots,7$. We have

$$T(\sigma) - T^i(\sigma) = \sum_{j=3}^{6} \sigma^j C_{i,j}(\sigma) + \sigma^7 C_{i,7}(\sigma^*),$$

which clearly converges to zero in $L^2(\pi(\boldsymbol{x}) \otimes \mathcal{N}(0,I))$ as $n \to \infty$. This result together with the Lipchitz continuity of $1 \wedge \exp(\cdot)$ [28] and Cauchy-Schwarz inequality shows that

$$\lim_{n\to\infty} n^{1/3}\mathbb{E}\left[(x_i - y_i)^2 \left(1 \wedge \exp(T(\sigma)) - 1 \wedge \exp\left(T^i(\sigma)\right)\right)\right] = 0. \tag{4.6}$$

Next, we decompose $n^{1/3}S$ as follows

$$n^{1/3}S = \underbrace{n^{1/3}\mathbb{E}\left[(x_i - y_i)^2 1 \wedge \exp\left(T^i(\sigma)\right)\right]}_{S_1} + \underbrace{n^{1/3}\mathbb{E}\left[(x_i - y_i)^2 \left(1 \wedge \exp(T(\sigma)) - 1 \wedge \exp\left(T^i(\sigma)\right)\right)\right]}_{S_2 \to 0 \text{ by } (4.6)}.$$

Then,

$$\lim_{n\to\infty} S_1 = \ell^2 \lim_{n\to\infty} \mathbb{E}_{x_i}\left[-\frac{\lambda_i^2}{g^2(x_i/\lambda_i)}\right] \mathbb{E}\left[1 \wedge \exp\left(T^i(\sigma)\right)\right] = \ell^2 \lim_{n\to\infty} \mathbb{E}_{x_i}\left[-\frac{\lambda_i^2}{g^2(x_i/\lambda_i)}\right] a(\ell),$$

due to (4.6) and the definition of $a(\ell)$ in (4.5). Therefore, we arrive at

$$S = \mathbb{E}_{x_i}\left[-\frac{\lambda_i^2}{g^2(x_i/\lambda_i)}\right] \ell^2 a(\ell)\, n^{-1/3} + o\left(n^{-1/3}\right). \tag{4.7}$$

Let us now summarize the above analysis.

THEOREM 4.3. *Let the proposal variance be $\sigma^2 = \ell^2 n^{-1/3}$ and assume that C_4 converges to $\ell^4 K_3$ in $L^2(\pi(\boldsymbol{x}) \otimes \mathcal{N}(0,I))$, the asymptotic average acceptance rate $a(\ell)$ is bounded away from zero and is given by (4.5). Furthermore, the square-jump distance S scales like (4.7) as $n \to \infty$, and in particular S is maximized if $\ell^2 a(\ell)$ attains its maximum.*

REMARK 4.4. *Ideally, one first seeks ℓ^* to maximize $\ell^2 a(\ell)$, then obtain the optimal acceptance rate as $a(\ell^*)$. Unlike the theories for RWMH [13, 28] and MALA [29], the dependence of the average acceptance rate (4.5) on four parameters ℓ, K_1, K_2 and K_3 prevents us from doing so. Our numerical results, however, suggest that ℓ be of order 1, as we shall show.*

Clearly, when f is a Gaussian our sSN algorithm becomes a preconditioned MALA with the following preconditioned matrix

$$\mathbf{A} = \operatorname{diag}\left(\lambda_1^2,\ldots,\lambda_n^2\right). \tag{4.8}$$

In this case, $2K_2 + K_1 = 0$ holds and $K_3 = 0$. Hence, the asymptotic average acceptance rate is 0.574 and the efficiency is the same as that of the i.i.d. case ($\lambda_i = 1$). In other words, the inefficiency factor, see [18] for example, is unity even though the target (4.1) is heterogeneous. It is the introduction of the inverse of Hessian as the preconditioned matrix that places different components on the same scale as if the target were an i.i.d. distribution. It should be further pointed out that, in practice, however, it would not be possible to determine the variances λ_i^2, for $i = 1, \ldots, n$, exactly due to the complexity of the target distribution $\pi(\boldsymbol{x})$. Alternatively, one can estimate them using a pilot MCMC run [4], but it could be costly. On the contrary, our choice of the preconditioner $\mathbf{A}$ in (2.2) explores the structure of the underlying $\pi(\boldsymbol{x})$ and implicitly computes the hidden scalings λ_i^2 exactly in an automatic manner. In particular, if the component variance λ_i^2 of the target density scales as $\lambda_i^2 = i^{-2\kappa}$, for some $\kappa \geq 0$, then the optimal scaling for MALA would be $\sigma^2 = \ell^2 n^{-2\kappa-1/3}$ [2, 3], while it is always $\sigma^2 = \ell^2 n^{-1/3}$ for sSN no matter what λ_i^2 is. Therefore, sSN is more effective than MALA.

One can remove the assumption on the convergence of C_4 in $L^2(\pi(\boldsymbol{x}) \otimes \mathcal{N}(0, I))$ by taking smaller proposal variance, e.g., $\sigma^2 = \ell^2 n^{-1/2}$. In this case, one can show that C_3, C_5, C_6 and C_7 converge to zero in $L^2(\pi(\boldsymbol{x}) \otimes \mathcal{N}(0, I))$, while C_4 converges to $\ell^4 \mathbb{E}[C_{\cdot,4}]$ in $L^2(\pi(\boldsymbol{x}) \otimes \mathcal{N}(0, I))$. As a result, $T(\sigma)$ converges to $\mathcal{N}\left(\ell^4 \mathbb{E}[C_{\cdot,4}], 0\right)$. This, together with the second assertion in Lemma 4.1, implies

$$a(\ell) = \begin{cases} 1 & \text{if } \mathbb{E}[C_{\cdot,4}] > 0 \\ \exp\left(\ell^4 \mathbb{E}[C_{\cdot,4}]\right) & \text{otherwise} \end{cases},$$

which is clearly bounded away from zero. Nevertheless, smaller proposal variance means less efficient algorithm as argued above.

4.3. Scaling limits in transient phase and for non-product targets. We have assumed that the MCMC chain ideally starts at stationarity and then studied the proposal variance (scaling) as the dimension n approaches infinity. All the above results are therefore only valid for chains after the burn-in period. A question immediately arise is what the scaling limit is in the transient phase. An attempt to answer this question has been carried out in [10] for standard Gaussian target $\pi(\boldsymbol{x})$, in which the authors show that the scaling limit for RWMH is $\sigma^2 = \mathcal{O}\left(n^{-1}\right)$ while it is $\sigma^2 = \mathcal{O}\left(n^{-1/2}\right)$ for MALA. Consequently, the MALA proposal variance must be smaller in the transient phase as compared to that in the stationary phase; otherwise, the chain would take a long time to escape the transient period.

The sSN method, when applied to standard Gaussian targets, is identical to the MALA, and hence all the theoretical results for MALA in [10] are carried over for sSN. In particular, as we shall demonstrate numerically, it is necessary to use small proposal variance $\sigma^2 = \mathcal{O}\left(n^{-1/2}\right)$ in the transient phase and then switch to larger one $\sigma^2 = \mathcal{O}\left(n^{-1/3}\right)$ in stationarity for efficiency.

Our interest is on Bayesian inversion problems in which the target density is the posterior obeying the following change of measure formula

$$\frac{d\pi(\boldsymbol{x})}{d\pi_0(\boldsymbol{x})} \propto \exp\left(-\Phi(\boldsymbol{x})\right),$$

where $\pi_0(\boldsymbol{x})$ is the prior density having the product structure (4.1), and $\exp\left(-\Phi(\boldsymbol{x})\right)$ the likelihood. Following [2, 3], let us consider a variant of sSN for $\pi(\boldsymbol{x})$ with the

following proposal

$$\boldsymbol{y} = \boldsymbol{x} + \frac{\sigma^2}{2}\boldsymbol{A}\nabla \log\left(\pi_0\left(\boldsymbol{x}\right)\right) + \sigma\mathcal{N}\left(0, \boldsymbol{A}\right), \tag{4.9}$$

with $\boldsymbol{A} = \boldsymbol{A} = \text{diag}\left[\lambda_1^2, \ldots, \lambda_n^2\right]$ being the Hessian of $-\log\left(\pi_0\left(\boldsymbol{x}\right)\right)$. Note that we have used the gradient and the Hessian of the logarithm of the prior instead of the posterior to define the sSN. With this simplification, one can show that the scaling limit is still $\sigma^2 = \ell^2 n^{-1/3}$ assuming that $\Phi\left(\boldsymbol{x}\right)$ is uniformly bounded from above [2,3].

