# Multilevel Stein variational gradient descent with applications to Bayesian inverse problems

Terrence Alsup, Luca Venturi, and Benjamin Peherstorfer
Courant Institute of Mathematical Sciences, New York University

August 2021

# Intro: Bayesian inverse problems

Infer an unknown parameter $\boldsymbol{\theta} \in \Theta$ from some noisy observed data

$$\boldsymbol{y} = G(\boldsymbol{\theta}^*) + \boldsymbol{e}$$

with forward PDE model $G : \Theta \to \mathcal{Y}$, Gaussian noise $\boldsymbol{e} \sim N(0, \Gamma)$

Given prior $\pi_0$, the posterior takes the form

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

Sequence of surrogate models (discretizations) $G^{(1)}, G^{(2)}, \ldots$ induce sequence of measures

$$(\pi^{(\ell)})_{\ell \geq 1} \longrightarrow \pi$$

**Classical approach**: choose high-fidelity approximation $G^{(L)}$ and sample w.r.t. $\pi^{(L)}$

# Intro: Stein variational gradient descent (SVGD)

Find an approximation $\mu$ to a target measure $\pi^{(L)}$ such that

$$\mathrm{KL}\left(\mu \mid\mid \pi^{(L)}\right) \leq \epsilon$$

- Evolve a density $\mu_t$ along a gradient flow that minimizes the KL divergence to the target
- KL divergence of density updated with map $\boldsymbol{g}$ given by functional

$$J_t(\boldsymbol{g}) = \mathrm{KL}\left((I - \boldsymbol{g})_{\#}\mu_t \mid\mid \pi^{(L)}\right)$$

- Evolve ensemble of particles $\boldsymbol{\theta}_t^{[1]}, \ldots, \boldsymbol{\theta}_t^{[M]} \sim \mu_t$

$$\dot{\boldsymbol{\theta}}_t^{[i]} = -\nabla J_t(0)\left(\boldsymbol{\theta}_t^{[i]}\right), \qquad i = 1, \ldots, M$$

- Discretize with forward Euler in time and approximate gradient using particles

**Classical approach**: pick $L \in \mathbb{N}$ and then integrate with SVGD w.r.t. $\pi^{(L)}$

$$\mu_0 \quad \xrightarrow[T]{\pi^{(L)}} \quad \mu^{\mathsf{SL}}$$

Integration time (cost) depends on divergence between starting density $\mu_0$ and target $\pi^{(L)}$

# Intro: Literature overview

**Multilevel methods for sampling**

- MCMC methods that exploit hierarchies of distributions
  **[Christen and Fox, 2005, Fox and Nicholls, 1997, Dodwell et al., 2015]**
- Multilevel variational methods that learn parametric transport maps
  **[Moselhy and Marzouk, 2012, Parno and Marzouk, 2018, Alsup and Peherstorfer, 2020]**
- Multilevel particle filters and multilevel sequential Monte Carlo
  **[Jasra et al., 2017, Beskos et al., 2017, Hoel et al., 2016, Latz et al., 2018, Wagner et al., 2020]**

**Stein variational gradient descent** **[Liu and Wang, 2016, Liu, 2017]**

- Analysis of SVGD in the mean-field limit **[Liu, 2017, Duncan et al., 2019]**
- Convergence rate analysis of SVGD **[Korba et al., 2020, Chewi et al., 2020]**
- Variants use Newton directions **[Detommaso et al., 2018]**, exploit geometry **[Chen et al., 2019]**, other acceleration techniques **[Liu et al., 2019]**

# MLSVGD: Multi-level preconditioning

single-level SVGD:

$$\mu_0 \xrightarrow[T]{\pi^{(L)}} \mu^{\mathrm{SL}}$$

proposed MLSVGD:

$$\mu_0 \xrightarrow[T_1]{\pi^{(1)}} \mu^{(1)}_{T_1} \xrightarrow[T_2]{\pi^{(2)}} \cdots \xrightarrow[T_L]{\pi^{(L)}} \mu^{\mathrm{ML}}$$

- Integration time depends on divergence of starting density $\mu_0$ from $\pi^{(L)}$
- Use surrogate models as preconditioners to find better starting densities for following levels
- Need to understand for how long to integrate on each level and what the corresponding cost complexity is

# MLSVGD: Assumptions for cost analysis

1. **Model cost**: Cost $c_\ell$ of evaluating model $G^{(\ell)}$ at level $\ell$ bounded as

$$c_\ell \lesssim s^{\gamma\ell}, \quad s > 1,\ \gamma > 0$$

2. **Discretization error**: Error of surrogate model $G^{(\ell)}$ at level $\ell$ bounded as

$$\|G^{(\ell)} - G\|_{L^2(\pi_0)} \lesssim s^{-\alpha\ell}, \quad \alpha > 0$$

3. **SVGD convergence**: Exponential convergence for any starting distribution $\nu_0$ and level $\ell$

$$\mathrm{KL}\left(\nu_t || \pi^{(\ell)}\right) \leq e^{-\lambda t}\mathrm{KL}\left(\nu_0 || \pi^{(\ell)}\right), \quad \lambda > 0,\ \forall t \geq 0$$

