Variance Reduction in Stochastic Particle-Optimization Sampling

Jianyi Zhang 1 Yang Zhao 2 Ruiyi Zhang 1 Lawrence Carin 1 Changyou Chen 2

Abstract
Stochastic particle-optimization sampling (SPOS) is a recently-developed scalable Bayesian sampling framework that unifies stochastic gradient MCMC (SG-MCMC) and Stein variational gradient descent (SVGD) algorithms based on Wasserstein gradient flows. With a rigorous non-asymptotic convergence theory developed recently, SPOS avoids the particle-collapsing pitfall of SVGD. Nevertheless, variance reduction in SPOS has never been studied. In this paper, we address this gap by presenting several variance-reduction techniques for SPOS. Specifically, we propose three variants of variance-reduced SPOS, called SAGA particle-optimization sampling (SAGA-POS), SVRG particle-optimization sampling (SVRG-POS) and a variant of SVRG-POS which avoids full gradient computations, denoted as SVRG-POS+. Importantly, we provide non-asymptotic convergence guarantees for these algorithms in terms of the 2-Wasserstein metric and analyze their complexities. The results show our algorithms yield better convergence rates than existing variance-reduced variants of stochastic Langevin dynamics, even though more space is required to store the particles in training. Our theory aligns well with experimental results on both synthetic and real datasets.

1. Introduction
Sampling has been an effective tool for approximate Bayesian inference, which has become increasingly important in modern machine learning. In the setting of big data, recent research has developed scalable Bayesian sampling algorithms such as stochastic gradient Markov Chain Monte Carlo (SG-MCMC) (Welling & Teh, 2011) and Stein variational gradient descent (SVGD) (Liu & Wang, 2016). These methods have facilitated important real-world applications and have achieved impressive results, such as topic modeling (Gan et al., 2015; Liu et al., 2016), matrix factorization (Chen et al., 2014; Ding et al., 2014; Simşekli et al., 2016), differential privacy (Wang et al., 2015; Li et al., 2017), Bayesian optimization (Springenberg et al., 2016), reinforcement learning (Haarnoja et al., 2018; Zhang et al., 2018a;b; 2019) and deep neural networks (Li et al., 2016). Generally speaking, these methods use gradient information of a target distribution to generate samples, leading to more effective algorithms compared to traditional sampling methods. Recently, (Chen et al., 2018) proposed a particle-optimization Bayesian sampling framework based on Wasserstein gradient flows, which unified SG-MCMC and SVGD in a new sampling framework called particle-optimization sampling (POS). Further, Zhang et al. discovered that SVGD has some unintended pitfalls, i.e., particles tend to collapse under some conditions. As a result, a remedy was proposed to inject random noise into SVGD update equations in the POS framework, leading to stochastic particle-optimization sampling (SPOS) algorithms (Zhang et al.). Remarkably, for the first time, non-asymptotic convergence theory was developed for SPOS (SVGD-type algorithms) in (Zhang et al.).

In order to deal with large-scale datasets, many gradient-based methods for optimization and sampling use stochastic gradients calculated on a mini-batch of a dataset, for computational feasibility. Unfortunately, this has the potential of adding extra variance into the algorithms, which may potentially degrade performance. Consequently, variance control has been an important and interesting direction of research. Efficient solutions such as SAGA (Defazio et al., 2014) and SVRG (Johnson & Zhang, 2013) were proposed to reduce variance in stochastic optimization. Subsequently, (Dubey et al., 2016) introduced these techniques in SG-MCMC for Bayesian sampling, which also has achieved great success in practice.

Since SPOS has enjoyed the best of both worlds by combining SG-MCMC and SVGD, it is of great value to further reduce its gradient variance. While both the algorithm and theory have been developed for SPOS, no work has been done to investigate its variance-reduction techniques. Compared with the research on SG-MCMC, where variance reduction has been well explored by recent work such
as (Dubey et al., 2016; Chatterji et al., 2018; Zou et al., 2018), it is much more challenging for SPOS to control the variance of stochastic gradients. This is because from a theoretical perspective, SPOS corresponds to nonlinear stochastic differential equations (SDE), where fewer existing mathematical tools can be applied for theoretical analysis. Furthermore, the fact that many particles are used in the algorithm makes it difficult to improve its performance by adding modifications to the way they interact with each other.

In this paper, we take the first attempt to study variance-reduction techniques in SPOS and develop corresponding convergence theory. We adopt recent ideas on variance reduction in SG-MCMC and stochastic-optimization algorithms, and propose three variance-reduced SPOS algorithms, denoted as SAGA particle-optimization sampling (SAGA-POS), SVRG particle-optimization sampling (SVRG-POS) and a variant of SVRG-POS without full-gradient computations, denoted as SVRG-POS*. For all these variants, we prove rigorous theoretical results on their non-asymptotic convergence rates in terms of 2-Wasserstein metrics. Importantly, our theoretical results demonstrate significant improvements in convergence rates over standard SPOS. Remarkably, when comparing our convergence rates with those of variance-reduced stochastic gradient Langevin dynamics (SGLD), our theory indicates faster convergence of variance-reduced SPOS when the number of particles is large enough. Our theoretical results are verified by a number of experiments on both synthetic and real datasets.

2. Preliminaries

2.1. Stochastic gradient MCMC

In Bayesian sampling, one aims at sampling from a posterior distribution $p(\theta | \mathbf{x}) \propto p(\mathbf{x} | \theta) p(\theta)$, where $\theta \in \mathbb{R}^d$ represents the model parameter, and $\mathbf{X} \triangleq \{ \mathbf{x}_j \}_{j=1}^N$ is the dataset. Let $p(\theta | \mathbf{X}) = (1/Z) \exp(-U(\theta))$, where

$$U(\theta) = - \log p(\mathbf{X} | \theta) - \log p(\theta)$$

$$\triangleq - \sum_{i=1}^N \log p(\mathbf{x}_i | \theta) - \log p(\theta)$$

is referred to as the potential energy function, and $Z$ is the normalizing constant. We further define the full gradient $F$ and individual gradient $F_j$ used in this paper:

$$F_j(\theta) \triangleq - \nabla_{\theta} \log p(\mathbf{x}_j | \theta) - \frac{1}{N} \nabla_{\theta} \log p(\theta)$$

$$F(\theta) \triangleq \nabla_{\theta} U(\theta) = \sum_{j=1}^N F_j(\theta)$$

We can define a stochastic differential equation, an instance of Itô diffusion, whose stationary distribution equals the target posterior distribution $p(\theta | \mathbf{X})$. For example, consider the following 1st-order Langevin dynamic:

$$d\theta_t = -\beta^{-1} F(\theta_t) dt + \sqrt{2\beta^{-1}} dW_t,$$

where $t$ is the time index, $W_t \in \mathbb{R}^d$ is $d$-dimensional Brownian motion, and $\beta$ a scaling factor. By the Fokker-Planck equation (Kolmogoroff, 1931; Risken, 1989), the stationary distribution of (1) equals $p(\theta | \mathbf{X})$.

SG-MCMC algorithms are discretized numerical approximations of Itô diffusions (1). To make algorithms efficient in a big-data setting, the computationally-expensive term $F$ is replaced with its unbiased stochastic approximation via a random subset of the dataset in each interation, e.g. $F$ can be approximated by a stochastic gradient:

$$G_k \triangleq \frac{N}{B} \sum_{j \in I_k} F_j(\theta_k) = -\nabla \log p(\theta_k) - \frac{N}{B} \sum_{j \in I_k} \nabla_{\theta} \log p(\mathbf{x}_j | \theta_k)$$

where $I_k$ is a random subset of $\{1, 2, \cdots, N\}$ with size $B$. The above definition of $G_k$ reflects the fact that we only have information from $B \ll N$ data points in each iteration. This is the source of the variance we seek to reduce. We also note that $G_k$ is used in standard SVGD and SPOS. As an example, SGLD is a numerical solution of (1), with update equation: $\theta_{k+1} = \theta_k - \beta^{-1} G_k \theta_k + \sqrt{2\beta^{-1}} \xi_k$, where $h$ means the step size and $\xi_k \sim \mathcal{N}(0, I)$.

2.2. Stein variational gradient descent

Different from SG-MCMC, SVGD initializes a set of particles, that are updated iteratively to approximate a posterior distribution. Specifically, we consider a set of particles $\{\theta^{(i)}\}_{i=1}^M$ drawn from some distribution $q$. SVGD tries to update these particles by doing gradient descent on the interactive particle system via

$$\theta^{(i)} \leftarrow \theta^{(i)} + h \phi(\theta^{(i)}), \quad \phi = \arg \max_{\phi \in \mathcal{F}} \{ \frac{\partial}{\partial h} \text{KL}(q_{[h \phi]} \| p) \}_{h=0}$$

where $\phi$ is a function perturbation direction chosen to minimize the KL divergence between the updated density $q_{[h \phi]}$ induced by the particles and the posterior $p(\theta | \mathbf{X})$. The standard SVGD algorithm considers $\mathcal{F}$ as the unit ball of a vector-valued reproducing kernel Hilbert space (RKHS) $\mathcal{H}$ associated with a kernel $\kappa(\theta, \theta')$. In such a setting, (Liu & Wang, 2016) shows that

$$\phi(\theta) = \mathbb{E}_{\theta \sim q} [\kappa(\theta, \theta') F(\theta') + \nabla_{\theta'} \kappa(\theta, \theta')].$$ (2)

When approximating the expectation $\mathbb{E}_{\theta \sim q} [\cdot]$ with an empirical distribution formed by a set of particles $\{\theta^{(i)}\}_{i=1}^M$ and adopting stochastic gradients $G_k^{(i)} \triangleq \frac{N}{B} \sum_{j \in I_k} F_j^{(i)}(\theta^{(i)}),$
we arrive at the following update for the particles:

$$\theta_{k+1}^{(i)} = \theta_k^{(i)} + \frac{h}{M} \sum_{q=1}^{M} \left[ \kappa(\theta_k^{(q)}, \theta_k^{(i)}) G_{k}^{(i)} + \nabla_{\theta_k^{(q)}} \kappa(\theta_k^{(q)}, \theta_k^{(i)}) \right]$$

(SM).