5. Numerical experiments. We compare the simplied manifold MALA [14] (abbreviated in this paper as mMALA), the Hessian-based Metropolis-Hasting (HMH) [26], the un-globalized[1] stochastic Newton method (USN) [20], and the sSN method for several examples. Choosing a reliable comparison criteria for different MCMC methods is not a trivial task. Indeed, one can use either of 13 convergence diagnostics reviewed in [11], but none of them assure with certainty that the Markov chain is representative of the stationary distribution $\pi\left(\boldsymbol{x}\right)$. For simplicity, we run each algorithm until it converges (by inspection), and then follow [14] to use the effective sample size (ESS) as the comparison criteria.

For completeness, let us recall the proposal of all methods here. The mMALA proposal is given by

$$\boldsymbol{y}_{\text{mMALA}} = \boldsymbol{x} + \frac{\sigma^2}{2}\boldsymbol{A}\left(\boldsymbol{x}\right)\nabla \log\left(\pi\left(\boldsymbol{x}\right)\right) + \sigma\mathcal{N}\left(0, \boldsymbol{A}\left(\boldsymbol{x}\right)\right),$$

where $\boldsymbol{A}\left(\boldsymbol{x}\right)$ is the expected Fisher information matrix at $\boldsymbol{x}$. The HMH proposal is defined as

$$\boldsymbol{y}_{\text{HMH}} = \boldsymbol{x} + \gamma\boldsymbol{H}^{-1}\left(\boldsymbol{x}\right)\nabla \log\left(\pi\left(\boldsymbol{x}\right)\right) + \mathcal{N}\left(0, \boldsymbol{H}^{-1}\left(\boldsymbol{x}\right)\right),$$

where $\boldsymbol{H}\left(\boldsymbol{x}\right)$ is the Hessian matrix of $-\log\left(\pi\left(\boldsymbol{x}\right)\right)$ at $\boldsymbol{x}$, and γ is the learning rate, a uniform random number with values in $[0, 1]$. The USN algorithm, on the other hand, specifies the proposal as

$$\boldsymbol{y}_{\text{USN}} = \boldsymbol{x} + \boldsymbol{H}^{-1}\left(\boldsymbol{x}\right)\nabla \log\left(\pi\left(\boldsymbol{x}\right)\right) + \mathcal{N}\left(0, \boldsymbol{H}^{-1}\left(\boldsymbol{x}\right)\right),$$

where $\boldsymbol{H}\left(\boldsymbol{x}\right)$ is the Hessian matrix of $-\log\left(\pi\left(\boldsymbol{x}\right)\right)$ at $\boldsymbol{x}$. Finally, our sSN method uses the following proposal

$$\boldsymbol{y}_{\text{sSN}} = \boldsymbol{x} + \frac{\sigma^2}{2}\boldsymbol{A}\nabla \log\left(\pi\left(\boldsymbol{x}\right)\right) + \sigma\mathcal{N}\left(0, \boldsymbol{A}\right),$$

where $\boldsymbol{A}$ is the inverse of the Hessian matrix $\boldsymbol{H} = -\nabla^2 \log\left(\pi\left(\boldsymbol{x}\right)\right)$ at $\boldsymbol{x}$. One of the key differences between sSN and the others is that $\boldsymbol{A}$, though a function of the current state $\boldsymbol{x}$, is freezed during the computation of the acceptance rate. That is, in the sSN approach we use the same Hessian matrix $\boldsymbol{H}\left(\boldsymbol{x}\right)$ to compute the probability of going from $\boldsymbol{x}$ to $\boldsymbol{y}$, namely $q(\boldsymbol{x}, \boldsymbol{y})$, and from $\boldsymbol{y}$ to $\boldsymbol{x}$, namely $q\left(\boldsymbol{y}, \boldsymbol{x}\right)$, whereas either the Hessian $\boldsymbol{H}$ or the expected Fisher information matrix $\boldsymbol{A}$ is recomputed at $\boldsymbol{y}$ for $q\left(\boldsymbol{y}, \boldsymbol{x}\right)$ in other methods. To further demonstrate the advantage of freezing the Hessian in the sSN method, we consider its natural variant, to be called sSNm,

[1] One of the ongoing researches [19] is to globalize the stochastic Newton that used in [20].

in which the Hessian is recomputed at $\boldsymbol{y}$ for $q(\boldsymbol{y}, \boldsymbol{x})$. The sSNm uses the following proposal

$$\boldsymbol{y}_{\text{sSNm}} = \boldsymbol{x} + \frac{\sigma^2}{2}\boldsymbol{H}^{-1}(\boldsymbol{x})\nabla \log(\pi(\boldsymbol{x})) + \sigma\mathcal{N}\left(0, \boldsymbol{H}^{-1}(\boldsymbol{x})\right), \tag{5.1}$$

with $\boldsymbol{H}(\boldsymbol{x}) = -\nabla^2 \log(\pi(\boldsymbol{x}))$. If one performs a similar optimal scaling analysis using the same technique as in Section 4, he will find that the optimal scaling in this case is $\sigma^2 = \ell^2 n^{-1}$ since only $C_0 = 0$ in (4.3). So freezing the Hessian during the Metropolized step in the sSN approach is both theoretically and computationally advantageous.

Note that we choose to compare the proposed sSN approach with mMALA, HMH, USN, and sSNm because they are similar in the sense that all ignore the term involving the change of curvature in (1.5). One can also view this comparison as a comparative study of five different local Hessian-aware preconditioned Langevin methods.

5.1. Verifying the sSN optimal scaling for i.i.d. targets. Before comparing sSN with other methods, let us examine the theoretical prediction on the stationary optimal scaling $\sigma^2 = \ell^2 n^{-1/3}$ for standard normal distributions as n increases. For standard normal distribution, the optimal scaling for sSN coincides with that of MALA, namely, $\sigma^2 = 1.65^2 n^{-1/3}$ and the optimal acceptance rate is about 0.574. Here, $\ell = 1.65$ is the stationary value for standard Gaussian target computed from the MALA optimal scaling theory [29]. We consider two cases; Case I with proposal variance of $\sigma^2 = 1$ and Case II with proposal variance of $\sigma^2 = 1.65^2 n^{-1/3}$. For both cases and for all dimensions considered, we run the sSN MCMC ten times each with 5000 samples starting at stationarity and present the average number of accepted proposals over ten runs in Table 5.1. As can be observed, Case I with fixed proposal variance has the acceptance rate converging to zero as the dimension n increases, while Case II with the optimal scaling of $\sigma^2 = 1.65^2 n^{-1/3}$ has the acceptance rate converging to $2887/5000 \approx 0.574$ as predicted by the theory.

