# Further analysis of multilevel Stein variational gradient descent with an application to the Bayesian inference of glacier ice models

Terrence Alsup[1], Tucker Hartland[2], Benjamin Peherstorfer[1*], Noemi Petra[2]

[1]Courant Institute of Mathematical Sciences, New York University, 251 Mercer Street, New York, 10012, NY, USA.
[2]Department of Applied Mathematics, University of California, Merced, 5200 North Lake Rd, Merced, 95343, CA, USA.

*Corresponding author(s). E-mail(s): pehersto@cims.nyu.edu;
Contributing authors: alsup@cims.nyu.edu; hartland1@llnl.gov;
npetra@ucmerced.edu;

**Abstract**

Multilevel Stein variational gradient descent is a method for particle-based variational inference that leverages hierarchies of surrogate target distributions with varying costs and fidelity to computationally speed up inference. The contribution of this work is twofold. First, an extension of a previous cost complexity analysis is presented that applies even when the exponential convergence rate of single-level Stein variational gradient descent depends on iteration-varying parameters. Second, multilevel Stein variational gradient descent is applied to a large-scale Bayesian inverse problem of inferring discretized basal sliding coefficient fields of the Arolla glacier ice. The numerical experiments demonstrate that the multilevel version achieves orders of magnitude speedups compared to its single-level version.

**Keywords:** Multi-fidelity and multilevel methods, surrogate modeling, Bayesian inference, Stein variational gradient descent, ice sheet inverse problems

# 1 Introduction

Bayesian inference is a ubiquitous and flexible tool for updating a belief (i.e., learning) about a quantity of interest when data are observed, which ultimately can be used to inform downstream decision-making. In particular, Bayesian inverse problems allow one to derive knowledge from data through the lens of physics-based models. These problems can be formulated as follows: given observational data, a physics-based model, and prior information about the model inputs, find a posterior probability distribution for the inputs that reflects the knowledge about the inputs in terms of the observed data and prior. Typically, the physics-based models are given in the form of an input-to-observation map that is based on a system of partial differential equations (PDEs). The computational task underlying Bayesian inference is approximating posterior probability distributions to compute expectations and to quantify uncertainties. There are multiple ways of computationally exploring posterior distributions to gain insights, reaching from Markov chain Monte Carlo to variational methods [1–3].

In this work, we make use of Stein variational gradient descent (SVGD) [4], which is a method for particle-based variational inference, to approximate posterior distributions. It builds on Stein's identity to formulate an update step for the particles that can be realized numerically in an efficient manner via a reproducing kernel Hilbert space. There are various extensions to SVGD such as exploiting curvature information of the target distribution with a corresponding Newton method [5] as well as using adaptive kernels as in [6, 7]. Specifically for Bayesian inverse problems, SVGD has been extended to take advantage of low-dimensional structure [8] in the posterior distribution [9] and the model states [10]. Much effort has been put into understanding the convergence and statistical properties of SVGD and its variants. The study of convergence of SVGD was sparked primarily by [11, 12], which showed that in the mean-field limit SVGD follows a gradient flow with respect to the Kullback-Leibler (KL) divergence. Similar results were later shown for the chi-squared divergence in [13]. Pre-asymptotic convergence results in both the number of samples and the discrete-time setting remains open, but progress in this direction has been made in [14]. There also has been work on understanding and improving the performance of SVGD in high dimensions [15].

We focus on the multilevel extension of SVGD (MLSVGD), which was introduced in [16] and leverages hierarchies of approximations of a target posterior distribution with varying costs and fidelity to computationally speed up inference. Such approximations can be obtained via, e.g., coarse and fine discretizations of the governing equations of the physics-based models as well as surrogate models [17] and simplified-physics models [18]. Multilevel methods have a long tradition in scientific computing and computational statistics. The MLSVGD approach is motivated by multi-fidelity and multilevel methods such as multilevel and multi-fidelity Monte Carlo [19–23] and Markov chain Monte Carlo (MCMC) methods [24–26]. The MLSVGD also shares similarities with multilevel sequential Monte Carlo [27–30] and importance sampling [17, 29, 31], multilevel particle filters [32], multilevel preconditioning [33–35], and multilevel ensemble Kalman methods [36, 37], which all use hierarchies of surrogate models to generate samples sequentially. The work [16] provides a cost complexity analysis of MLSVGD that shows speedups compared to single-level SVGD; but it relies on an

exponential convergence rate of SVGD with a fixed parameter and thus is limited in scope.

In this work, we contribute an analysis of MLSVGD that applies when the parameter of the convergence rate depends on the MLSVGD iteration. The finding is that the same cost complexity is achieved as in the fixed-parameter setting as long as mild conditions on the parameter can be made. We also show how the constants in the cost complexity change and that MLSVGD achieves speedups over single-level SVGD when the constants in the convergence rate of SVGD lead to a slow error decay. This is directly applicable to Bayesian inverse problems, where we show that the assumptions of the cost complexity analysis are satisfied in typical settings.

We numerically demonstrate MLSVGD on a Bayesian inverse problem of inferring a discretized basal sliding coefficient field from velocity observations at the surface of the Arolla glacier [38]; see also [39]. The numerical setup builds on FEniCS [40] and hIPPYlib [41–44], which allows for fast gradient-based inference via adjoints. The numerical results show that MLSVGD performs inference at a fraction of the cost of inference with SVGD and that it leads to higher quality particles with respect to the maximum mean discrepancy (MMD) [45] than samples obtained with a variant of MCMC.

The manuscript is organized as follows. In Section 2 we outline preliminaries on Bayesian inverse problems, SVGD, and previous work on MLSVGD. Section 3 introduces extended cost complexity bounds for MLSVGD that apply in more general settings. In Section 4 we demonstrate improvements by several factors in terms of computational savings of MLSVGD over SVGD for inferring the basal sliding coefficient in the Arolla glacier ice model. We conclude in Section 5.

# 2 Preliminaries

In this section, Bayesian inverse problems are reviewed and it is discussed how they are related to sampling from a target distribution with, e.g., SVGD and MLSVGD.

## 2.1 Bayesian inverse problems

Let $G : \Theta \to \mathbb{R}^q$ denote a parameter-to-observable map and consider noisy data $\mathbf{y} = G(\boldsymbol{\theta}^*) + \boldsymbol{\eta}$, where $\boldsymbol{\eta} \sim N(\mathbf{0}, \Gamma)$ with known noise covariance matrix $\Gamma \in \mathbb{R}^{q\times q}$ and $\boldsymbol{\theta}^* \in \Theta \subset \mathbb{R}^d$. Given a prior $\pi_0 : \Theta \to \mathbb{R}$, the target posterior density is

$$\pi(\boldsymbol{\theta}) \propto \exp\left(-\frac{1}{2}\left\|\mathbf{y} - G(\boldsymbol{\theta})\right\|^2_{\Gamma^{-1}}\right)\pi_0(\boldsymbol{\theta})\,, \tag{1}$$

where $\|\boldsymbol{v}\|^2_{\Gamma^{-1}} = \langle \boldsymbol{v},\ \Gamma^{-1}\boldsymbol{v}\rangle$. In many computational science and engineering applications, the parameter-to-observable map $G$ depends on the solution of an underlying system of PDEs, which means that it cannot be evaluated directly. Instead, one must resort to a numerical method that discretizes the underlying PDE problem to approximately evaluate $G$. Let $G^{(\ell)}$ be such an approximate parameter-to-observable map, where the index $\ell$ denotes the fidelity and corresponds to, e.g., the mesh width or number of grid points. The larger $\ell$, the more accurate the approximation $G^{(\ell)}$ of $G$

in the following. The corresponding low-fidelity posterior is

$$\pi^{(\ell)}(\boldsymbol{\theta}) \propto \exp\left(-\frac{1}{2}\left\|\mathbf{y} - G^{(\ell)}(\boldsymbol{\theta})\right\|^2_{\Gamma^{-1}}\right)\pi_0(\boldsymbol{\theta})\,. \tag{2}$$

Increasing the level $\ell$, gives rise to a sequence of densities $(\pi^{(\ell)})_{\ell=1}^{\infty}$ that converges pointwise $\pi^{(\ell)}(\boldsymbol{\theta}) \to \pi(\boldsymbol{\theta})$ for every $\boldsymbol{\theta} \in \Theta$ so that the sequence of random variables $\boldsymbol{\theta}^{(\ell)} \sim \pi^{(\ell)}$ converges weakly to $\boldsymbol{\theta} \sim \pi$.

Our aim is to compute quantities of interest of the form

$$\mathbb{E}_{\pi}[f] = \int_{\Theta} f(\boldsymbol{\theta})\mathrm{d}\pi(\boldsymbol{\theta})\,, \tag{3}$$

for given test functions $f : \Theta \to \mathbb{R}$. Because $\pi$ is not readily available, one typically selects a sufficiently accurate $G^{(L)}$ and approximates the quantity of interest with respect to the corresponding density $\pi^{(L)}$,

$$\mathbb{E}_{\pi^{(L)}}[f] = \int_{\Theta} f(\boldsymbol{\theta})\mathrm{d}\pi^{(L)}(\boldsymbol{\theta})\,. \tag{4}$$

A well-established approach to estimate (4) using Monte Carlo involves drawing samples $\boldsymbol{\theta}^{[1]}, \ldots, \boldsymbol{\theta}^{[N]}$ of the distribution with density $\pi^{(L)}$ and computing

$$\hat{f} = \frac{1}{N}\sum_{i=1}^{N} f(\boldsymbol{\theta}^{[i]})\,. \tag{5}$$

For example, the samples $\boldsymbol{\theta}^{[1]}, \ldots, \boldsymbol{\theta}^{[N]}$ may be i.i.d. or come from a realization of an ergodic Markov chain. This gives rise to two sources of error with respect to the quantity of interest (3). The first source of error is the Monte Carlo error of estimating the expectation in (4) with (5), while the second source of error is due to using the deterministic approximation $G^{(L)}$ of $G$, and thus $\pi^{(L)}$ instead of $\pi$. The Monte Carlo error can be controlled with the number of samples $N$. The second error is controlled by the level $L$, which can be selected via, e.g., the Hellinger distance, whose square is given by

$$\mathrm{d}_{\mathrm{Hell}}\left(\mu_1, \mu_2\right)^2 = \frac{1}{2}\int_{\Theta}\left(\sqrt{\mu_1(\boldsymbol{\theta})} - \sqrt{\mu_2(\boldsymbol{\theta})}\right)^2 \mathrm{d}\boldsymbol{\theta}\,,$$

so that

$$\mathrm{d}_{\mathrm{Hell}}\left(\pi^{(L)}, \pi\right) \leq \epsilon$$

holds for some tolerance $\epsilon > 0$. The Hellinger distance is particularly useful because it is a metric on the space of probability measures, allowing to separate the deterministic error due to the fidelity and the statistical error due to sampling, and can be bounded from above by the KL divergence, defined by

$$\mathrm{KL}\left(\mu_1 \,||\, \mu_2\right) = \int_{\Theta} \log\left(\frac{\mu_1(\boldsymbol{\theta})}{\mu_2(\boldsymbol{\theta})}\right)\mu_1(\boldsymbol{\theta})\,\mathrm{d}\boldsymbol{\theta}\,,$$

using Pinsker's inequality

$$2\mathrm{d}_{\mathrm{Hell}}\left(\mu_1, \mu_2\right)^2 \leq \mathrm{KL}\left(\mu_1 \mid\mid \mu_2\right) . \tag{6}$$

## 2.2 Stein variational gradient descent

We now briefly review SVGD [4] that aims to derive a sequence of distributions to minimize the KL divergence with respect to the target density $\pi^{(L)}$. Once convergence has been reached, the quantity of interest (3) can be estimated using particles of the distribution.