4. **Envelope assumption**: SVGD densities are bounded by the prior density

$$\mu_t^{(\ell)} \lesssim \pi_0, \quad \forall t \geq 0,\ \ell \geq 0$$

# MLSVGD: Cost complexity

**Cost complexity of single-level SVGD**
*The integration time $T$ to reach*

$$\mathrm{KL}\left(\mu_T \;||\; \pi^{(L)}\right) \leq \epsilon$$

*is*

$$T = \frac{1}{\lambda} \log\left(\frac{\mathrm{KL}(\mu_0 \;||\; \pi^{(L)})}{\epsilon}\right),$$

*and the computational complexity for single-level SVGD scales as*

$$\mathcal{O}(\epsilon^{-\gamma/\alpha} \log \epsilon^{-1}).$$

**Cost complexity of MLSVGD** [A., V., P., (2021)]
*The integration times $T_\ell$ needed at each level are $\mathcal{O}(1)$. The computational complexity for multi-level SVGD scales as*

$$\mathcal{O}(\epsilon^{-\gamma/\alpha}).$$

# MLSVGD: Algorithm

$$\mu_0 \quad \xrightarrow[T_1]{\pi^{(1)}} \quad \mu^{(1)}_{T_1} \quad \xrightarrow[T_2]{\pi^{(2)}} \quad \cdots \quad \xrightarrow[T_L]{\pi^{(L)}} \quad \mu^{\mathrm{ML}}$$

- Draw $M$ particles $\boldsymbol{\theta}_0^{[1]}, \dots, \boldsymbol{\theta}_0^{[M]}$ from a reference distribution $\mu_0$
- For level $\ell = 1, \dots, L$, integrate w.r.t. $\pi^{(\ell)}$ by computing the gradient at each step

$$\boldsymbol{g}_t^{[i]} = \frac{1}{M}\left(\sum\nolimits_{j=1}^{M} \nabla_1 K(\boldsymbol{\theta}_t^{[j]}, \boldsymbol{\theta}_t^{[i]}) + \sum\nolimits_{j=1}^{M} K(\boldsymbol{\theta}_t^{[j]}, \boldsymbol{\theta}_t^{[i]}) \nabla \log \pi^{(\ell)}(\boldsymbol{\theta}_t^{[j]})\right)$$

  for $i = 1, \dots, M$ and updating with step-size $\delta_t > 0$

$$\boldsymbol{\theta}_{t+\delta_t}^{[i]} = \boldsymbol{\theta}_t^{[i]} + \delta_t \boldsymbol{g}_t^{[i]}, \qquad i = 1, \dots, M$$

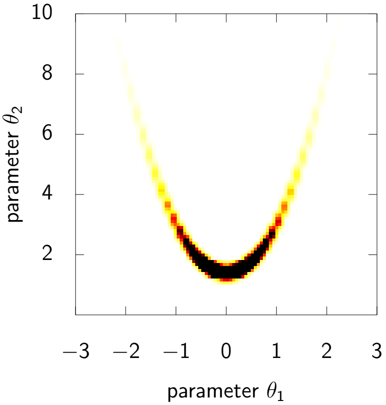
- In practice, cannot monitor the KL divergence, so switch to next level $\ell + 1$ whenever the norm of the gradient is below predetermined threshold $\epsilon$

# Numerical results: Nonlinear diffusion reaction

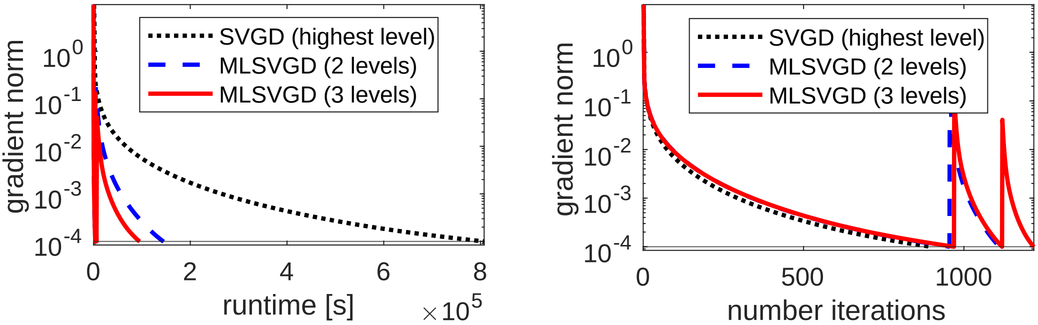
Diffusion-reaction in $\Omega = [0,1]^2$ with nonlinear reaction

$$f(u,\boldsymbol{\theta}) = (0.1\sin(\theta_1) + 2)\mathrm{e}^{-2.7\theta_1^2}(\mathrm{e}^{1.8\theta_2 u} - 1)$$

- Infer reaction parameters $\boldsymbol{\theta} = [\theta_1, \theta_2]^T$
- Finite differences with mesh width $2^{-5}$
- Surrogate with mesh widths $2^{-3}, 2^{-4}$
- Gaussian prior and 0.5% noise

