Math Geosci (2021) 53:375–393
https://doi.org/10.1007/s11004-021-09937-x

SPECIAL ISSUE

p-Kernel Stein Variational Gradient Descent for Data Assimilation and History Matching

Andreas S. Stordal[1,2] · Rafael J. Moraes[3] · Patrick N. Raanes[1] · Geir Evensen[1,4]

Received: 14 February 2020 / Accepted: 8 February 2021 / Published online: 17 March 2021

Abstract A Bayesian method of inference known as "Stein variational gradient descent" was recently implemented for data assimilation problems, under the heading of "mapping particle filter". In this manuscript, the algorithm is applied to another type of geoscientific inversion problems, namely history matching of petroleum reservoirs. In order to combat the curse of dimensionality, the commonly used Gaussian kernel, which defines the solution space, is replaced by a p-kernel. In addition, the ensemble gradient approximation used in the mapping particle filter is rectified, and the data assimilation experiments are re-run with more relevant settings and comparisons. Our experimental results in data assimilation are rather disappointing. However, the results from the subsurface inverse problem show more promise, especially as regards the use of p-kernels.

Keywords Ensemble methods · Bayesian inversion · Data assimilation · History matching · Adjoint computation

✉ Andreas S. Stordal
asto@norceresearch.no

1 NORCE, Norwegian Research Center, Nygårdsporten 112, 5008 Bergen, Norway

2 Department of Mathematics, University of Bergen, Postboks 7803, 5020 Bergen, Norway

3 CENPES, Petrobras Research and Development Center, Av. Haracio Macedo 950, Cidade Universitaria, Rio de Janeira, RJ 21941-915, Brazil

4 NERSC, Nansen Environmental and Remote Sensing Center, Thormœhlensgate 47, 5006 Bergen, Norway

1 Introduction

Bayesian inference in data assimilation (DA) has been researched for several decades and is frequently applied in the petroleum industry today. Due to the typically vast computational cost of the forward simulation problem, classic Bayesian Monte Carlo methods, such as Markov chain Monte Carlo (MCMC) and importance sampling, are not applicable. Therefore, approximate Bayesian methods, which require fewer computational resources, are applied in most real-world cases. It is convenient to separate these methods into derivative-based and derivative-free approaches. Derivative-based approximations include Randomized Maximum Likelihood (RML, Kitanidis 1995; Oliver et al. 1996), distributed Gauss–Newton solvers (Gao et al. 2017), and EDA-4DVar (Carrassi et al. 2018).

However, it is more common to use derivative-free methods due to the common lack of adjoint code in both commercial simulators and large scale models in general. Many of the modern derivative-free methods are based on the ensemble Kalman filter (EnKF, Evensen 2004). During the last decade the method of choice has become the iterative ensemble methods such as iterative EnKF/EnKS (Bocquet and Sakov 2014), the ensemble RML (Chen and Oliver 2013) and the ensemble smoother with multiple data assimilations (ESMDA, Emerick and Reynolds 2013). Unfortunately, in the general (non-linear) case, none of these ensemble methods converge to the true posterior distribution in the limit of an infinite sample size. It is possible to nest some of these methods in the importance sampling framework to achieve asymptotic optimality (Stordal and Elsheikh 2015; Stordal 2015; Stordal and Karlsen 2017), but at a considerable cost. Other approximate approaches to Bayesian sampling that go beyond the traditional approach of MCMC include sampling via optimal transport (El Moshely and Marzouk 2012; Reich 2013; Marzouk et al. 2017) and implicit sampling (Skauvold et al. 2019)

In this work, the recently published Stein variational gradient descent method (SVGD, Liu and Wang 2016) is applied in history matching for the first time. The kernels introduced in the algorithm are more appropriate for high dimensional applications. In addition, an alternative derivative-free implementation is discussed. Previous applications of SVGD includes Bayesian logistic regression (Liu and Wang 2016), training of neural nets (Feng et al. 2017), sequential filtering of the Lorenz-63 and -96 models (Pulido and van Leeuwen 2019; Pulido et al. 2019), inference on a simple linear and nonlinear partial differential equation (PDE) using a subspace Hessian projection (Chen et al. 2019) and a Gauss–Newton formulation (Detommaso et al. 2018).

The outline of this paper is as follows: Sect. 2 introduces the SVGD theory. In Sect. 3, the implementation is presented with and without an adjoint code, and the choice of kernels for high dimensional applications. Examples with a toy model, with the Lorenz systems, and with a reservoir model, are given in Sect. 4. A summary and discussion of future work concludes the paper in Sect. 5.

2 Stein Variational Gradient Descent

Let $X \in \mathscr{X} \subseteq \mathbb{R}^{N_x}$ be the unknown vector of interest. In Bayesian statistics, it is assumed random. Denote its (prior) probability density function (PDF), $p(x)$. It is observed through the noise-corrupted measurements

$$Y = \mathcal{H}(X) + \epsilon, \tag{1}$$

where $\epsilon \in \mathbb{R}^{N_y}$ is a random noise vector, and $\mathcal{H}$ is a (possibly nonlinear) mapping $\mathcal{H} : \mathscr{X} \to \mathscr{Y} \subseteq \mathbb{R}^{N_y}$. The likelihood, $p(y|x)$, follows from the distribution of ϵ, whose framing as an *additive* error is mere convention. The posterior is given by Bayes' rule: $p(x|y) \propto p(y|x)\, p(x)$. For complex problems, such as large scale inversion and DA, Bayesian inference often takes the form of (approximate) sampling from the posterior.

Given two densities p and q, the Stein discrepancy can be used to quantify their difference. It is defined as

$$\mathbb{S}(q, p) = \max_{f \in \mathscr{F}} \left(\mathbb{E}_q[(\partial_x \log p(X))\, f(X) + \partial_x f(X)] \right), \tag{2}$$

where $\mathscr{F}$ is a set of test functions, $f : \mathscr{X} \to \mathbb{R}$, that are sufficiently smooth and satisfy

$$\mathbb{E}_p[\left(\partial_x \log p(X)\right) f(X) + \partial_x f(X)] = 0. \tag{3}$$

That is, f is in the Stein class of p (Liu et al. 2016). The variational problem Eq. 2 is computationally intractable in most cases, and therefore Chwialkowski et al. (2016) and Liu et al. (2016) introduced the *kernelized* Stein discrepancy, where the function space $\mathscr{F}$ is set to the unit ball within a reproducible kernel Hilbert space (RKHS), $\mathscr{H}$, for which analytical solutions may be obtained. The Kullback–Leibler (KL) divergence from p to q is given by

$$D_{\mathrm{KL}}(q \| p) = \int q(x) \log \left(\frac{q(x)}{p(x)} \right) dx. \tag{4}$$

It was related to kernelized Stein discrepancy by Liu and Wang (2016), who showed that if X has density q, then the KL divergence to p from the PDF $q_{\mathcal{T}}$ of the transformation

$$\mathcal{T}(x) = x + \epsilon \phi(x), \tag{5}$$

has a derivative, $\frac{d}{d\epsilon} D_{\mathrm{KL}}(q_{\mathcal{T}} \| p)$, that is maximized at $\epsilon = 0$ for

$$\phi(\cdot) = \mathbb{E}_q[K(X, \cdot)\, \partial_x \log p(X) + \partial_x K(X, \cdot)], \tag{6}$$

where $K(x, x')$ is the unique kernel defining $\mathscr{H}$.