Let $\mathcal{H}$ be a reproducing kernel Hilbert space (RKHS) with positive definite kernel $K : \Theta \times \Theta \to \mathbb{R}$ of functions $g : \Theta \to \mathbb{R}$ and let $\mathcal{H}^d \simeq \mathcal{H} \times \cdots \times \mathcal{H}$ be the corresponding RKHS of vector fields $\mathbf{g} = (g_1, \ldots, g_d) : \Theta^d \to \mathbb{R}^d$. Define the KL functional

$$J_\mu(\mathbf{g}) = \mathrm{KL}\left((\mathbf{I} - \mathbf{g})_{\#}\mu \mid\mid \pi^{(L)}\right) ,$$

where $(\mathbf{I} - \mathbf{g})_{\#}\mu$ denotes the pushforward measure of $\mu$ under the map $\mathbf{I} - \mathbf{g}$, so that if $\boldsymbol{\theta} \sim \mu$, then $\boldsymbol{\theta} - \mathbf{g}(\boldsymbol{\theta}) \sim (\mathbf{I} - \mathbf{g})_{\#}\mu$, with $\mathbf{I} : \mathbb{R}^d \to \mathbb{R}^d$ being the identity map and $\mathbf{g} \in \mathcal{H}^d$. From the particle point of view, SVGD starts with an initial particle $\boldsymbol{\theta}_0 \sim \mu_0$ and evolves it according to the gradient dynamics, also known as the mean-field characteristic flow [12],

$$\dot{\boldsymbol{\theta}}_t = -\nabla J_{\mu_t}(\mathbf{0})(\boldsymbol{\theta}_t) , \tag{7}$$

where $\mu_t$ denotes the density of $\boldsymbol{\theta}_t$ at time $t \geq 0$. The gradient $\nabla J_\mu(\mathbf{0})$ can be computed using the following relation derived in [4]

$$\nabla J_\mu(\mathbf{0})\left(\boldsymbol{\theta}\right) = -\mathbb{E}_{\mathbf{z}\sim\mu}\left[K(\mathbf{z}, \boldsymbol{\theta})\nabla \log \pi^{(L)}(\mathbf{z}) + \nabla_1 K(\mathbf{z}, \boldsymbol{\theta})\right] , \tag{8}$$

where $\nabla_1$ denotes the gradient with respect to the first argument. The density $\mu_t$ is the solution of the nonlinear Fokker-Planck equation corresponding to the particle evolution (7)

$$\partial_t \mu_t(\boldsymbol{\theta}) = -\nabla \cdot \left(\mu_t(\boldsymbol{\theta})\mathbb{E}_{\mathbf{z}\sim\mu_t}\left[K(\mathbf{z}, \boldsymbol{\theta})\nabla \log \pi^{(L)}(\mathbf{z}) + \nabla_1 K(\mathbf{z}, \boldsymbol{\theta})\right]\right) . \tag{9}$$

Much of the analysis of SVGD revolves around understanding the solution $\mu_t$ to the, potentially high-dimensional, nonlinear PDE (9). One key result that arises due to the gradient flow dynamics (9) is that the KL divergence $\mathrm{KL}\left(\mu_t \mid\mid \pi^{(L)}\right)$ converges to zero and it was shown in [11, Theorem 3.4] that for a solution $\mu_t$ of (9) with $\mathrm{KL}\left(\mu_0 \mid\mid \pi^{(L)}\right) < \infty$, it holds that

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathrm{KL}\left(\mu_t \mid\mid \pi^{(L)}\right) = -\mathbb{D}(\mu_t \mid\mid \pi^{(L)})^2 , \tag{10}$$

where

$$\mathbb{D}(\mu \mid\mid \nu) = \max_{\mathbf{g}\in\mathcal{H}^d} \left\{\mathbb{E}_{\boldsymbol{\theta}\sim\mu}[\nabla \log \nu(\boldsymbol{\theta})^\top \mathbf{g}(\boldsymbol{\theta}) + \nabla \cdot \mathbf{g}(\boldsymbol{\theta})] : \|\mathbf{g}\|_{\mathcal{H}} \leq 1\right\}$$

is the Stein discrepancy, guaranteeing that the KL divergence from the target decreases monotonically. The result (10) provides motivation for considering a monotone convergence behavior as in Assumption 5 later. The Stein discrepancy $\mathbb{D}(\mu \mid\mid \nu) = 0$ if $\mu = \nu$, but the converse may only be valid if the space $\mathcal{H}$ is sufficiently rich and can otherwise result in a biased estimate of the quantity of interest (3).

**Remark 1.** *There is a strong connection between SVGD and the unadjusted Langevin algorithm [46] in the sense that the Langevin algorithm evolves a density that minimizes the KL divergence in the Wasserstein metric as opposed to a SVGD that uses a kernelized Wasserstein metric [13].*

## 2.3 Multilevel Stein variational gradient descent

The work [16] introduced a multilevel variant of SVGD and showed that one can achieve a cost complexity reduction by integrating the continuous-time mean-field flow (7) with successively more accurate and more expensive-to-evaluate low-fidelity densities $\pi^{(1)}, \ldots, \pi^{(L)}$ as opposed to integrating only with respect to the high-fidelity density $\pi^{(L)}$. The analysis in [16] of the cost complexity relied on the following assumptions.

**Assumption 1.** *The costs $c_\ell$ of integrating (9) in time with target density $\pi^{(\ell)}$ for a unit time interval are bounded as*

$$c_\ell \leq c_0 s^{\gamma \ell}, \qquad \ell \in \mathbb{N},$$

*with constants $c_0, \gamma > 0$ independent of $\ell$ and $s > 1$.*

**Assumption 2.** *There exists $\alpha, k_0, k_1 > 0$ independent of $\ell$ such that $\mathrm{KL}\left(\mu_0 \mid\mid \pi^{(\ell)}\right) \leq k_0$ for all $\ell \in \mathbb{N}$ and*

$$\mathrm{KL}\left(\pi^{(\ell)} \mid\mid \pi\right) \leq k_1 s^{-\alpha \ell}, \qquad \ell \in \mathbb{N},$$

*where $s$ is the same constant independent of $\ell$ as in Assumption 1 and $\mu_0$ is the initial distribution.*

**Assumption 3.** *There exists a rate $\lambda > 0$ such that for any initial distribution $\nu_0$*

$$\mathrm{KL}\left(\nu_t \mid\mid \pi^{(\ell)}\right) \leq \mathrm{e}^{-\lambda t} \mathrm{KL}\left(\nu_0 \mid\mid \pi^{(\ell)}\right), \qquad \ell \in \mathbb{N}, \tag{11}$$

*holds, where $\nu_t$ solves the mean-field SVGD equation (9) at time $t$.*

Single-level SVGD derives an approximation $\mu^{\mathrm{SL}}$ such that $\mathrm{d}_{\mathrm{Hell}}\left(\mu^{\mathrm{SL}}, \pi\right) \leq \epsilon$, by selecting a high-fidelity approximation $\pi^{(L)}$ with

$$\mathrm{d}_{\mathrm{Hell}}\left(\pi^{(L)}, \pi\right) \leq \epsilon/2 \tag{12}$$

and then integrating (9) with respect to $\pi^{(L)}$ for time $T_{\mathrm{SL}}(\epsilon)$

$$T_{\mathrm{SL}}(\epsilon) = \min\left\{t \geq 0 \ : \ \mathrm{d}_{\mathrm{Hell}}\left(\mu_t, \pi^{(L)}\right) \leq \frac{\epsilon}{2}\right\}. \tag{13}$$

This leads to the cost of single-level SVGD

$$c_{\mathrm{SL}}(\epsilon) = c_{L(\epsilon)} T_{\mathrm{SL}}(\epsilon)\,,$$

where the cost $c_{L(\epsilon)}$ depends on $\epsilon$ through the level $L$ that is selected such that (12) holds. In the remainder of this manuscript, for brevity, we drop the explicit dependence $L = L(\epsilon)$ and similarly $T_{\mathrm{SL}} = T_{\mathrm{SL}}(\epsilon)$ when $\epsilon$ is fixed. The following upper bound for the cost complexity of single-level SVGD was derived in [16, Proposition 2].

**Proposition 1.** *If Assumptions 1–3 hold, then the costs of single-level SVGD to obtain $\mu^{\mathrm{SL}}$ with*

$$\mathrm{d}_{\mathrm{Hell}}\left(\mu^{\mathrm{SL}}, \pi\right) \leq \epsilon\,,$$

*is bounded as*

$$c_{\mathrm{SL}}(\epsilon) \leq \frac{2c_0 s^\gamma}{\lambda} \left(\frac{\sqrt{2k_1}}{\epsilon}\right)^{2\gamma/\alpha} \log\left(\frac{\sqrt{\mathrm{KL}\left(\mu_0 \mid\mid \pi^{(L)}\right)}}{\sqrt{2}\epsilon}\right)\,, \tag{14}$$

*with high-fidelity level*

$$L = \left\lceil \frac{1}{2\alpha} \log_s\left(\frac{\sqrt{2k_1}}{\epsilon}\right) \right\rceil\,. \tag{15}$$

From (14), a higher initial KL divergence $\mathrm{KL}\left(\mu_0 \mid\mid \pi^{(L)}\right)$ or a slower convergence rate (small $\lambda$) for SVGD will result in a larger cost complexity to obtain the single-level SVGD approximation of $\pi^{(L)}$.

In contrast to single-level SVGD, the MLSVGD method introduced in [16] first integrates with respect to the cheapest and least accurate lowest fidelity density $\pi^{(1)}$ for time $T_1 > 0$ to obtain density $\mu^{(1)}_{T_1}$, which serves as an initial density for the next level and so on until the highest level $L$ is reached. For $\ell = 1, \ldots, L$, let $\mu^{(\ell)}_{T_\ell}$ be the solution of (9), with the low-fidelity density $\pi^{(\ell)}$ replacing the target $\pi$, at time $T_\ell$ with initial density $\mu^{(\ell)}_0 = \mu^{(\ell-1)}_{T_{\ell-1}}$ where the times $T_\ell$ are given by

$$T_\ell = \min\left\{ t \geq 0 \;:\; \mathrm{KL}\left(\mu^{(\ell)}_t \mid\mid \pi^{(\ell)}\right) \leq \frac{\epsilon_\ell^2}{2} \right\}\,, \tag{16}$$

where $\epsilon_1 \geq \epsilon_2 \geq \ldots \geq \epsilon_L$ and $\epsilon_L \leq \epsilon$ is a sequence of tolerances. Then, the continuous-time MLSVGD approximation is defined as

$$\mu^{\mathrm{ML}} = \mu^{(L)}_{T_L}\,,$$

which gives the cost of MLSVGD as

$$c_{\mathrm{ML}}(\epsilon) = \sum_{\ell=1}^{L} c_\ell T_\ell\,,$$

where both $L$ and $T_\ell$ will depend on $\epsilon$. Since the KL divergence does not satisfy the triangle inequality, the following assumption for MLSVGD ensures that the KL divergence between levels converges as well, which is different from Assumption 2.

**Assumption 4.** *There exists a constant $k_2 > 0$ independent of $\ell$ such that $\mathrm{KL}\left(\pi^{(\ell-1)} \,||\, \pi^{(\ell)}\right) \leq k_2 s^{-\alpha\ell}$, where $\alpha$ is the same rate as in Assumption 2.*

With these additional assumptions one can derive the cost complexity for MLSVGD [16, Proposition 4] below.

**Proposition 2.** *If Assumptions 1–4 hold and $R_\ell \leq k_3 s^{-\alpha\ell}$ and*

$$\epsilon_\ell = \sqrt{2k_1} s^{-\alpha\ell/2}\,,$$

*where*

$$R_\ell = \int_{\mathbb{R}^d} \left(\mu^{(\ell-1)}_{T_{\ell-1}}(\boldsymbol{\theta}) - \pi^{(\ell-1)}(\boldsymbol{\theta})\right) \log\left(\frac{\pi^{(\ell-1)}(\boldsymbol{\theta})}{\pi^{(\ell)}(\boldsymbol{\theta})}\right) \mathrm{d}\boldsymbol{\theta}\,, \tag{17}$$

*then the costs of MLSVGD to have $\mathrm{d_{Hell}}\left(\mu^{\mathrm{ML}}, \pi\right) \leq \epsilon$ can be bounded as*

$$c_{\mathrm{ML}}(\epsilon) \leq \frac{c_0 s^{2\gamma}}{\lambda\gamma \log(s)} \log\left(s^\alpha + \frac{k_2 + k_3}{k_1}\right) \left(\frac{\sqrt{2k_1}}{\epsilon}\right)^{2\gamma/\alpha}. \tag{18}$$

The cost complexity of MLSVGD scales at most as $\mathcal{O}(\epsilon^{-2\gamma/\alpha})$, whereas the cost complexity for single-level SVGD has an additional $\log \epsilon^{-1}$ factor. Furthermore, the bound (18) is independent of the KL divergence of the initial density $\mu_0$ and instead only depends on the constant $k_2$ that measures the KL divergence between successive levels.

# 3 Further analysis of MLSVGD

We now extend the analysis of MLSVGD to apply in settings where SVGD exhibits an exponential convergence rate with a varying parameter.

## 3.1 Cost bound for MLSVGD

We now consider a relaxed assumption on the convergence rate that includes having $\lambda(t) \geq 0$ depend on time $t$ so that the multiplicative factor in (11) becomes $\mathrm{e}^{-\lambda(t)t}$. The following assumption formalizes the time-dependent convergence factor as $r(t)$, which includes the case with factor $\mathrm{e}^{-\lambda(t)t}$.

**Assumption 5.** *There exists a decreasing function $r : [0, \infty) \to [0, 1]$ such that $r(0) = 1$, $\lim_{t\to\infty} r(t) = 0$, and for an initial distribution $\nu_0$*

$$\mathrm{KL}\left(\nu_t \,||\, \pi^{(\ell)}\right) \leq r(t)\mathrm{KL}\left(\nu_0 \,||\, \pi^{(\ell)}\right), \qquad \ell \in \mathbb{N}\,,$$

*holds, where $\nu_t$ is the solution of the mean-field SVGD equation (9) at time $t$.*

When Equation (10) holds, the KL divergence is monotone decreasing, and for any fixed $\ell \in \mathbb{N}$ and initial distribution $\nu_0$, the inequality in Assumption 5 is satisfied.

