Title

Variational particle smoothers and their localization

Journal

Quarterly Journal of the Royal Meteorological Society, 144(712)

ISSN

0035-9009

Authors

Morzfeld, M
Hodyss, D
Poterjoy, J

Publication Date

2018-04-01

DOI

10.1002/qj.3256

Peer reviewed

Received: 19 July 2017 | Revised: 18 December 2017 | Accepted: 22 December 2017

DOI: 10.1002/qj.3256

RESEARCH ARTICLE

Quarterly Journal of the Royal Meteorological Society RMetS

Variational particle smoothers and their localization

M. Morzfeld[1] | D. Hodyss[2] | J. Poterjoy[3]

[1]Depertment of Mathematics, University of Arizona, Tucson
[2]Naval Research Laboratory, Monterey, California
[3]NOAA Atlantic Oceanographic and Meteorological Laboratory, Miami, Florida

Correspondence
M. Morzfeld, Department of Mathematics, University of Arizona, 617 N. Santa Rita Avenue, PO Box 210089, Tucson, AZ 85721.
Email: mmo@math.arizona.edu

Funding information
Office of Naval Research grants N00173-17-2-C003 and PE-0601153N; National Science Foundation grant DMS-1619630; the Alfred P. Sloan Foundation, and the National Research Council Research Associateship Program,

Given the success of 4D-variational methods (4D-Var) in numerical weather prediction, and recent efforts to merge ensemble Kalman filters with 4D-Var, we revisit how one can use importance sampling and particle filtering ideas within a 4D-Var framework. This leads us to variational particle smoothers (varPS) and we study how weight-localization can prevent the collapse of varPS in high-dimensional problems. We also discuss the relevance of (localized) weights in near-Gaussian problems. We test our ideas on the Lorenz'96 model of dimensions $n = 40$, $n = 400$, and $n = 2,000$. In our numerical experiments the localized varPS does not collapse and yields results comparable to ensemble formulations of 4D-Var, while tuned EnKFs and the local particle filter lead to larger estimation errors. Additional numerical experiments suggest that using localized weights may not yield significant advantages over unweighted or linearized solutions in near-Gaussian problems.

KEYWORDS
data assimilation, particle filters, variational methods

1 | INTRODUCTION

In numerical weather prediction (NWP), and in many other applications in science and engineering, one wants to update the state of a numerical model based on noisy observations of the state, e.g. Kalnay (2003), van Leeuwen (2009), Bocquet *et al.* (2010) and Fournier *et al.* (2010). Accounting for errors in the numerical model and in the observations naturally leads to a Bayesian formulation of this problem in terms of prior probabilities, likelihoods and posterior probabilities. An important feature of such "data assimilation" problems in NWP is their size. A typical global atmospheric model has more than 600 million state variables and several million atmospheric observations are assimilated into such a model during a 6 hr cycle. Numerical data assimilation methods have been developed and refined over the past decades and can be divided into three main groups: variational methods (e.g. Talagrand and Courtier, 1987; Bennet *et al.*, 1993), Kalman filters (e.g. Tippet *et al.*, 2003; Evensen, 2006), and particle filters (PFs; e.g. Gordon *et al.*, 1993; Doucet *et al.*, 2001; Arulampalam *et al.*, 2002; van Leeuwen, 2009). Given the immense size of the problem in NWP, it is imperative that useful numerical methods scale favourably with the size of the problem.

Ensemble Kalman filters (EnKFs) have been implemented for full-scale global atmospheric models. Their success with extremely small ensemble size (50–100) is made possible by covariance localization, as described in Gaspari and Cohn (1999), Hamill *et al.* (2001), Houtekamer and Mitchell (2001), Houtekamer *et al.* (2005), Anderson (2007; 2012). During localization one makes use of the fact that observations have only a local effect: observations of the weather collected in Australia do not have an immediate effect on estimates of the weather in North America. To enforce the locality of observations, ensemble estimates of prior errors are de-correlated by setting the corresponding elements in the error covariance matrix to zero. This leads to sparse and banded forecast and posterior covariances, which allows for effective EnKF implementations with small ensemble sizes, e.g. Morzfeld *et al.* (2017) and Bickel and Levina (2008). Variational methods also have been applied to full-scale global atmospheric models. Their implementations exploit the same sparse/banded problem structure during optimization, by using nonlinear least-squares algorithms and adjoint equations for gradient computations. Many recent works merge EnKF and 4D-Var methods to create hybrid schemes that can combine strengths of Kalman filter and variational

approaches, e.g. Lorenc (2003), Buehner (2005), Liu *et al.* (2008), Sakov *et al.* (2012), Bonavita *et al.* (2012), Bocquet and Sakov (2013; 2014); Lorenc *et al.* (2015), Poterjoy and Zhang (2015), Bocquet (2016) and Hodyss *et al.* (2016).

PFs are rarely used in NWP. The reason is that many PFs require an ensemble size that scales exponentially with dimension, e.g. Bickel *et al.* (2008), Bengtsson *et al.* (2008), Chorin and Morzfeld (2013), Snyder *et al.* (2008; 2015), Snyder (2011) and Morzfeld *et al.* (2017). This effect is often called the "collapse of PFs". The collapse for a class of PFs (see below), is unavoidable for generic problems, i.e. problems without any additional "structure". Thus, while PFs may collapse on any given, generic, high-dimensional problem, they may work fine on some problems, characterized by specific problem structure, such as bandedness or sparsity of forecast covariances. The main idea of "localizing" PFs is to exploit banded problem structure to avoid PF collapse (section 2.4 below). Several methods to localize PFs have been invented and have been shown to "work well", mostly on relatively simple models (Lei and Bickel, 2011; Reich, 2013; Penny and Miyoshi, 2015; Poterjoy, 2015; Tödter and Ahrens, 2015; Lee and Majda, 2016; Poterjoy *et al.*, 2017), but Poterjoy and Anderson (2016) and Robert *et al.* (2017) present results in a realistic NWP context.