In the context of Bayesian inference, Bayes' rule may be computed by a KL minimizing algorithm (Liu and Wang 2016): the SVGD starts with a sample $\{x_0^i\}_{i=1}^N$ from the prior density $p(x)$ and iteratively updates x_k^i using the empirical (Monte Carlo) version of Eq. 6 in Eq. 5

$$x_{k+1}^i = x_k^i + \epsilon_k N^{-1} \sum_{j=1}^{N} \left(K(x_k^j, x_k^i) \partial_x \log p(x_k^j|y) + \partial_x K(x_k^j, x_k^i) \right), \tag{7}$$

for all particles, or ensemble members, $i = 1, \ldots, N$. If the kernels are chosen to be Gaussian, i.e. $K(x, x') = \exp(-\frac{1}{2} \left\| x - x' \right\|_2^2)$, then Eq. 7 reduces to

$$x_{k+1}^i = x_k^i + \epsilon_k N^{-1} \sum_{j=1}^{N} K(x_k^j, x_k^i) \left(\partial_x \log p(x_k^j|y) + (x_k^i - x_k^j) \right). \tag{8}$$

The last term may be seen as a weighted average of the gradient of the log posterior and of its similarity to the other members. The first term guides the sample points towards the maximum of the log posterior, while the second term repulses sample points that are too close. Equation 8 presupposes the use of simple gradient descent optimization In practice, Liu and Wang (2016); Pulido and van Leeuwen (2019) both used an adaptive subgradient variation of this (ADAGRAD, Duchi et al. 2011), which is also the choice made here.

It is interesting to note that if the kernel is degenerate, then the SVGD algorithm produces N copies of the MAP estimate (or N local optima in the general non-convex case). It was shown in Liu (2017) that the continuous-time infinite-sample limit of the density induced by SVGD satisfies the Vlasov equation (Vlasov 1961)

$$\partial_t \mu_t = -\nabla \cdot (\phi \mu_t)(\mu_t). \tag{9}$$

Using Stein's lemma (Stein 1972) it may be shown that $\partial \mu_t = 0$ for $\mu_t = p(x|y)$, meaning that the SVGD algorithm can be viewed as particle flow. Furthermore, Zhang et al. (2018) showed that SVGD can be combined with Langevin dynamics to obtain a stochastic particle flow version of SVGD that may avoid the issue of particles collapsing to a single point or getting stuck in a local mode. This will not be discussed further here as it did not have any impact on the problems presented here. For more details the reader is referred to Zhang et al. (2018) and references therein.

3 Implementation

In the following it is assumed that the prior and the measurement error are both Gaussian. Hence

$$\partial_x \log p(x|y) = \mathbf{C}_x^{-1}(\mu - x) + \mathbf{H}^\top \mathbf{C}_\epsilon^{-1}(y - \mathcal{H}(x)), \tag{10}$$

where $\mathbf{H}$ is the sensitivity matrix (the derivative of $\mathcal{H}$ with respect to x) and $\mathbf{C}_x$ and $\mathbf{C}_\epsilon$ are the covariance matrices of X and ϵ, respectively. Computing $\mathbf{H}$ is challenging since it requires an adjoint code. It is therefore of great interest to study alternative implementations of SVGD using approximate gradients. It should also be mentioned that Han and Liu (2018) presented a weighted SVGD wherein the gradient is computed using a surrogate density instead of the target density. In the following description, however, the focus is to use either an adjoint code, or an ensemble approximation of $\mathcal{H}$.

3.1 Adjoint Implementation

In large scale PDE-constrained optimization for the solution of inverse problems (e.g. in reservoir history matching) it is convenient to formulate $\mathcal{H}$ as depending on two separate variables, $\mathcal{H} = \mathcal{H}(\xi, x)$, where the dynamic ones, ξ, depend deterministically on the fixed (but unknown) parameter fields x through $g(\xi, x) = 0$, where g is the set of (discretized) forward model equations. Implicit differentiation yields $\partial_x \xi = -(\partial_\xi g)^{-1}\, \partial_x g$, so that the (total) derivative of $\mathcal{H}$ with respect to x, required in Eq. 10, is given by

$$\mathbf{H} = \partial_\xi \mathcal{H}\, \partial_x \xi + \partial_x \mathcal{H} \tag{11}$$

$$= -\partial_\xi \mathcal{H}\, (\partial_\xi g)^{-1}\, \partial_x g + \partial_x \mathcal{H}. \tag{12}$$

Using the shorthand

$$w = \mathbf{C}_\epsilon^{-1}(y - \mathcal{H}), \tag{13}$$

the last term in Eq. 10 becomes

$$\mathbf{H}^\top w = \big(- \partial_\xi \mathcal{H}\, (\partial_\xi g)^{-1} \partial_x g + \partial_x \mathcal{H}\big)^\top w \tag{14}$$

$$= -(\partial_x g)^\top \Big\{ (\partial_\xi g)^{-\top} \big[(\partial_\xi \mathcal{H})^\top w \big] \Big\} + (\partial_x \mathcal{H})^\top w. \tag{15}$$

The use of brackets in Eq. 15 seems heavy-handed. However, in contrast to the explicit computation of $\mathbf{H}$ of Eq. 12 (known in the literature as the forward, or direct, method (Rodrigues 2006; Anterion et al. 1989), or gradient simulator (Oliver et al. 2008)), the ordering of the brackets in Eq. 15 yields a sequence of computations, now involving the transposed system, with a cost proportional to a single simulation. This sequence of operations is known in the literature as adjoint, or backward, method (Rodrigues 2006; Chavent et al. 1975).

A major hurdle in applying the adjoint method (as well as in the Forward/Direct method) is writing the code for the computation of the partial derivatives of g and $\mathcal{H}$ with regard to both ξ and (mainly) x. In this study, automatic differentiation (Bendtsen and Stauning 1996) is applied.

3.2 Adjoint-Free Implementation

Zhou (2008) showed that, by the reproducing property of $f \in \mathscr{H}$,

$$\partial_x f(x) = \langle f, \partial_{x'} K(x, \cdot) \rangle_{\mathscr{H}}. \tag{16}$$

However, the inner product on $\mathscr{H}$ is given by

$$\langle f, g \rangle_{\mathscr{H}} = \sum_{i=1}^{\infty} \frac{\langle f, \phi_i \rangle_{L_2} \langle g, \phi_i \rangle_{L_2}}{\lambda_i}, \tag{17}$$

where ϕ and λ are the eigenfunctions and eigenvalues of K. This is not trivial to compute due to the high-dimensional integrals and the computation of eigenfunctions.