Assumption 5 is stronger in that it requires the inequality to hold uniformly for all levels $\ell$ and initial distributions $\nu_0$. The uniformness implies that there exists a $\lambda > 0$ with $\lambda(t) \geq \lambda$ so that the analysis of [16] applies. However, we obtain a tighter bound in terms of constants in the following. In the case where $r$ is not invertible due to a discontinuity, we define

$$r^{-1}(\epsilon) = \min \{t \in [0, \infty) \ : \ r(t) \leq \epsilon\} \ . \tag{19}$$

We now derive a result analogous to Proposition 1.

**Proposition 3.** *If Assumptions 1,2,5 hold, then the costs of SVGD to obtain $\mu^{\mathrm{SL}}$ with*

$$\mathrm{d}_{\mathrm{Hell}}\left(\mu^{\mathrm{SL}}, \pi\right) \leq \epsilon \, ,$$

*is bounded as*

$$\mathrm{c}_{\mathrm{SL}}(\epsilon) \leq c_0 s^{\gamma L} T_{\mathrm{SL}} \leq c_0 s^{\gamma} (2k_1)^{\gamma/\alpha} r^{-1}\left(\frac{\epsilon^2}{2\mathrm{KL}\left(\mu_0 \mid\mid \pi^{(L)}\right)}\right) \epsilon^{-2\gamma/\alpha} \, . \tag{20}$$

*Proof.* By the triangle inequality for the Hellinger distance we have that

$$\mathrm{d}_{\mathrm{Hell}}\left(\mu^{\mathrm{SL}}, \pi\right) \leq \mathrm{d}_{\mathrm{Hell}}\left(\mu^{\mathrm{SL}}, \pi^{(L)}\right) + \mathrm{d}_{\mathrm{Hell}}\left(\pi^{(L)}, \pi\right),$$

so we will bound both of these terms independently by $\epsilon/2$. By inequality (6), it is sufficient to bound the KL divergence because

$$\mathrm{d}_{\mathrm{Hell}}\left(\mu^{\mathrm{SL}}, \pi^{(L)}\right) \leq \sqrt{\frac{\mathrm{KL}\left(\mu^{\mathrm{SL}} \mid\mid \pi^{(L)}\right)}{2}}, \tag{21}$$

and similarly for $\mathrm{d}_{\mathrm{Hell}}\left(\pi^{(L)}, \pi\right)$. By Assumption 2 choose $L$ to be

$$L = \left\lceil \frac{1}{\alpha} \log_s\left(\frac{2k_1}{\epsilon^2}\right) \right\rceil \leq \frac{1}{\alpha} \log_s\left(\frac{2k_1}{\epsilon^2}\right) + 1, \tag{22}$$

so that

$$\mathrm{d}_{\mathrm{Hell}}\left(\pi^{(L)}, \pi\right) \leq \sqrt{\frac{\mathrm{KL}\left(\pi^{(L)} \mid\mid \pi\right)}{2}} \leq \sqrt{\frac{k_1 s^{-\alpha L}}{2}} \leq \frac{\epsilon}{2}. \tag{23}$$

The time needed to integrate with SVGD to achieve $\mathrm{d}_{\mathrm{Hell}}\left(\mu^{\mathrm{SL}}, \pi^{(L)}\right) \leq \epsilon/2$ is

$$T_{\mathrm{SL}} = \min\left\{t \geq 0 \ : \ \mathrm{d}_{\mathrm{Hell}}\left(\mu_t, \pi^{(L)}\right) \leq \frac{\epsilon}{2}\right\} \, .$$

Again by inequality (6),

$$T_{\mathrm{SL}} \leq \min\left\{t \geq 0 \ : \ \mathrm{KL}\left(\mu_t \mid\mid \pi^{(L)}\right) \leq \frac{\epsilon^2}{2}\right\}.$$

Now by Assumption 5, the rate function $r$ is invertible, or by applying the definition (19) of $r^{-1}$, and the time needed to integrate with SVGD to achieve $\mathrm{d}_{\mathrm{Hell}}\left(\mu^{\mathrm{SL}}, \pi^{(L)}\right) \leq \epsilon/2$ is bounded as

$$T_{\mathrm{SL}} \leq r^{-1}\left(\frac{\epsilon^2}{2\mathrm{KL}\left(\mu_0 \mid\mid \pi^{(L)}\right)}\right). \tag{24}$$

With Assumption 1, the total cost to integrate until time $T_{\mathrm{SL}}$ at level $L$ is therefore bounded as

$$\mathrm{c}_{\mathrm{SL}}(\epsilon) \leq c_0 s^{\gamma L} T_{\mathrm{SL}} \leq c_0 s^{\gamma} (2k_1)^{\gamma/\alpha} r^{-1}\left(\frac{\epsilon^2}{2\mathrm{KL}\left(\mu_0 \mid\mid \pi^{(L)}\right)}\right) \epsilon^{-2\gamma/\alpha}.$$

□

**Remark 2.** *If the Hellinger distance is split differently so that*

$$\mathrm{d}_{\mathrm{Hell}}\left(\pi^{(L)}, \pi\right) \leq \delta\epsilon$$

*and*

$$\mathrm{d}_{\mathrm{Hell}}\left(\mu^{SL}, \pi^{(L)}\right) \leq (1-\delta)\epsilon$$

*for $\delta \in (0,1)$, as opposed to $\delta = 1/2$, then by following the same steps as in the proof of Proposition 3, one can derive a bound that is analogous to (20) but with different constants. Since $r(t)$ and the initial KL divergence are unknown in general, it is not practical to optimize $\delta$ and we choose $\delta = 1/2$ for simplicity.*

As in Proposition 1 we see that the cost complexity depends on the tolerance $\epsilon$, the KL divergence of the initial distribution $\mu_0$ from the high-fidelity density $\pi^{(L)}$, as well as the SVGD convergence rate. Because the rate function $r$ is decreasing, its inverse $r^{-1}$ is also decreasing and so a larger initial KL divergence will require a longer integration time. We also derive a new cost complexity for the more general convergence behavior for MLSVGD in the following proposition.

**Proposition 4.** *If Assumptions 1, 2, 4, and 5 hold and $R_\ell \leq k_3 s^{-\alpha\ell}$, then by setting $\epsilon_\ell = \sqrt{2k_1}\, s^{-\alpha\ell/2}$ for $\ell = 1, \ldots, L$, the costs of MLSVGD to have $\mathrm{d}_{\mathrm{Hell}}\left(\mu^{\mathrm{ML}}, \pi\right) \leq \epsilon$ can be bounded as*

$$c_{\mathrm{ML}}(\epsilon) \leq \frac{c_0 s^{2\gamma} (2k_1)^{\gamma/\alpha}}{s^{\gamma} - 1} r^{-1}\left(\frac{1}{s^{\alpha} + (k_2 + k_3)/k_1}\right) \epsilon^{-2\gamma/\alpha}. \tag{25}$$

*Proof.* As in Equation (22) in the proof of Proposition 3 we select the level $L$ as

$$L = \left\lceil \frac{1}{\alpha} \log_s \left( \frac{2k_1}{\epsilon^2} \right) \right\rceil \leq \frac{1}{\alpha} \log_s \left( \frac{2k_1}{\epsilon^2} \right) + 1, \tag{26}$$

so that $\mathrm{d}_{\mathrm{Hell}} \left( \pi^{(L)}, \pi \right) \leq \epsilon/2$. Note that by setting $\epsilon_\ell = \sqrt{2k_1} s^{-\alpha \ell / 2}$ for $\ell = 1, \ldots, L$ in (16) we have that

$$\frac{\epsilon_L^2}{2} = k_1 s^{-\alpha L} \leq \frac{\epsilon}{2},$$

by the choice of the high-fidelity level $L$ (22).

By Assumption 1, the total cost for MLSVGD is bounded by

$$c_{\mathrm{ML}}(\epsilon) \leq \sum_{\ell=1}^{L} c_0 s^{\gamma \ell} T_\ell \,, \tag{27}$$

where it remains to bound the integration times $T_\ell$ at each level. By Assumption 5 and Equation (28), we have

$$\begin{aligned} \mathrm{KL} \left( \mu_{T_\ell}^{(\ell)} \mid\mid \pi^{(\ell)} \right) &\leq r(T_\ell) \mathrm{KL} \left( \mu_{T_{\ell-1}}^{(\ell-1)} \mid\mid \pi^{(\ell)} \right) \\ &= r(T_\ell) \left( \mathrm{KL} \left( \mu_{T_{\ell-1}}^{(\ell-1)} \mid\mid \pi^{(\ell-1)} \right) + \mathrm{KL} \left( \pi^{(\ell-1)} \mid\mid \pi^{(\ell)} \right) + R_\ell \right) , \end{aligned} \tag{28}$$

giving a recursive bound on the KL divergence in terms of the KL divergence at the previous level. By the definition (16) of the integration times $T_\ell$ at level $\ell$, we know that

$$\mathrm{KL} \left( \mu_{T_\ell}^{(\ell)} \mid\mid \pi^{(\ell)} \right) \leq \frac{\epsilon_\ell^2}{2} \,, \tag{29}$$

is satisfied for each level $\ell = 1, \ldots, L$. Using (29) at level $\ell - 1$ gives

$$\mathrm{KL} \left( \mu_{T_\ell}^{(\ell)} \mid\mid \pi^{(\ell)} \right) \leq r(T_\ell) \left( \frac{\epsilon_{\ell-1}^2}{2} + \mathrm{KL} \left( \pi^{(\ell-1)} \mid\mid \pi^{(\ell)} \right) + R_\ell \right) . \tag{30}$$

Note that by (29) we know that the left-hand-side of (30) is guaranteed to be bounded above by $\epsilon_\ell^2/2$, but the same is not necessarily true for the right-hand-side which is an upper bound. Instead define $T'_\ell$ as

$$T'_\ell = \min \left\{ t \geq 0 \;:\; r(t) \left( \frac{\epsilon_{\ell-1}^2}{2} + \mathrm{KL} \left( \pi^{(\ell-1)} \mid\mid \pi^{(\ell)} \right) + R_\ell \right) \leq \frac{\epsilon_\ell^2}{2} \right\} , \tag{31}$$

for each level $\ell = 1, \ldots, L$, which is finite by the assumption that $r(t) \to 0$ (Assumption 5). By (30) and because $r$ is monotonically decreasing we know that $T_\ell \leq T'_\ell$. Solving directly gives

$$T'_\ell \leq r^{-1} \left( \frac{\epsilon_\ell^2}{\epsilon_{\ell-1}^2 + 2\mathrm{KL} \left( \pi^{(\ell-1)} \mid\mid \pi^{(\ell)} \right) + 2R_\ell} \right) . \tag{32}$$

We now use the fact that $r^{-1}$ is decreasing as well as Assumption 4 and the assumption that $R_\ell \leq k_3 s^{-\alpha\ell}$ to bound

$$r^{-1}\left(\frac{\epsilon_\ell^2}{\epsilon_{\ell-1}^2 + 2\mathrm{KL}\left(\pi^{(\ell-1)} \mid\mid \pi^{(\ell)}\right) + 2R_\ell}\right) \leq r^{-1}\left(\frac{\epsilon_\ell^2}{\epsilon_{\ell-1}^2 + 2k_2 s^{-\alpha\ell} + 2k_3 s^{-\alpha\ell}}\right).$$

Therefore, by substituting $\epsilon_\ell = \sqrt{2k_1}s^{-\alpha\ell/2}$ (and similarly for $\epsilon_{\ell-1}$) we can bound $T'_\ell$, and hence $T_\ell$, with

$$T_\ell \leq r^{-1}\left(\frac{2k_1 s^{-\alpha\ell}}{2k_1 s^\alpha s^{-\alpha\ell} + 2k_2 s^{-\alpha\ell} + 2k_3 s^{-\alpha\ell}}\right).$$

Simplifying gives the bound

$$T_\ell \leq r^{-1}\left(\frac{1}{s^\alpha + (k_2+k_3)/k_1}\right), \tag{33}$$

which is independent of the tolerance $\epsilon$. The total cost can now be bounded by

$$c_{\mathrm{ML}}(\epsilon) \leq \sum_{\ell=1}^{L} c_0 s^{\gamma\ell} r^{-1}\left(\frac{1}{s^\alpha + (k_2+k_3)/k_1}\right), \tag{34}$$

which we may again compute explicitly

$$\begin{aligned} c_{\mathrm{ML}}(\epsilon) &\leq c_0 s^\gamma r^{-1}\left(\frac{1}{s^\alpha + (k_2+k_3)/k_1}\right)\frac{s^{\gamma L}-1}{s^\gamma - 1} \\ &\leq c_0 s^\gamma r^{-1}\left(\frac{1}{s^\alpha + (k_2+k_3)/k_1}\right)\frac{s^{\gamma L}}{s^\gamma - 1}, \end{aligned} \tag{35}$$

and we have again added 1 in the numerator of the last term for convenience. Substituting the upper bound (22) on the level $L$ and simplifying terms gives the final upper bound on the improved cost complexity of the MLSVGD approximation $\mu^{\mathrm{ML}}$

$$c_{\mathrm{ML}}(\epsilon) \leq \frac{c_0 s^{2\gamma}(2k_1)^{\gamma/\alpha}}{s^\gamma - 1} r^{-1}\left(\frac{1}{s^\alpha + (k_2+k_3)/k_1}\right)\epsilon^{-2\gamma/\alpha}. \tag{36}$$

□

By setting $r(t) = \mathrm{e}^{-\lambda t}$ as in Assumption 3, one can recover the cost complexities stated in Section 2.3. When compared to (20), if SVGD is slow to converge then the MLSVGD can spend most of the integration time at the lower levels, which can be faster to integrate, in order to find a good initial density for integrating with respect to the highest level $L$ and so potentially achieve speedups. In contrast, if SVGD converges quickly then the low-fidelity densities will be less beneficial and both costs will be comparable.