TABLE 5.1
sSN accepted proposals as the dimension n increases for standard normal distributions.

		dimension n					
		1	10	100	200	500	100000
Case	I ($\sigma^2 = 1$)	4614	3494	1075	397	21	0
	II ($\sigma^2 = 1.65^2 n^{-1/3}$)	3361	2906	2896	2884	2863	2887

We next consider a non-Gaussian target, but still i.i.d. , given by

$$g(x_i) = \begin{cases} -\frac{1}{2}x_i^2 - 0.1\exp\left(-\frac{1}{2x_i^2}\right) & \text{if } x_i \geq 0 \\ -\frac{1}{2}x_i^2 & \text{if } x_i < 0 \end{cases}. \tag{5.2}$$

For this example, we shall compare the acceptance rate of all Hessian-aware methods discussed above. For sSN, we respect our theory by taking $\sigma^2 = 1.65^2 n^{1/3}$, where ℓ is again taken to be the stationary value computed for standard Gaussian target, namely $\ell = 1.65$. Note that mMALA coincides with sSN for this example. For sSNm, we also take $\sigma^2 = 1.65^2 n^{1/3}$. For each method, we run the corresponding MCMC simulation 10 times each with 5000 samples (excluding 1000 "burn-ins"), and then compute the average number of accepted proposals over ten runs in Table 5.2. As can be seen, the acceptance rate is bounded away from zero for sSN while it converges

to zero as the dimension n increases for sSNm and USN. In fact, using sSN, the stationary constant for standard Gaussian target $\ell = 1.65$ is quite conservative for target (5.2) since the acceptance rate increases as the dimension grows. It is surprising that the average acceptance rate for HMH seems to converge to 0.37, and this begs for future investigation on the HMH method. Again, the only difference between sSN and sSNm is that sSN freezes the Hessian evaluated at the current state $\boldsymbol{x}$ while sSNm re-computes the Hessian at the proposal $\boldsymbol{y}$. The numerical results agree with the theory that freezing the Hessian during the Metropolized step is advantageous both theoretically and numerically.

TABLE 5.2
Accepted proposals for different Hessian-aware methods as the dimension n increases for non-Gaussian target distribution (5.2).

		dimension n					
		100	500	1000	2000	10000	100000
Methods	sSN	2979	3237	3449	3752	4743	5000
	sSNm	1970	462	85	6	0	0
	USN	1837	212	23	0	0	0
	HMH	2048	1889	1871	1864	1878	1850

5.2. Bayesian logistic regression. In this section, we study the Bayesian logistic regression example using five data sets—Pima Indian, Australian credit, Germann credit, Heart, and Ripley—considered in Section 7 of [14]. The target density is given by

$$\pi(\boldsymbol{x}) \propto \exp\left(\mathbf{f}^T\mathbf{t} - \sum_{i=1}^{n} \log\left(1+\exp\left(f_i\right)\right) - \frac{1}{2\alpha^2}\|\boldsymbol{x}\|^2\right),$$

where $\alpha = 100$, $\mathbf{f} = \mathbf{W}\boldsymbol{x}$; $\mathbf{W} \in \mathbb{R}^{n\times n}$ and $\mathbf{t} \in \mathbb{R}^n$ take different values for different data sets.

For all data sets, we perform 10 runs, each of which is with 10000 MCMC iterations and with different random initialization. We then discard the first 5000 iterations as the burn-ins, and compute the minimum, mean, and maximum ESS. Let us define the effective time to generate a sample, time_{eff}, as the ratio of the minimum ESS to the overall simulation time measured by Matlab's `tic-toc` command.

We choose the scaling for mMALA and sSN as follows. For mMALA, we simply take $\sigma^2 = 1$ as suggested in [14]. Our asymptotic theoretical results in Section 4 suggests $\sigma^2 = \ell^2 n^{-1/3}$ for high dimensional problems. For the Bayesian logistic regression problems considered here, the highest dimension is 24 while the smallest one is 2. We therefore choose ℓ such that $\sigma^2 = 1$ to be fair with other methods, and this amounts to taking $\ell^2 = 2.88$ for $n = 24$ and $\ell^2 = 1.26$ for $n = 2$.

The results are shown in Tables 5.3–5.7. As can be observed, the USN method seems to least efficient for all data sets except for the Pima. Recall that USN sampling is exact if the target is Gaussian. On the one hand, this suggests that the target $\pi(\boldsymbol{x})$ be close to Gaussian so that USN is the most efficient. On the other hand, the results imply that USN is not recommended for target that is far away from Gaussian. The sSN method seems to be most efficient in general. In addition, sSN method in general takes less time than other methods since we compute the Hessian and its inverse only for accepted proposal $\boldsymbol{y}$.

TABLE 5.3
Comparison of sSN, mMALA, sSNm, USN, and HMH on the Bayesian logistic regression example using Ripley data set (n = 2).

Method	Time (s)	ESS (min, mean, max)	time_{eff}	speed
sSN	2.1147	(372, 656, 904)	0.0057	4.21
mMALA	2.057	(265, 386, 504)	0.0078	3.08
sSNm	2.4154	(239, 361, 477)	0.0101	2.38
HMH	2.325	(238, 318, 401)	0.0098	2.45
USN	2.3986	(100, 180, 280)	0.024	1.0

TABLE 5.4
Comparison of sSN, mMALA, sSNm, USN, and HMH on the Bayesian logistic regression example using Pima data set (n = 7).

Method	Time (s)	ESS (min, mean, max)	time_{eff}	speed
sSN	3.2733	(1233, 1387, 1537)	0.0027	1.41
mMALA	3.1841	(1008, 1155, 1280)	0.0032	1.19
sSNm	3.8727	(1010, 1179, 1330)	0.0038	1.0
HMH	3.5082	(979, 1118, 1265)	0.0036	1.56
USN	3.6069	(1497, 1909, 2274)	0.0024	1.58

TABLE 5.5
Comparison of sSN, mMALA, sSNm, USN, and HMH on the Bayesian logistic regression example using Heart data set (n = 13).

Method	Time (s)	ESS (min, mean, max)	time_{eff}	speed
sSN	2.7742	(930, 1128, 1305)	0.003	13.4
mMALA	3.2326	(368, 477, 590)	0.0088	4.57
sSNm	3.382	(352, 471, 587)	0.0096	4.19
HMH	3.1031	(257, 365, 461)	0.0121	3.32
USN	3.2947	(82, 196, 350)	0.0402	1.0

TABLE 5.6
Comparison of sSN, mMALA, sSNm, USN, and HMH on the Bayesian logistic regression example using Australian data set (n = 14).

Method	Time (s)	ESS (min, mean, max)	time_{eff}	speed
sSN	3.736	(787, 1011, 1194)	0.0047	3.28
mMALA	4.3378	(474, 596, 724)	0.0092	1.67
sSNm	4.789	(464, 612, 757)	0.0103	1.5
HMH	4.5858	(384, 506, 633)	0.0111	1.39
USN	4.5599	(297, 506, 744)	0.0154	1.0

In order to access the level of mixing and correlation of each Markov chain we study the typical trace plot of the first component x_1 of the random vector $\boldsymbol{x}$ and the corresponding average autocorrelation function (ACF) over 10 runs for each of the method. The results for other components show similar behavior and hence omitted. We present the trace plot and average ACF for the German data—the highest dimension case—in Figure 5.1. As can be seen, sSN chain seems to mix better than the others and its samples are least correlated. Chains generated by mMALA

TABLE 5.7
Comparison of sSN, mMALA, sSNm, USN, and HMH on the Bayesian logistic regression example using German data set (n = 24).

Method	Time (s)	ESS (min, mean, max)	time_{eff}	speed
sSN	4.7812	(669, 908, 1085)	0.0071	3.62
mMALA	7.7772	(464, 619, 752)	0.0168	1.53
sSNm	6.5261	(428, 590, 742)	0.0152	1.69
HMH	6.217	(342, 507, 636)	0.0182	1.41
USN	6.3339	(246, 542, 826)	0.0257	1.0

method are generally better than those of USN and HMH methods, while HMH and USN chains seem to be comparable to each other. These observations are consistent with the above comparison using ESS.