An alternative formulation was used by Pulido et al. (2019) where the gradient of $\mathcal{H}$ was approximated by Monte Carlo integration

$$\tilde{\mathbf{H}}(x) = N^{-1} \sum_{i=1}^{N} \mathcal{H}(x^i) K(x, x^i), \tag{18}$$

using the sample available from the SVGD algorithm at each iteration. A further normalization was later introduced to improve the approximation by dividing each value of $\tilde{\mathbf{H}}$ by the sum of the kernels at that point. The approximation was based on the claim that Eq. 18 would converge to

$$\widehat{\mathbf{H}}(x) = \int \mathcal{H}(x') \, \partial_{x'} K(x, x') \, dx', \tag{19}$$

for $\mathcal{H} \in \mathscr{H}$ as N goes to infinity. However, this uses the plain L_2 inner product,

$$\langle \mathcal{H}, K \rangle = \int \mathcal{H}(x') \, K(x, x') \, dx', \tag{20}$$

which is not a RKHS, so Eq. 19 is incorrect. However, if it assumed that $\mathcal{H}(x) K(x, x') = 0$ in the limits or at the boundary (in case of finite support), then integration by parts yields

$$\int \mathcal{H}(x') \, \partial_{x'} K(x, x') \, dx' = -\int \mathbf{H}(x') \, K(x, x') \, dx' = -\mathbb{E}_{K(x,\cdot)}[\mathbf{H}]. \tag{21}$$

Hence, Eq. 19 can be seen as a kernel-smoothed derivative of $\mathcal{H}$ around the point x, provided the kernel is normalized (integrates to one).

Furthermore, their unsatisfactory results are a consequence, not only of the approximation in Eq. 19, but also of the fact that the sample is drawn from a particular distribution and not uniformly in space. Hence the notion that Eq. 18 will converge to $\int \mathcal{H} K$ is also incorrect. This can be corrected by noting that the samples are drawn

from a distribution q, hence, Eq. 18 converges to $\mathbb{E}_q[\mathcal{H}K]$ unless the integrand is normalized by q. An alternative approach is therefore to draw the sample from the kernel density, $\widehat{q}(x) = N^{-1}\sum_i K(x, x^i)$ at each iteration, yielding the Monte Carlo estimate of Eq. 19 given by

$$\mathbf{H}(x) \approx -\sum_i \mathcal{H}(x^i)\frac{\partial_{x'}K(x, x^i)}{\sum_j K(x^i, x^j)}. \tag{22}$$

Alternatively, Pulido et al. (2019) suggest using an average ensemble approximation for the derivative, as in the EnKF. This does not allow for local information, and the algorithm falls into the ensemble-based category where quasi-linearity of the model is assumed.

Finally, since $\tilde{\mathbf{H}}$ is estimated using a sample, it will suffer from Monte Carlo errors. This can, just like in the EnKF and its variants, be reduced by introducing localization. A user defined correlation matrix ρ can be used to taper $\mathcal{H}(x)$ in Eq. 22 via the Schur product $\rho \circ \mathcal{H}(x)$.

3.3 Extension to p-Kernels

In the literature discussed above, almost all applications selected $K(x, x')$ to be a Gaussian kernel. The choice of the kernel may not be critical in low dimensional problems. For high dimensional problems, however, the Gaussian kernel is not able to separate points locally, due to the curse of dimensionality. It will produce degeneracy, as illustrated in importance sampling. To overcome this issue, scaled kernels were used both in Liu and Wang (2016) and Detommaso et al. (2018). In many cases, a simple scaling of the kernel bandwidth is not sufficient to separate points in high dimensions (increasing the bandwidth in high dimensions results in almost uniform weights) and an alternative approach of using p-kernels (Francois et al. 2005) is therefore chosen here. The kernel is specified by

$$K(x, x') = \exp\left(-\left(d(x, x')/\sigma\right)^p\right). \tag{23}$$

The special case $p = 2$ and $d(x, x') = \left\|x - x'\right\|_2$ reduces the p-kernel to a Gaussian kernel. As the dimension increases, the distances ($i \neq j$), which are χ^2 if X is Gaussian, tend to cluster away from zero, hence the term with $i = j$ in Eq. 8 increasingly dominates unless the bandwidth of the kernel is chosen to be very large. In this case the kernel will not separate points, and the local property of the kernel is lost. To overcome this, p-kernels force the kernel value to stay close to 1 until it reaches the lower tail of the distance distribution of the sample, and then decay appropriately. Specifically, the parameters p and σ are here set to match the α-percentile, d_α, and the $(1-\alpha)$-percentile, $d_{1-\alpha}$, of the empirical distribution of distances, meaning

$$K(0, d_\alpha) = 1 - \alpha, \quad K(0, d_{1-\alpha}) = \alpha, \tag{24}$$

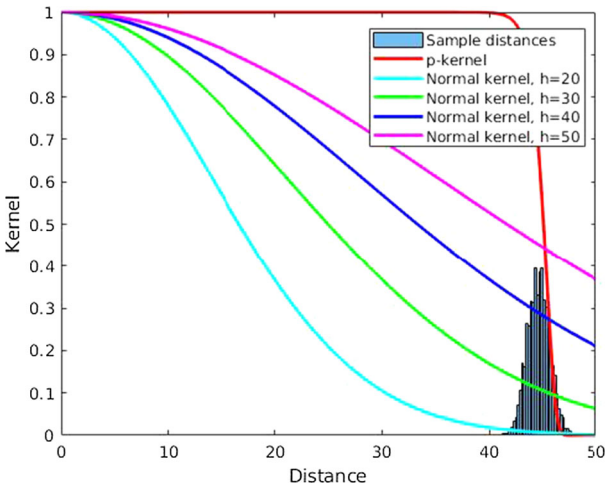


Fig. 1 p-kernel (red) and Gaussian kernels with different bandwidths together with the sample distribution of distances

which has the following solution

$$p = \frac{\log\left(\frac{\log\alpha}{\log(1-\alpha)}\right)}{\log\left(\frac{d_{1-\alpha}}{d_\alpha}\right)}; \quad \sigma = \frac{d_{1-\alpha}}{(-\log\alpha)^{1/p}}. \tag{25}$$

In Fig. 1, the distribution of Euclidean distances for a standard Gaussian random vector of dimension 1000 with $\alpha = 0.05$ is plotted. Also included is a set of Gaussian kernels with different bandwidths and the p-kernel. The problem of using Gaussian kernels is evident: either the kernel function dies out before it reaches the sampled distances, or the bandwidth is so large that it does not separate the points. Hence uniform weights could in practice replace the kernel.