### 3.2 Cost complexity for Bayesian inverse problems

The results from Section 3.1 are applicable in Bayesian inverse problem settings. Recall that typically in Bayesian inverse problems, the sequence of posterior distributions $(\pi^{(\ell)})$ is obtained via a sequence of approximate parameter-to-observable maps $(G^{(\ell)})_{\ell=1}^{\infty}$ with $G^{(\ell)}(\boldsymbol{\theta}) \to G(\boldsymbol{\theta})$ pointwise for every $\boldsymbol{\theta} \in \Theta$, so that the sequence of densities $(\pi^{(\ell)})$ converges pointwise as well. As shown in [16], the following assumption on the parameter-to-observable maps ensures that the KL divergences of the densities converges as required in Assumptions 2 and 4.

**Assumption 6.** *The error of the approximate parameter-to-observable $G^{(\ell)}$ map at level $\ell \geq 1$ is bounded by*

$$\left\| G(\boldsymbol{\theta}) - G^{(\ell)}(\boldsymbol{\theta}) \right\|_{L^2(\pi_0)} \leq b_0 s^{-\alpha \ell}\,, \tag{37}$$

*where $\alpha, b_0 > 0$ and $s > 1$ are constants with $s$ the same as in Assumption 1 and $\|\cdot\|_{L^2(\pi_0)}$ is the $L^2$ norm over $\pi_0$.*

As long as the SVGD approximations $\mu_{T_\ell}^{(\ell)}$ remain absolutely continuous at each level with respect to the prior density $\pi_0$, the remainders $R_\ell$ defined in (17) can be bounded and thus the cost bound with the same rate as in Proposition 4 applies for approximating the Bayesian posterior; see [16, Theorem 1]. We now state this result formally.

**Proposition 5.** *Let Assumptions 1, 5,and 6 hold. Furthermore, assume that there exists a constant $b_3 > 0$ independent of $\ell$ such that*

$$\mu_{T_\ell}^{(\ell)}(\boldsymbol{\theta}) \leq b_3 \pi_0(\boldsymbol{\theta})\,, \quad \boldsymbol{\theta} \in \Theta\,, \tag{38}$$

*for all $\ell \geq 1$. Then, the cost complexity of finding $\mu^{\mathrm{ML}}$ with $\mathrm{d}_{\mathrm{Hell}}\left(\mu^{\mathrm{ML}}, \pi\right) \leq \epsilon$ is*

$$c_{\mathrm{ML}}(\epsilon) \leq \frac{c_0 s^{2\gamma} (3 b_1 b_2 b_0)^{\gamma/\alpha}}{s^\gamma - 1} r^{-1} \left( \frac{1}{s^\alpha + (1+s^\alpha)(4 + 3b_3/b_2)} \right) \epsilon^{-2\gamma/\alpha}\,, \tag{39}$$

*where the constants*

$$b_1 = \sup_{\ell \geq 1} \left\| \Gamma^{-1} \left( 2\mathbf{y} - G^{(\ell-1)} - G^{(\ell)} \right) \right\|_{L^2(\pi_0)} \tag{40}$$

*and*

$$b_2 = \sup_{\ell \geq 1} \frac{1}{Z_\ell} \tag{41}$$

*are independent of $\epsilon$.*

*Proof.* Because Assumption 6 holds, by [16, Lemmas 7 and 8] we know that Assumptions 2 and 4 hold with $k_1 = \frac{3}{2} b_0 b_1 b_2$ and $k_2 = \frac{3}{2} b_0 b_1 b_2 (1 + s^\alpha)$. Thus, we just need to verify that $R_\ell \leq k_3 s^{-\alpha \ell}$ holds for a constant $k_3$ to apply Proposition 4. Using

definitions given in (1) and (17), we make the following transformations

$$\begin{aligned} R_\ell &= \int_\Theta \left(\mu_{T_{\ell-1}}^{(\ell-1)}(\boldsymbol{\theta}) - \pi^{(\ell-1)}(\boldsymbol{\theta})\right) \log\left(\frac{\pi^{(\ell-1)}(\boldsymbol{\theta})}{\pi^{(\ell)}(\boldsymbol{\theta})}\right) \,\mathrm{d}\boldsymbol{\theta} \\ &= \int_\Theta \left(\mu_{T_{\ell-1}}^{(\ell-1)}(\boldsymbol{\theta}) - \pi^{(\ell-1)}(\boldsymbol{\theta})\right) \log\left(\frac{Z_\ell \exp\left(-\frac{1}{2}\|\mathbf{y} - G^{(\ell-1)}(\boldsymbol{\theta})\|_{\Gamma^{-1}}^2\right)}{Z_{\ell-1} \exp\left(-\frac{1}{2}\|\mathbf{y} - G^{(\ell)}(\boldsymbol{\theta})\|_{\Gamma^{-1}}^2\right)}\right) \,\mathrm{d}\boldsymbol{\theta} \\ &= \int_\Theta \left(\mu_{T_{\ell-1}}^{(\ell-1)}(\boldsymbol{\theta}) - \pi^{(\ell-1)}(\boldsymbol{\theta})\right) \log\left(\frac{\exp\left(-\frac{1}{2}\|\mathbf{y} - G^{(\ell-1)}(\boldsymbol{\theta})\|_{\Gamma^{-1}}^2\right)}{\exp\left(-\frac{1}{2}\|\mathbf{y} - G^{(\ell)}(\boldsymbol{\theta})\|_{\Gamma^{-1}}^2\right)}\right) \,\mathrm{d}\boldsymbol{\theta}, \end{aligned} \tag{42}$$

where $Z_\ell$ and $Z_{\ell-1}$ are the normalizing constants of $\pi^{(\ell)}$ and $\pi^{(\ell-1)}$, respectively, so that

$$Z_\ell = \int_\Theta \exp\left(-\frac{1}{2}\|\mathbf{y} - G^{(\ell)}(\boldsymbol{\theta})\|_{\Gamma^{-1}}^2\right) \pi_0(\boldsymbol{\theta}) \,\mathrm{d}\boldsymbol{\theta}\,. \tag{43}$$

The last line of (42) follows from the fact that

$$\int_\Theta \left(\mu_{T_{\ell-1}}^{(\ell-1)}(\boldsymbol{\theta}) - \pi^{(\ell-1)}(\boldsymbol{\theta})\right) \log\left(\frac{Z_\ell}{Z_{\ell-1}}\right) \,\mathrm{d}\boldsymbol{\theta} = 0 \tag{44}$$

since $\frac{Z_\ell}{Z_{\ell-1}}$ is constant in $\boldsymbol{\theta}$ and $\pi^{(\ell-1)}$ and $\mu_{T_{\ell-1}}^{(\ell-1)}$ both integrate to one. Simplifying the expression for $R_\ell$ gives

$$R_\ell = \frac{1}{2}\int_\Theta \left(\mu_{T_{\ell-1}}^{(\ell-1)}(\boldsymbol{\theta}) - \pi^{(\ell-1)}(\boldsymbol{\theta})\right) \left(\|\mathbf{y} - G^{(\ell)}(\boldsymbol{\theta})\|_{\Gamma^{-1}}^2 - \|\mathbf{y} - G^{(\ell-1)}(\boldsymbol{\theta})\|_{\Gamma^{-1}}^2\right) \,\mathrm{d}\boldsymbol{\theta}\,. \tag{45}$$

By taking the absolute value and applying the triangle inequality we have that

$$\begin{aligned} R_\ell \leq \frac{1}{2}\int_\Theta \left|\|\mathbf{y} - G^{(\ell)}(\boldsymbol{\theta})\|_{\Gamma^{-1}}^2 - \|\mathbf{y} - G^{(\ell-1)}(\boldsymbol{\theta})\|_{\Gamma^{-1}}^2\right| \mu_{T_{\ell-1}}^{(\ell-1)}(\boldsymbol{\theta}) \,\mathrm{d}\boldsymbol{\theta} \\ + \frac{1}{2}\int_\Theta \left|\|\mathbf{y} - G^{(\ell)}(\boldsymbol{\theta})\|_{\Gamma^{-1}}^2 - \|\mathbf{y} - G^{(\ell-1)}(\boldsymbol{\theta})\|_{\Gamma^{-1}}^2\right| \pi^{(\ell-1)}(\boldsymbol{\theta}) \,\mathrm{d}\boldsymbol{\theta}\,. \end{aligned} \tag{46}$$

Additionally, since

$$\exp\left(-\frac{1}{2}\|\mathbf{y} - G^{(\ell)}(\boldsymbol{\theta})\|_{\Gamma^{-1}}^2\right) \leq 1\,,$$

we have

$$\pi^{(\ell)}(\boldsymbol{\theta}) = \frac{1}{Z_\ell}\exp\left(-\frac{1}{2}\|\mathbf{y} - G^{(\ell)}(\boldsymbol{\theta})\|_{\Gamma^{-1}}^2\right)\pi_0(\boldsymbol{\theta}) \leq \frac{1}{Z_\ell}\pi_0(\boldsymbol{\theta})\,. \tag{47}$$

Therefore, by combining (47) for $\pi^{(\ell-1)}(\boldsymbol{\theta}) \leq \pi_0(\boldsymbol{\theta})/Z_{\ell-1}$ with (38) we get

$$\begin{aligned} R_\ell &\leq \frac{1}{2}\int_\Theta \left| \|\mathbf{y} - G^{(\ell)}(\boldsymbol{\theta})\|^2_{\Gamma^{-1}} - \|\mathbf{y} - G^{(\ell-1)}(\boldsymbol{\theta})\|^2_{\Gamma^{-1}} \right| \mu^{(\ell-1)}_{T_{\ell-1}}(\boldsymbol{\theta})\, \mathrm{d}\boldsymbol{\theta} \\ &\quad + \frac{1}{2}\int_\Theta \left| \|\mathbf{y} - G^{(\ell)}(\boldsymbol{\theta})\|^2_{\Gamma^{-1}} - \|\mathbf{y} - G^{(\ell-1)}(\boldsymbol{\theta})\|^2_{\Gamma^{-1}} \right| \pi^{(\ell-1)}(\boldsymbol{\theta})\, \mathrm{d}\boldsymbol{\theta} \\ &\leq \frac{b_3}{2}\int_\Theta \left| \|\mathbf{y} - G^{(\ell)}(\boldsymbol{\theta})\|^2_{\Gamma^{-1}} - \|\mathbf{y} - G^{(\ell-1)}(\boldsymbol{\theta})\|^2_{\Gamma^{-1}} \right| \pi_0(\boldsymbol{\theta})\, \mathrm{d}\boldsymbol{\theta} \\ &\quad + \frac{1}{2Z_{\ell-1}}\int_\Theta \left| \|\mathbf{y} - G^{(\ell)}(\boldsymbol{\theta})\|^2_{\Gamma^{-1}} - \|\mathbf{y} - G^{(\ell-1)}(\boldsymbol{\theta})\|^2_{\Gamma^{-1}} \right| \pi_0(\boldsymbol{\theta})\, \mathrm{d}\boldsymbol{\theta}\,. \end{aligned} \tag{48}$$