SVGD then applies (3) repeatedly for all the particles.

### 2.3. Stochastic particle-optimization sampling

In this paper, we focus on RBF kernel $\kappa(\theta, \theta') = \exp(-\frac{||\theta - \theta'||^2}{2\kappa^2})$ due to its wide use in both theoretical analysis and practical applications. Hence, we can use a function $K(\theta) = \exp(-\frac{||\theta||^2}{2\kappa^2})$ to denote the kernel $\kappa(\theta, \theta')$. According to the work of (Chen et al., 2018; Zhang et al.), the stationary distribution of the $\rho_t$ in the following partial differential equation equals $p(\theta|X)$.

$$\partial_t \rho_t = \nabla_\theta \cdot (\rho_t \beta^{-1} F(\theta) + \rho_t E_x \sim \mu_t K(\theta - X) F(X)) - \rho_t (\nabla K \ast \rho_t) + \beta^{-1} \nabla \rho_t.$$  

(4)

When approximating the $\rho_t$ in (4) with an empirical distribution formed by a set of particles $\{\theta^{(i)}\}_{i=1}^{M}$, (Zhang et al.) derive the following diffusion process characterizing the SPOS algorithm.

$$d\theta_k^{(i)} = -\beta^{-1} F(\theta_k^{(i)}) dt - \frac{1}{M} \sum_{q=1}^{M} K(\theta_k^{(i)} - \theta_k^{(q)}) F(\theta_k^{(q)}) dt$$

$$+ \frac{1}{M} \sum_{q=1}^{M} \nabla K(\theta_k^{(i)} - \theta_k^{(q)}) dt + \sqrt{2\beta^{-1}t} dW_k^{(i)} \forall i$$

(5)

Note that if we set the initial distribution of all the particles $\theta_0^{(i)}$ to be the same, the system of these $M$ particles is exchangeable. So the distributions of all the $\theta_k^{(i)}$ are identical and can be denoted as $\rho_t$. When solving the above diffusion process with a numerical method and adopting stochastic gradients $G_k^{(i)}$, one arrives at the SPOS algorithm of (Zhang et al.) with the following update equation:

$$\theta_{k+1}^{(i)} = \theta_k^{(i)} - h\beta^{-1} G_k^{(i)} - \frac{h}{M} \sum_{j=1}^{M} K(\theta_k^{(i)} - \theta_k^{(j)}) G_k^{(j)}$$

$$+ \frac{h}{M} \sum_{j=1}^{M} \nabla K(\theta_k^{(i)} - \theta_k^{(j)}) + \sqrt{2\beta^{-1}h} \xi_k^{(i)}$$

(6)

where $\xi_k^{(i)} \sim N(0, I)$, and SPOS applies an update of (6) repeatedly for all the particles $\theta_k^{(i)}$. Detailed theoretical results for SPOS are reviewed in the Supplementary Material (SM).

### 3. Variance Reduction in SPOS

In standard SPOS, each particle is updated by adopting $G_k^{(i)} = \frac{N}{M} \sum_{j \in I_k} F_j(\theta_k^{(j)})$. Because one can only access $B \ll N$ data points in each step, the increased variance of the “noisy gradient” $G_k^{(i)}$ causes a slower convergence rate. A simple way to alleviate this is to increase $B$ by using larger minibatches. Unfortunately, this brings more computational costs, an undesired side effect. Thus more effective variance-reduction methods are needed for SPOS.

Inspired by recent work on variance reduction in SGLD, e.g., (Dubey et al., 2016; Chatterji et al., 2018; Zou et al., 2018), we develop three different variance-reduction algorithms for SPOS based on SAGA (Defazio et al., 2014) and SVRG (Johnson & Zhang, 2013) from stochastic optimization.

#### 3.1. SAGA-POS

SAGA-POS generalizes the idea of SAGA (Defazio et al., 2014) to an interactive particle-optimization system. For each particle $\theta_k^{(i)}$, we use $\{g_{k,j}^{(i)}\}_{j=1}^{B}$ as an approximation for each individual gradient $F_j(\theta_k^{(i)})$. An unbiased estimate of the full gradient $F(\theta_k^{(i)})$ is calculated as:

$$G_k^{(i)} = \sum_{j=1}^{N} g_{k,j}^{(i)} + \frac{N}{B} \sum_{j \in I_k} (F_j(\theta_k^{(i)}) - g_{k,j}^{(i)}), \forall i$$

(7)

where $I_k$ represents the set of data in mini-batch $k$. In each iteration, $\{g_{k,j}^{(i)}\}_{j=1}^{N}$ will be partially updated under the following rule: $g_{k+1,j}^{(i)} = F_j(\theta_k^{(i)})$ if $j \in I_k$, and $g_{k+1,j}^{(i)} = g_{k,j}^{(i)}$ otherwise. The algorithm is described in Algorithm 3.1.

Compared with standard SPOS, SAGA-POS also enjoys high computational efficiency, as it does not require calculation of each $F_j(\theta_k^{(i)})$ to get the full gradient $F(\theta_k^{(i)})$ in each iteration. Hence, the computation time of SAGA-POS is almost the same as that of POS. However, our analysis in Section 4 shows that SAGA-POS is endowed with a better convergence rate.

From another perspective, SAGA-POS has the same drawback as SAGA-based algorithms, which requires memory scaling at a rate of $O(MNd)$ in the worst case. For each particle $\theta_k^{(i)}$, one needs to store $N$ gradient approximations $\{g_{k,j}^{(i)}\}_{j=1}^{N}$. Fortunately, as pointed out by (Dubey et al., 2016; Chatterji et al., 2018), in some applications the memory cost scales only as $O(N)$ for SAGA-LD, which corresponds to $O(MN)$ for SAGA-POS as $M$ particles are used.

**Remark 1** When compared with SAGA-LD, note that $M$ particles are used in both SPOS and SAGA-POS. This makes the memory complexity $M$ times worse than SAGA-LD in training, thus SAGA-POS does not seem to bring any advantages over SAGA-LD. However, this intuition is not correct. As indicated by our theoretical results in Section 4, when the number of particles $M$ is large enough, the convergence
Although SVRG-POS alleviates the storage requirements.

Algorithm 1 SAGA-POS

**Input:** A set of initial particles \( \{\theta_0^{(i)}\}_{i=1}^M \), each \( \theta_0^{(i)} \in \mathbb{R}^d \), step size \( h_k \), batch size \( B \).

**Initialize** \( \{g_{0,j}^{(i)}\}_{j=1}^N = \{F_j(\theta_0^{(i)})\}_{j=1}^N \) for all \( i \in \{1, \ldots, M\} \);

1: for iteration \( k=0,1,\ldots,T \) do
2: Uniformly sample \( I_k \) from \( \{1,2,\ldots,N\} \) randomly with replacement such that \( |I_k| = B \);
3: Sample \( \xi_k^{(i)} \sim N(0, I_{d \times d}) \), \( \forall i \);
4: Update \( G_k^{(i)} = \sum_{j=1}^N g_{k,j}^{(i)} + \frac{N}{B} \sum_{j \in I_k} (F_j(\theta_k^{(i)})) - g_{k,j}^{(i)} \) for all \( i \);
5: Update each \( \theta_k^{(i)} \) with Eq.(6);
6: Uniformly sample \( I_k \) from \( \{1,2,\ldots,N\} \) randomly with replacement such that \( |I_k| = B \);
7: Sample \( \xi_k^{(i)} \sim N(0, I_{d \times d}) \), \( \forall i \);
8: Update \( G_k^{(i)} = \sum_{j \in I_k} (F_j(\theta_k^{(i)})) - g_{k,j}^{(i)} \) for all \( i \);
9: Update each \( \theta_k^{(i)} \) with Eq.(6);
10: end for

**Output:** \( \{\theta_T^{(i)}\}_{i=1}^M \)

rates of our algorithms are actually better than those of variance-reduced SGLD counterparts.