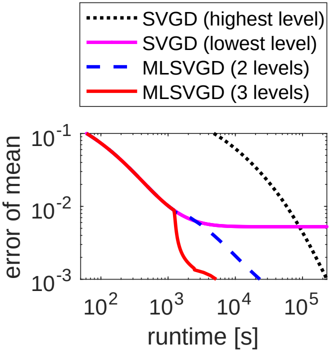


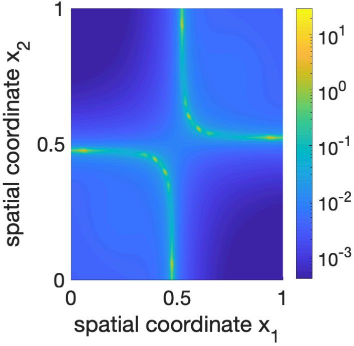
# Numerical results: SVGD vs. MLSVGD



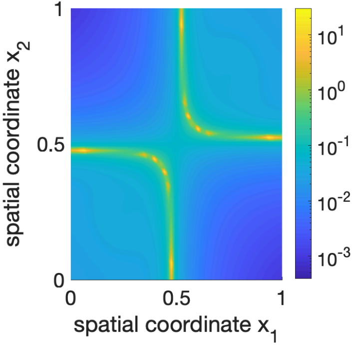

- SVGD converges in fewer total iterations but ...
- ... MLSVGD off-loads the bulk of the cost onto the lower levels making it more efficient

# Numerical results: Performance of MLSVGD



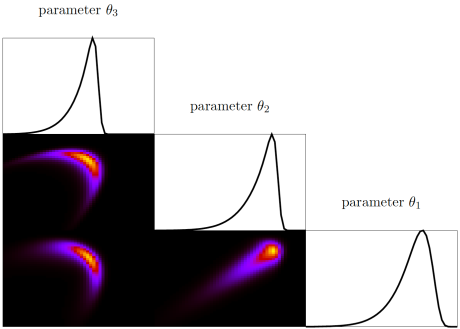

MLSVGD




SVGD

- SVGD on lowest level alone is inaccurate, highest level alone is expensive
- MLSVGD achieves one order of magnitude speedup and is accurate
- For same costs, MLSVGD leads to more accurate inferred solution than SVGD on highest level

# Numerical results: Euler Bernoulli beam

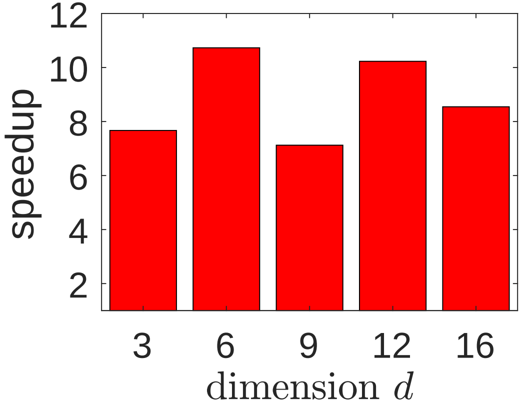



Infer 16 dimensional parameter $\boldsymbol{\theta}$ that determines stiffness $S(x; \boldsymbol{\theta})$ of Euler Bernoulli beam with displacement $u$ and load $f$ over domain $x \in (0, 1)$ **[Parno and Marzouk, 2018]**

$$\frac{\partial^2}{\partial x^2}\left(S(x;\boldsymbol{\theta})\frac{\partial^2}{\partial x^2}u(x;\boldsymbol{\theta})\right) = f(x)$$

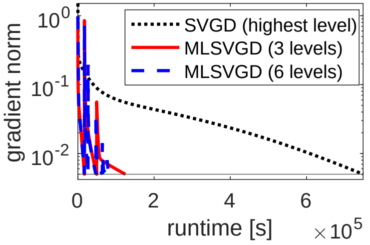
- Forward model $G$ solves PDE with finite differences on a mesh of 601 equally-spaced points; surrogates $G^{(\ell)}$ use $51, 101, \ldots, 501$ points
- Prior is log-normal and data $\boldsymbol{y}$ is solution $u$ observed at 41 equally-spaced points and polluted with 0.01% Gaussian noise

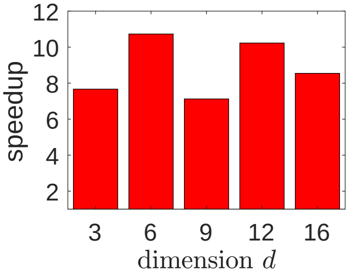
# Numerical results: MLSVGD speedup




Speedup of MLSVGD over SVGD is consistent across dimension

# Conclusion




- MLSVGD exploits a hierarchy of distributions to achieve speedup over single-level SVGD for Bayesian inference
- Analysis conducted in mean-field limit shows a cost complexity reduction of MLSVGD compared to single-level SVGD
- Numerical experiments conducted in discrete-time and finite-particle regime demonstrate up to one order of magnitude speedup