On the other hand, it is important to realize that localization should not be viewed as a "cure for all problems" with PFs. After localization, one can think of a data assimilation problem as a collection of loosely coupled sub-problems, and localized PFs solve each sub-problem individually. It is thus not the number of sub-problems, or the overall dimension, or the overall number of (independent) observations that define the performance bounds for localized PFs, but the characteristics of each sub-problem. It is yet to be determined whether localized PFs can indeed solve some of the high-dimensional data assimilation problems that arise in NWP, where the number of observations per sub-problem can be huge, and possibly leads to collapse of even localized PFs. Moreover, there are PFs, e.g. the equivalent weights PF (van Leeuwen, 2010; Ades and van Leeuwen, 2013), which avoid filter collapse by judicious choice of proposal distributions, and these PFs, and their (non-)collapse are not described by the "typical" theory for the collapse of PF as described, e.g. in Chorin and Morzfeld (2013), Snyder *et al.* (2008; 2015), Bickel *et al.* (2008), Bengtsson *et al.* (2008), Snyder (2011) and Morzfeld *et al.* (2017).

We do not attempt to address all of the above issues in this article, and focus our attention on how importance sampling and PF ideas can be used within a 4D-Var framework and, more specifically, what role weight-localization plays in this context. When studying importance sampling for data assimilation in NWP, it becomes apparent that it matters which posterior distribution one considers for sampling – the distribution of the state at observation time, $p(\mathbf{x}_k|\mathbf{y}_k)$, as is typical in particle filtering, or the distribution of the initial condition of a deterministic model at an earlier time, $p(\mathbf{x}_{k-1}|\mathbf{y}_k)$, as is typical in variational methods; see also Bocquet and Sakov (2014) and Weir *et al.* (2013). These ideas leads us to revisit implicit sampling and variational particle smoothers, which were also considered by Atkins *et al.* (2013), how these methods can be localized, and what role the localized weights play in near-Gaussian problems.

2 | BACKGROUND AND NOTATION

2.1 | Data assimilation problem formulation

We consider data assimilation problems defined by

$$\mathbf{x}_k = \mathbf{f}_k(\mathbf{x}_{k-1}), \tag{1}$$

$$\mathbf{y}_k = \mathbf{h}_k(\mathbf{x}_k) + \varepsilon_k, \tag{2}$$

where $k = 1, 2, \ldots$ is discrete time, the state at time k, $\mathbf{x}_k$, is an n_x-dimensional vector, $\mathbf{y}_k$ is a noisy observation of the state, and ε_k are independent identically distributed (iid) Gaussian random variables with means $E[\varepsilon_k] = \mathbf{0}$, and covariance matrices $\mathbf{R_k} = E\left[\varepsilon_k \varepsilon_k^{\mathrm{T}}\right]$. Here the numerical model $\mathbf{f}_k$ is a known n_x-dimensional vector function, and the observation function $\mathbf{h}_k$ is an n_y-dimensional vector function. Note that we exclude model error and stochastic models from our study. We touch, briefly, on stochastic models and "optimal" particle filters in the Appendix, but defer a more thorough study to future work.

The goal in data assimilation is to estimate the state at time k, given the data up to time k. This estimate can be based on the posterior distribution at time k, given observations up to time k

$$p(\mathbf{x}_k|\mathbf{y}_{1:k}) \propto p(\mathbf{x}_k|\mathbf{y}_{1:k-1})p(\mathbf{y}_k|\mathbf{x}_k), \tag{3}$$

which is the foundation for (ensemble) Kalman filters, as explained by Evensen (2006) and Tippet *et al.* (2003), and particle filters, e.g. van Leeuwen (2009) and Arulampalam *et al.* (2002). Here and below, $\mathbf{y}_{1:m}$ denotes the set of vectors $\{\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_m\}$.

Alternatively, one can consider the state at time $k-1$, given the data up to time k, described by the posterior distribution

$$p(\mathbf{x}_{k-1}|\mathbf{y}_{1:k}) \propto p(\mathbf{y}_k|\mathbf{x}_{k-1})p(\mathbf{x}_{k-1}|\mathbf{y}_{1:k-1}). \tag{4}$$

The above posterior distribution is fundamental to four-dimensional variational (4D-Var) methods, e.g. Talagrand and Courtier (1987) and below. An estimate of $\mathbf{x}_k$ can be based on $p(\mathbf{x}_{k-1}|\mathbf{y}_{1:k})$ by using the model to evolve this distribution to time k. Of course, one can also go back further in time and consider, e.g. the distribution $p(\mathbf{x}_{k-L}|\mathbf{y}_{1:k})$. Such extensions, sometimes called "lag-L smoothers", are conceptually simple but the details are intricate and we choose not to discuss them here.

2.2 | 4D-Var

In 4D-Var methods, the distribution $p(\mathbf{x}_{k-1}|\mathbf{y}_{1:k-1})$ in Equation 4 is approximated by a Gaussian with mean μ and

covariance $\mathbf{B}$, called the background and background covariance respectively. Replacing $p(\mathbf{x}_{k-1}|\mathbf{y}_{1:k-1})$ by the Gaussian $\widetilde{p}(\mathbf{x}_{k-1}|\mathbf{y}_{1:k-1}) = \mathcal{N}(\mu, \mathbf{B})$ in Equation 4, generates an approximate posterior distribution,

$$\widehat{p}(\mathbf{x}_{k-1}|\mathbf{y}_{1:k}) \propto \widetilde{p}(\mathbf{x}_{k-1}|\mathbf{y}_{1:k-1})p(\mathbf{y}_k|\mathbf{x}_{k-1}), \tag{5}$$

which can be written as

$$\widehat{p}(\mathbf{x}_{k-1}|\mathbf{y}_{1:k}) \propto \exp\left(-\mathcal{J}(\mathbf{x}_{k-1})\right), \tag{6}$$

where

$$\begin{aligned}\mathcal{J}(\mathbf{x}_{k-1}) =& \frac{1}{2}\left(\mathbf{x}_{k-1}-\mu\right)^{\mathrm{T}}\mathbf{B}^{-1}\left(\mathbf{x}_{k-1}-\mu\right) \\ &+\frac{1}{2}\left\{\mathbf{h}\left[f(\mathbf{x}_{k-1})\right]-\mathbf{y}_k)\right\}^{\mathrm{T}}\mathbf{R}^{-1}\left\{\mathbf{h}[f(\mathbf{x}_{k-1})\right]-\mathbf{y}_k\}.\end{aligned} \tag{7}$$