3.2. SVRG-POS

Under limited memory, we propose SVRG-POS, which is based on the SVRG method of (Johnson & Zhang, 2013). For each particle \( \theta_k^{(i)} \), one needs to store a stale parameter \( \bar{\theta}^{(i)} \), and update it occasionally every \( \tau \) iterations. At each update, we need to further conduct a global evaluation of full gradients at \( \bar{\theta}^{(i)} \), i.e., \( G^{(i)} = F(\theta_k^{(i)}) = F(\bar{\theta}^{(i)}) \). An unbiased gradient estimate is then calculated by leveraging both \( G^{(i)} \) and \( \bar{\theta}^{(i)} \): as:

\[
G_k^{(i)} = G^{(i)} + \frac{N}{B} \sum_{j \in I_k} (F_j(\theta_k^{(i)})) - F_j(\bar{\theta}^{(i)})
\] (8)

The algorithm is depicted in Algorithm 3.2, where one only needs to store \( \bar{\theta}^{(i)} \) and \( G^{(i)} \), instead of gradient estimates of all the individual \( F_j \). Hence the memory cost scales as \( O(Md) \), almost the same as that of standard SPOS.

Although SVRG-POS alleviates the storage requirements of SAGA-POS significantly, it also has the downside that the full gradients, \( F(\bar{\theta}^{(i)}) = \sum_{j=1}^N F(\theta_k^{(i)}) \), need to be recomputed every \( \tau \) iterations, leading to high computation cost in a big-data scenario.

**Remark 2** i) Similar to SAGA-POS, according to our theory in Section 4, SVRG-POS enjoys a faster convergence rate than SVRD-LD – its SGLD counterpart, although \( M \) times more space are required for the particles. This provides a trade-off between convergence rates and space complexity. ii) Previous work has shown that SAGA typically outperforms SVRG (Dubey et al., 2016; Chatterji et al., 2018) in terms of convergence speed. This conclusion applies to our case, which will be verified both by theoretical analysis in Section 4 and experiments in Section 5.

3.3. SVRG-POS+

The need for full gradient computation in SVRG-POS motivates the development of SVRG-POS+. Our algorithm is also inspired by the recent work of SVRG-LD+ on reducing the computational cost in SVRG-LD (Zou et al., 2018). The main idea in SVRG-POS+ is to replace the full gradient computation every \( \tau \) iterations with a subsampled gradient, i.e., to uniformly sample \( |J_k| = b \) data points where \( J_k \) is random samples from \( \{1,2,\ldots,N\} \) with replacement. Given the sub-sampled data, \( \bar{\theta}^{(i)} \) and \( G^{(i)} \) are updated as:

\[
\bar{\theta}^{(i)} = \theta_k^{(i)}, \quad G^{(i)} = \frac{N}{b} \sum_{j \in J_k} F_j(\theta_k^{(i)})
\]

The full algorithm is shown in Algorithm 3.3.

4. Convergence Analysis

We prove non-asymptotic convergence rates for the SAGA-POS, SVRG-POS and SVRG-POS+ algorithms under the 2-Wasserstein metric, defined as:

\[
\mathcal{W}_2(\mu, \nu) = \left( \inf_{\zeta \in \Gamma(\mu, \nu)} \int_{\mathbb{R}^d \times \mathbb{R}^d} \|X_\mu - X_\nu\|^2 d\zeta(X_\mu, X_\nu) \right)^{\frac{1}{2}}
\]
Assumption 1 Let $F$ and $K$ satisfy the following conditions:

- There exists a positive constant $m_F$ such that $\langle F(\theta) - F(\theta'), \theta - \theta' \rangle \geq m_F \| \theta - \theta' \|^2$; $F$ is $L_F$-Lipschitz continuous, i.e., $\| F(\theta) - F(\theta') \| \leq L_F \| \theta - \theta' \|$;
- $K$ is $L_K$-Lipschitz continuous for some $L_K > 0$, i.e., $\| K(\theta) - K(\theta') \| \leq L_K \| \theta - \theta' \|$ and $\nabla K$ is $\nabla L_K$-Lipschitz continuous for some $\nabla L_K \geq 0$, i.e., $\| \nabla K(\theta) - \nabla K(\theta') \| \leq L_K \| \theta - \theta' \|$;
- $K$ is an even function, i.e., $K(-\theta) = K(\theta)$
- The initial probability law of each particle has a bounded and strictly positive density $\nu_0$ with respect to the Lebesgue measure on $\mathbb{R}^d$, and $\gamma_0 \triangleq \log \int_{\mathbb{R}^d} e^{\| \theta \|^2/2} \nu_0(\theta) d\theta < \infty$

Assumption 2 There exists a constant $D_F > 0$ such that $\| \nabla F(\theta) - \nabla F(\theta') \| \leq D_F \| \theta - \theta' \|$.

Assumption 3 There exists a constant $\sigma$ such that for all $j \in \{1, 2, ..., N\}$,

$$\mathbb{E}[\| F_j(\theta) - \frac{1}{N} \sum_{j=1}^{N} F_j(\theta') \|^2] \leq \sigma \delta^2 / N^2$$

Assumption 4 There exist some positive constants $H_1, H_2$ such that $\| F(\theta_1) K(\theta_1 - \theta_2) - F(\theta_2) K(\theta_1 - \theta_2) \| \leq H_1 \| L_K \| \| \theta_1 - \theta_2 \| (\| \theta_1 - \theta_2 \| + \| \theta_1 - \theta_2 \|)$ and $\| F(\theta_2) \nabla K(\theta_1 - \theta_2) - F(\theta_2) \nabla K(\theta_1 - \theta_2) \| \leq H_2 \| L_{\nabla K} \| \| \theta_1 - \theta_2 \| (\| \theta_1 - \theta_2 \| + \| \theta_1 - \theta_2 \|)$. 

Remark 3 i) Assumption 1 is adopted from (Zhang et al.). The first bullet of Assumption 1 suggests $U(\cdot)$ is a strongly convex function, which is the general assumption in analyzing SGLD (Dalalyan & Karagulyan, 2017; Durmus & Moulines, 2016) and its variance-reduced variants (Zou et al., 2018; Chatterji et al., 2018). Although some work has been done on investigating the non-convex case, there is still significant utility in analyzing the convex case, which is more instructive and meaningful for addressing practical issues (Dalalyan & Karagulyan, 2017; Durmus & Moulines, 2016; Zou et al., 2018; Chatterji et al., 2018). ii) All of the $m_F, L_F$ and $D_F$ can scale linearly with $N$. iii) $K(\theta) = \exp(-\| \theta \|^2/2)$ can satisfy the above assumptions by setting the bandwidth large enough. Then $K$ can also be a Hessian Lipschitz with some positive constant $D_{\nabla^2 K}$ and $\| \nabla K \|$ can be bounded by some positive constant $H_{\nabla K}$. iv) Assumption 4 can be viewed as an extension to the Lipschitz continuity mentioned in Assumption 1, and it is used to bridge the work of (Chatterji et al., 2018) and (Zhang et al.). We assume $H_1, H_2$ are related with $F$ and they can scale linearly with $N$.

Now we present a convergence analysis for our algorithms, where $\alpha$ is some positive constant independent of $T$.

Theorem 1 Let $\mu_\cdot$ denote the distribution of the particles after $T$ iterations with SAGA-POS, and consider step size $h < \frac{B}{\sqrt{C_2}}$ and batch size $B \geq 9$. Under Assumptions 1 and 2, the convergence rate of SAGA-POS is bounded as

$$\mathcal{W}_2(\mu_\cdot, \mu^*) \leq C_1 \sqrt{\frac{T}{M}} + 5 \exp(-\frac{C_3 h}{4}) \mathcal{W}_2(\mu_0, \mu^*)$$

Assumption 4 can be viewed as an extension to the Lipschitz continuity mentioned in Assumption 1, and it is used to bridge the work of (Chatterji et al., 2018) and (Zhang et al.). We assume $H_1, H_2$ are related with $F$ and they can scale linearly with $N$.

Theorem 2 Let $\mu_\cdot$ denote the distribution of the particles after $T$ iterations with SVRG-POS in Algorithm 3.2. Under Assumptions 1 and 2, if we choose Option 1 and set the step size $h < \frac{1}{\sqrt{C_2}}$, the batch size $B \geq 2$ and the epoch length $\tau = 1 / \sqrt{C_2}$, the convergence rate of SVRG-POS is bounded for all $T$, which mod $\tau = 0$, as

$$\mathcal{W}_2(\mu_\cdot, \mu^*) \leq C_1 \sqrt{\frac{T}{M}} + \exp(-\frac{C_3 h}{50}) \mathcal{W}_2(\mu_0, \mu^*)$$
If we choose Option II and set the step size \( h < \frac{\sqrt{B}}{2M^2} \), the convergence rate of SVRG-POS is bounded for all \( T \) as

\[
W_2(\mu_T, \mu^*) \leq C_1 \sqrt{M} \exp(-\frac{C_3 h}{4} T) W_2(\mu_0, \mu^*)
\]

\[
+ \frac{\sqrt{2hC_d M^{1/2-\alpha}}}{C_3} + \frac{5hC_2^{3/2}}{C_3 M^{\alpha}} + \frac{9hC_2 T \sqrt{d}}{M^{\alpha} \sqrt{BC_3}}
\]

\[(11)\]