Re-writing the expression inside the absolute value in the integrand gives

$$\begin{aligned} &\left| \|\mathbf{y} - G^{(\ell)}(\boldsymbol{\theta})\|^2_{\Gamma^{-1}} - \|\mathbf{y} - G^{(\ell-1)}(\boldsymbol{\theta})\|^2_{\Gamma^{-1}} \right| \\ &= \left| \left\| G^{(\ell)}(\boldsymbol{\theta}) - G^{(\ell-1)}(\boldsymbol{\theta}) + G^{(\ell-1)}(\boldsymbol{\theta}) - \mathbf{y} \right\|^2_{\Gamma^{-1}} - \|\mathbf{y} - G^{(\ell-1)}(\boldsymbol{\theta})\|^2_{\Gamma^{-1}} \right| \\ &= \Big| \left\langle G^{(\ell)}(\boldsymbol{\theta}) - G^{(\ell-1)}(\boldsymbol{\theta}),\ \Gamma^{-1}(G^{(\ell)}(\boldsymbol{\theta}) - G^{(\ell-1)}(\boldsymbol{\theta})) \right\rangle \\ &\qquad\qquad + 2\left\langle G^{(\ell)}(\boldsymbol{\theta}) - G^{(\ell-1)}(\boldsymbol{\theta}),\ \Gamma^{-1}(G^{(\ell-1)}(\boldsymbol{\theta}) - \mathbf{y}) \right\rangle \Big| \\ &= \left| \left\langle G^{(\ell)}(\boldsymbol{\theta}) - G^{(\ell-1)}(\boldsymbol{\theta}),\ \Gamma^{-1}(G^{(\ell)}(\boldsymbol{\theta}) + G^{(\ell-1)}(\boldsymbol{\theta}) - 2\mathbf{y}) \right\rangle \right| \\ &\leq \left\| G^{(\ell)}(\boldsymbol{\theta}) - G^{(\ell-1)}(\boldsymbol{\theta}) \right\| \cdot \left\| \Gamma^{-1}(2\mathbf{y} - G^{(\ell)}(\boldsymbol{\theta}) - G^{(\ell-1)}(\boldsymbol{\theta})) \right\|\,, \end{aligned} \tag{49}$$

where we have applied the Cauchy-Schwarz inequality to obtain the last line. From the Cauchy-Schwarz inequality on $L^2(\pi_0)$

$$\begin{aligned} &\int_\Theta \left| \|\mathbf{y} - G^{(\ell)}(\boldsymbol{\theta})\|^2_{\Gamma^{-1}} - \|\mathbf{y} - G^{(\ell-1)}(\boldsymbol{\theta})\|^2_{\Gamma^{-1}} \right| \pi_0(\boldsymbol{\theta})\, \mathrm{d}\boldsymbol{\theta} \\ &\leq \int_\Theta \left\| G^{(\ell)}(\boldsymbol{\theta}) - G^{(\ell-1)}(\boldsymbol{\theta}) \right\| \cdot \left\| \Gamma^{-1}(2\mathbf{y} - G^{(\ell)}(\boldsymbol{\theta}) - G^{(\ell-1)(\boldsymbol{\theta})}) \right\| \pi_0(\boldsymbol{\theta})\, \mathrm{d}\boldsymbol{\theta} \\ &\leq \left\| G^{(\ell)} - G^{(\ell-1)} \right\|_{L^2(\pi_0)} \cdot \left\| \Gamma^{-1}(2\mathbf{y} - G^{(\ell)} - G^{(\ell-1)}) \right\|_{L^2(\pi_0)} \\ &\leq b_1 \left\| G^{(\ell)} - G^{(\ell-1)} \right\|_{L^2(\pi_0)} \\ &\leq b_1 \left\| G - G^{(\ell)} \right\|_{L^2(\pi_0)} + b_1 \left\| G - G^{(\ell-1)} \right\|_{L^2(\pi_0)}\,. \end{aligned} \tag{50}$$

From Assumption 6 we have

$$b_1 \left\| G - G^{(\ell)} \right\|_{L^2(\pi_0)} + b_1 \left\| G - G^{(\ell-1)} \right\|_{L^2(\pi_0)} \leq b_0 b_1 s^{-\alpha\ell}(1 + s^\alpha)\,,$$

and therefore

$$R_\ell \leq \frac{1}{2} b_0 b_1 (1 + s^\alpha) (b_3 + b_2) s^{-\alpha \ell}, \tag{51}$$

so that $R_\ell \leq k_3 s^{-\alpha \ell}$ with $k_3 = \frac{b_0 b_1}{2} (b_2 + b_3)(1 + s^\alpha)$. Thus, Proposition 4 applies and gives the bound (39). □

# 4 Numerical example: Ice sheet modeling of the Arolla glacier

To demonstrate the applicability and performance of MLSVGD, we formulate and solve an inverse problem governed by a Stokes ice sheet model. In particular, we infer the basal sliding coefficient field from pointwise surface velocity observations. The problem formulation and adjoint-based derivatives computation follows the work in [38]. The numerical computations are carried out in Python using FEniCS [40] and hIPPYlib [41–44]. All reported runtimes were measured on Intel Xeon Platinum 8268 24C 205W 2.9GHz Processor. The computation of the gradients $\nabla \log \pi^{(\ell)}(\boldsymbol{\theta}_t^{[j]})$ was parallelized over 32 cores.

## 4.1 Nonlinear Stokes forward model

For the numerical studies, we use an ice sheet model problem that uses the Arolla (Haut Glacier d'Arolla) geometry and setup from the ISMIP-HOM benchmark collection [39]. That is, the glacier is considered a sliding mass of ice whose velocity is determined primarily by the force of gravity and the friction against the underlying rock. The ice flow is modeled as a non-Newtonian, viscous, incompressible fluid. The velocity field $\boldsymbol{u}$ over the domain $\Omega \subset \mathbb{R}^2$, as shown in Figure 1, is governed by the following Stokes equations

$$\begin{aligned} \nabla \cdot \boldsymbol{u} &= 0, \quad \text{in } \Omega\,, \\ -\nabla \cdot \boldsymbol{\sigma}_u &= \rho \boldsymbol{g}, \quad \text{in } \Omega\,. \end{aligned} \tag{52}$$

The boundary conditions along the top and bottom of the glacier are given as

$$\begin{aligned} \boldsymbol{n}^\top (\boldsymbol{\sigma}_u \boldsymbol{n} + \omega \boldsymbol{u}) &= 0, \quad \text{on } \Gamma_b\,, \\ \mathbf{T}\boldsymbol{\sigma}_u \boldsymbol{n} + \exp(\beta)\mathbf{T}\boldsymbol{u} &= \mathbf{0}, \quad \text{on } \Gamma_b\,, \\ \boldsymbol{\sigma}_u \boldsymbol{n} &= \mathbf{0}, \quad \text{on } \Gamma_t\,. \end{aligned} \tag{53}$$

The density of the ice is $\rho = 910$ [kg/m$^3$] and the downwards gravitational force is $\boldsymbol{g} = (0, -9.81)$ [m/s$^2$]. For the boundary conditions, $\Gamma_b$ represents the bottom part of the domain where the ice slides across the bedrock and $\Gamma_t$ represents the top part of the domain; see Figure 1. The vector $\boldsymbol{n}$ represents the outward unit normal vector and $\mathbf{T} = \mathbf{I} - \boldsymbol{n}\boldsymbol{n}^\top$ is the tangential projection. In the first boundary condition where $\boldsymbol{n}^\top (\boldsymbol{\sigma}_u \boldsymbol{n} + \omega \boldsymbol{u}) = 0$ on $\Gamma_b$ we set the parameter $\omega = 10^6$ and is meant to approximate the no out-flow condition $\boldsymbol{u} \cdot \boldsymbol{n} = 0$, which is difficult to enforce directly due to the

curvature of the domain $\Omega$. The stress tensor is

$$\boldsymbol{\sigma}_{\boldsymbol{u}} = \boldsymbol{\tau}_{\boldsymbol{u}} - \mathbf{I}p,$$

with pressure $p$ and deviatoric stress tensor

$$\boldsymbol{\tau}_{\boldsymbol{u}} = 2\eta(\boldsymbol{u})\dot{\boldsymbol{\epsilon}}(\boldsymbol{u})\,,$$

with effective viscosity

$$\eta(\boldsymbol{u}) = \frac{1}{2}A^{-\frac{1}{n}}\dot{\epsilon}_{\text{II}}^{\frac{1-n}{2n}}\,.$$

The constants are Glen's flow law exponent $n = 3$ and the flow rate factor $A = 10^{-16}$ [$\text{Pa}^{-n}\text{a}^{-1}$] (Pascals and years, respectively). The strain rate tensor is

$$\dot{\boldsymbol{\epsilon}} = \frac{1}{2}\left(\nabla\boldsymbol{u} + \nabla\boldsymbol{u}^{\top}\right)\,,$$

as well as the second invariant

$$\dot{\epsilon}_{\text{II}} = \frac{1}{2}\text{tr}\left(\dot{\boldsymbol{\epsilon}}_{\boldsymbol{u}}^2\right)\,,$$

where tr denotes the trace operator. The parameter of interest is the log basal sliding coefficient field $\beta : [0, 5000] \to \mathbb{R}$, which models the friction of the ice sheet across the underlying bedrock and relates tangential traction to the tangential velocity.

To solve (52), we discretize (52)–(53) using Taylor-Hood finite elements on a triangular mesh where the velocity is discretized with quadratic Lagrange elements and the pressure is discretized with linear Lagrange elements. We consider one high-fidelity model and two low-fidelity models by coarsening the mesh. The high-fidelity forward model $F^{(3)}$ ($L = 3$) maps the log basal sliding coefficient field $\beta$ to the velocity field solution $\boldsymbol{u}$ using 3,602 and 501 degrees of freedom for the velocity and pressure components, respectively. The coarsest low-fidelity model $F^{(1)}$ uses 448 and 73 degrees of freedom for the velocity and pressure and the second low-fidelity model $F^{(2)}$ uses 1002 and 151 degrees of freedom, respectively. To solve the discretized PDE we use a constrained Newton solver with the gradient tolerance set to $10^{-6}$. The wall-clock time for evaluating each model (gradient of log posterior descibed in the next section) is estimated by averaging over 100 total evaluations and is approximately 0.100 seconds for $\ell = 1$, 0.206 seconds for $\ell = 2$, and 1.106 seconds for the high-fidelity model $L = 3$.

## 4.2 Problem setup

We are interested in inferring a discretized log basal sliding coefficient field $\beta$, which effectively determines the velocity of the ice as it slides along the bedrock. We discretize the coefficient field $\beta : [0, 5000] \to \mathbb{R}$ with a vector $\boldsymbol{\beta} \in \mathbb{R}^d$ ($d = 25$) that we aim to infer from data of the parameter-to-observable map. The parameter vector $\boldsymbol{\beta} \in \mathbb{R}^{25}$ corresponds to 25 equally-spaced pointwise evaluations of the coefficient field $\beta$ throughout the domain $[0, 5000]$. In particular, let $\mathcal{I}^{\text{int}}$ denote the interpolation

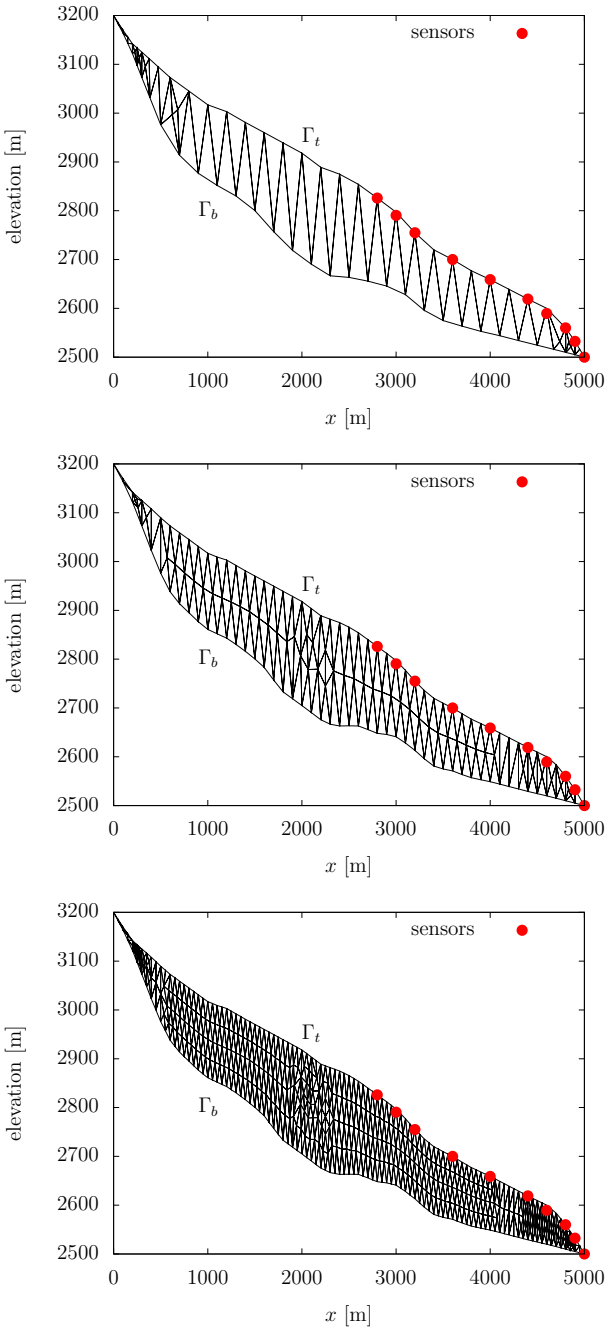



**Fig. 1** The domain $\Omega$ of Haut Glacier d'Arolla from the ISMIP-HOM benchmark collection [39]. The red dots represent the location of the measurements. (**Top**) The coarsest mesh used by the lowest fidelity model with $\ell = 1$. (**Middle**) A refined mesh used by the low-fidelity model with $\ell = 2$. (**Bottom**) The finest mesh used by the high-fidelity model $L = 3$.