The cost function $\mathcal{J}$ is minimized by 4D-Var methods, e.g. Talagrand and Courtier (1987). The minimizer of the cost function, $\mathbf{x}^*_{k-1}$ is an estimate of $\mathbf{x}_{k-1}$ given the observations $\mathbf{y}_{1:k}$ up to time k. An estimate of $\mathbf{x}_k$ can be obtained by evolving $\mathbf{x}^*_{k-1}$ forward to time k using the model (1). It is important to realize that the approximate posterior distribution $\widehat{p}$ is not a Gaussian. Non-Gaussian aspects are introduced by the nonlinear model or observation functions.

In many "traditional" 4D-Var schemes, the background covariance matrix $\mathbf{B}$ is "static", i.e. it does not change from one cycle to the next. Updating background matrices in view of the observations is the main idea of ensemble formulations of 4D-Var. We refer to Lorenc *et al.* (2015) and Hodyss *et al.* (2016) for the definitions of the various flavours of ensemble formulations of 4D-Var and recall the iterative ensemble Kalman filter (IEnKF; Sakov *et al.*, 2012) and the iterative ensemble Kalman smoother (IEnKS; Bocquet and Sakov, 2013; 2014; Bocquet, 2016) as specific, well-studied and theoretically well-justified examples of ensemble formulations of 4D-Var.

There are three main approaches to blending flow-dependent ensemble background covariances with 4D-Var. In ensemble-4DVar (E4D-Var; e.g. Lorenc, 2003; Buehner, 2005; Poterjoy and Zhang, 2015), a flow dependent background is obtained by coupling an EnKF system to a variational system. In 4D-ensemble var (4DEnVar), a 4D ensemble is used to to replace the tangent linear and adjoint model operators in 4D-Var, e.g. Liu *et al.* (2008) and Poterjoy and Zhang. An ensemble of 4D-Vars method (EDA; e.g. Bonavita *et al.*, 2012) generates an ensemble by solving a variational problem N_e times with perturbed observations and perturbed states. The IEnKF and IEnKS (Sakov *et al.*, 2012; Bocquet and Sakov, 2013; 2014; Bocquet, 2016) are also ensemble-based variational methods, derived and inspired by Bayes' rule, which, as opposed to E4D-Var, are self-sufficient and do not require any additional data assimilation system. The IEnKF and the IEnKS do not necessarily require tangent linear and adjoint models, but tangent linear and adjoint models can be used for some implementations of IEnKF/IEnKS. Below, we will test some of these techniques on a Lorenz'96 (L96) model described in Lorenz (1996), and compare EDA and E4D-Var results to results obtained by a variational particle smoother (varPS). We also discuss connections of varPS to ensemble formulations of 4D-Var, in particular with the IEnKF and the IEnKS (Sakov *et al.*, 2012; Bocquet and Sakov, 2013; 2014; Bocquet, 2016).

2.3 | Particle filters

A PF draws weighted samples from the posterior distribution

$$p(\mathbf{x}_k|\mathbf{y}_{1:k}) \propto p(\mathbf{x}_k|\mathbf{y}_{1:k-1})p(\mathbf{y}_k|\mathbf{x}_k), \tag{8}$$

by drawing samples from a proposal distribution $q(\mathbf{x}_k)$ (e.g. Doucet *et al.*, 2001; Arulampalam *et al.*, 2002; van Leeuwen, 2009). Attached to each sample is a weight

$$w_k \propto \frac{p(\mathbf{x}_k|\mathbf{y}_{1:k})}{q(\mathbf{x}_k)}. \tag{9}$$

The weighted ensemble $\left\{\mathbf{x}^j_k, w^j_k\right\}$, $j = 1, \ldots, N_e$, approximates the posterior distribution $p(\mathbf{x}_k|\mathbf{y}_{1:k})$ in the sense that weighted averages over the ensemble converge to expected values with respect to the posterior distribution as $N_e \to \infty$. We will use the "standard particle filter" with proposal distribution and weights given by

$$q_k(\mathbf{x}_k) = p(\mathbf{x}_k|\mathbf{y}_{1:k-1}), \qquad w_k \propto p(\mathbf{y}_k|\mathbf{x}_k).$$

This means that the standard PF amounts to generating a forecast ensemble by running the numerical model (just as EnKF), and then attaching weights proportional to the likelihood to each ensemble member.

The "quality" of a PF can be assessed by computing the effective sample size (e.g. Doucet *et al.*, 2001),

$$N_{\mathrm{eff}} = \frac{N_e}{G}, \qquad G = 1 + \frac{var(w_k)}{E(w_k)^2} = \frac{E(w_k^2)}{E(w_k)^2}. \tag{10}$$

The effective sample size is a heuristic quantity that describes the sample size of an unweighted ensemble equivalent to the weighted ensemble. If the variance of the weights is large, then G is large and the effective sample size is small. In an extreme case, N_{eff} may be one, and the particle filter has "collapsed" and produces statistical estimates with an accuracy equivalent to having only one sample. For a particle filter to be "useful", G cannot be too large.