**Theorem 3** Let \( \mu_T \) denote the distribution of particles after \( T \) iterations with SVRG-POS. Under Assumptions 1, 2 and 3, if we set the step size \( h \leq \min\{\frac{(BC_3)^2}{2h(C^3/3hC_2)}, \frac{1}{hT(C^3/3hC_2)}\} \), then the convergence rate of SVRG-POS is bounded for all \( T \) as

\[
W_2(\mu_T, \mu^*) \leq C_1 \sqrt{M} \exp(-\frac{C_3 h}{4} T) W_2(\mu_0, \mu^*)
\]

\[
+ \frac{3C_3 h^{1/2}}{M^{\alpha} C_3^{1/2}} (b \leq N) + \frac{2h(C_d M^{1/2-\alpha})}{C_3}
\]

\[
+ \frac{2hC_2^{3/2}}{C_3 M^{\alpha}} + \frac{4hC_2 (rd)^{1/2}}{M^{\alpha} \sqrt{BC_3}}
\]

\[(12)\]

Since the complexity has been discussed in Section 3, we mainly focus on discussing the convergence rates here. Due to space limits, we move the comparison between convergence rates of the standard SPOS and its variance-reduced counterparts (such as SAGA-POS) into the SM. Specifically, adopting the standard framework of comparing different variance-reduction techniques in SGLD (Dubey et al., 2016; Chatterji et al., 2018; Zou et al., 2018), we focus on the scenario where \( m_f, L_f, H_f \) and \( D_f \) all scale linearly with \( N \) with \( N \gg d \). In this case, the last term in Theorem 1 dominates for SAGA-POS, \( O\left(\frac{h\sqrt{N}}{M^4} \right) \approx O\left(\frac{hN\sqrt{d}}{M^4} \right) \). Thus, to achieve an accuracy of \( \varepsilon \), we need the stepsizes

\[
h_{ag} = O\left(\frac{\varepsilon M^4}{N^{2}} \right).
\]

For SVRG-POS, the dominant term in Theorem 2 is \( O\left(\frac{\sqrt{hN\sqrt{d}}}{M^4} \right) \) for Option I and \( O\left(\frac{hN\sqrt{d}}{M^4 \sqrt{B}} \right) \) for Option II. Hence, for an accuracy of \( \varepsilon \), the corresponding step sizes are

\[
h_{v1} = O\left(\frac{\varepsilon M^4}{N^{2}} \right)
\]

and

\[
h_{v2} = O\left(\frac{\varepsilon M^4}{N^{2}} \right)
\]

respectively. Due to the fact that the mixing time \( T \) for these methods is roughly proportional to the reciprocal of step size (Chatterji et al., 2018), it is seen that when \( \varepsilon \) is small enough, one can have \( h_{v1} \ll h_{ag} \), which causes SAGA-POS to converge faster than SVRG-POS (Option I). Similar results hold for Option II since the factor \( \frac{1}{\sqrt{B}} \) in \( h_{v2} \) would make the step size even smaller. More theoretical results are given in the SM.

**Remark 4** We have provided a theoretical analysis to support the statement of i) in Remark 2. Moreover, we should also notice in SAGA-POS, stepsize \( h_{ag} = O\left(\frac{\varepsilon M^4}{N^{2}} \right) \) has an extra factor, \( M^{\alpha} \), compared with the step size \( O\left(\frac{\varepsilon B}{N^{2} \sqrt{d}} \right) \) used in SAGA-LD (Chatterji et al., 2018). This means SAGA-POS with more particles (\( M \) is large) would outperform SAGA-LD. SVRG-POS and SVRG-POS yield similar conclusions. This provides theoretical support for the statements of Remark 1 and i) in Remark 2. Furthermore, an interesting result from the above discussion is that when \( h_{v1} = O\left(\frac{\varepsilon M^4}{N^{2d}} \right) \) in SVRG-POS, there is an extra factor \( M \) compared to the stepsize \( O\left(\frac{\varepsilon B}{N^{2d}} \right) \) in SVRG-LD (Chatterji et al., 2018). Since the order of \( M^{2\alpha} \) is higher than \( M^{\alpha} \), one expects that the improvement of SVRG-POS over SVRG-LD is much more significant than that of SAGA-POS over SAGA-LD. This conclusion is verified in our experiments.

5. Experiments

We conduct experiments to verify our theory, and compare SAGA-POS, SVRG-POS and SVRG-POS with existing representative Bayesian sampling methods with/without variance-reduction techniques, e.g. SGLD and SPOS without variance reduction; SAGA-LD, SVRG-LD and SVRG-LD with variance reduction. For SVRG-POS, we focus on Option I in Algorithm 3.2 to verify our theory.

5.1. Synthetic log-normal distribution

We first evaluate our proposed algorithms on log-normal synthetic data, drawn from \( p(x|\mu) = \frac{1}{x \sqrt{2\pi} \sigma \exp(-\frac{(\ln x - \mu)^2}{2\sigma^2})} \) where \( x, \mu \in \mathbb{R}^{10} \). We calculate log-MSE of the samples “mean” w.r.t. the true value, and plot the log-MSE versus number of passes through data (Chatterji et al., 2018), similar to other variance-reduction algorithms in Figure 3, which shows that SAGA-POS and SVRG-POS converge the fastest among other algorithms. It is also interesting to see SPOS even outperforms both SAGA-LD and SVRG-LD.

5.2. Bayesian logistic regression

Following related work in (Dubey et al., 2016), we test the proposed algorithms for Bayesian-logistic-regression (BLR) on four publicly available datasets from the UCI machine learning repository: *Australasian* (690-14), *Pima* (768-8), *Diabetic* (1151-20) and *Susy* (100000-18), where \( (N - d) \) means a dataset of \( N \) data points with dimensionality \( d \). The first three datasets are relatively small, and the last one is large and is suitable for evaluating scalable Bayesian sampling algorithms.

Consider a dataset \( \{x_i, y_i\}_{i=1}^{N} \) with \( N \) samples, where \( x_i \in \mathbb{R}^{d} \) and \( y_i \in \{0, 1\} \). The likelihood of a BLR model is written as \( p(y_i = 1|x_i, \alpha) = \text{sigmoid}(\alpha^T X_i) \) with regression coefficient \( \alpha \in \mathbb{R}^{d} \), which for simplicity is assumed to be sampled from a standard multivariate Gaussian prior \( N(0, I) \). The datasets are split into 80% training data and 20% testing data. Optimized constant stepsizes are applied

---

1 For fair comparisons with our algorithms, we consider variance-reduced versions of SGLD with \( M \) independent chains.

---
for each algorithm via grid search. Following existing work, we report testing accuracy and log-likelihood versus the number of data passes for each dataset, averaging over 10 runs with 50 particles. The minibatch size is set to 15 for all experiments.

5.2.1. **VARIANCE-REDUCED SPOS VERSUS SPOS**

We first compare SAGA-POS, SVRG-POS and SVRG-POS + with SPOS without the variance reduction proposed in (Zhang et al.). The testing accuracies and log-likelihoods versus number of passes through data on the four datasets are plotted in Figure 4. It is observed that SAGA-POS converges faster than both SVRG-POS and SVRG-POS +, all of which significantly outperform SPOS. On the largest dataset SUSY, SAGA-POS starts only after one pass through the data, which then converges quickly, outperforming the other algorithms. SVRG-POS + outperforms SVRG-POS because the dataset SUSY is so large. All of these phenomena are consistent with our theory.

5.2.2. **VARIANCE-REDUCED SPOS VERSUS VARIANCE-REDUCED SGLD**

Next we compare the three variance-reduced SPOS algorithms with SGLD counterparts, i.e., SAGA-LD, SVRG-LD and SVRG-LD +, with results plotted in Figure 1. Similar phenomena are observed, where both SAGA-POS and SVRG-POS outperform SAGA-LD and SVRG-LD, respectively, consistent with our theoretical results discussed in
Variance Reduction in Stochastic Particle-Optimization Sampling

Figure 4. Testing accuracy and log-likelihood vs. the number of data pass for SPOS and its variance-reduction variants.

Remarks 1 and 2. Interestingly, for the PIMA dataset, SVRG-LD is observed to perform even worse (converges slower) than standard SGLD. Furthermore, as discussed in Remark 4, our theory indicates that the improvement of SVRG-POS over SVRG-LD is more significant than that of SAGA-POS over SAGA-LD. This is indeed true by inspecting the plots in Figure 1.

5.2.3. Impact of Number of Particles

Finally, we examine the impact of the number of particles on the convergence rates. As indicated by Theorems 1-3, for a fixed number of iterations $T$, the convergence error in terms of 2-Wasserstein distance decreases with increasing number of particles. To verify this, we run SAGA-POS and SVRG-POS for BLR with the number of particles ranging between $\{1, 2, 4, 8, 16\}$. The test log-likelihoods versus iteration numbers are plotted in Figure 2, demonstrating consistency with our theory.