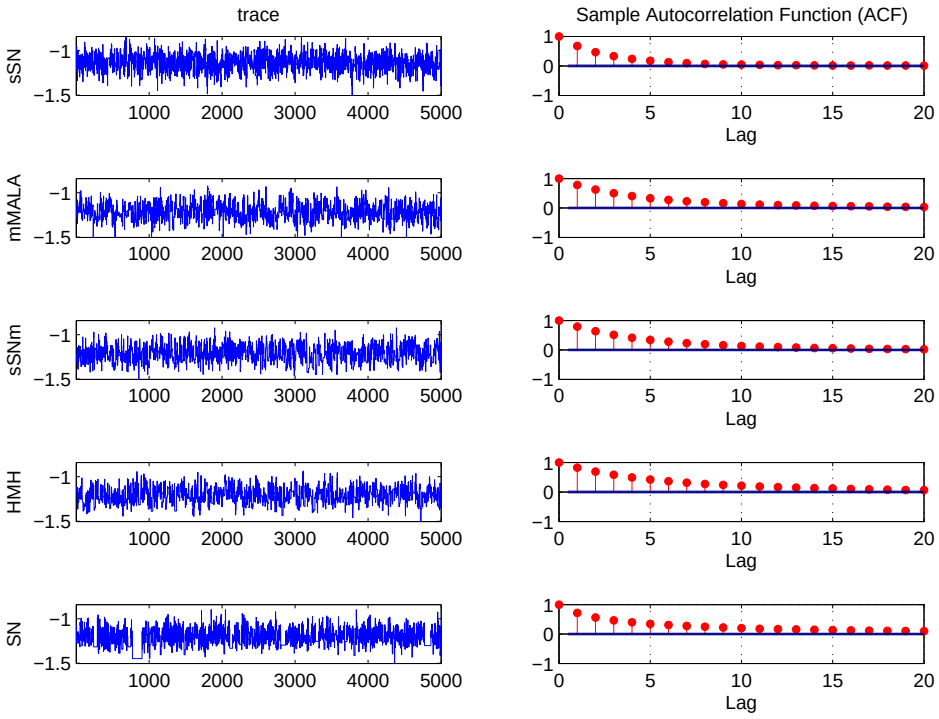

FIG. 5.1. *Typical trace plot of the first random variable x_1 and the average autocorrelation function using German data set (n = 24) for sSN, mMALA, sSNm, USN, and HMH methods.*

5.3. Log-Gaussian Cox point process. Our next example is the log-Gaussian Cox point process studied in [23] and then used in [14] to compare various MCMC methods. We use the same data and Matlab code for the simplified manifold MALA as provided in [14]. In particular we would like to make inference on the $n = 4096$ latent variables $\boldsymbol{x}$ from the following target density

$$\pi(\boldsymbol{x}) \propto \exp\left(\mathbf{t}^T\boldsymbol{x} - m\exp\left(\boldsymbol{x}^T\mathbf{1}\right) - (\boldsymbol{x}-\mu\mathbf{1})^T\Sigma^{-1}(\boldsymbol{x}-\mu\mathbf{1})\right),$$

where $\Sigma_{ij} = \gamma^2 \exp\left(-\delta\left(i,i',j,j'\right)/(64\beta)\right)$, $\delta\left(i,i',j,j'\right) = \sqrt{(i-i')^2+(j-j')^2}$, $m = 1/4096$, $\beta = 1/33$, $\gamma^2 = 1.91$, $\mu = \log(196) - \gamma^2/2$, and $\mathbf{1} = (1,\ldots,1)^T$. For all Markov chains considered in this section, the starting state is $\boldsymbol{x}_0 = \mu\mathbf{1}$, unless otherwise stated.

We first attemp to run USN and HMH methods on $\pi(\boldsymbol{x})$, but all the proposals rejected. A natural idea borrowed from optimization literature is to truncate the Newton step if the target is locally very different from the Gaussian. Doing so would avoid the risk of being trapped in regions of low probability density due to full Newton steps [26]. Unfortunately, no matter how small the truncated Newton step is—in fact up to machine zero step size is used—both USN and HMH reject all the proposals. This implies that straightforward application of optimization techniques to MCMC simulations may not be successful. The reason is that the proposal consists of two terms. The deterministic term $\boldsymbol{H}^{-1}(\boldsymbol{x})\nabla\log(\pi(\boldsymbol{x}))$ attempts to drift the proposal to region of high probability density, which is most likely the case if the target is locally close to Gaussian. The stochastic term $\mathcal{N}\left(0,\boldsymbol{H}^{-1}(\boldsymbol{x})\right)$ draws Gaussian random variables described by the covariance matrix $\boldsymbol{H}^{-1}(\boldsymbol{x})$. Consequently, the proposal is rejected if either of the terms is unreasonable. The truncating strategy adjusts the former to make it reasonable, but the latter needs to be adjusted as well if the target is far from Gaussian. With this in mind and following the spirit of $mMALA$ and sSN methods, we consider the following modification of the stochastic Newton (and hence HMH), abbrebriated as USNm,

$$\boldsymbol{y}_{\mathrm{USNm}} = \boldsymbol{x} + \sigma^2\boldsymbol{H}^{-1}(\boldsymbol{x})\nabla\log(\pi(\boldsymbol{x})) + \mathcal{N}\left(0,\sigma^2\boldsymbol{H}^{-1}(\boldsymbol{x})\right). \tag{5.3}$$

This seems to be sensible since if one believes that the deterministic Newton step should be cut down due to non-Gaussianity of the target, the variance of the Gaussian term should be reduced too. The modification can be also viewed as the Euler-Maruyama discretization, with step size $\Delta t = \sigma^2$, of the following stochastic differential equation

$$d\boldsymbol{x}(t) = \boldsymbol{H}^{-1}(\boldsymbol{x})\nabla\log(\pi(\boldsymbol{x}))\,dt + \sqrt{\boldsymbol{H}^{-1}(\boldsymbol{x})}d\boldsymbol{W}(t),$$

for which it is not clear what the invariant density is even when $\boldsymbol{H}^{-1}(\boldsymbol{x})$ is a constant matrix. On the other hand, it should be pointed out that USNm is almost identical to sSNm except for the unity factor in the drift term instead of $\frac{1}{2}$. Nevertheless, when $\boldsymbol{H}^{-1}(\boldsymbol{x})$ is constant the stochastic differential equation corresponding to the sSNm approach has $\pi(\boldsymbol{x})$ as its invariant distribution; this seems to make sSNm chains better than USN chains, as the following numerical results show.

Note that if one performs a similar optimal scaling analysis for the USN method given by (5.3) using the same technique as in Section 4, he will find that the optimal scaling in this case is $\sigma^2 = \ell^2 n^{-1}$ since only $C_0 = 0$ in (4.3). That is, USNm has a similar asymptotic optimal scaling to that of sSNm, and hence both are generally less efficient than sSN as the numerical results have showed. To further confirm this fact, we show similar results in Table 5.8 to those in Table 5.2 but now for USN and sSNm with $\sigma^2 = n^{-1}$ and $\sigma^2 = n^{-1/2}$. As can be seen, the limit scaling for both USN and sSNm is $\sigma^2 = \ell^2 n^{-1}$, agreeing with our theoretical prediction.

We now attempt to run USNm using the proposal variance of 0.07, a default value for mMALA provided in the Matlab code of [14]. We observe that USNm rejects all the proposal steps. On the other hand, if we use sSNm with $\sigma^2 = 0.07$, the average acceptance rate for each 50 MCMC steps is observed to be in between 0.46 and 0.8. This shows the importance of the factor $\frac{1}{2}$ in the drift term of sSNm (5.1) as opposed

TABLE 5.8
Accepted proposals for USN and sSNm as the dimension n increases for non-Gaussian target distribution (5.2).

		dimension n					
		100	500	1000	2000	10000	100000
USN	$\sigma^2 = n^{-1}$	2029	1676	1571	1533	1495	1555
	$\sigma^2 = n^{-1/2}$	426	13	2	0	0	0
sSNm	$\sigma^2 = n^{-1}$	2683	2133	2024	1948	1862	1972
	$\sigma^2 = n^{-1/2}$	2180	375	66	8	0	0

to unity in the USNm approach (5.3). That is, convergence to the target density on the differential level seems to facilitate the convergence on the discrete level via MH algorithm.