2.4 | Localization of particle filters

It has been shown that the required ensemble size of a PF can grow exponentially with dimension, (e.g. Bengtsson *et al.*, 2008; Snyder *et al.*, 2008; 2015; Snyder, 2011; Chorin and Morzfeld, 2013; Rebeschini and van Handel, 2015; Morzfeld *et al.*, 2017). This is certainly true for generic problems and for a certain class of PFs, however the collapse of PFs can be prevented by making use of the locality of observations, i.e. the fact that an observation has a local, not a global effect.

This is the main idea behind localization of PFs, first discussed by Bengtsson *et al.* (2003) and van Leeuwen (2003), which typically consists of the following two steps (e.g. Lei and Bickel, 2011; Reich, 2013; Penny and Miyoshi, 2015; Poterjoy, 2015; Tödter and Ahrens, 2015; Lee and Majda, 2016; Poterjoy and Anderson, 2016; Poterjoy *et al.*, 2017 give specific localization strategies):

1. find a way to compute weights in Equation 9 locally;
2. make use of these local weights without upsetting the complex multivariate relationships between variables (model "balance").

A diagonal problem is characterized by a diagonal model,

$$[\mathbf{f}_k(\mathbf{x}_{k-1})]_i = [\mathbf{f}_k]_i([\mathbf{x}_{k-1}]_i),$$

a diagonal observation function,

$$[\mathbf{h}_k(\mathbf{x}_k)]_i = [\mathbf{h}_k]_i([\mathbf{x}_k]_i),$$

and a diagonal observation-error covariance $\mathbf{R}$. Here and below, $[\mathbf{a}]_j$ denotes the jth component of a vector $\mathbf{a}$. Block-diagonal problems, which consist of "blocks" of independent variables, can be defined similarly. In (block-) diagonal problems one can achieve step (a) by computing the weights for each coordinate and step (b) by resampling separately in each coordinate because there are no multivariate relationships between variables. Using the diagonalizing approach in a coupled problem amounts to neglecting correlation, which avoids PF collapse, but introduces additional errors if correlations among the variables are indeed important. There is thus a trade-off between preventing PF collapse by localization and the additional errors introduced by localization.

Localization strategies for data assimilation problems that are not diagonal typically introduce tuning parameters to define the localization and then adjust these parameters such that a mean square error (MSE) is on the order of a predicted average variance (spread; technical definitions of MSE and spread are given below). As a specific example, consider the localization schemes created by Poterjoy (2015) and Poterjoy and Anderson (2016). The PF weights vary with location (or state variable), and weights at a certain location depend only on observations in the neighborhood of a given location. A posterior ensemble is generated by merging prior particles and particles that are weighted with the spatially varying weights using a localization function. Parameters that define the localization function are then tuned to yield small MSE and an appropriate spread. Bias introduced by this procedure is assumed to be small. It is not known what the localization scheme does to the asymptotic behavior of the PF as the ensemble size goes to infinity. However, such issues are not specific to PFs and their localization. Similar statements are also true for localization of EnKFs – there are several localization strategies in use, it is not clear which is best, and, in general, different localization schemes lead to different asymptotic behaviour of EnKF, (e.g. Mitchell and Houtekamer, 2002; Lorenc, 2003; Kepert, 2009; Greybush *et al.*, 2011; Le Gland *et al.*, 2011).

3 | VARIATIONAL PARTICLE SMOOTHERS

Motivated by the success of 4D-Var methods in NWP, and the recent advances in making PFs more applicable via localization, we revisit how localized PFs (more generally, importance sampling) can be used within a 4D-Var framework, and the specific role and advantages of weight localization in this context. As indicated above, 4D-Var methods work with the posterior distribution $p(\mathbf{x}_{k-1}|\mathbf{y}_{1:k})$, particle filters usually work with $p(\mathbf{x}_k|\mathbf{y}_{1:k})$. "Smoothers", on the other hand, also work with $p(\mathbf{x}_{k-1}|\mathbf{y}_{1:k})$, i.e. there is a natural connection between smoothers and 4D-Var (also Sakov *et al.*, 2012; Atkins *et al.*, 2013; Bocquet and Sakov, 2013; 2014; Bocquet, 2016). We thus consider "particle smoothers", their localization and their connections with 4D-Var and ensemble formulations of 4D-Var. Note that we adopt typical Monte Carlo literature terminology, as in Doucet *et al.* (2001), and we define a particle smoother to be a sampling method for $p(\mathbf{x}_{k-1}|\mathbf{y}_{1:k})$, in the same vein as a PF is a sampling method for $p(\mathbf{x}_k|\mathbf{y}_{1:k})$. One can use results, obtained by a smoother at time $k-1$, to compute state estimates at observation time, k, by using the numerical model to evolve the smoother-ensemble to observation time, as is routinely done in 4D-Var (section 2.2; also Bocquet and Sakov, 2013; 2014).

Particle smoothers thus work as follows. We pick a proposal distribution $q(\mathbf{x}_{k-1};\mathbf{y}_{1:k})$, draw samples from it, and then attach to each sample a weight

$$w \propto \frac{p(\mathbf{y}_k|\mathbf{x}_{k-1})p(\mathbf{x}_{k-1}|\mathbf{y}_{1:k-1})}{q(\mathbf{x}_{k-1};\mathbf{y}_{1:k})}. \tag{11}$$

The weighted ensemble $\left\{\mathbf{x}_{k-1}^j, w^j\right\}, j = 1,\dots,N_e$, approximates the posterior distribution $p(\mathbf{x}_{k-1}|\mathbf{y}_{1:k})$ in the sense that weighted averages over the ensemble converge to expected values with respect to the posterior distribution $p(\mathbf{x}_{k-1}|\mathbf{y}_{1:k})$ as $N_e \to \infty$. In practice one runs into the problem that these weights cannot be evaluated, because $p(\mathbf{x}_{k-1}|\mathbf{y}_{1:k-1})$ in the numerator of Equation 11 is generally not known. The exception are linear/Gaussian problems, for which $p(\mathbf{x}_{k-1}|\mathbf{y}_{1:k-1})$ can be computed explicitly. The same difficulty arises when one considers particle filters, as is discussed section 3.5.