6. Additional theoretical discussion for SAGA-POS, SVRG-POS and SVRG-POS$^+$

We discuss the mixing time and gradient complexity of our algorithms. The mixing time is the number of iterations needed to provably have error less than $\varepsilon$ measured in $W_2$ distance (Chatterji et al., 2018). The gradient complexity (Zou et al., 2018), which is almost the same as the computational complexity in (Chatterji et al., 2018), is defined as the required number of stochastic gradient evaluations to achieve a target accuracy $\varepsilon$. In Table 1 we present the mixing time and gradient complexity of several related algorithms, and we focus on Option I of SVRG-POS. This result for SVRG-LD$^+$ and SVRG-POS$^+$ may be a little different from that in (Zou et al., 2018) since we adopt different definitions for $F_j$.

Note that the result for SVRG-POS$^+$ is derived by adopting that $B = 1$ and $b = O(d\sigma^2/\mu^2\varepsilon^2)$ from (Zou et al., 2018), which also sheds light on the optimal choice of $b$ and $B$ in our SVRG-POS$^+$. For fair comparisons with our algorithms, we consider variance-reduced versions of SGLD with $M$ independent chains. Hence, the gradient complexities of the SAGA-LD, SVRG-LD and SVRG-LD$^+$ need to times $M$, consistent with the discussion in Section 3 and our experiment results. Since the convergence guarantees in Theorems 1, 2 and 3 are developed with respect to both iteration $T$ and the number $M$, we define the “threshold-particle,” which means the number of particles needed to provably have error less than $\varepsilon$ measured in $W_2$ distance. We will “threshold-particle” for our algorithms.

Note that the $M$ in the mixing time from Table 1 also should satisfy the result that $M \geq C_1^2/\varepsilon^2$. In practice, since $C_1 = 2(HF + HK) \sqrt{M(\beta + \frac{\varepsilon}{2} H_L FK - LF - 2L \psi K)}$, we set $\beta$ to be small enough to avoid the threshold-particle to be too large. However, in our experiments, since $HF, LK$ and $LF$ is not large, we do not need to worry about this issues.

Finally, we give explanations concerning the SAGA-POS algorithm, where it may be noted that we must store elements like $G_k^{(i)}$, since in Algorithm 1 we need to store them each iteration. But the $\{G_k^{(i)}\}_{i=1}^M$ only scales as $O(Md)$. Taking the dataset Susy as an example, we have $N = 10000$ and $d = 18$. However, we only use $M \leq 40$ particles. Hence, we do not take $\{G_k^{(i)}\}_{i=1}^M$ into consideration.

7. Conclusions

We have proposed several variance-reduction techniques for stochastic particle-optimization sampling, and for the
first time develop nonasymptotic convergence theory for the algorithms in terms of 2-Wasserstein metrics. Our theoretical results indicate the improvement of convergence rates for the proposed variance-reduced SPOS compared to both standard SPOS and the variance-reduced SGLD algorithms. Our theory is verified by a number of experiments on both synthetic data and real data for Bayesian logistic regression. Leveraging both our theory and empirical findings, we recommend the following algorithm choices in practice: i) SAGA-POS is preferable when storage is not a concern; ii) SVRG-POS is a better choice when storage is a concern and full gradients are feasible to calculate; iii) Otherwise, SVRG-POS$^+$ is a good choice and works well in practice.

Acknowledgements

The research performed at Duke University was supported in part by DARPA, DOE, NSF and ONR.

References


Zhang, J., Zhang, R., and Chen, C.


A. More details about the notations

- You may notice the different use of \( \theta \) and \( \theta \). \( \theta \) is mostly used for the interpretation of the theory. However, \( \theta \) is only used for the interpretation of algorithms, which means \( \theta \) often appears with \( k \) (which stands for the \( k \)th iteration) like \( \theta_k \). We design these differences to help you have a better understanding of our results.

The above rules still apply for the results in Appendix.

- The symbol \( 1(H_1 \leq H_2) \) in Theorem 3 means

\[
1(H_1 \leq H_2) = \begin{cases} 
1 & H_1 \leq H_2 \\
0 & H_1 > H_2
\end{cases}
\]  

(13)

and the symbol \( H_3 \wedge H_4 \) means \( \min\{H_3, H_4\} \)

- The relationship between RBF kernel \( \kappa(\theta, \theta') = \exp(-\frac{\|\theta - \theta\|^2}{2\sigma^2}) \) and the function \( K(\theta) = \exp(-\frac{\|\theta\|^2}{2\eta^2}) \) can be interpreted as \( \kappa(\theta, \theta') = K(\theta - \theta') \) in detail.

We moved the above details about the notations to the appendix due to the space limit.

B. The positive constants in Theorem 1, Theorem 2 and Theorem 3

For the sake of clarity, we present the following constants which are used in our theorems.

\[
C_1 = \frac{2(H_{\nabla K} + H_\theta)}{\sqrt{M}(\beta^{-1} - \frac{3}{2}H_\theta L_K - L_F - 2L_{\nabla K})}
\]

\[
C_2 = \sqrt{2(\beta^{-1}L_F + 2L_K H_1 + H_K L_F + L_{\nabla K})^2 + 2}
\]

\[
C_3 = \beta^{-1}m_F - 2L_F - 3H_1 L_K
\]

\[
C_4 = \beta^{-1}D_F + 4D_{\nabla K} + 4H_2 L_{\nabla K} + 2L_F H_{\nabla K} + 2H_1 L_K + L_F H_K
\]

\[
C_5 = 2\beta^{-1}\sigma^2 + 2H^2_{\nabla K}\sigma^2
\]

C. Convergence guarantees for SAGA-LD, SVRG-LD and SVRG-LD^+

In this section we present the convergence guarantees for SAGA-LD, SVRG-LD and SVRG-LD+ from (Chatterji et al., 2018; Zou et al., 2018)

Assumption 5 • (Sum-decomposable) The \( F(\theta) \) is decomposable i.e. \( F(\theta) = \sum_{j=1}^{N} F_j(\theta) \)

- (Smoothness) \( F(\theta) \) is Lipschitz continuous with some positive constant, i.e. for all \( \theta_1, \theta_2 \in \mathbb{R}^d, \|F(\theta_1) - F(\theta_2)\| \leq L_F \|\theta_1 - \theta_2\| \)

- (Strong convexity) \( F(\theta) \) is a \( m_F \)-strongly convex function, i.e. \( (F(\theta_1) - F(\theta_2)) (\theta_1 - \theta_2) \geq m_F \|\theta_1 - \theta_2\| \)

- (Hessian Lipschitz) There exists such a positive constant such that \( \|\nabla F(\theta_1) - \nabla F(\theta_2)\| \leq D_F \|\theta_1 - \theta_2\| \)

Assumption 6 (Bound Variance)\(^2\) There exists a constant \( \sigma \geq 0 \), such that for all \( j \)

\[
\mathbb{E}[\|F_j(\theta) - \frac{1}{N} \sum_{j=1}^{N} F_j(\theta)\|^2] \leq \sigma^2 / N^2
\]

Theorem 4 Under Assumption 5, let the step size \( h < \frac{B}{\sqrt{N}L_F} \) and the batch size \( B \geq 9 \), then we can have the bound for \( \mathcal{W}_2(\mu_T, \mu^*) \) in the SAGA-LD algorithm

\(^2\)This assumption is a little different from that in (Zou et al., 2018) since we adopt different definition of \( F_j \)
\[ W_2(\mu_T, \mu^*) \leq 5 \exp(-\frac{mFh}{4}T)W_2(\mu_0, \mu^*) + \]
\[ \frac{2hD_Fd}{m_F} + \frac{2hL_F^2\sqrt{d}}{m_F} + \frac{4hL_F^2N}{m_F \sqrt{B}} \]

**Theorem 5** Under Assumption 5, if we choose Option I and set the step size \( h < \frac{1}{mF} \), the batch size \( B \geq 2 \) and the epoch length \( \tau \geq \frac{8}{mFh} \), then we can have the bound for all \( T \) in the SVRG-LD algorithm

\[ W_2(\mu_T, \mu^*) \leq \exp\left(-\frac{mFh}{56}T\right)W_2(\mu_0, \mu^*) + \]
\[ \frac{2hD_Fd}{m_F} + \frac{2hL_F^2\sqrt{d}}{m_F} + \frac{64L_F^3\sqrt{d}}{m_F \sqrt{B}} \]

If we choose Option II and set the step size \( h < \frac{\sqrt{N}}{4 \tau^{3/2}} \), then we can have the bound for all \( T \) in the SVRG-LD algorithm

\[ W_2(\mu_T, \mu^*) \leq \exp\left(-\frac{mFh}{4}T\right)W_2(\mu_0, \mu^*) + \]
\[ \sqrt{2hD_Fd} + \frac{5hL_F^2\sqrt{d}}{m_F} + \frac{9hL_F\sqrt{d}}{\sqrt{Bm_F}} \]

**Theorem 6** Under Assumption 5 and Assumption 6, if we set the step size \( h \leq \min\{\left(\frac{BhC_3}{24C_4^2\tau}\right)^{\frac{1}{2}}, \frac{1}{8(\tau(C_3)^2/h+C_2)}\} \), then we can have the bound for all \( T \) in the SVRG-LD\(^+\) algorithm

\[ W_2(\mu_T, \mu^*) \leq (1-hmF/4)^T W_2(\mu_0, \mu^*) + \]
\[ \frac{3\sigma d^{1/2}}{m_F b^{1/2}} 1(b \leq N) + \frac{2hD_A d}{m_F} + \frac{2hL_F^3/d^{1/2}}{m_F} \]
\[ + \frac{4hL_F(\tau d)^{1/2} + 3h^{1/2}d^{1/2}\sigma}{\sqrt{Bm_F}} \]