4 Examples

4.1 Univariate Example

The first test is to reproduce the toy example in Pulido et al. (2019) in order to validate our derivative approximation (Eq. 22) and demonstrate the sampling properties of SVGD. In the simple toy problem of Pulido et al. (2019), the prior is given by a univariate Gaussian density with mean 0.5 and variance of 1. The measurement model is $y = x^2 + \epsilon$, where ϵ is a zero-mean Gaussian random variable with a variance of 0.5. The actual measurement is set to $y = 3$. Although x^2 is not in the Gaussian RKHS (Minh 2010), the reproducing property is still valid for the derivative in this particular case. That is, $\int u^2 \partial_x K(x, u)\, du = 2x$.

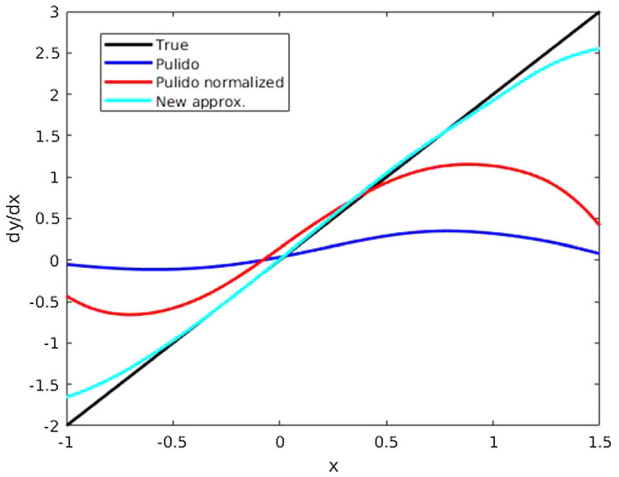
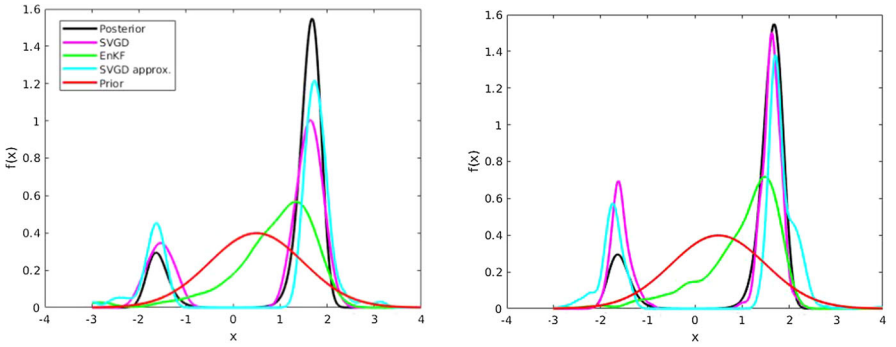


Fig. 2 The exact and approximate derivatives of the model $y = x^2$



(a) N=100 **(b)** N=1000

Fig. 3 Posterior and posterior estimates in the toy model, using two different ensemble sizes, N

Figure 2 shows the derivative approximation of Eq. 18 and its normalized version (blue and red), reproduced from Pulido et al. (2019), together with the true derivative (black) and our proposed method (cyan) in Eq. 22 of the prior sample. The results clearly show the inadequacy of the approximation (blue line) in Eq. 18, which in this case is not even monotonic. The normalized version (red line) is just a damped version and exhibits the same non-monotonic behavior. However, it can also be seen that the corrected Monte Carlo estimate is slightly biased due to deficiency the of kernel density estimation (Silverman 1986) for the tails of the prior density.

In Fig. 3 we plot density estimates, including SVGD using the exact (magenta) and the new approximate (cyan) derivative of Eq. 22. The standard EnKF solution (green) and the prior (red) are also included. The sample size is 100 and 1000. The error in SVGD is due to the sampling error and bias in the kernel density estimation. The SVGD with the (new) approximate derivative has an additional bias due to the error

in the derivative estimate, as seen by the left and right skewness in Fig. 3. Note that this error is reduced with increased sample size.

4.2 Data Assimilation Tests with the Lorenz Systems

Pulido and van Leeuwen (2019) tested the SVGD in the sequential filtering setting (described below) of data assimilation (DA, Wikle and Berliner 2007), applying it in the analysis step by constituting the prior from the ensemble using Gaussian mixtures. They called this the mapping particle filter (MPF), however, the SVGD label is maintained here. Their results suggest that the SVGD filter achieves the accuracy of the (bootstrap) particle filter on the Lorenz-63 system, and the EnKF on the Lorenz-96 system.

Unfortunately, their result lacks relevance because the Lorenz systems, used as coarse prototypes of atmospheric dynamics, are inundated with noise (see appendix A), so that the extended Kalman filter (EKF, Jazwinski 1970), or even 3D-Var, can achieve optimal accuracy, at much lower costs. This issue is rectified here by benchmarking the SVGD filter (i.e. MPF) with more standard settings.

The Lorenz-63 system (Lorenz 1963; Sakov et al. 2012) is given by the $N_x = 3$ coupled ordinary differential equations

$$\dot{x} = \sigma(y - x), \tag{26}$$

$$\dot{y} = rx - y - xz, \tag{27}$$

$$\dot{z} = xy - bz, \tag{28}$$

with parameter values $r = 28$, $\sigma = 10$, and $b = 8/3$. The true trajectory, $x(t)$, is computed using the fourth-order Runge–Kutta scheme, with time steps of 0.01 time units, and no model noise. Observations of the entire state vector (i.e. $\mathbf{H} = \mathbf{I}_{N_x}$) are taken $\Delta t_{\text{Obs}} = 0.25$ time units apart with a noise variance of 2 (i.e. $\mathbf{C}_\epsilon = 2\mathbf{I}_{N_x}$).

The Lorenz-96 system (Lorenz 1996; Ott et al. 2004) is given by the coupled ordinary differential equations

$$\dot{x}_m = (x_{m+1} - x_{m-2})\, x_{m-1} - x_m + F, \tag{29}$$

for $m = 1, \ldots, N_x$, with periodic boundary conditions, and a constant "force" of $F = 8$. These are integrated using the fourth-order Runge–Kutta scheme, with time steps of 0.05 time units, and no model noise. The state dimension is set to $N_x = 10$ (rather than the typical value of 40) so that the particle filter is practical. Observations of the entire state vector (i.e. $\mathbf{H} = \mathbf{I}_{N_x}$) are taken $\Delta t_{\text{Obs}} = 0.1$ time units apart with unit noise variance (i.e. $\mathbf{C}_\epsilon = \mathbf{I}_{N_x}$).