In Figure 5.2 we present, in the first row from left to right, the true latent field, the true process, and the observed data to infer the latent field, respectively. On the third row we show the sample posterior mean, process, and variance, from left to right respectively, for the sSN method with $\sigma^2 = 0.07$ (this is equivalent to taking $\ell^2 = 1.12$), denoted as sSN1, with 5000 samples after discarding 1000 samples as burn-ins. We show similar results for sSNm and mMALA in the fourth and the fifth row. As can be seen, the posterior mean and process are almost identical for sSN1, sSNm, and mMALA. The variances are large at the top right corner since the data is very sparse there, and the sSNm result seems to have less variability elsewhere compared to others.

It turns out that proposal variance of 0.07 is quite small for the sSN method since the acceptance rate for each 50 MCMC steps is greater than 0.8 (in fact is is greater than 0.9 most of the time). This suggests that we should use bigger proposal variance to explore the stationary distribution faster. We choose $\sigma^2 = 0.12$ to have the average acceptance rate between 0.6 and 0.8, this corresponds to $\ell^2 = 1.9$. We denote this particular combination as sSN2 and show the results in the second row of Figure 5.2. As can be observed, while the sample posterior mean and process are similar to other methods, there is less variability away from the top right corner compared to sSN1. Note that $\sigma^2 = 0.12$ is too big for sSNm and mMALA methods, and hence the resulting MCMC chains explore the invariant density very slowly due to very small acceptance rate. In particular, the sample posterior, process, and variance of sSNm are completely wrong while mMALA has large variance in patchy areas (not shown here due to limited space), and both are due to high correlations in the samples.

Next, we consider the transient behavior of the scaling limit. Recall the result in Section 4.3 that the scaling needs to be $\sigma^2 = \mathcal{O}\left(n^{-1/2}\right)$ in the transient phase. Theoretical results for standard Gaussian target in [10] show that the optimal scaling in stationarity is $\sigma^2 = \ell^2 4096^{-1/3} = 0.17$ where $\ell^2 = 1.65$, while it is $\sigma^2 = \ell^2 4096^{-1/2} = 0.03125$ where $\ell^2 = 2$ for transient phase. We use the following two starting states: 1) $\boldsymbol{x}_0 = \mu \mathbf{1}$, and 2) $\boldsymbol{x}_0 = \mathbf{a}$, where $\mathbf{a}$ solves the equation

$$t_i - \exp(a_i) - (a_i - \beta)/\gamma^2 = 0,$$

for which $\mathbf{a}$ is near the posterior mode.

For these two starting states, the mMALA (which becomes MALA in this case) rejects all the proposals using the stationary scaling $\sigma^2 = 0.17$. We observe the same problem for USN and sSNm. The sSN method, while rejecting all the proposals for

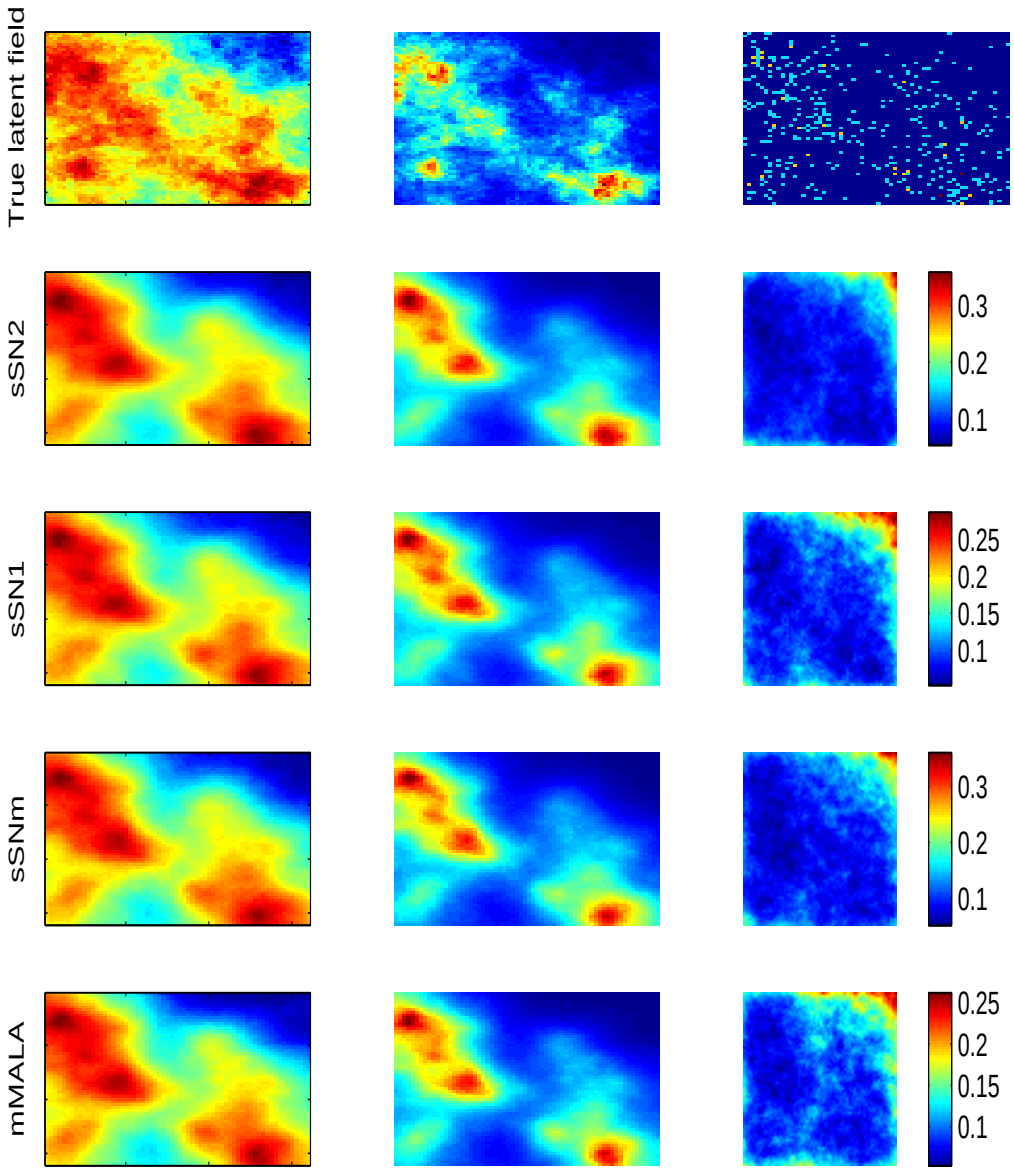

FIG. 5.2. *Log-Gaussian Cox process example with sSN1 ($\sigma^2 = 0.07$), sSN2 ($\sigma^2 = 0.12$), sSNm ($\sigma^2 = 0.07$), and mMALA ($\sigma^2 = 0.07$). Here, the sample posterior means, processes, and variances using 5000 MCMC samples after discarding 1000 samples as burn-ins.*

the second starting state, converges to stationarity after about 1400 iterations using the first starting state as shown on the left of Figure 5.3. As opposed to mMALA, USN, and sSNm, the sSN method is able to find the stationary region and then converges quickly, though staying put for a large number of initial iterations with a few unpredictable jumps (see [10] for a similar behavior when using MALA on a standard Gaussian target).

Now let us use the transient scaling $\sigma^2 = 0.03125$, the USN method again rejects

all the samples while the others converge very fast to equilibrium after less than 100 iterations using the first starting state. The second starting state is, however, more difficult for all the methods. Numerical results show that both USN and sSNm also fail to converge to the equilibrium though sSN and mMALA do converge. The convergence of sSN and mMALA chains is shown (in the middle and on the right) in Figure 5.3, for which the trace of $-\log(\pi(\boldsymbol{x}))$ is plotted after 100 iterations. As can be seen, the sSN chain seems to get to the stationarity faster. Both of the chains have the average acceptance rate about 0.96, which is expected since the transient scaling is too conservative for the stationary phase.