### D. Proof of the theorems in Section 4

In this section, we give proofs to the theorems in Section 4. We are sorry that the proof of our theorems is a little long since we want to make it more easy to understand. However, this does not affect that fact that our proof is credible. Our proof is based on the idea of (Zhang et al.) and borrow some results from (Chatterji et al., 2018; Zou et al., 2018)

\[ d\theta_t^{(i)} = -\beta^{-1}F(\theta_t^{(i)})dt - \frac{1}{M} \sum_{q=1}^{M} K(\theta_t^{(i)} - \theta_t^{(q)})F(\theta_t^{(q)})dt \]
\[ + \frac{1}{M} \sum_{q=1}^{M} \nabla K(\theta_t^{(i)} - \theta_t^{(q)})dt + \sqrt{2\beta^{-1}dW_t^{(i)}} \quad \forall i \quad (14) \]

As mention is Section 2.3 we denote the distribution of \( \theta_t^{(i)} \) in Eq.(14) as \( \nu_t \). From the proof of Theorem 5 in (Zhang et al.) we can derive that

\[ W_2(\nu_{\infty}, \mu^*) \leq \frac{2(H_{\nabla K} + H_\theta)}{\sqrt{M(\beta^{-1} - \frac{3}{2}H_\theta L_K - L_F - 2L_{\nabla K})}} \quad (15) \]

In order to bound \( W_2(\mu_T, \mu^*) \), we need to bound \( W_2(\mu_T, \nu_{\infty}) \) next. Now we borrow the idea in (Zhang et al.) , concatenating the particles at each time into a single vector representation, We define a new parameter at time \( t \) as \( \Theta_t \triangleq [\theta_t^{(1)}, \cdots, \theta_t^{(M)}] \in \mathbb{R}^{Md} \). Consequently, \( \Theta_t \) is driven by the following linear SDE:

\[ d\theta_t = -F^\theta(\theta_t)dt + \sqrt{2\beta^{-1}dW_t^{(Md)}}, \quad (16) \]
\[ F^\Theta(\Theta_t) \triangleq [\beta^{-1}F(\theta_t^{(1)}) - \frac{1}{M} \sum_{q=1}^M \nabla K(\theta_t^{(1)} - \theta_t^{(q)}) + \frac{1}{M} \sum_{q=1}^M K(\theta_t^{(M)} - \theta_t^{(q)})F(\theta_t^{(q)})] \]

is a vector function \( \mathbb{R}^M \to \mathbb{R}^M \), and \( W_t^{(M)} \) is Brownian motion of dimension \( Md \).

Now we define \( F_j^\Theta(\Theta_t) \triangleq [\beta^{-1}F_j(\theta_t^{(1)}) - \frac{1}{MN} \sum_{q=1}^M \nabla K(\theta_t^{(1)} - \theta_t^{(q)}) + \frac{1}{M} \sum_{q=1}^M K(\theta_t^{(M)} - \theta_t^{(q)})F(\theta_t^{(q)})] \). We can find the \( F^\Theta(\Theta_t) \) and \( F_j^\Theta(\Theta_t) \) defined above satisfy the following theorem.

**Theorem 7**

(Sum-decomposable) The \( F^\Theta(\Theta) \) is decomposable i.e. \( F^\Theta(\Theta) = \sum_{j=1}^N F_j^\Theta(\Theta) \)

- (Strong convexity) \( F^\Theta(\Theta) \) is a \((\beta^{-1}m_F - 2L_F - 3H_1 L_K)\)-strongly convex function, i.e. \((F^\Theta(\Theta_1) - F^\Theta(\Theta_2))(\Theta_1 - \Theta_2) \leq (\beta^{-1}m_F - 2L_F - 3H_1 L_K)\|\Theta_1 - \Theta_2\|\)

- (Hessian Lipschitz) The function \( F^\Theta(\Theta) \) is Hessian Lipschitz, i.e., \( \| \nabla F^\Theta(\Theta_1) - \nabla F^\Theta(\Theta_2) \| \leq (\beta^{-1}D_F + 4D_K + 4H_2 L_N K + 2L_F H_N K + 2H_1 L_K + L_F H_K) \| \Theta_1 - \Theta_2 \| \)

- (Smoothness) \( F^\Theta(\Theta) \) is Lipschitz continuous with some positive constant, i.e. for all \( \Theta_1, \Theta_2 \in \mathbb{R}^{Md}, |F^\Theta(\Theta_1) - F^\Theta(\Theta_2)| \leq \sqrt{2(3\beta^{-1}L_F + 2H_1 L_F H_N K + H_2 L_K + L_F H_K)^2 + 2} \| \Theta_1 - \Theta_2 \| \)

- (Bound Variance) There exits a constant, \( \sigma \geq 0 \), such that for all \( j \),

\[
\mathbb{E} [\| F_j^\Theta(\Theta) - \frac{1}{N} \sum_{j=1}^N F_j^\Theta(\Theta) \|^2 ] \leq Md(2\beta^{-1} + 2H_2 K)^2/N^2
\]

**Proof**

- The sum-decomposable property of \( F^\Theta(\Theta) \) is easy to verify.

- (Strong convexity)

\[
(F^\Theta(\Theta_1) - F^\Theta(\Theta_2))(\Theta_1 - \Theta_2) = \frac{1}{M} \sum_{i,q} (\xi_{iq} + \xi_{iq}^2 + \xi_{iq}^3 + \xi_{iq}^4)
\]

where

\[
\xi_{iq}^1 = \beta^{-1} \left( F(\theta_1^{(i)}) - F(\theta_2^{(i)}) \right) \cdot (\Theta_1^{(i)} - \Theta_2^{(i)})
\]

\[
\xi_{iq}^2 = - \left( \nabla K(\theta_1^{(i)} - \theta_1^{(q)}) - \nabla K(\theta_2^{(i)} - \theta_2^{(q)}) \right) \cdot (\Theta_1^{(i)} - \Theta_2^{(i)})
\]

\[
\xi_{iq}^3 = \left( F(\theta_1^{(q)}) K(\theta_1^{(i)} - \theta_1^{(q)}) - F(\theta_2^{(q)}) K(\theta_2^{(i)} - \theta_2^{(q)}) \right) \cdot (\Theta_1^{(i)} - \Theta_2^{(i)})
\]

\[
\xi_{iq}^4 = \left( F(\theta_2^{(q)}) K(\theta_1^{(i)} - \theta_1^{(q)}) - F(\theta_1^{(q)}) K(\theta_2^{(i)} - \theta_2^{(q)}) \right) \cdot (\Theta_1^{(i)} - \Theta_2^{(i)})
\]

For the \( \xi_{iq}^1 \) terms, applying the convex condition for \( F \), we have

\[
\sum_{iq} \xi_{iq}^1 = \sum_{iq} \beta^{-1} \left( F(\theta_1^{(i)}) - F(\theta_2^{(i)}) \right) \cdot (\Theta_1^{(i)} - \Theta_2^{(i)}) \geq \beta^{-1} m_F M \sum_i \| \theta_1^{(i)} - \theta_2^{(i)} \|^2
\]
For the $\xi_{iq}^2$ term, applying the concave condition for $K$ and $\nabla K$ is odd, we have $\sum_{iq} \xi_{iq}^2 = -$

$$
\sum_{iq}^M \left( \nabla K(\theta_1^{(i)} - \theta_1^{(q)}) - \nabla K(\theta_2^{(i)} - \theta_2^{(q)}) \right) \cdot \left( \theta_1^{(i)} - \theta_2^{(i)} \right)
$$

$$
= -\frac{1}{2} \sum_{iq}^M \sum_{iq}^M \left( \nabla K(\theta_1^{(i)} - \theta_1^{(q)}) - \nabla K(\theta_2^{(i)} - \theta_2^{(q)}) \right) \cdot \left( \theta_1^{(i)} - \theta_1^{(q)} - (\theta_2^{(i)} - \theta_2^{(q)}) \right)
$$

$$
= \frac{1}{2} L_{\nabla K} \sum_{iq} \left\| \theta_1^{(i)} - \bar{\theta}_1^{(i)} - (\bar{\theta}_2^{(i)} - \bar{\theta}_2^{(j)}) \right\|^2 \geq -2L_{\nabla K} ME \sum_i \left\| \theta_1^{(i)} - \bar{\theta}_1^{(i)} \right\|^2 \tag{19}
$$

For the $\xi_{iq}^3$ terms, after applying the $L_F$-Lipschitz property of $F$, we have $\sum_{iq} \xi_{iq}^3 =$

$$
\sum_{iq} (F(\theta_1^{(i)} - \theta_1^{(q)}) - F(\theta_2^{(i)} - \theta_2^{(q)})) \cdot \left( \theta_1^{(i)} - \theta_2^{(i)} \right)
$$

$$
\geq -\sum_{iq} L_F \left\| \theta_1^{(q)} - \theta_2^{(q)} \right\| \left\| \theta_1^{(i)} - \theta_2^{(i)} \right\|
$$

$$
\geq -2L_F M \sum_{i} \left\| \theta_1^{(i)} - \theta_2^{(i)} \right\|^2 \tag{20}
$$