The methods are assessed by their accuracy, as measured by root-mean squared error (RMSE)

$$\text{RMSE}(t) = \sqrt{\frac{1}{N_x} \left\| x(t) - \bar{x}(t) \right\|_2^2}, \tag{30}$$

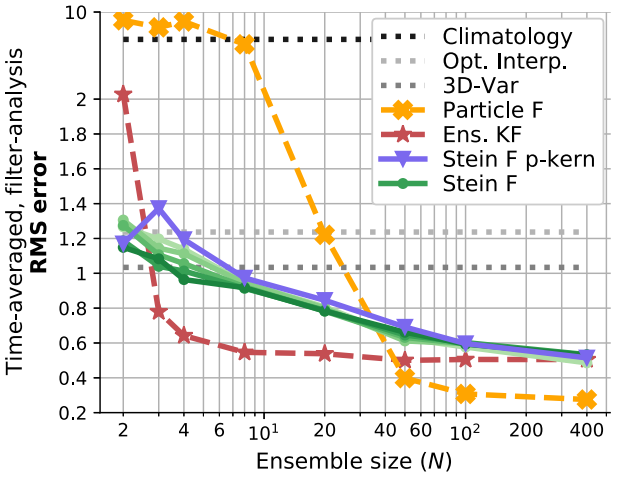


Fig. 4 Benchmarks of filter accuracy (RMSE) from synthetic DA experiments on the *Lorenz-63* system, plotted as functions of the ensemble size, N. For the SVGD (label "Stein") filter, one curve is plotted for each tuning value tested for the kernel bandwidth. Note that the vertical scale is compressed above 2

which is recorded following each analysis of the latest observation $y(t)$. After the experiment, the instantaneous RMSE(t) are averaged for all $t > 20$.

Comparison of the benchmark performance of SVGD is made to that of the EnKF (Hunt et al. 2004) and the bootstrap particle filter (with universal resampling, triggered if $\|w\|^{-2} < N/2$, where w is the vector of weights). In addition, baseline methods are included for context. Their analysis estimates, x_a, are computed as follows: $\bar{x}$ for Climatology, $\bar{x} + \mathbf{K}(\overline{\mathbf{C}})\,[y - \bar{x}]$ for Optimal Interpolation, and $x_f + \mathbf{K}(c\mathbf{I})\,[y - x_f]$ for 3D-Var. Here, $\bar{x}$ and $\overline{\mathbf{C}}$ are the mean and covariance of the (invariant measure of the) system dynamics, $\mathbf{K}(\mathbf{C}) = \mathbf{C}\mathbf{H}^\top(\mathbf{H}\mathbf{C}\mathbf{H}^\top + \mathbf{R})^{-1}$ is a gain matrix, x_f is the model forecast of the previous x_a, and c is a scaling factor subject to tuning.

The RMSE averages of each method are tabulated for a range of ensemble sizes, N, and plotted as curves in Figs. 4 and 5. The plotted scores represent the lowest obtained among a large number of tuning settings, selected for optimality at each point. For the PF the tuning parameter is the bandwidth (scaling) of the regularizing post-resample jitter, whose covariance is computed from the weighted ensemble. For the EnKF (Hunt et al. 2004) the tuning parameters are (i) the post-analysis inflation factor and (ii) whether or not to apply random, covariance-preserving, post-analysis rotations (Sakov and Oke 2008). For the SVGD the tuned parameters are: (i) the number of iterations (maximum: 100) and (ii) the step size for ADAGRAD, (iii) the bandwidth of the isotropic Gaussian kernels of the priors' mixture, and (iv) the bandwidth of the isotropic Gaussian kernels defining the RKHS. In the case of p-kernels, the latter bandwidth is determined automatically by Eq. 25, along with the p parameter.

As can be seen in both Figs. 4 and 5, the abscissa (N-axis) can be roughly partitioned into three segments, with regard to the ordering of the method scores: For intermediate ensemble sizes, N, the EnKF obtains the lowest RMSE score among all of the methods. For the largest ensemble size, $N = 400$, the particle filter obtains the best score, while

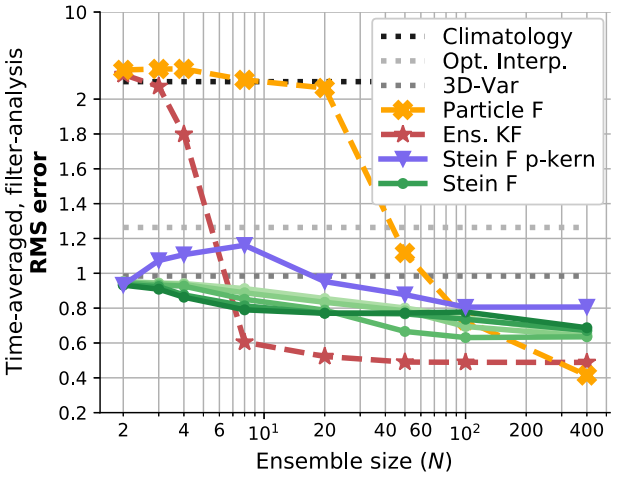


Fig. 5 Benchmarks of filter accuracy (RMSE) from synthetic DA experiments on the *Lorenz-96* system, plotted as functions of the ensemble size, N. For the SVGD (label "Stein") filter, one curve is plotted for each tuning value tested for the kernel bandwidth. Note that the vertical scale is compressed above 2

in the case of Lorenz-63, the SVGD filter achieves the same score as the EnKF (which it already obtained with $N = 8$). Tests with $N > 400$ were not affordable due to the cost of SVGD. For small N, the SVGD filter obtains lower RMSE than both the particle filter and the EnKF. However, as this score is not better than 3D-Var using a background matrix proportional to identity, it does not hold much relevance.

Furthermore, the p-kernel modification generally scores worse than SVGD with Gaussian kernels and tuned bandwidths. This is not very surprising, because the dimensionality is low, preempting their rationale. Moreover, an investigation reveals that, for small ensemble sizes, the p-kernel approach has a tendency to use very large values of p. For example, the corresponding curve in Fig. 4 spikes at $N = 3$. The reason is that, sometimes, the three distances between the three ensemble members are almost equal, but far from zero, requiring, on average, $p = 22$. By contrast, this does not occur at $N = 2$, because then there is only one distance, nor does it occur when N is very large and the distances rarely coincide.

In summary, the SVGD filter of Pulido and van Leeuwen (2019), with or without our p-kernel modification, while functional, does not achieve the same accuracy as standard DA methods. It is important to remember that the SVGD filter involves many iterations, quite a few tuning parameters, and linear algebra with $N_x \times N_x$ matrices, all of which add to its cost. It seems likely that the disappointing performance of SVGD stems from several issues: Firstly, it is limited by the precision of the prior approximations, which come from Gaussian mixtures, and may suffer from the curse of dimensionality, or from using simple, identity covariances. Secondly, the optimization process may not be very successful. In cursory experiments, other optimization routines were tested, and testw were conducted using more iterations and the use of pre-conditioning, all of which sometimes could yield improved performance. A thorough examination is left for future work.