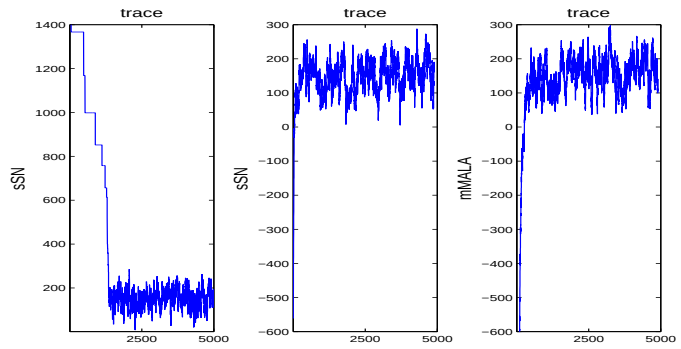

FIG. 5.3. *The trace of* $-\log(\pi(\boldsymbol{x}))$ *for the Log-Gaussian Cox process example using: on the left, sSN with* $\sigma^2 = 0.17$ *and the first starting state, in the middle, sSN with* $\sigma^2 = 0.03125$ *and the second starting state, and on the right, mMALA with* $\sigma^2 = 0.03125$ *and the second starting state.*

Note that the severe converging problem in transient period with starting point close to the posterior mode is not surprising since it has already predicted by the theory [10].

5.4. Inverse thermal fin problem. In this section, we consider a thermal fin problem adopted from [5, 25] with geometry given in Figure 5.4. The temperature u distributed on the thermal fin Ω is governed by the following partial differential equation

$$-\nabla \cdot (e^{\gamma} \nabla u) = 0 \qquad \text{in } \Omega, \tag{5.4a}$$

$$-e^{\gamma} \nabla u \cdot \mathbf{n} = Bi\, u \qquad \text{on } \partial\Omega \setminus \Gamma_{\text{Root}}, \tag{5.4b}$$

$$-e^{\gamma} \nabla u \cdot \mathbf{n} = -1 \qquad \text{on } \Gamma_{\text{Root}}, \tag{5.4c}$$

where Γ_{Root} is the root of the thermal fin, Bi the Biot number, γ the logarithm of the conductivity, and $\mathbf{n}$ the unit normal vector.

We discretize (5.4) using the first order standard conforming finite element method on the mesh in Figure 5.4 so that the discretized equation has the form

$$\mathcal{A}(\boldsymbol{\gamma})\, U = F, \tag{5.5}$$

where $U \in \mathbb{R}^{1333}$ is the vector of nodal values of u, $F \in \mathbb{R}^{1333}$ the right-hand side resulting from the boundary condition (5.4c), and $\mathcal{A}(\gamma) \in \mathbb{R}^{1333\times 1333}$ the stiffness matrix.

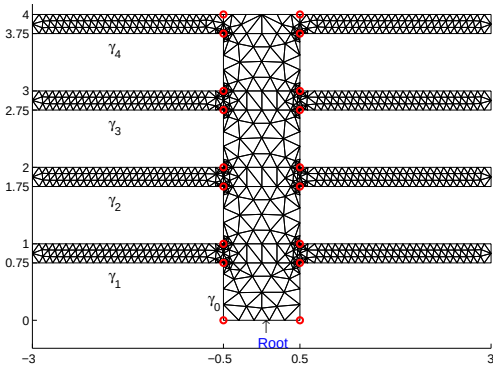

FIG. 5.4. *A finite element mesh of the thermal fin and its dimensions.*

Similar to [25], we set $\boldsymbol{\gamma} = (\gamma_0, \gamma_1, \gamma_2, \gamma_3, \gamma_4) \in \mathbb{R}^5$, where $\exp(\gamma_0)$ is the thermal conductivity of the base $\Omega_0 = [-0.5, 0.5] \times [0, 4]$ and $\exp(\gamma_i)$, $i = 1, \ldots, 4$, the thermal conductivities of the fins, respectively, as shown in Figure 5.4.

The temperature is measured at 18 observation locations x_i^{obs} specified by red circles in Figure 5.4 by the following additive noise model

$$u_i^{obs} = u(x_i^{obs}) + \eta_i, \quad , i = 1, \ldots, 18,$$

with $\eta_i \sim \mathcal{N}\left(0, 0.001^2\right)$ denoting the i.i.d. Gaussian noise. Thus, the likelihood model can be written as

$$\pi\left(\mathbf{u}^{obs}|\boldsymbol{\gamma}\right) = \mathcal{N}\left(\mathbf{u}^{obs}, 0.001^2\mathbf{I}_{18}\right),$$

with $\mathbf{I}_{18}$ denoting the 18×18 identity matrix. To avoid the inverse crime, we synthesize the observations $\mathbf{u}^{obs}$ with

$$\boldsymbol{\gamma} = (0, 0.25, 0.5, 0.75, 1) \tag{5.6}$$

on a mesh that is twice finer than that given in Figure 5.4.

We would like to solve the inverse problem of finding $\boldsymbol{\gamma}$ given $\mathbf{u}^{obs}$ using the Bayesian framework. To this end, we need to specify a prior distribution $\pi_{\text{prior}}(\boldsymbol{\gamma})$ for $\boldsymbol{\gamma}$, and we choose

$$\pi_{\text{prior}}(\boldsymbol{\gamma}) \propto \exp\left(-\frac{\beta}{2}\boldsymbol{\gamma}^T\boldsymbol{\gamma}\right),$$

with $\beta = 0.1$ as the regularization constant.

The Bayesian posterior solution is therefore given by

$$\pi_{\text{post}}\left(\boldsymbol{\gamma}|\mathbf{u}^{obs}\right) \propto \exp\left\{-\frac{1}{2 \times 0.001^2}\sum_{i=1}^{18}\left[u\left(x_i^{obs}\right) - u_i^{obs}\right]^2 - \frac{\beta}{2}\boldsymbol{\gamma}^T\boldsymbol{\gamma}\right\}, \tag{5.7}$$

where $u\left(x_i^{obs}\right)$ are nodal solutions extracted from the solution of (5.5). Using the notation in Section 4.3, we have

$$\Phi(\boldsymbol{\gamma}) = \frac{1}{2 \times 0.001^2}\sum_{i=1}^{18}\left[u\left(x_i^{obs}\right) - u_i^{obs}\right]^2.$$

Now, we attempt to apply the simplified sSN method (4.9) to the Bayesian posterior solution (5.7). Since this is a five-dimensional problem, it is not clear whether the asymptotic optimal scaling result applies or not. The initial state $\boldsymbol{\gamma}_0 = (0, 0, 0, 0, 0)$ is used. Let us take a conservative step $\sigma^2 = 2.23e-06$ so that the average acceptance rate is about 0.234, which is the best value for RWMH. We then run 105000 MCMC simulations (arbitrarily chosen) and discard the first 100000 runs as "burn-ins". We denote this experiment as SimsSN1. On the first row of Figure 5.5, we show the mean of $\boldsymbol{\gamma}$, the trace plot of $\boldsymbol{\gamma}$ (γ_1 to γ_5 from bottom to top), and the autocorrelation function for γ_5, respectively. As can be seen, the sample mean seems to be very accurate compared to the exact solution (5.6). However, it is not clear in the trace plot whether the MCMC chain has already converged. Moreover, the correlation length is large. Let us now take $\sigma^2 = 3.72e-07$ so that the average acceptance rate is about 0.573, which is the best value for the Langevin MCMC. We again discard the first

100000 runs as "burn-ins" and denote this experiment as SimsSN2, for which we obtain similar results on the second row of Figure 5.5. As can be seen, the results do not seem to be better than those of SimpsSN1. Note that this does not conflict with the discussion in Section 4.3. First, we have not yet been able to show the boundedness of $\Phi(\boldsymbol{\gamma})$, and hence the scaling limit $\sigma^2 = \ell^2 n^{-1/3}$ may not be valid. Second, even when it holds, the result applies only for stationary phase, and it is not clear in Figure 5.5 when the chains get to stationarity. This is a manifestation of the difficulty of inverse problems and care must be taken when using MCMC.