For the $\xi_{iq}^4$ terms, we have $\sum_{iq} \xi_{iq}^4 =$

$$
\sum_{iq} (F(\theta_1^{(i)} - \theta_1^{(q)}) - F(\theta_2^{(i)} - \theta_2^{(q)})) \cdot \left( \theta_1^{(i)} - \theta_2^{(i)} \right)
$$

$$
\geq -H_1 L_K \sum_{iq} \left\| \theta_1^{(i)} - \theta_1^{(q)} - (\theta_2^{(i)} - \theta_2^{(q)}) \right\| \left\| \theta_1^{(i)} - \theta_2^{(i)} \right\|
$$

$$
\geq -3H_1 L_K M \sum_{i} \left\| \theta_1^{(i)} - \theta_2^{(i)} \right\|^2 \tag{21}
$$

Then we finally arrive at:

$$
(F(\Theta_1) - F(\Theta_2)) (\Theta_1 - \Theta_2)
$$

$$
\geq (\beta^{-1} m_F - 2L_{\nabla K} - 2L_F - 3H_1 L_K) \sum_{i} \left\| \theta_1^{(i)} - \theta_2^{(i)} \right\|
$$

$$
\geq (\beta^{-1} m_F - 2L_{\nabla K} - 2L_F - 3H_1 L_K) \left\| \Theta_1 - \Theta_2 \right\| \tag{22}
$$

• Now, we will prove the third result:

$$
\left\| \nabla F(\Theta_1) - \nabla F(\Theta_2) \right\|
$$

$$
\leq \beta^{-1} \sum_{i=1}^M \left\| \nabla F(\theta_1^{(i)}) - \nabla F(\theta_2^{(i)}) \right\| + \sum_{i=1}^M \frac{2}{M} \sum_{q=1}^M \left\| \nabla^2 K(\theta_1^{(i)} - \theta_1^{(q)}) - \nabla^2 K(\theta_2^{(i)} - \theta_2^{(q)}) \right\|
$$

$$
+ \frac{2}{M} \sum_{i=1}^M \sum_{q=1}^M \left\| \nabla K(\theta_1^{(i)} - \theta_1^{(q)}) F(\theta_1^{(q)}) - \nabla K(\theta_2^{(i)} - \theta_2^{(q)}) F(\theta_2^{(q)}) \right\|
$$

$$
+ \sum_{i=1}^M \sum_{q=1}^M \frac{1}{M} \left\| K(\theta_1^{(i)} - \theta_1^{(q)}) \nabla F(\theta_1^{(q)}) - K(\theta_2^{(i)} - \theta_2^{(q)}) \nabla F(\theta_2^{(q)}) \right\|
$$
We apply Euler-Maruyama discretization to Eq.(16) and substitute $G_k$ for $F(\Theta_k)$ to derive the following equation:

$$\Theta_{k+1} = \Theta_k - G_k^\Theta h + \sqrt{2\beta^{-1}h} \Xi_k, \quad \Xi_k \sim N(0, M_d \times M_d)$$

Hence, with different $G_k^\Theta$, we can perform different algorithm of $\Theta_k$, like SAGA-LD, SVRG-LD and SVRG-LD$^+$ algorithm of $\Theta_k$. It is worth noting that the SAGA-LD, SVRG-LD and SVRG-LD$^+$ algorithm of $\Theta_k$ is actually the corresponding SAGA-POS, SVRG-POS and SVRG-POS$^+$ algorithm of $\{\theta_k\}$. 

Similarly, we can easily verify that

$$\|F(\Theta_1) - F(\Theta_2)\| \leq \sqrt{2(\beta^{-1}L_F + 2L_K H_1 + H_K L_F + L_{\nabla K})^2 + 2 \|\Theta_1 - \Theta_2\|}$$

• Now, we will prove the last result.

$$\mathbb{E}[[F_j^\Theta(\theta) - \frac{1}{N} \sum_{j=1}^{N} F_j^\Theta(\theta)]^2] =$$

$$\sum_{i=1}^{M} \frac{1}{N} \sum_{j=1}^{N} F_j(\theta^{(i)}) - \frac{1}{N} \sum_{j=1}^{N} F_j(\theta^{(i)}) - \frac{1}{M} \sum_{i=1}^{M} K(\theta^{(i)} - \theta^{(q)}) F_j(\theta^{(q)}) - \frac{1}{M} \sum_{i=1}^{M} \sum_{i=1}^{M} K(\theta^{(i)} - \theta^{(q)}) F_j(\theta^{(q)})|^2]$$

$$\leq \sum_{i=1}^{M} \frac{1}{N} \sum_{j=1}^{N} F_j(\theta^{(i)}) - \frac{1}{N} \sum_{j=1}^{N} F_j(\theta^{(i)})|^2 + 2 \frac{H_K}{M^2} \mathbb{E} \| \sum_{q=1}^{M} \left( F_j(\theta^{(q)}) - \frac{1}{N} \sum_{j=1}^{N} F_j(\theta^{(q)}) \right) \|^2$$

$$\leq \sum_{i=1}^{M} (2d\sigma^2 + 2H_K d\sigma^2)/N^2$$

$$\leq Md(2\sigma^2 + 2H_K^2 \sigma^2)/N^2$$
This result is extremely important for our proof and bridges the gap between the variance reduction in stochastic gradient Langevin dynamics (SGLD) and variance reduction in stochastic particle-optimization sampling (SPOS). And thanks to the Theorem 7, we can can find \( F(\Theta) \) satisfies the Assumption 5 and Assumption 6. (Please notice the \( F(\Theta) \) corresponds to the \( \nabla F \) in (Chatterji et al., 2018)). Hence, we can borrow the theorems in (Chatterji et al., 2018; Zou et al., 2018) and derive some thrilling results for the variance reduction techniques in stochastic particle-optimization sampling (SPOS).

We denotes the distribution of \( \Theta \) in Eq.(16) and the distribution of \( \Theta_i \) in Eq.(25) as \( \Gamma_i \) and and \( \Lambda_i \). Now we can derive the following theorems. \( (C_1, C_2, C_3, C_4 \text{ and } C_5 \text{ are defined in Section 4}) \)

**Theorem 8** Let the step size \( h < \frac{B}{8NC^1} \) and the batch size \( B \geq 9 \), then we can have the bound for \( W_2(\Lambda_T, \Gamma_\infty) \) in the SAGA-LD algorithm of \( \Theta_k \).

\[
W_2(\Lambda_T, \Gamma_\infty) \leq 5 \exp\left(-\frac{C_3 h}{4}T\right)W_2(\Lambda_0, \Gamma_\infty) + \frac{2hC_4Md}{C_3} + \frac{2hC_2 \cdot \sqrt{Md}}{C_3} + \frac{24hC_2 \cdot \sqrt{MdN}}{\sqrt{C_2B}}
\]

**Theorem 9** If we choose Option I and set the step size \( h < \frac{1}{\sqrt{C_2}} \), the batch size \( B \geq 2 \) and the epoch length \( \tau \geq \frac{8}{C_2} \), then we can have the bound for all \( T \text{ mod } \tau = 0 \) in the SVRG-LD algorithm of \( \Theta_k \).

\[
W_2(\Lambda_T, \Gamma_\infty) \leq \exp\left(-\frac{C_3 h}{56}T\right)W_2(\Lambda_0, \Gamma_\infty) + \frac{2hC_4Md}{C_3} + \frac{2hC_2 \cdot \sqrt{Md}}{C_3} + \frac{64C_2 \cdot \sqrt{hMd}}{\sqrt{BC_3}}
\]

If we choose Option II and set the step size \( h < \frac{\sqrt{B}}{4C_2} \), then we can have the bound for all \( T \) in the SVRG-LD algorithm of \( \Theta_k \).

\[
W_2(\Lambda_T, \Gamma_\infty) \leq \exp\left(-\frac{C_3 h}{4}T\right)W_2(\Lambda_0, \Gamma_\infty) + \frac{\sqrt{2hC_4Md}}{C_3} + \frac{5hC_2 \cdot \sqrt{Md}}{C_3} + \frac{9hC_2 \cdot \sqrt{Md}}{\sqrt{BC_3}}
\]

**Theorem 10** If we set the step size \( h \leq \min\{\left(\frac{BC_2}{24C^2 \cdot \tau}\right)^{1/2}, \frac{1}{6\tau(C^2 \cdot b + C_2)}\} \), then we can have the bound for all \( T \) in the algorithm SVRG-LD\(^{+} \) of \( \Theta_k \).

\[
W_2(\Lambda_T, \Gamma_\infty) \leq (1 - hC_2 / 4)^T W_2(\mu_0, \mu^*) + \frac{3C_5(Md)^{1/2}}{C_3 b^{1/2}} \cdot 1(b \leq N) + \frac{2h(C_4Md)}{C_3} + \frac{2hC_2 \cdot 3/2(Md)^{1/2}}{C_3} + \frac{4hC_2(Md)^{1/2} \wedge 3h^{1/2}(Md)^{1/2}C_5}{\sqrt{BC_3}}
\]

Now we will give a proposition which will be useful in connecting the \( W_2(\Lambda_T, \Gamma_\infty) \) and \( W_2(\mu_T, \nu_\infty) \) mentioned above.