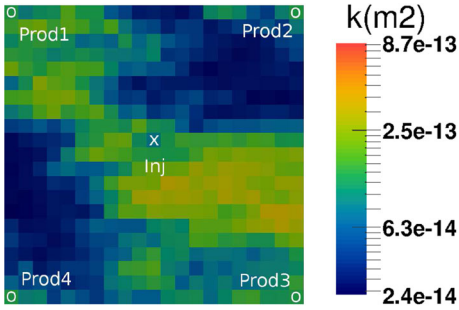


Fig. 6 The synthetic inverted five-spot model used in the numerical experiments. One of the 1000 permeability realizations is shown

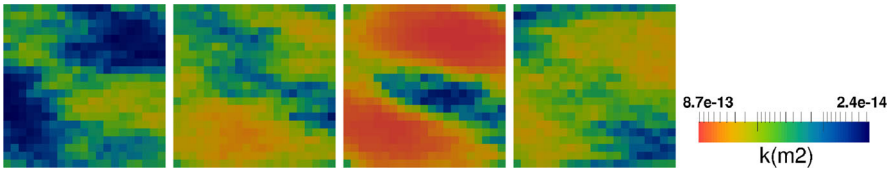


Fig. 7 Four different permeability realizations from the ensemble of 1000 members

4.3 Reservoir History Matching Example

This section presents the results of SVGD with Gaussian kernels and p-kernels on a synthetic reservoir case. The reservoir is a two-dimensional model with 21 by 21 grid cells consisting of incompressible two-phase flow (oil-water). In each corner cell there is a production well and in the center an injection well (see Fig. 6). The porosity is assumed to be known in each grid cell with a value of 0.3. The forward model equations, as well as the adjoint model for this example, are fully depicted in de Moraes et al. (2018).

The permeability field is considered unknown with a prior distribution that is Gaussian. There are 1000 realizations available for this model (Jansen 2011), which specify the prior mean and covariance. Figure 7 shows four different permeability realizations from the ensemble. This is the same set up as in Stordal and Elsheikh (2015).

The observed data are generated from a synthetic truth (of the permeability field), chosen at random as one realization from the ensemble. Specifically, the observations are the water rates resulting from the simulation of 10 years of the truth, observed every six months, with a 5% white noise level. For more details on the reservoir settings, the reader is referred to Jansen (2011) and de Moraes et al. (2018).

The ADAGRAD method for optimization was implemented with a step size of 0.1 and an autocorrelation of 0.9, which are the standard settings for ADAGRAD. The maximum number of iterations was set to 50. The experiments were conducted with both Gaussian and p-kernels, and with ensemble sizes $N = 100$ and $N = 1000$.

The results for the data match with 100 members are shown in Fig. 8, as well as the data match with 1000 members in Fig. 9.

As can be observed from the results, both history matching using 100 members (Fig. 8) and 1000 members (Fig. 9) provide a reasonably good match for each individual well. Even though the prior ensemble either mostly underestimates the

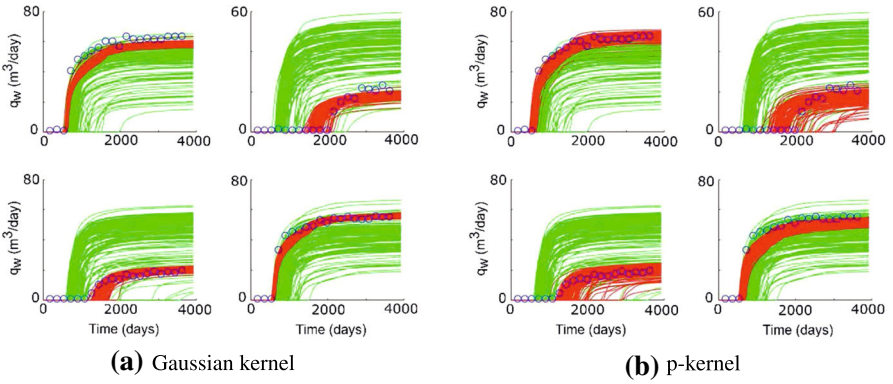


Fig. 8 Data predictions from prior (green) and posterior (red) ensemble with 100 members. The blue circles represent the observed data. The y-axis represents water rate (q_w) in m^3/day and the x-axis represents time in days. Data match using Gaussian kernel (**a**) and p-kernel (**b**)

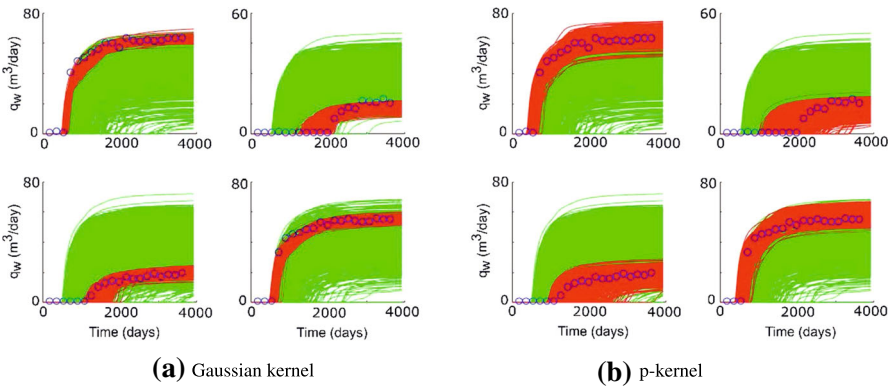


Fig. 9 Data predictions from prior (green) and posterior (red) ensemble with 1000 members. The blue circles represent the observed data. The y-axis represents water rate (q_w) in m^3/day and the x-axis represents time in $days$. Data match using Gaussian kernel (**a**) and p-kernel (**b**)

water rate for PROD1 and PROD3 wells (top-left and bottom-right panels, respectively) or overestimates the PROD2 and PROD4 water rates (top-right and bottom-left panels, respectively), the posterior data predictions capture the observed data well. Additionally, better data predictions are obtained with 1000 members. Furthermore, underestimation of uncertainty is often a problem in history matching. It therefore is interesting to see that the posterior ensemble using the p-kernel contains a much larger uncertainty when compared to the posterior ensemble using the Gaussian kernel. This is even clearer if one observes the uncertainty around the water break-through of wells PROD2 and PROD4 (top-right and bottom-left panels on Fig. 9b).

Next, the history matching results are further investigated by comparing the permeability marginal PDFs conditioned to production data, shown in Fig. 10. In general, the conditioned marginal PDFs for both Gaussian kernel and p-kernel provide a reasonably good representation of uncertainty. While subtle, one may note that the conditioned

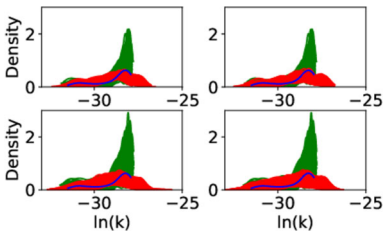


Fig. 10 Marginal PDFs of permeability. Each curve represents one ensemble member. Top: 100 members. Bottom: 1000 members. Left: Gaussian kernel. Right: p-kernel. The prior is shown in green, and the posterior in red. The blue curves represent the true (reference) marignal pdf

PDFs using p-kernel (right-side panels of Fig. 10) better represent the uncertainty when compared to the conditioned PDFs using Gaussian kernel, mainly when looking at the permeability range (horizontal axis) and the spread of the PDFs at the bottom, that use a 1000 member ensemble.

Comparisons with RML (Oliver et al. 1996) in order to further investigate the usage of p-kernels to better represent different high-dimensional geological settings will be investigated in the future.