3.1 | Sampling the past by a variational particle smoother

The above deficiency can be overcome by using a Gaussian approximation for $p(\mathbf{x}_{k-1}|\mathbf{y}_{1:k-1})$, as is common in 4D-Var (section 2.2) and the IEnKF or the IEnKS (Sakov *et al.*, 2012; Bocquet and Sakov, 2013; 2014; Bocquet, 2016). Thus, we replace $p(\mathbf{x}_{k-1}|\mathbf{y}_{1:k-1})$ in Equation 4 by the Gaussian $\widetilde{p}(\mathbf{x}_{k-1}|\mathbf{y}_{1:k-1}) = \mathcal{N}(\mu, \mathbf{B})$, to construct the approximate posterior distribution $\widehat{p}$ in Equation 5. One *can* evaluate the

approximate posterior $\widehat{p}$, and thus construct importance sampling methods for $\widehat{p}$.

A natural choice for a proposal distribution is the Gaussian

$$q(\mathbf{x}_{k-1};\mathbf{y}_{1:k}) = \mathcal{N}(\mathbf{x}^*,\mathbf{J}^{-1}) \propto \exp\left[-\frac{1}{2}(\mathbf{x}_{k-1}-\mathbf{x}^*)^{\mathrm{T}}\mathbf{J}(\mathbf{x}_{k-1}-\mathbf{x}^*)\right], \tag{12}$$

where $\mathbf{x}^*$ is the minimizer of $\mathcal{J}$, and $\mathbf{J}$ is the (approximate) Hessian of $\mathcal{J}$, evaluated at the minimizer $\mathbf{x}^*$. For example, one can use the Gauss–Newton approximation of the Hessian, which requires first derivatives of $\mathcal{J}$, computed by tangent linear and adjoint models (e.g. Talagrand and Courtier, 1987). Strategies for implementing this sampling method in high-dimensional problems using existing software infrastructure are discussed in Auligné *et al.* (2016), and the ensemble of the IEnKF and the IEnKS is constructed similarly (Sakov *et al.*, 2012; Bocquet and Sakov, 2013; 2014; Bocquet, 2016; and our discussion below).

With this proposed distribution, the weights become

$$w \propto \frac{p(\mathbf{y}_k|\mathbf{x}_{k-1})\widetilde{p}(\mathbf{x}_{k-1}|\mathbf{y}_{1:k-1})}{q(\mathbf{x}_{k-1};\mathbf{y}_{1:k})} \propto \frac{\exp\left[-\mathcal{J}(\mathbf{x}_{k-1})\right]}{\exp\left[-\frac{1}{2}(\mathbf{x}_{k-1}-\mathbf{x}^*)^{\mathrm{T}}\mathbf{J}(\mathbf{x}_{k-1}-\mathbf{x}^*)\right]}. \tag{13}$$

To avoid under- or overflow, one may want to consider computing the negative logarithm of the weights, $\widehat{w} = -\log(w)$. Once all N_e negative-log weights $\widehat{w}$ are computed, one can subtract their minimum value from all of them, then take the exponential, then normalize so that the weights sum to one.

The weighted ensemble $\left\{\mathbf{x}^j_{k-1}, w^j\right\}, j = 1,\dots,N_e$ approximates the posterior distribution $\widehat{p}$ in Equation 5, in the sense that weighted averages converge to expected values as $N_e \to \infty$. Generating samples in this way is an implementation of "implicit sampling", (e.g. Chorin *et al.*, 2010), whose connections with variational data assimilation have been discussed in Atkins *et al.* (2013). Implicit sampling in the context of smoothing has first been discussed by Weir *et al.* (2013), where it was also shown that considering a smoothing density can prevent collapse. An application of this method to geomagnetic data assimilation can be found in Morzfeld *et al.* (2017). Its use in inverse problems is discussed in Morzfeld *et al.* (2015). A recent application of this sampling method can also be found in Liu *et al.* (2017). We discuss connections of this sampling method, applied to the posterior distribution $p(\mathbf{x}_k|\mathbf{y})$ with ensemble formulations of 4D-Var, specifically with the IEnKF and the IEnKS, in section 3.3. For the remainder of this article we refer to this method as the "variational particle smoother" (varPS).

Note that the varPS proposal distribution in Equation 12 is Gaussian, but the posterior distribution $\widehat{p}$ in Equation 5 is not necessarily a Gaussian. The unweighted varPS ensemble (all weights equal to $w = 1/N_e$) is distributed according to the varPS proposal distribution, but the varPS weights in Equation 13 account for the non-Gaussian aspects of $\widehat{p}$ and "transform" samples from the Gaussian proposal distribution, into weighted samples of the posterior distribution. In a linear/Gaussian problem, the proposal distribution is *exactly* equal to the posterior distribution, so that all weights are equal. Since all weights are equal, varPS does not collapse, even if the dimension is high. These issues are carefully discussed in Weir *et al.* (2013). However, in practice, varPS can be expected to collapse due to nonlinearity and/or approximations of the proposal covariances. The collapse of varPS and how it can be prevented by weight-localization is discussed in detail in section 4.

Finally, note that the Gaussian approximation used to define the approximate posterior distribution $\widehat{p}$ in Equation 5 causes distributional errors in the sense that the varPS produces weighted samples of the approximate posterior distribution $\widehat{p}$, rather than the posterior distribution p in Equation 4. Such errors vanish when the problem is linear and Gaussian. The success of variational methods in practice, which rely on the same Gaussian approximation as varPS, indicates that distributional errors may be small in "near-Gaussian problems" of practical importance. The success of the IEnKF and the IEnKS, which also rely on a Gaussian approximation of $\widehat{p}$, is another indication that this approximation is indeed appropriate (Sakov *et al.*, 2012; Bocquet and Sakov, 2013; 2014; Bocquet, 2016)

3.2 | Cycling varPS

The varPS performs the following three steps:

1. solve the 4D-Var problem;
2. generate samples by perturbing the posterior mode;
3. compute weights by Equation 13.