**Proposition 11** (For simplicity of notations, we directly use \( \theta \) and \( \Theta \) themselves to denote their own distributions.) If \( \Theta_1 \) and \( \Theta_2 \) are defined as \( \Theta_1 \triangleq [\theta_1^{(1)}, \cdots, \theta_1^{(M)}] \in \mathbb{R}^M \) and \( \Theta_2 \triangleq [\theta_2^{(1)}, \cdots, \theta_2^{(M)}] \in \mathbb{R}^M \), we can derive the following result

\[
\sum_{i=1}^{M} W_2^2(\theta_1^{(i)}, \theta_2^{(i)}) \leq W_2^2(\Theta_1, \Theta_2)
\] (25)
Variance Reduction in Stochastic Particle-Optimization Sampling

Proof According to the Eq.(4.2) in (Soheil Feizi & Tse), we can write the $W_2(\theta_1^{(i)}, \theta_2^{(i)})$ in the following optimization:

$$
W_2^2(\theta_1^{(i)}, \theta_2^{(i)}) = \mathbb{E}\|\theta_1^{(i)}\|^2 + \mathbb{E}\|\theta_2^{(i)}\|^2 + 2\sup_{\phi: \text{convex}}\{-\mathbb{E}[\phi(\theta_1^{(i)})] - \mathbb{E}[\phi^*(\theta_2^{(i)})]\} 
$$

(26)

where $\phi^*(\theta) = \sup_{\nu \in \Pi} (\nu^T \theta - \phi(\theta))$ is the convex-conjugate of the function $\phi$. We assume $\phi_i$ is the optimal function of Eq.26. Then it is trivial to verify that $\Psi(\Theta) = \sum_{i=1}^M \phi_i(\theta^{(i)})$ is a convex function. Due to the property of conjugate functions, we need to notice $\Psi(\Theta)^* = \sum_{i=1}^M \phi_i^*(\theta^{(i)})$. Now we can derive the following result:

$$
\sum_{i=1}^M W_2^2(\theta_1^{(i)}, \theta_2^{(i)}) = \sum_{i=1}^M \{\mathbb{E}\|\theta_1^{(i)}\|^2 + \mathbb{E}\|\theta_2^{(i)}\|^2 + 2(-\mathbb{E}[\phi_i(\theta_1^{(i)})] - \mathbb{E}[\phi_i^*(\theta_2^{(i)})])\} 
$$

$$
= \mathbb{E}\|\Theta_1\|^2 + \mathbb{E}\|\Theta_2\|^2 + 2(-\mathbb{E}[\Psi(\Theta_1)] - \mathbb{E}[\Psi^*(\Theta_2)]) 
$$

$$
\leq W_2^2(\Theta_1, \Theta_2) 
$$

Then we finish our proof.

We should notice due to the exchangeability of the M-particles system $\{\theta_k^{(i)}\}$ in our SPOS-type sampling, the distribution of each particle $\theta_k^{(i)}$ at the same time is identical. Hence, using Proposition 11, we can derive

$$
W_2(\mu_T, \nu_\infty) \leq \frac{1}{\sqrt{M}} W_2(\Lambda_T, \Gamma_\infty) 
$$

(27)

Now we will introduce a mild assumption that $W_2(\mu_T, \nu_\infty) \leq \frac{1}{\sqrt{M\alpha}} W_2(\Lambda_T, \Gamma_\infty)$. We wish to make some comments on the additional assumption. This assumption is reasonable. With this assumption, our theory can be verified by the experiment results, e.g. the improvement of SVRG-POS over SVRG-LD is much more significant than that of SAGA-POS over SAGA-LD, which imply the correctness and effectiveness of our assumption. Moreover, this assumption does not conflict with what you mentioned, since $W_2(\mu_T, \nu_\infty) \leq \frac{1}{M^{1/2+\alpha}} W_2(\Lambda_T, \Gamma_\infty) \leq \frac{1}{\sqrt{M}} W_2(\Lambda_T, \Gamma_\infty)$. Furthermore, this assumption can be supported theoretically. Please consider the continuous function $\log M \left( W_2(\Theta_1, \Theta_2) M / \sum_{i=1}^M W_2(\theta_1^{(i)}, \theta_2^{(i)}) \right) - 1/2$. We often care about bounded space in practice, which means we can find a positive minimum for that function in most cases. Since in practice we cannot use infinite particles, the required $\alpha$ does exist within the positive minima for every $M$ mentioned above. Although we do not aim at giving an explicit expression for it, the existence is enough to explain the experiment results in our paper. Last, this assumption is supported in the algorithm itself. Please notice the fact that SPOS can be viewed as the combination of SVGD and SGLD. The SVGD part can let it satisfy some good properties which SGLD does not endow.

Proof of Theorem 1, Theorem 2 and Theorem 3 Applying the results for $W_2(\Lambda_T, \Gamma_\infty)$ in Theorem 8, Theorem 9 and Theorem 10, we can get the corresponding results for $W_2(\mu_T, \nu_\infty)$ in the SAGA-POS, SVRG-POS and SVRG-POS. Then we can bound $W_2(\mu_T, \mu^*)$, which is what we desire, with the following fact

$$
W_2(\mu_T, \mu^*) \leq W_2(\mu_T, \nu_\infty) + W_2(\nu_\infty, \mu^*) 
$$

(28)

Note that from the proof of Theorem 3 and Remark 1 in (Zhang et al.), we can get that

$$
W_2(\nu_\infty, \mu^*) \leq \frac{C_1}{\sqrt{M}} 
$$

(29)

Apply the results in Theorem 8, Theorem 9 and Theorem 10 above, we can prove the Theorem 1, Theorem 2 and Theorem 3.
E. Comparison between SPOS and its variance-reduction counterpart

In (Zhang et al.), they use the distance \( \tilde{B}_T \) defined as \( \tilde{B}_T \equiv \sup |E_{\mu_T}[f(\theta)] - E_{\mu^*}[f(\theta)]| \). When \( ||f||_{lip} \leq 1 \), \( \tilde{B}_T \) is another definition of \( W_1(\mu_T, \mu^*) \). Actually, according to the proof in (Zhang et al.), they did give a bound in terms of \( W_1(\mu_T, \mu^*) \).

Then according to the results in (Zhang et al.), we can get the following theorem,

**Theorem 12 (Fixed Stepsize)**  Under Assumption 1, there exit some positive constants \((c_1, c_2, c_3, c_4, c_5, c_6)\) such that the bound for \( W_1(\mu_T, \mu^*) \) in the SPOS algorithm satisfies:

\[
W_1(\mu_T, \mu^*) \leq \frac{c_1}{\sqrt{M(\beta^{-1} - c_2)}} + c_3 \exp \left\{ -2 \left( \beta^{-1} m_F - 2L_K - L_F \right) Th \right\} + c_6 Md \beta^{-3}(c_4 \beta^2 B^{-1} + c_5 h) \frac{1}{2} T \frac{1}{2} h \frac{1}{2} .
\]

Firstly, we should notice that the third term \( c_3 Md \beta^{-3}(c_4 \beta^2 B^{-1} + c_5 h) \frac{1}{2} T \frac{1}{2} h \frac{1}{2} \) on the right side increases with \( T \) and \( M \).

However, the bound for SAGA-POS, SVRG-POS and SVRG-POS\(^+\) in our paper decrease with with \( T \) and \( M \), which means that the bound for SAGA-POS, SVRG-POS and SVRG-POS\(^+\) is much tighter than the bound for SPOS. Furthermore, the convergence of SPOS is characterized in \( W_1(\mu_T, \mu^*) \) but the convergence of SAGA-POS, SVRG-POS and SVRG-POS\(^+\) are characterized by \( W_2(\mu_T, \mu^*) \). Due to the well-known fact that \( W_1(\mu_T, \mu^*) \leq W_1(\mu_T, \mu^*) \), we can verify that SAGA-POS, SVRG-POS and SVRG-POS\(^+\) can outperform SPOS in the theoretical perspective. Although the result for SPOS in (Zhang et al.) may be improved in the future, but we believe that there is no doubt that SAGA-POS, SVRG-POS and SVRG-POS\(^+\) are better than it in performance, which has been verified in experiments in our paper.

F. More experiments results

We further examine the impact of number of particles to the convergence rates of variance-reduced SGLD and SPOS. As indicated by Theorems 1-3 (discussed in Remark 1 and 2), when the number of particles are large enough, the convergence rates of SAGA-POS and SVRG-POS would both outperform their SGLD counterparts. In addition, the performance gap would increase with increasing \( M \), as indicated in Remark 4. We conduct experiments on the *Australian* dataset by varying its particle numbers among \( \{1, 8, 16, 32\} \). The results are plotted in Figure 5, which are roughly aligned with our theory.
Variance Reduction in Stochastic Particle-Optimization Sampling

Figure 5. Testing accuracy and log-likelihood vs the number of data pass for SPOS with varying number of particles.