Even though the results presented here indicate that the SVGD is a promising alternative for reservoir uncertainty quantification, it is only feasible, even considering this relatively small example, with an efficient gradient computation strategy. However, other data assimilation/uncertainty quantification strategies, such as RML (Chen and Oliver 2012), are also only feasible combined with efficient gradient computation strategies. Even though derivative-free formulations have been devised to overcome the adjoint implementation challenges (e.g. EnRML), the performance of both formulations (derivative and derivative-free) in an SVGD setting should be investigated in reservoir history matching problems.

5 Conclusions

The Stein variational gradient descent (SVGD) algorithm was extended to p-kernels, and we discussed derivative and derivative-free implementations. With an adjoint code, the algorithm was applied to subsurface petroleum history matching for the first time. The results showed that it can obtain reasonable data match with both Gaussian kernels and p-kernels, but that the posterior uncertainty was larger using the p-kernels, hence demonstrating the potential usefulness of p-kernels in higher dimensions.

The SVGD was also tested on two small chaotic dynamical systems. For these models the measurement operator is linear, thus the sensitivity matrix is trivial. However, as the prior distribution is unknown, it has to be approximated using kernel density estimation. The results showed that the SVGD is outperformed by the EnKF for intermediate ensemble sizes and the particle filter for large ensemble sizes, tempering the impression given by Pulido and van Leeuwen (2019). This contradiction is most likely a result of using a less noisy system (the dynamics in Pulido and van Leeuwen (2019) were entirely driven by noise) and the fact that the posterior sample from SVGD cannot overcome the deficiencies of the density estimate of the prior. Further analysis and improvements are left for future work.

The reservoir example indicated more potential for the SVGD algorithm. In particular, the uncertainty quantification of the posterior seemed to improve when using the p-kernel instead of the Gaussian kernel. A more detailed investigation of the uncertainty quantification properties of SVGD, in particular when compared to RML or ensemble smoothers, is left for future work. In addition, the ensemble approximation of the derivative for reservoir models will also be investigated in the future, in combination with localization.

Acknowledgements The first, third and fourth author acknowledge financial support from the DIGIRES project, supported by the Norwegian research Council and industry partners (in random order), Equinor, Petrobras, Aker BP, Neptune Energy, Wintershall Dea, Lundin Norway and Vår Energi. The second author acknowledges Petrobras for the support and permission to publish this work.

Funding Open access funding provided by University of Bergen (incl Haukeland University Hospital).

Appendix A: The Magnitude of the Noise Used to Test the MPF

The model noise covariance used by Pulido and van Leeuwen (2019) is $\mathbf{Q} = 0.3\mathbf{I}_{N_x}$ for each $\Delta t_{\text{Obs}} = 0.01$ (Lorenz-63) or $\Delta t_{\text{Obs}} = 0.05$ (Lorenz-96) (Pulido and van Leeuwen (2019) state that the value of Q used for the Lorenz-63 system is based on climatological variances. However, this is probably inaccurate, judging by the code the authors received in email communication with the principal author. The paper does not clearly state how frequently the noise with covariance Q should be applied, but the code indicates that it is every DA cycle, Δt_{Obs}).

A.1 The Relative Importance of Noise/Model

The near-optimal RMSE scores that can be achieved for each twin experiment are around

- 0.50 (resp. 0.03) for Lorenz-63, with (resp. without) noise.
- 0.50 (resp. 0.12) for Lorenz-96, with (resp. without) noise.

The noiseless scores were obtained with the PF and the EnKF, with large ensemble sizes. The RMSE (The RMSE is not defined by Pulido and van Leeuwen (2019), but their code indicates that the square root is taken after averaging in time (as well as space). This is different from the DA literature standard, but our investigations show that it makes little numerical difference in these experiments) score of 0.5 can be gleaned from Fig. 7 and Fig. 9 of Pulido and van Leeuwen (2019). The huge

difference in RMSE between the noisy and noiseless cases indicates that the systems are dominated by noise.

A.2 Predicting the RMSE Without Experiments

It is no coincidence that the RMSE value of 0.5 occurs in the noisy case both (!) for Lorenz-63 and -96. Consider the stationary Riccati recursion with constant system matrices (with conventional DA notation)

$$(\mathbf{P}^a_\infty)^{-1} = (\mathbf{M}\mathbf{P}^a_\infty\mathbf{M}^\top + \mathbf{Q})^{-1} + (\mathbf{H}^\top\mathbf{R}\mathbf{H})^{-1}, \tag{31}$$

which can be used to predict the squared error, $\mathbf{P}^a_\infty$, if the dynamics, $\mathbf{M}$, are approximately constant. Pulido and van Leeuwen (2019) use $\mathbf{H} = \mathbf{I}$, and a diagonal model noise covariance, $\mathbf{Q}$. Now, if the error growth is entirely dominated by the noise, then the impact of the dynamics is negligible, i.e. $\mathbf{M} \approx \mathbf{I}$. Thus the system dimensions (for either Lorenz-63 or -96) becomes decoupled and homogeneous, and Eq. 31 reduces to a set of identical scalar systems, each of which yields the quadratic equation

$$(P^a_\infty)^2 + Q(P^a_\infty) - QR = 0. \tag{32}$$

Inserting the settings of the Pulido and van Leeuwen (2019), namely $Q = 0.3$ and $R = 0.5$, yields $P^a_\infty = 0.26$, of which the square-root is 0.5. This indicates that, indeed, the model, $\mathbf{M}$, has negligible bearing on the experiments.

References

Anterion F, Eymard R, Karcher B (1989) Use of parameter gradients for reservoir history matching. In: SPE symposium on reservoir simulation. Society of Petroleum Engineers

Bendtsen C, Stauning O (1996) Fadbad, a flexible c++ package for automatic differentiation. Technical report, Technical Report IMM-REP-1996-17, Department of Mathematical Modelling

Bocquet M, Sakov P (2014) An iterative ensemble Kalman smoother. Q J R Meteorol Soc 140(682):1521–1535

Carrassi A, Bocquet M, Bertino L, Evensen G (2018) Data assimilation in the geosciences: an overview of methods, issues, and perspectives. Wiley Interdiscip Rev Clim Change 9(5):e535

Chavent G, Dupuy M, Lemmonier P (1975) History matching by use of optimal theory. Soc Petrol Eng J 15(01):74–86

Chen P, Wu K, Chen J, O'Leary-Roseberry T, Ghattas O (2019) Projected stein variational newton: a fast and scalable Bayesian inference method in high dimensions. arXiv:1901.08659

Chen Y, Oliver DS (2012) Ensemble randomized maximum likelihood method as an iterative ensemble smoother. Math Geosci 44(1):1–26

Chen Y, Oliver DS (2013) Levenberg–Marquardt forms of the iterative ensemble smoother for efficient history matching and uncertainty quantification. Comput Geosci 17:689–703