To be able to start and to cycle a varPS, one needs to compute and update the background state μ and background covariance $\mathbf{B}$. To start the algorithm, one can use a "climatological" covariance and mean, or some other initial guess. With this choice, one solves the 4D-Var problem of minimizing $\mathcal{J}$ in Equation 7, to find the most likely state $\mathbf{x}^*$ and approximate Hessian $\mathbf{J}$, which define the proposal distribution (12). We draw N_e samples, $\left\{\mathbf{x}^j_{k-1}\right\}$, $j = 1,\dots,N_e$, from the Gaussian proposal and compute their weights by Equation 13, to obtain a weighted ensemble of the approximate posterior (5). One can then resample, and replace particles with low weights by particles with larger weights to obtain an unweighted ensemble (e.g. Doucet *et al.*, 2001; Arulampalam *et al.*, 2002 for resampling algorithms). Each ensemble member is then evolved to time k using the model (1), which leads to an ensemble at observation time $\left\{\mathbf{x}^j_k\right\}, j = 1,\dots,N_e$. The background state and background covariance at the next assimilation cycle are obtained by computing the ensemble mean and ensemble covariance. Localization and inflation of this updated background covariance can be tuned just like localization and

inflation in EnKF. It may also be necessary to localize the weights, as discussed in section 4. Pseudo-code for the varPS is provided in algorithm 1.

Algorithm 1 Variational particle smoother (varPS)

Solve the variational problem: minimize $\mathcal{J}(\mathbf{x}_{k-1})$
Result: minimizer $\mathbf{x}^*$ and Hessian $\mathbf{J}$
Localize/inflate proposal covariance $\mathbf{J}^{-1}$
Sampling: draw an ensemble of N_e particles from the proposal: $\mathbf{x}_{k-1}^j \sim \mathcal{N}(\mathbf{x}^*, \mathbf{J}^{-1})$
Compute and store the corresponding states at time k (running the model N_e times)
for $j = 1, \dots, N_e$ **do**
  Compute weight: $w_j \propto \frac{\exp\left[-\mathcal{J}(\mathbf{x}_{k-1})\right]}{\exp\left[-\frac{1}{2}(\mathbf{x}_{k-1}-\mathbf{x}^*)^{\mathrm{T}}\mathbf{J}(\mathbf{x}_{k-1}-\mathbf{x}^*)\right]}$
end for
Normalize weights: $w_j \leftarrow w_j / \sum_{l=1}^{N_e} w_l$
Resample states at time k using these weights
Update background state μ and background covariance $\mathbf{B}$ from resampled states
Localize/inflate background covariance $\mathbf{B}$
Set $k \leftarrow k+1$ and repeat

3.3 | Connections of varPS with ensemble formulations of 4D-Var

One can interpret the varPS as a weighted sampling method for computing a flow-dependent background-covariance matrix in 4D-Var. Compared to EDA, the varPS is computationally less demanding because it requires only one optimization. Moreover, because of its weights, the varPS can account for additional aspects of nonlinearity and non-Gaussianity when generating the analysis ensemble. Such weights can, in principle, also be generated for EDA by borrowing ideas from the "Bayesian inverse problem" literature, where this technique is known as "Randomize-then-optimize" (RTO; Bardsley *et al.*, 2014). A related approach is "randomized maximum likelihood", used in oil-reservoir modelling (Oliver *et al.*, 2008). However, such weights require localization or else varPS or weighted EDA/RTO collapses when the dimension of the problem is large. Weight-localization for varPS is discussed in detail in section 4.

Compared to E4D-Var the varPS does not require an EnKF system. In our numerical experiments with L96 models (Lorenz, 1996) in section 5 we compare E4D-Var and EDA to the varPS and find that they give comparable results. In our implementation, the varPS differs from 4DEnVar, because we use tangent linear and adjoint model operators during the solution of the variational problem. In the future, one can experiment with using ideas from 4DEnVar for practical implementation of the varPS on large-scale NWP problems.

The varPS also has connections with the IEnKF and the IEnKS (Sakov *et al.*, 2012; Bocquet and Sakov, 2013; 2014; Bocquet, 2016). The IEnKF/IEnKS ensemble is generated similarly to how the varPS generates its unweighted ensemble. In fact, if one views the IEnKF/IEnKS as the "concept" of using a Gaussian approximation of $p(\mathbf{x}_{k-1}|\mathbf{y}_{1:k})$ to generate an ensemble (setting aside details of numerical implementation or additional approximations), then the ensemble of the IEnKF/IEnKS is identical to the proposal ensemble of varPS. Thus, the most significant difference between varPS and the IEnKF/IEnKS are weights. These weights, and their localization are the focus of this article. Below, we also study varPS "with equal weights", i.e. we study what happens when one sets all weights equal to $1/N_e$, and this varPS with equal weights is in fact an IEnKF/IEnKS (section 4). In the context of ensemble formulations of 4D-Var and IEnKF/IEnKS, it is important to realize that IEnKF/IEnKS make use of the same Gaussian approximation of $p(\mathbf{x}_{k-1}|\mathbf{y}_{1:k-1})$ as 4D-Var, varPS and (other) ensemble formulations of 4D-Var. However, IEnKF/IEnKS do not require a "separate" data assimilation system for updating background covariances, since such an update is built into the algorithms.