operator that maps a vector $\boldsymbol{\beta} \in \mathbb{R}^{25}$ to its piecewise linear interpolant $\bar{\beta} : [0, 5000] \to \mathbb{R}$ defined at the nodes $x_i = 5000(i-1)/24$ by $\bar{\beta}(x_i) = \boldsymbol{\beta}_i$ for $i = 1, \ldots, 25$. Given the piecewise linear interpolant $\bar{\beta}$, the forward models $F^{(\ell)}$ for $\ell = 1, 2, 3$ map the parameter to the corresponding velocity field $\boldsymbol{u}$. Finally, the observation operator $\mathcal{B}^{\text{obs}}$ maps the solution $\boldsymbol{u}$ of (52), given by the output of the forward models $F^{(\ell)}$, to a 20 dimensional vector of horizontal and vertical velocity measurements at 10 sensor locations throughout the right side of the domain along the top of the glacier as shown in Figure 1. The full parameter-to-observable map $G^{(\ell)} : \mathbb{R}^{25} \to \mathbb{R}^{20}$ is

$$G^{(\ell)} = \mathcal{B}^{\text{obs}} \circ F^{(\ell)} \circ \mathcal{I}^{\text{int}}, \ \ell = 1, 2, 3.$$

Now consider the true parameter vector $\boldsymbol{\beta}^* = [\beta_1^*, \ldots, \beta_{25}^*]^\top \in \mathbb{R}^{25}$ as given by taking pointwise evaluations

$$\beta_i^* = \beta_{\text{true}}(x_i)\,, \qquad x_i = 5000(i-1)/24\,, \quad i = 1, \ldots, 25\,, \tag{54}$$

where

$$\beta_{\text{true}}(x) = \log \begin{cases} 1000 + 1000 \sin\left(\frac{3\pi x}{5000}\right) + \zeta & \text{if } 0 \le x < 2500, \\ 1000\left(16 - \frac{x}{250}\right) + \zeta & \text{if } 2500 \le x < 4000, \\ 1000 + \zeta & \text{if } 4000 \le x < 5000\,, \end{cases}$$

and $\zeta = 10^{-6}$ is a small positive constant to ensure that the log basal coefficient field remains bounded. We generate synthetic observations $\mathbf{y} \in \mathbb{R}^{20}$ with

$$\mathbf{y} = G^{(L+1)}(\boldsymbol{\beta}^*) + \boldsymbol{\eta}, \quad \boldsymbol{\eta} \sim N(\mathbf{0}, \Gamma)\,,$$

where the noise covariance matrix $\Gamma$ is diagonal with $\sigma_{\text{vertical}} = 3$ and $\sigma_{\text{horizontal}} = 18$ corresponding to the vertical and horizontal velocity measurements, respectively. The Euclidean norm of the observation is $\|\mathbf{y}\|_2 \approx 623.254$ and the Frobenius norm of the covariance matrix is $\|\Gamma\|_F \approx 57.706$, leading to a signal-to-noise ratio of approximately $\|\mathbf{y}\|_2 / \|\Gamma\|_F \approx 10.8$. Here the level $L+1$ (a further refinement of the high-fidelity mesh) is used to compute the observed data $\mathbf{y}$. The prior $\pi_0$ is Gaussian with diagonal covariance matrix $0.05\mathbf{I}_{25\times 25}$, where $\mathbf{I}_{25\times 25} \in \mathbb{R}^{25\times 25}$ is the identity matrix. The mean of the prior is perturbed from the true parameters $\boldsymbol{\beta}^* \in \mathbb{R}^{25}$ by adding a mean-zero normal random vector with covariance equal to the prior's covariance. The starting distribution for SVGD and MLSVGD is the 25-dimensional standard normal distribution. The gradients of the log posterior density are computed using adjoints with hIPPYlib [41–43]. Our quantity of interest is the mean of the posterior distribution $\mathbb{E}_{\pi^{(L)}}[\boldsymbol{\beta}]$ and we compute a reference value $\hat{\boldsymbol{\beta}}^{\text{Ref}}$ by using the preconditioned Crank-Nicolson (pCN) method [47]. We run 100 independent chains and use a burn-in period of 10,000 samples for each chain to obtain $10^7$ total samples. The parameter (denoted by $\beta$ in the work [47]) in the pCN algorithm, which scales the variance of the proposal distribution, is set to $10^{-2}$. The parameter was chosen by a grid search over values

ranging from $10^{-3}$ to 0.5 to find the value giving an asymptotic acceptance rate closest to 25% (the acceptance rate for 0.01 approaches 24.38%).

### 4.3 SVGD algorithm with approximate gradients

Consider an empirical measure

$$\hat{\mu}_\tau^{(N)} = \frac{1}{N}\sum_{i=1}^{N} \delta_{\boldsymbol{\theta}_\tau^{[i]}}\,, \tag{55}$$

given by an ensemble of particles $\{\boldsymbol{\theta}_\tau^{[i]}\}_{i=1}^N$ with $\{\boldsymbol{\theta}_0^{[i]}\}_{i=1}^N \sim \mu_0$ and where $\delta_{\boldsymbol{x}}$ represents the Dirac-mass at $\boldsymbol{x}$. Practical SVGD implementations alternate between using the ensemble of particles $\{\boldsymbol{\theta}_\tau^{[i]}\}_{i=1}^N$ at time $\tau$ to estimate the gradient (8) and using the estimated gradient to update the ensemble to obtain $\hat{\mu}_{\tau+1}$. The gradient is estimated with Monte Carlo from the current ensemble of particles as

$$\hat{\mathbf{g}}_\tau(\boldsymbol{\theta};\ \{\boldsymbol{\theta}_\tau^{[1]},\dots,\boldsymbol{\theta}_\tau^{[N]}\}) = -\frac{1}{N}\sum_{i=1}^{N} K(\boldsymbol{\theta}_\tau^{[i]},\boldsymbol{\theta})\nabla \log \pi^{(L)}(\boldsymbol{\theta}_\tau^{[i]}) + \nabla_1 K(\boldsymbol{\theta}_\tau^{[i]},\boldsymbol{\theta})\,. \tag{56}$$

The SVGD algorithm then reuses the ensemble of particles and updates them according to the approximate gradient with step size $\delta$ as

$$\boldsymbol{\theta}_{\tau+1}^{[j]} = \boldsymbol{\theta}_\tau^{[j]} - \delta\hat{\mathbf{g}}_\tau(\boldsymbol{\theta};\ \{\boldsymbol{\theta}_\tau^{[1]},\dots,\boldsymbol{\theta}_\tau^{[N]}\})\,, \quad j=1,\dots,N\,. \tag{57}$$

Because the Hellinger distance from the high-fidelity density $\pi^{(L)}$ at iteration $\tau$ is unknown, the integration time given by (13) cannot be determined practically. Instead the stopping criteria is that the average norm of the gradient $\bar{g}_\tau$, defined as

$$\bar{g}_\tau = \frac{1}{N}\sum_{j=1}^{N} \left\| \hat{\mathbf{g}}_\tau(\boldsymbol{\theta}_\tau^{[j]};\ \{\boldsymbol{\theta}_\tau^{[1]},\dots,\boldsymbol{\theta}_\tau^{[N]}\}) \right\|\,,$$

decreases below the predetermined threshold $\epsilon$. The convergence of $\hat{\mu}_\tau$ to $\pi^{(L)}$ can no longer be measured in the KL divergence because at each iteration the measure $\hat{\mu}_\tau$ is no longer absolutely continuous with respect to the target $\pi^{(L)}$. Moreover, the convergence properties as the number of particles $N \to \infty$ remains an open question.

For a practical MLSVGD algorithm, an outer loop is performed over the levels $\ell = 1,\dots,L$ with the inner loop given by the SVGD updates (57). At each intermediate level $\ell < L$ the gradients (56) are obtained by replacing the high-fidelity density $\pi^{(L)}$ with the low-fidelity density $\pi^{(\ell)}$. Again, we cannot monitor the KL divergence to the target $\pi^{(\ell)}$ at each level $\ell$ as required by (16). Thus, the stopping criteria for when to terminate the SVGD iterations at the current level and proceed to the next level is that the norm of the gradient $\bar{g}_\tau$ decreases below the threshold $\epsilon$.

### 4.4 Numerical results

In the following we compare the performance of MLSVGD and SVGD. We run both SVGD and MLSVGD with $N = 1,000$ particles, set a step size of $\delta = 0.05$, and use a Gaussian radial basis function kernel with the bandwidth parameter set to $h = 0.1$. The bandwidth parameter is kept constant, but is comparable to the one obtained from using the median heuristic presented in [4].

#### 4.4.1 Number of iterations and runtime of SVGD and MLSVGD

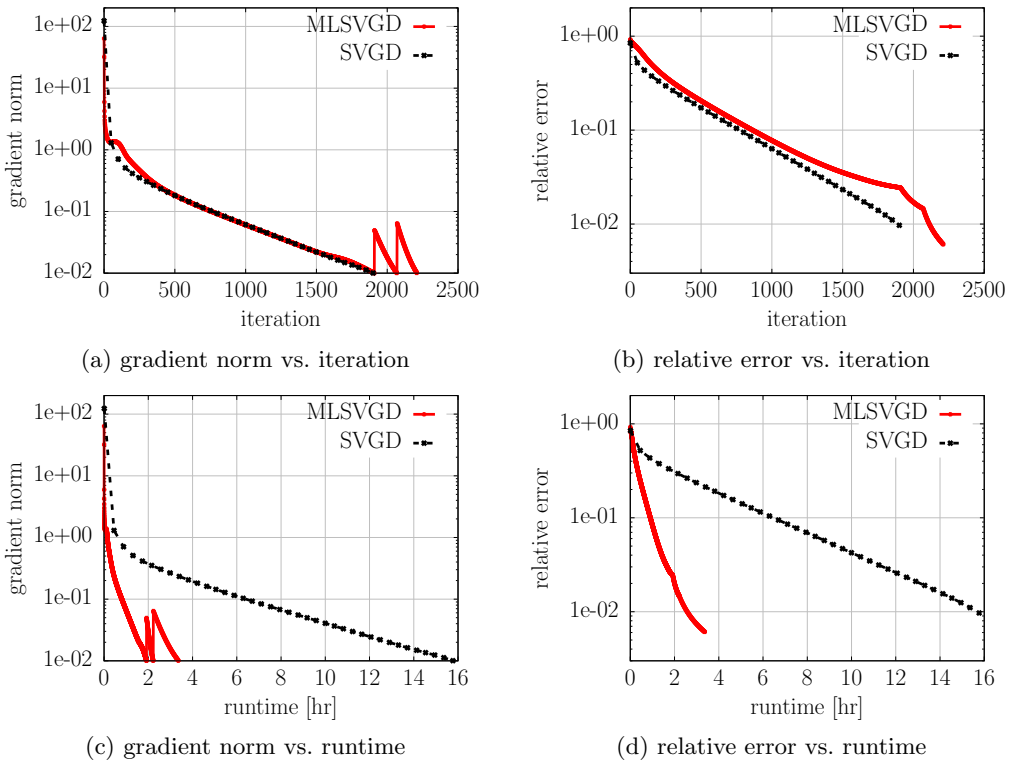
Figure 2 shows that with a gradient tolerance of $\epsilon = 10^{-2}$, MLSVGD achieves a speedup of a factor of five over SVGD despite requiring more iterations. Note that reducing the gradient norm below $\epsilon = 10^{-2}$ corresponds to a relative reduction of the gradient norm of more than four orders of magnitude. The results presented in Figure 2 are consistent with the numerical examples presented in [16]. The runtime improvement of MLSVGD over SVGD is a result of most of the iterations being performed on the lowest fidelity model with the coarsest mesh. MLSVGD quickly converges to the low-fidelity posterior $\pi^{(1)}$, which serves as a good initial distribution for the following two levels whereas SVGD requires many iterations at the high-fidelity level resulting in high computational costs. The two plots in the right column of Figure 2 show that both algorithms give accurate estimates of the quantity of interest in terms of the relative error

$$\text{rel}(\boldsymbol{\beta}) = \frac{\left\| \boldsymbol{\beta} - \hat{\boldsymbol{\beta}}^{\text{Ref}} \right\|_2}{\left\| \hat{\boldsymbol{\beta}}^{\text{Ref}} \right\|_2}, \tag{58}$$

where $\boldsymbol{\beta}$ is the mean of the particles and $\hat{\boldsymbol{\beta}}^{\text{Ref}}$ is the reference posterior mean computed with MCMC. The results suggest that the mean of the distributions of particles $\{\boldsymbol{\theta}_t^{[j]}\}_{j=1}^N$ is converging to the mean of the high-fidelity target posterior $\pi^{(L)}$.