Chwialkowski K, Strathmann H, Gretton A (2016) A kernel test of goodness of fit. In: Proceedings of the 33rd international conference on machine learning. JMRL

de Moraes RJ, Rodrigues JR, Hajibeygi H, Jansen JD (2018) Computing derivative information of sequentially coupled subsurface models. Comput Geosci 22(6):1527–1541

Detommaso G, Cui T, Spantini A, Marzouk Y, Scheichl R (2018) A stein variational Newton method. In: 32nd Conference on neural information processing systems, Montreal, Canada. NeurIPS 2018

Duchi J, Hazan E, Singer Y (2011) Adaptive subgradient methods for online learning and stochastic optimization. J Mach Learn Res 12:2121–2159

El Moshely TA, Marzouk YM (2012) Bayesian inference with optimal maps. J Comput Phys 231(23):7815–7850

Emerick A, Reynolds A (2013) Ensemble smoother with multiple data assimilation. Comput Geosci 55:3–15

Evensen G (2004) Sampling strategies and square root analysis schemes for the EnKF. Ocean Dyn 54(6):539–560

Feng Y, Wang D, Liu Q (2017) Learning to draw samples with amortized stein variational gradient descent. arXiv:1707.06626v2

Francois D, Wertz V, Verleysen M (2005) About locality of kernels in high-dimensional spaces. In: International symposium on applied stochastic models and data analysis. ASDMA

Gao G, Jiang H, Van Hagen P, Vink JC, Wells T (2017) A gauss-newton trust region solver for large scale history matching problems. In: SPE reservoir simulation conference, 20–22 Feb, Montgomery, Texas. SPE-182602

Han J, Liu Q (2018) Stein variational gradient descent without gradient. arXiv:1806.02775v1

Hunt BR, Kalnay E, Kostelich EJ, Ott E, Patil DJ, Sauer T, Szunyogh I, Yorke JA, Zimin AV (2004) Four-dimensional ensemble Kalman filtering. Tellus A 56(4):273–277

Jansen JD (2011) SimSim: a simple reservoir simulator. Departement of Geotechnology, TU, Delft

Jazwinski AH (1970) Stochastic processes and filtering theory, vol 63. Academic Press, London

Kitanidis PK (1995) Quasi-linear geostatistical theory for inversing. Water Resour Res 31(10):2411–2419

Liu Q (2017) Stein variational gradient descent as gradient flow. In: 31st Conference on neural information processing systems. NIPS

Liu Q, Wang D (2016) 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. Curran Associates, Inc. https://proceedings.neurips.cc/paper/2016/file/b3ba8f1bee1238a2f37603d90b58898d-Paper.pdf

Liu Q, Lee JD, Jordan M (2016) A kernelized stein discrepancy for goodness of fit tests. In: Proceedings of the 33rd international conference on machine learning. JMRL

Lorenz EN (1963) Deterministic nonperiodic flow. J Atmos Sci 20(2):130–141

Lorenz EN (1996) Predictability: a problem partly solved. In: Proceedings ECMWF seminar on predictability, vol 1, pp 1–18, Reading, UK

Marzouk Y, Moselhy T, Parno M, Spantini A (2017) Sampling via measure transport: an introduction. Springer International Publishing, Cham, pp 785–825

Minh HQ (2010) Some properties of gaussian reproducing kernel hilbert spaces and their implications for function approximation and learning theory. Constr Approx 32(2):307–338. https://doi.org/10.1007/s00365-009-9080-0

Oliver DS, He N, Reynolds AC (1996) Conditioning permeability fields to pressure data. In: Conference proceedings, ECMOR V - 5th European conference on the mathematics of oil recovery, Sep 1996, cp-101-00023. European Association of Geoscientists & Engineers (EAGE). https://doi.org/10.3997/2214-4609.201406884

Oliver DS, Reynolds AC, Liu N (2008) Inverse theory for petroleum reservoir characterization and history matching. Cambridge University Press, Cambridge

Ott E, Hunt BR, Szunyogh I, Zimin AV, Kostelich EJ, Corazza M, Kalnay E, Patil DJ, Yorke JA (2004) A local ensemble Kalman filter for atmospheric data assimilation. Tellus A 56(5):415–428

Pulido M, van Leeuwen PJ (2019) Sequential Monte Carlo with kernel embedded mappings: the mapping particle filter. J Comput Phys 396:400–415

Pulido M, van Leeuwen PJ, Posselt DJ (2019) Kernel embedded nonlinear observational mappings in the variational mapping particle filter. In: Rodrigues JMF, Cardoso PJS, Monteiro J, Lam R, Krzhizhanovskaya VV, Lees MH, Dongarra JJ, Sloot PM (eds) Computational science—ICCS. Springer International Publishing, Cham, pp 141–155

Reich S (2013) A guided sequential Monte Carlo method for theassimilationof data into stochastic dynamical systems. In: Johann A, Kruse HP, Rupp F, Schmitz S (eds) Recent trends in dynamical systems. Springer proceedings in mathematics & statistics, vol 35. Springer, Basel

Rodrigues JRP (2006) Calculating derivatives for automatic history matching. Comput Geosci 10(1):119–136

Sakov P, Oke PR (2008) Implications of the form of the ensemble transformation in the ensemble square root filters. Mon Weather Rev 136(3):1042–1053

Sakov P, Oliver DS, Bertino L (2012) An iterative EnKF for strongly nonlinear systems. Mon Weather Rev 140(6):1988–2004

Silverman BW (1986) Density estimation for statistics and data analysis. Chapman and Hall, Boca Raton
Skauvold J, Eidsvik J, van Leeuwen PJ, Amezcua J (2019) A revised implicit equal-weights particle filter. Q J R Meteorol Soc. https://doi.org/10.1002/qj.3506
Stein C (1972) A bound for the error in the normal approximation to the distribution of a sum of dependent random variables. In: Proceedings of the sixth Berkeley symposium on mathematical statistics and probability, vol 2: probability theory. The Regents of the University of California
Stordal A (2015) Iterative Bayesian inversion with gaussian mixtures: finite sample implementation and large sample asymptotics. Comput Geosci 19(1):1–15. https://doi.org/10.1007/s10596-014-9444-9
Stordal A, Elsheikh A (2015) Iterative ensemble smoothers in the annealed importance sampling framework. Adv Water Resour 86:231–239
Stordal AS, Karlsen HA (2017) Large sample properties of the adaptive gaussian mixture filter. Mon Weather Rev 145(7):2533–2553. https://doi.org/10.1175/MWR-D-15-0372.1
Vlasov AA (1961) Many-particle theory and its application to plasma. Gordon & Breach Science Publishers, Inc
Wikle CK, Berliner LM (2007) A Bayesian tutorial for data assimilation. Physica D 230(1–2):1–16
Zhang J, Zhang R, Chen C (2018) Stochastic particle-optimization sampling and the non-asymptotic convergence theory. arXiv:1809.01293v2
Zhou D (2008) Derivative reproducing properties for kernel methods in learning theory. Journal od computational and Applied Mathematics 220:456–463