3.4 | Benchmarking varPS against EnKF and localized PF

We benchmark the varPS by numerical experiments with the linear problem

$$\mathbf{x}_k = \mathbf{x}_{k-1}, \tag{14}$$

$$\mathbf{y}_k = \mathbf{x}_k + \varepsilon_k, \tag{15}$$

where ε_k are iid Gaussians with diagonal covariance matrices $\mathbf{R_k} = \mathbf{I}$. This problem has been used in the context of the collapse of PFs before by Chorin and Morzfeld (2013), Snyder *et al.* (2008; 2015), Bickel *et al.* (2008), Bengtsson *et al.* (2008), Snyder (2011) and Morzfeld *et al.* (2017). We pick the dimension to be $n_x = n_y = 100$, which is large enough to make unlocalized PFs collapse. In fact, the results obtained at this dimension are qualitatively the same as those of higher-dimensional problems.

We apply several data assimilation methods (see below) and assess their performance by the mean square error (MSE) and the spread. The MSE is defined by

$$\mathrm{MSE} = \frac{1}{n}\sum_{j=1}^{n}([\mathbf{x}_k^t]_j - [\bar{\mathbf{x}}_k]_j)^2, \tag{16}$$

where $[\mathbf{a}]_j$ denotes the jth element of a vector $\mathbf{a}$, where $\mathbf{x}_k$ is the "true" state at time k, and where $\bar{\mathbf{x}}_k$ is the estimate of the state at time k of a data assimilation algorithm. For the localized PFs and EnKF, we use the weighted ensemble mean and ensemble mean, respectively, as the estimate. The varPS yields an "analysis" ensemble at time $k-1$, which we propagate to time k using the model (here the identity matrix). The varPS estimate is the average over the ensemble at time k, which is obtained by applying the model (14) to the ensemble generated at time $k-1$. The spread is defined as the

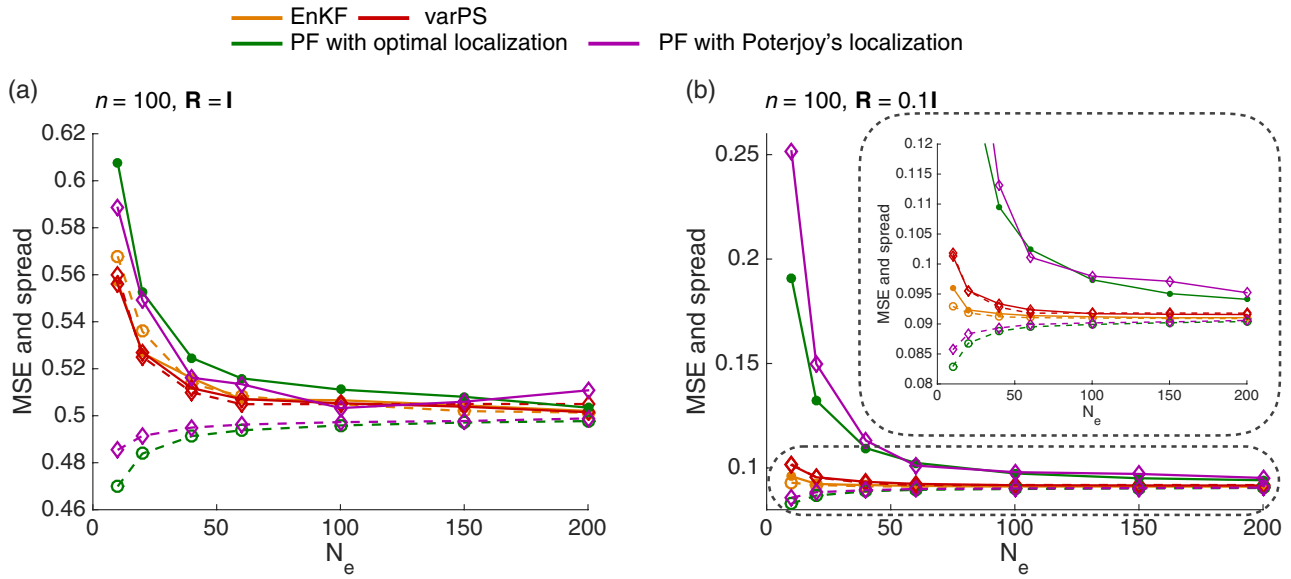

FIGURE 1 MSE (solid lines) and spread (dashed lines) as a function of the ensemble size for EnKF, PFs, and varPS with (a) larger noise in the observation, **R**=**I**, and (b) smaller noise in the observation, **R**=0.1 **I** [Colour figure can be viewed at wileyonlinelibrary.com]

normalized trace of the posterior covariance matrix at time k:

$$\text{spread} = \frac{1}{n}\text{trace}(\mathbf{P}_a). \tag{17}$$

For the localized PFs, $\mathbf{P}_a$ is the covariance of the weighted ensemble, and for EnKF, $\mathbf{P}_a$ is the covariance of the analysis ensemble. For varPS $\mathbf{P}_a$ is computed from the ensemble at time k (not at time $k-1$). This ensemble is generated in using the observation at time k. Note that we use the MSE (Equation 16) and not its square root (root mean square error) to assess performances of the various methods. For that reason, we use the spread as defined in Equation (17), and not its square root (which is also common).

We apply a localized and inflated stochastic EnKF, localized by the identity matrix. Inflation is tuned to achieve an MSE roughly equal to the spread. We also apply a localized PF (section 2.3) to this problem. Localization of the PF is straightforward in this diagonal problem because we can compute weights separately in each variable, and also perform the resampling step separately in each variable (section 2.4). One may argue that this problem is too trivial to test localization methods because this problem lacks complex multivariate relationships between the various state components (balance); (see also Rebeschini and van Handel, 2015; Snyder *et al.*, 2015; Morzfeld *et al.*, 2017). Nevertheless, this problem can serve as a benchmark and best-case-scenario for localized PF.