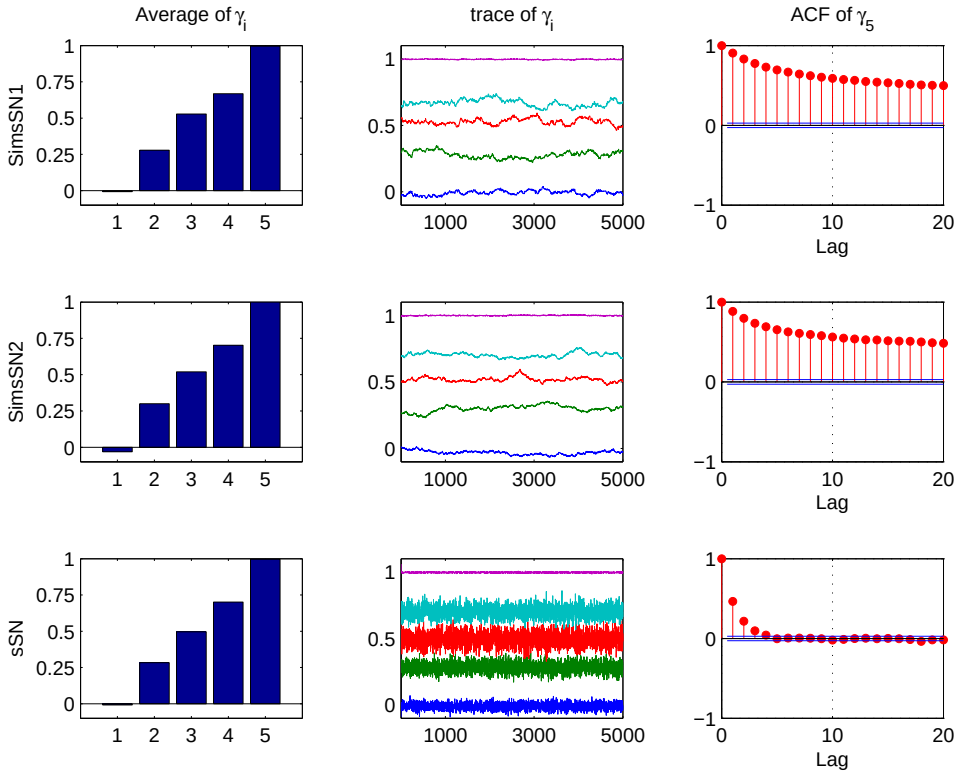

FIG. 5.5. *Inverse thermal fin example with SimpsSN1 ($\sigma^2 = 2.23e-06$), SimpsSN2 ($\sigma^2 = 3.7181e-07$), and sSN ($\sigma^2 = 1.17$). Here, we discard the first 100000 samples as burn-ins for both SimpsSN1 and SimpsSN2, but no burn-in for sSN.*

Since this is a low dimensional problem with 5 parameters, it is feasible for us to compute the Hessian of the posterior (5.7) exactly. Indeed, using an adjoint method we can compute the gradient $\nabla \log(\pi_{\text{post}})$ by solving an adjoint of forward equation (5.4). On the other hand, the action of the Hessian $\nabla^2 \log(\pi_{\text{post}})$ on a vector can be computed efficiently by solving a pair of incremental forward and incremental adjoint equations (see e.g. [6]). Therefore, computing the full Hessian amounts to acting $\nabla^2 \log(\pi_{\text{post}})$ on the 5×5 identity matrix. We take $\ell^2 = 2$, and hence $\sigma^2 = \ell^2 n^{-1/3} = 1.17$. The sample mean, trace plots, and ACF of γ_5 using a sSN chain with 5000 samples with zero burn-in are shown on the third row of Figure 5.5. The average acceptance rate in this case is about 0.74. As can be observed, the chain mixes very well and has correlation length about 4. This implies that using the actual gradient and Hessian of the posterior for the sSN method leads to MCMC chains with faster

convergence and excellent mixing.

We compute the sample variances and use them as uncertainty estimates associated with the sample mean and show them on the first row of Table 5.9. The uncertainty for each thermal conductivity seems to be small and is about the same order as the noise standard deviation. This is not surprising since we have more observations (18) than the number of unknown parameters (5). This is most likely an over-determined case, for which a linear theory in [38] shows that the posterior distribution converges to the Dirac delta distribution concentrated at the maximum a posterior point in the limit of no noise. Let us confirm this numerically for our nonlinear inverse problem by additionally taking the noise standard deviation to be $1.e-4$ and $1.e-5$. As can be observed in Table 5.9, the estimated uncertainties converges to zero as the noise level diminishes.

TABLE 5.9
Uncertainty estimates for the conductivity $\boldsymbol{\gamma}$.

		Thermal conductivity				
		γ_1	γ_2	γ_3	γ_4	γ_5
noise deviation	$1.e-3$	0.47e-3	1.12e-3	2.17e-3	1.55e-3	0.01e-3
	$1.e-4$	0.07e-4	0.12e-4	0.32e-4	0.17e-4	0.01e-4
	$1.e-5$	0.25e-5	0.04e-5	1.00e-5	0.17e-5	0.09e-5

6. Conclusions. Based on a connection between Euler-Maruyama discretization of the Langevin diffusion on Riemann manifold and gradient-based optimization techniques we have proposed a scaled stochastic Newton algorithm (sSN) algorithm for local Metropolis-Hastings Markov chain Monte Carlo simulation. The sSN proposal consists of a deterministic and a stochastic part. The former corresponds to a Newton step which attempts to drift the current state to region of higher probability, hence potentially increasing the acceptance probability. The latter is distributed by a Gaussian tailored to the local Hessian as the inverse covariance matrix. The proposal step is then corrected by the standard Metropolization to guarantee that the target density is the stationary distribution of the sSN chain.

The sSN method can be considered as an Euler-Maruyama discretization of the Langevin diffusion on Riemann manifold with piecewise constant Hessian of the negative logarithm of the target density as the metric tensor. Its distinct feature is that the Hessian evaluated at a current state is freezed during the Metropolization. This facilitates not only theoretical analysis but also computation as we have shown. We have provided a quite complete picture of the sSN method by studying its asymptotic convergence, geometric ergodicity, optimal scaling analysis in both stationary and transient phases, and an extension to Bayesian inverse problems. Numerical results on various examples have confirmed our theoretical findings. We also compare the sSN method against other Hessian-aware approaches, and the numerical results show that the sSN method in general outperforms the others in giving Markov chains with small burn-in and small correlation length. We have applied the sSN MCMC method to perform a simple uncertainty quantification on an inverse thermal problem with five parameters.

Nevertheless, there are a few issues that need to be addressed in future studies. On the theoretical side, even though the numerical results suggest that the constant ℓ in the optimal proposal scaling $\sigma^2 = \ell^2 n^{-1/3}$ be of order 1, we are not able to show this rigorously except for i.i.d. Gaussian targets. Extension to Bayesian inverse

problems using the actual gradient and Hessian of the posterior density remains an open question though numerical results support this idea. On the computational side, computing the full Hessian is infeasible for large scale problems such as inverse problems governed by PDEs, since it amounts to solving a pair incremental forward and adjoint PDEs n times. Even so, the Hessian may not be always positive definite. In this case, one can replace the negative spectrum with a threshold as suggested in [20]. Alternatively, one can approximate the Hessian of the data misfit $\Phi(\boldsymbol{x})$ using its Gauss-Newton approximation, which is always semi-positive. Since the Gauss-Newton Hessian is typically a compact operator [7, 8], one can use low rank approximation techniques to approximate it efficiently, and this is under investigation. Another way to deal with the indefiniteness of the Hessian is to approximate it using a trust region idea. Some work in this direction has been carried out [16,26], though a complete and effective method still requires further study. Note that using any approximation to the actual Hessian breaks down the theory and a new theory is necessary to establish (possibly new) optimal scaling for sSN chains.

REFERENCES

[1] Yves F. Atchadé, *An adaptive version for the Metropolis adjusted Langevin algorithm with a truncated drift*, Methodology and Computing in Applied Probability, 8 (2006), pp. 235–254.

[2] Alexandros Beskos, Gareth Roberts, and Andrew Stuart, *Optimal scalings for local Metropolis-Hastings chains on nonproduct targets in high dimensions*, The Annals of Applied Probability, 19 (2009), pp. 863–898.