#### 4.4.2 Speedups

MLSVGD recovers an approximation of the parameter $\boldsymbol{\beta}^*$, with relative error below $10^{-2}$, in less than a quarter of the time compared to SVGD because the low fidelity posteriors provide a good initialization. Figure 3 compares the inferred parameter means from MLSVGD and SVGD after fixed amounts of training time. After two hours MLSVGD has recovered the parameters whereas SVGD has not recovered them even after eight hours. We also note that the coordinates of $\boldsymbol{\beta}$ corresponding to the right side of the domain are recovered much faster due to the location of the velocity observation points shown in Figure 1. Figure 4 shows the final parameter uncertainty estimates with shaded regions indicating $\pm 2$ standard deviations for both MLSVGD and SLSVGD. The MLSVGD standard deviations match the SLSVGD standard deviations well. Moreover, Figure 5 shows the inferred velocity field $\boldsymbol{u}$ by solving (52) with the inferred parameter mean after approximately eight hours of run time over 32 cores. We see that the velocity field obtained with the MLSVGD inferred parameter




(a) gradient norm vs. iteration

(b) relative error vs. iteration

(c) gradient norm vs. runtime

(d) relative error vs. runtime

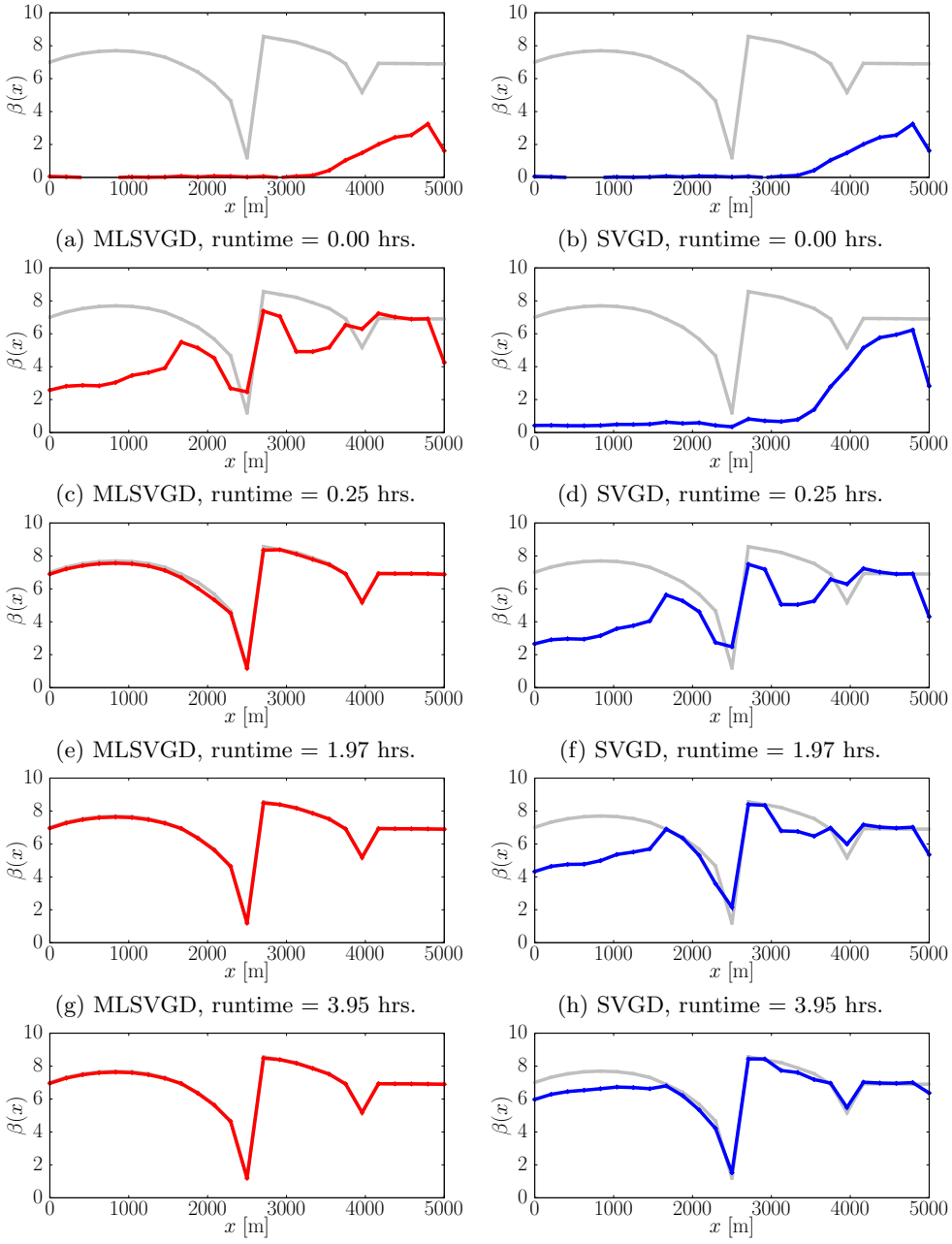
**Fig. 2** **(a)** The average gradient norm $\bar{g}_\tau$ vs. iteration for MLSVGD and SVGD with a tolerance of $\epsilon = 10^{-2}$. **(b)** The relative error (58) of MLSVGD and SVGD compared to an MCMC reference vs. iteration. **(c)** The average gradient norms vs. the actual runtime in hours over 32 cores. **(d)** The relative error vs. actual runtime.

mean closely matches the velocity field obtained with the ground truth reference value of the mean. On the other hand, SVGD fails to recover the correct velocity field within the same amount of time. Again we see that the left side of the domain is inaccurate due to the parameter in this region not yet being accurate. Note that the magnitude of the velocity is overestimated for SVGD which is consistent with the fact that the parameter is underestimated since the parameter controls the frictional forces to resist the downward pull of gravity.

### 4.4.3 Sample quality

Particles obtained with SVGD tend to be evenly spread out due to the repulsive interaction between particles given by the kernel. We measure sample quality with the maximum mean discrepancy (MMD)

$$\mathrm{MMD}[\mu,\nu]^2 = \sup_{\|f\|_{\mathcal{H}}\leq 1} \left(\mathbb{E}_\mu[f] - \mathbb{E}_\nu[f]\right)^2 ,$$




(a) MLSVGD, runtime = 0.00 hrs.

(b) SVGD, runtime = 0.00 hrs.

(c) MLSVGD, runtime = 0.25 hrs.

(d) SVGD, runtime = 0.25 hrs.

(e) MLSVGD, runtime = 1.97 hrs.

(f) SVGD, runtime = 1.97 hrs.

(g) MLSVGD, runtime = 3.95 hrs.

(h) SVGD, runtime = 3.95 hrs.

(i) MLSVGD, runtime = 7.90 hrs.

(j) SVGD, runtime = 7.90 hrs.

**Fig. 3** (**Left**) Snapshots of the MLSVGD inferred parameter mean (red) at different times. (**Right**) Snapshots of the SVGD inferred parameter mean (blue) at the same times. In each plot the solid light gray curve shows the reference value.

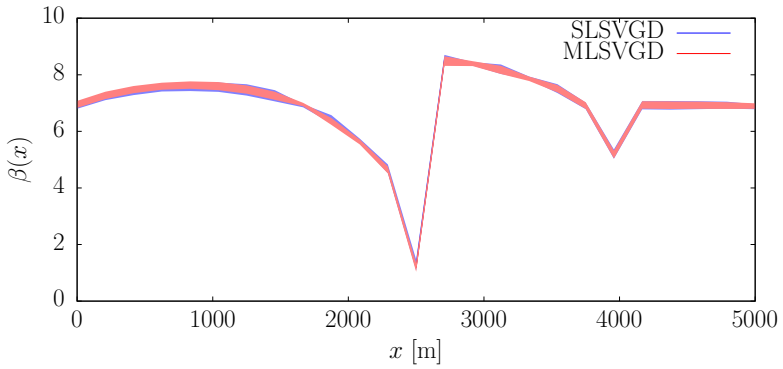



**Fig. 4** The final MLSVGD and SLSVGD parameter estimates with shaded regions indicating $\pm 2$ estimated standard deviations.

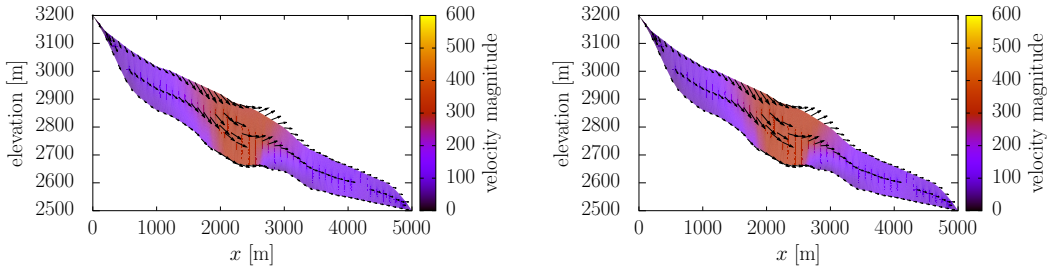
where $\mathcal{H}$ is the reproducing kernel Hilbert space with kernel $K$ [45]. The MMD is zero if and only if the distributions $\mu = \nu$. In practice one cannot evaluate the expectations exactly, so the following estimator [45, Eq. 5] is often used instead

$$\widehat{\mathrm{MMD}}\left(\{\mathbf{x}_i\}_{i=1}^N, \{\mathbf{y}_j\}_{j=1}^M\right)^2 = \frac{1}{N^2}\sum_{i=1}^N\sum_{i'=1}^N K(\mathbf{x}_i, \mathbf{x}_{i'}) + \frac{1}{M^2}\sum_{j=1}^M\sum_{j'=1}^M K(\mathbf{y}_j, \mathbf{y}_{j'}) - \frac{2}{NM}\sum_{i=1}^N\sum_{j=1}^M K(\mathbf{x}_i, \mathbf{y}_j)\,, \quad (59)$$

where $\{\mathbf{x}_i\}_{i=1}^N \sim \mu$ and $\{\mathbf{y}_j\}_{j=1}^M \sim \nu$. To compute the MMD from the target distribution $\pi^{(L)}$ we use pCN with $\beta = 0.01$ again to draw samples. We use a burn-in period of 20,000 samples and then run 100,000 more iterations taking every 5th sample for 20,000 samples total. These 20,000 samples serve as proxy samples from target posterior $\pi^{(L)}$. Figure 6 shows the estimated MMD for MLSVGD, SVGD, and MCMC. We see that MLSVGD gives samples with comparable quality to SVGD due to the repulsive interaction between particles, and both SVGD and MLSVGD outperform MCMC (pCN) with the same sample size ($N = 1,000$).

# 5 Conclusion

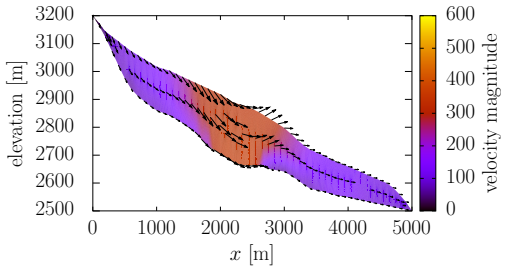
We provided an extension of the analysis of the MLSVGD method and demonstrated with a Bayesian inverse problem of inferring a discretized basal sliding coefficient field that MLSVGD scales well to larger settings than the ones considered in prior work [16]. In particular, MLSVGD provides particles of comparable quality as SVGD but at greatly reduced computational costs in our numerical example. There are several avenues of future research. One is combining MLSVGD with the likelihood-informed projections introduced in [9], which is especially useful in high-dimensional Bayesian inverse problems, where typically data inform only low-dimensional subspaces of the potentially high-dimensional spaces of the quantities of interest. Another direction of



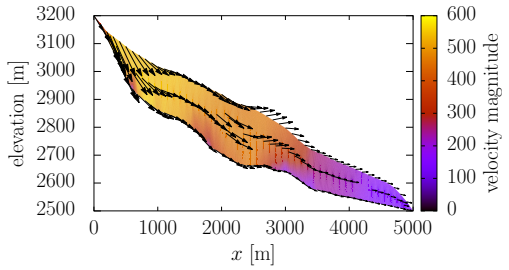

(a) true velocity field




(b) reference velocity field




(c) MLSVGD inferred velocity field




(d) SVGD inferred velocity field

**Fig. 5** (**a**) The true velocity field given by $\boldsymbol{\beta}^*$, which is defined (54). The color indicates the magnitude of the velocity in [m $a^{-1}$] (meters per year). (**b**) The reference velocity field computed using $\hat{\boldsymbol{\beta}}^{\text{Ref}}$ of the posterior mean. (**c**) The velocity field corresponding to the inferred parameters using MLSVGD after eight hours. (**d**) The velocity field corresponding to the inferred parameters using SVGD with equivalent costs as MLSVGD (eight hours of runtime).

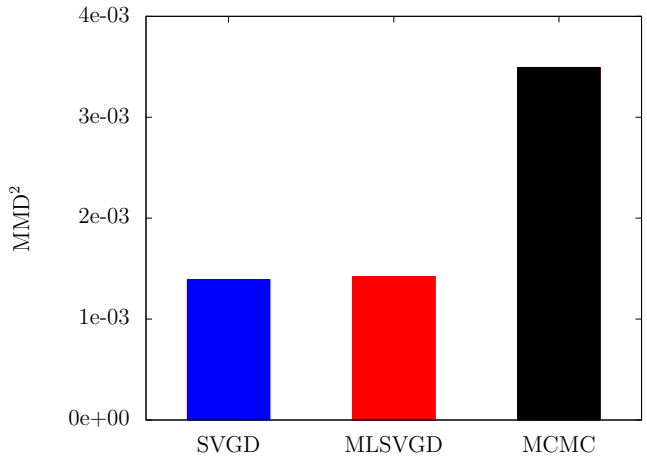



**Fig. 6** The estimated squared MMD using the estimator (59). The MLSVGD approximation has a comparable MMD to the single-level SVGD with the high-fidelity model only. Both have a lower MMD than MCMC suggesting higher quality samples.

future work concerns balancing the number of particles on each level. In the presented form, MLSVGD uses the same number of particles on each level, in contrast to multilevel Monte Carlo methods that use more samples on coarse levels and fewer on finer levels. It remains an open question how to realize different number of samples on each level in MLSVGD.

# Declarations


### *Funding*

The first and third author acknowledge support from the Air Force Office of Scientific Research under Award Number FA9550-21-1-0222 (Dr. Fariba Fahroo) and the National Science Foundation (NSF) under award IIS-1901091. The second and fourth author acknowledge support provided by the NSF under Grant No. CAREER-1654311. The second author acknowledges further support provided by the NSF under Grant No. DMS-1840265.


### *Competing interests*

The authors have no competing interests to declare that are relevant to the content of this article.

# References


[1] Kaipio, J., Somersalo, E.: Statistical inverse problems: Discretization, model reduction, and inverse crimes. Journal of Computational and Applied Mathematics **198**(2), 493–504 (2007)

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

[3] Latz, J.: On the well-posedness of Bayesian inverse problems. SIAM/ASA Journal on Uncertainty Quantification **8**(1), 451–482 (2020)

[4] Liu, Q., Wang, D.: Stein variational gradient descent: A general purpose Bayesian inference algorithm. In: Lee, D., Sugiyama, M., Luxburg, U., Guyon, I., Garnett, R. (eds.) Advances in Neural Information Processing Systems, vol. 29, pp. 2378–2386 (2016)

[5] Detommaso, G., Cui, T., Marzouk, Y., Spantini, A., Scheichl, R.: A Stein variational Newton method. In: Bengio, S., Wallach, H., Larochelle, H., Grauman, K., Cesa-Bianchi, N., Garnett, R. (eds.) Advances in Neural Information Processing Systems, vol. 31, pp. 9169–9179 (2018)

[6] Duncan, A., Nüsken, N., Szpruch, L.: On the geometry of stein variational gradient descent. Journal of Machine Learning Research **24**(56), 1–39 (2023)

[7] Wang, D., Tang, Z., Bajaj, C., Liu, Q.: Stein variational gradient descent with matrix-valued kernels. In: Wallach, H., Larochelle, H., Beygelzimer, A., Alché-Buc, F., Fox, E., Garnett, R. (eds.) Advances in Neural Information Processing Systems, vol. 32 (2019)

[8] Cui, T., Law, K.J.H., Marzouk, Y.M.: Dimension-independent likelihood-informed MCMC. Journal of Computational Physics **304**, 109–137 (2016)

[9] Chen, P., Wu, K., Chen, J., Leary-Roseberry, T.O., Ghattas, O.: Projected Stein variational Newton: A fast and scalable Bayesian inference method in high dimensions. In: Wallach, H., Larochelle, H., Beygelzimer, A., Alché-Buc, F., Fox, E., Garnett, R. (eds.) Advances in Neural Information Processing Systems, vol. 32, pp. 15130–15139 (2019)

[10] Chen, P., Ghattas, O.: Stein variational reduced basis Bayesian inversion. SIAM Journal on Scientific Computing **43**(2), 1163–1193 (2021)

[11] Liu, Q.: Stein variational gradient descent as gradient flow. In: Guyon, I., Luxburg, U.V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., Garnett, R. (eds.) Advances in Neural Information Processing Systems, vol. 30, pp. 3115–3123 (2017)

[12] Lu, J., Lu, Y., Nolen, J.: Scaling limit of the Stein variational gradient descent: The mean field regime. SIAM Journal on Mathematical Analysis **51**(2), 648–671 (2019)

[13] Chewi, S., Gouic, T.L., Lu, C., Maunu, T., Rigollet, P.: SVGD as a kernelized Wasserstein gradient flow of the chi-squared divergence. In: Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M.F., Lin, H. (eds.) Advances in Neural Information Processing Systems, vol. 33 (2020)

[14] Korba, A., Salim, A., Arbel, M., Luise, G., Gretton, A.: A non-asymptotic analysis for Stein variational gradient descent. In: Advances in Neural Information Processing Systems, vol. 33 (2020)

[15] Ba, J., Erdogdu, M., Ghassemi, M., Suzuki, T., Wu, D.: Towards characterizing the high-dimensional bias of kernel-based particle inference algorithms. In: 2nd Symposium on Advances in Approximate Bayesian Inference, pp. 1–17 (2019)

[16] Alsup, T., Venturi, L., Peherestorfer, B.: Multilevel Stein variational gradient descent with applications to Bayesian inverse problems. In: Bruna, J., Hesthaven, J.S., Zdeborova, L. (eds.) Proceedings of Machine Learning Research. 2nd Annual Conference on Mathematical and Scientific Machine Learning, vol. 145, pp. 1–25 (2021)

[17] Peherstorfer, B., Willcox, K., Gunzburger, M.: Survey of multifidelity methods in uncertainty propagation, inference, and optimization. SIAM Review **60**(3),

550–591 (2018)

[18] Konrad, J., Farcaş, I.-G., Peherstorfer, B., Di Siena, A., Jenko, F., Neckel, T., Bungartz, H.-J.: Data-driven low-fidelity models for multi-fidelity Monte Carlo sampling in plasma micro-turbulence analysis. Journal of Computational Physics **451**, 110898 (2022)

[19] Heinrich, S.: Multilevel Monte Carlo methods. In: Proceedings of the Third International Conference on Large-Scale Scientific Computing-Revised Papers, LSSC '01, pp. 58–67 (2001)

[20] Giles, M.: Multilevel Monte Carlo path simulation. Operations Research **56**(3), 607–617 (2008)

[21] Cliffe, K.A., Giles, M., Scheichl, R., Teckentrup, A.L.: Multilevel Monte Carlo methods and applications to elliptic PDEs with random coefficients. Computing and Visualization in Science **14**(1), 3–15 (2011)

[22] Peherstorfer, B., Gunzburger, M., Willcox, K.: Convergence analysis of multifidelity Monte Carlo estimation. Numerische Mathematik **139**(3), 683–707 (2018)

[23] Peherstorfer, B., Beran, P.S., Willcox, K.: Multifidelity Monte Carlo estimation for large-scale uncertainty propagation. In: 2018 AIAA Non-Deterministic Approaches Conference (2018)

[24] Dodwell, T.J., Ketelsen, C., Scheichl, R., Teckentrup, A.L.: A hierarchical multilevel Markov chain Monte Carlo algorithm with applications to uncertainty quantification in subsurface flow. SIAM/ASA Journal on Uncertainty Quantification **3**(1), 1075–1108 (2015)

[25] Lykkegaard, M.B., Dodwell, T.J., Fox, C., Mingas, G., Scheichl, R.: Multilevel delayed acceptance MCMC. SIAM/ASA Journal on Uncertainty Quantification **11**(1), 1–30 (2023)

[26] Peherstorfer, B., Marzouk, Y.: A transport-based multifidelity preconditioner for Markov chain Monte Carlo. Advances in Computational Mathematics **45**, 2321–2348 (2019)

[27] Beskos, A., Jasra, A., Law, K., Tempone, R., Zhou, Y.: Multilevel sequential Monte Carlo samplers. Stochastic Processes and their Applications **127**(5), 1417–1440 (2017)

[28] Latz, J., Papaioannou, I., Ullmann, E.: Multilevel sequential$^2$ Monte Carlo for Bayesian inverse problems. Journal of Computational Physics **368**, 154–178 (2018)

[29] Wagner, F., Latz, J., Papaioannou, I., Ullmann, E.: Multilevel sequential importance sampling for rare event estimation. SIAM Journal on Scientific Computing **42**(4), 2062–2087 (2020)

[30] Peherstorfer, B., Kramer, B., Willcox, K.: Multifidelity preconditioning of the cross-entropy method for rare event simulation and failure probability estimation. SIAM/ASA Journal on Uncertainty Quantification **6**(2), 737–761 (2018)

[31] Alsup, T., Peherstorfer, B.: Context-aware surrogate modeling for balancing approximation and sampling costs in multi-fidelity importance sampling and Bayesian inverse problems. SIAM/ASA Journal on Uncertainty Quantification (2022). (accepted)

[32] Gregory, A., Cotter, C.J., Reich, S.: Multilevel ensemble transform particle filtering. SIAM Journal on Scientific Computing **38**(3), 1317–1338 (2016)

[33] Briggs, W., Henson, V.E., McCormick, S.: A Multigrid Tutorial, Second Edition, 2nd edn. Society for Industrial and Applied Mathematics, Philadelphia (2000)

[34] Hackbush, W.: Multi-Grid Methods and Applications. Springer, Berlin (1985)

[35] Li, Z., Fan, Y., Ying, L.: Multilevel fine-tuning: Closing generalization gaps in approximation of solution maps under a limited budget for training data. Multiscale Modeling & Simulation **19**(1) (2021)

[36] Hoel, H., Law, K., Tempone, R.: Multilevel ensemble Kalman filtering. SIAM Journal on Numerical Analysis **54**(3), 1813–1839 (2016)

[37] Chada, N., Jasra, A., Yu, F.: Multilevel ensemble Kalman–Bucy filters. SIAM/ASA Journal on Uncertainty Quantification **10**(2), 584–618 (2022)

[38] Petra, N., Martin, J., Stadler, G., Ghattas, O.: A computational framework for infinite-dimensional Bayesian inverse problems, part II: Stochastic Newton MCMC with application to ice sheet flow inverse problems. SIAM J. Sci. Comput. **36**(4), 1525–1555 (2014)

[39] Pattyn, F., Perichon, L., Aschwanden, A., Breuer, B., Smedt, B., Gagliardini, O., Gudmundsson, G.H., Hindmarsh, R.C.A., Hubbard, A., Johnson, J.V., Kleiner, T., Konovalov, Y., Martin, C., Payne, A.J., Pollard, D., Price, S., Ruckamp, M., Saito, F., Soucek, O., Sugiyama, S., , Zwinger, T.: Benchmark experiments for higher-order and full-Stokes ice sheet models (ISMIP-HOM). The Cryosphere **2**, 95–108 (2008)

[40] Alnaes, M.S., Blechta, J., Hake, J., Johansson, A., Kehlet, B., Logg, A., Richardson, C., Ring, J., Rognes, M.E., Wells, G.N.: The FEniCS Project Version 1.5 (2015)

[41] Villa, U., Petra, N., Ghattas, O.: Documentation to "hIPPYlib: an Extensible Software Framework for Large-scale Deterministic and Bayesian Inverse Problems". http://hippylib.github.io (2016)

[42] Villa, U., Petra, N., Ghattas, O.: hIPPYlib: an Extensible Software Framework for Large-scale Deterministic and Bayesian Inverse Problems. Journal of Open Source Software **3**(30) (2018)

[43] Villa, U., Petra, N., Ghattas, O.: HIPPYlib: An Extensible Software Framework for Large-Scale Inverse Problems Governed by PDEs: Part I: Deterministic Inversion and Linearized Bayesian Inference. ACM Trans. Math. Softw. **47**(2) (2021)

[44] Kim, K.-T., Villa, U., Parno, M., Marzouk, Y., Ghattas, O., Petra, N.: hIPPYlib-MUQ: A Bayesian inference software framework for integration of data with complex predictive models under uncertainty. arXiv **2112.00713** (2021)

[45] Gretton, A., Borgwardt, K., Rasch, M., Schölkopf, B., Smola, A.: A kernel two-sample test. Journal of Machine Learning Research **13**, 723–773 (2012)

[46] Jordan, R., Kinderlehrer, D., Otto, F.: The variational formulation of the Fokker–Planck equation. SIAM Journal on Mathematical Analysis **29**(1) (1998)

[47] Cotter, S.L., Roberts, G.O., Stuart, A.M., White, D.: MCMC methods for functions: modifying old algorithms to make them faster. Statist. Sci. **28**(3), 424–446 (2013)