[3] A. Beskos and A.M. Stuart, *MCMC methods for sampling function space*, in Invited Lectures: Sixth International Congress on Industrial and Applied Mathematics, ICIAM 2007, R. Jeltsch and G. Wanner, eds., European Mathematical Society, 2009, pp. 337–364.

[4] Stephen P. Brooks, *Markov chain Monte Carlo method and its application*, Journal of the Royal Statistical Society. Series D (The Statistician), 47 (1998), pp. 69–100.

[5] Tan Bui-Thanh, *Model-Constrained Optimization Methods for Reduction of Parameterized Large-Scale Systems*, PhD thesis, MIT, 2007.

[6] T. Bui-Thanh, C. Burstedde, O. Ghattas, J. Martin, G. Stadler, and L. C. Wilcox, *Extreme-scale UQ for Bayesian inverse problems governed by PDEs*, in SC12: Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis, 2012.

[7] Tan Bui-Thanh and Omar Ghattas, *Analysis of the Hessian for inverse scattering problems. Part I: Inverse shape scattering of acoustic waves*, Inverse Problems, 28 (2012), p. 055001.

[8] ———, *Analysis of the Hessian for inverse scattering problems. Part II: Inverse medium scattering of acoustic waves*, Inverse Problems, 28 (2012), p. 055002.

[9] Fergal P. Casey, Joshua J. Waterfall, Ryan N. Gutenkunst, Christopher R. Myers, and James P. Sethna, *Variational method for estimating the rate of convergence of Marko-chain Monte Carlo algorithms*, Phy. Rev. E., 78 (2008), p. 046704.

[10] Ole F. Christensen, Gareth O. Roberts, and Jeffrey S. Rosenthal, *Scaling limits for the transient phase of local Metropolis–Hastings algorithms*, Journal of Royal Statistical Society. Series B (Statistical Methodology), 67 (2005), pp. 253–268.

[11] Mary Kathryn Cowles and Bradley P. Carlin, *Markov chain Monte Carlo convergence diagnostics: A comparative review*, Journal of the American Statistical Association, 91 (1996), pp. 883–904.

[12] Rick Durret, *Probability: theory and examples*, Cambidge University Press, 2010.

[13] A Gelman, G. O. Roberts, and W. R. Gilks, *Efficient Metropolis jumping rules*, in Bayesian Statistics 5, J. M. Bernardo, J. O. Berger, A. P. Dawid, and A. F. M. Smith, eds., Oxford Univ Press, 1996, pp. 599–607.

[14] Mark Girolami and Ben Calderhead, *Riemann manifold Langevin and Hamiltonian Monte Carlo methods*, Journal of the Royal Statistical Society: Series B (Statistical Methodology), 73 (2011), pp. 123–214.

[15] W. K. Hastings, *Monte Carlo sampling methods using Markov chains and their applications*, Biometrika, 57 (1970), pp. 97–109.

[16] Edward Herbst, *Gradient and Hessian-based MCMC for DSGE models*, (2010). Unpublished

manuscript.

[17] Soren F. Jarner and Ernst Hansen, *Geometric ergodicity of Metropolis algorithms*, Stochastic Processes and their Applications, 85 (2000), pp. 341–361.

[18] Tristan Marshall and Gareth O. Roberts, *An adaptive approach to Langevin MCMC*, Statistics and Computing, (2011), pp. 1–17.

[19] James Martin, *Private communication*, (2012).

[20] James Martin, Lucas C. Wilcox, Carsten Burstedde, and Omar Ghattas, *A stochastic Newton MCMC method for large scale statistical inverse problems with application to seismic inversion*, SIAM Journal on Scientific Computing, (2012). To appear.

[21] Nicholas Metropolis, Arianna W. Rosenbluth, Marshall N. Rosenbluth, Augusta H. Teller, and Edward Teller, *Equation of state calculations by fast computing machines*, The Journal of Chemical Physics, 21 (1953), pp. 1087–1092.

[22] Sean Meyn and Richard L. Tweedie, *Markov chains and stochastic stability*, Springer-Verlag, London, 1993.

[23] Jesper Moller, Anne Randi Syversveen, and Rasmus Plenge Waagepetersen, *Log Gauss Cox processes*, Scandinavian Journal of Statistics, 25 (1998), pp. 451–482.

[24] Jorge Nocedal and Stephen J. Wright, *Numerical Optimization*, Springer Verlag, Berlin, Heidelberg, New York, second ed., 2006.

[25] MIT OpenCourseWare, *Numerical methods for partial differential equations*, 2003. http://ocw.mit.edu/OcwWeb/Aeronauticsand-Astronautics/16-920JNumerical-Methods-for-Partial-Differential-EqationsSpring2003/CourseHome/.

[26] Yuan Qi and Thomas P. Minka, *Hessian-based Markov chain Monte-Carlo algorithms*, in First Cape Cod Workshop on Monte Carlo Methods, Cape Cod, MA, USA, September 2002.

[27] Christian P. Robert and George Casella, *Monte Carlo Statistical Methods (Springer Texts in Statistics)*, Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2005.

[28] Gareth O. Roberts, A. Gelman, and W. R. Gilks, *Weak convergence and optimal scaling of random walk Metropolis algorithms*, The Annals of Applied Probability, 7 (1997), pp. 110–120.

[29] Gareth O. Roberts and Jeffrey S. Rosenthal, *Optimal scaling of discrete approximations to Langevin diffusions*, J. R. Statist. Soc. B, 60 (1997), pp. 255–268.

[30] Gareth O. Roberts and Jeffrey S. Rosenthal, *Optimal scaling for various metropolis-hastings algorithms*, Statistical Science, 16 (2001), pp. pp. 351–367.

[31] Gareth O. Roberts and Jeffrey S. Rosenthal, *General state space Markov chains and MCMC algorithms*, Probability Surveys, 1 (2004), pp. 20–71.

[32] Gareth O. Roberts and Osnat Stramer, *Langevin diffussions and Metropolis-Hastings algorithms*, Methodology and Computing in Applied Probability, 4 (2003), pp. 337–357.

[33] Gareth O. Roberts and Richard L. Tweedie, *Exponential convergence of Langevin distributions and their discrete approximations*, Bernoulli, 2 (1996), pp. 341–363.

[34] ———, *Geometric convergence and central limit theorems for multidimensional Hastings and Metropolis algorithms*, Biometrika, 83 (1996), pp. 95–110.

[35] Jeffrey Rosenthal, *Optimal proposal distributions and adaptive MCMC*, in Handbook of Markov chain Monte Carlo, Steve Brooks, Andrew Gelman, Galin L. Jones, and Xiao-Li Meng, eds., CRC Press, 2011, ch. 4, pp. 93–112.

[36] P. J. Rossky, J. D. Doll, and H. L. Friedman, *Brownian dynamics as smart Monte Carlo simulation*, J. Chem. Phys., 69 (1978), p. 4268.

[37] A.M. Stuart, P. Wiberg, and J. Voss, *Conditional path sampling of SDEs and the Langevin MCMC method*, Communications in Mathematical Sciences, 2 (2004), pp. 685–697.

[38] Andrew M. Stuart, *Inverse problems: A Bayesian perspective*, Acta Numerica, 19 (2010), pp. 451–559.

[39] Luke Tierney, *Markov chains for exploring posterior distributions*, The Annals of Statistics, 22 (1994), pp. 1701–1762.

[40] Cornelia Vacar, Jean-Francois Giovannelli, and Yannick Berthoumieu, *Langevin and Hessian with Fisher approximation stochastic sampling for parameter estimation of structured covariance*, IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), (2011), pp. 3964–3967.

[41] Yichuan Zhang and Charles Sutton, *Quasi-Newton methods for Markov chain Monte Carlo*, in Advances in Neural Information Processing Systems, J. Shawe-Taylor, R. S. Zemel, P. Bartlett, F. C. N. Pereira, and K. Q. Weinberger, eds., vol. 24, 2011.