We compare the results we obtain by the "optimal" localization strategy of decoupling to the localization methods described by Poterjoy (2015) and Poterjoy and Anderson (2016). Here we use a small localization radius, as is required by this diagonal problem, and tune the localization scheme to achieve small MSE and comparable spread. We also apply the varPS, which does not require localization because the problem is linear, and, therefore, all weights are equal (section 4 gives benchmark results with nonlinear problems). Inflation of varPS is tuned to achieve small MSE and comparable spread.

For each method, we vary the ensemble size and record MSE and spread. We perform each experiment 5,000 times and average over the number of experiments. The results are shown in Figure 1; (a) shows the results with **R** being the identity matrix, and (b) when $\mathbf{R} = 0.1\mathbf{I}$, i.e. when the accuracy of the observations is increased.

We note that EnKF (in orange) exhibits the smallest MSE and that MSE is approximately equal to the spread even for small ensembles. The varPS (in red) yields comparable results. Both localized PFs (green and purple) exhibit larger MSE and a small spread, unless the ensemble size is larger than the dimension of the problem. Moreover, localization by Poterjoy's method yields results that are similar to the results one obtains by localization via decoupling, indicating that the localization strategy is effective.

We now consider a variation of this problem and decrease the observation-noise covariance by setting $\mathbf{R} = 0.1\,\mathbf{I}$. We observe qualitatively similar results as before, i.e. EnKF and varPS errors are smaller than PF errors, and localization by Poterjoy's method is as effective as an "optimally" localized PF. However, the PFs now yield significantly larger MSE than EnKF or varPS. We also note that localized PFs underestimate the spread in both experiments, unless the ensemble size is large. This suggests that "inflation" is needed by PFs in addition to localization.

If these examples were indeed indicative of how data assimilation algorithms perform in meteorological problems, then we conclude that localized PFs may not perform as well as localized EnKF or varPS in Gaussian or "nearly" Gaussian problems. The numerical experiments with localized PFs presented in Poterjoy (2015) and Poterjoy and Anderson (2016) confirm this conclusion – localized PFs are found

to perform no better than localized EnKF unless the nonlinearity/non-Gaussianity is significant due to a nonlinear observation function. Taking our simple examples with "perfect" localization and the more realistic simulations with a doable localization strategy into account, it appears unlikely that even a localized PF can perform as well as EnKF when the ensemble size is small and when the problem is Gaussian or nearly Gaussian.

The varPS performs as well as the EnKF. This may not be surprising because varPS exploits linearity of the model, while the PF, localized or not, does not make use of this linearity. However, our numerical examples give no indication that PF or varPS would be more appropriate than a tuned EnKF.

3.5 | Why smoothing and not filtering?

We wish to explain in more detail why we use the varPS to sample the distribution $p(\mathbf{x}_{k-1}|\mathbf{y}_{1:k})$, rather than adopting the "usual" PF approach and sampling $p(\mathbf{x}_k|\mathbf{y}_{1:k})$. Recall that the "standard" PF samples the proposal distribution $q(\mathbf{x}_k) = p(\mathbf{x}_k|\mathbf{y}_{1:k-1})$ by using the model to evolve an ensemble from time $k-1$ to time k. Its weights are the ratio of the posterior distribution and the proposal distribution:

$$w \propto \frac{p(\mathbf{x}_k|\mathbf{y}_{1:k})}{q(\mathbf{x}_k)} \propto \frac{p(\mathbf{x}_k|\mathbf{y}_{1:k-1})p(\mathbf{y}_k|\mathbf{x}_k)}{p(\mathbf{x}_k|\mathbf{y}_{1:k-1})} \propto p(\mathbf{y}_k|\mathbf{x}_k).$$

Note that $p(\mathbf{x}_k|\mathbf{y}_{1:k-1})$, which is generally unknown, cancels in the calculation of the weights.

In principle, other choices of proposal distributions are possible. In particular, one may choose a proposal distribution $q(\mathbf{x}_k|\mathbf{x}_{k-1}, \mathbf{y}_{1:k})$ that depends on the observations at time k, rather than only on observations up to time $k-1$. Suppose we have such a proposal and can draw N_e samples from it. The weights of the particles are given by

$$w \propto \frac{p(\mathbf{x}_k|\mathbf{y}_{1:k-1})p(\mathbf{y}_k|\mathbf{x}_k)}{q(\mathbf{x}_k|\mathbf{x}_{k-1}, \mathbf{y}_{1:k})}.$$

These weights cannot be evaluated because the probability distribution $p(\mathbf{x}_k|\mathbf{y}_{1:k-1})$ is, in general, not known. The exception are linear/Gaussian problems for which $p(\mathbf{x}_k|\mathbf{y}_{1:k-1})$ is known. Since the weights cannot be computed, using a particle filter with proposal distributions other than the standard choice is not possible without further approximation, e.g. the one presented in Klaas *et al.* (2005). Alternatively, one could use a Gaussian approximation for $p(\mathbf{x}_k|\mathbf{y}_{1:k-1})$, which would lead to particle filter algorithms similar to 3D-Var or EnKF methods.

Note that the varPS runs into the same problem: the posterior distribution $p(\mathbf{x}_k|\mathbf{y}_{1:k})$ *cannot* be evaluated. For this reason, a Gaussian approximation of $p(\mathbf{x}_{k-1}|\mathbf{y}_{1:k-1})$ is used to define the approximate posterior distribution $\widehat{p}$ (Equation 5). This approximate posterior can be evaluated, which is the key to computing weights for the varPS. Note that IEnKF/IEnKF also make use of the approximate posterior distribution $\widehat{p}$, and uses this approximation, and not the "true" posterior distribution for ensemble generation (Sakov *et al.*, 2012; Bocquet and Sakov, 2013; 2014; Bocquet, 2016).

Therefore, the main difference between particle filtering (sampling the distribution $p(\mathbf{x}_k|\mathbf{y}_{1:k})$) and smoothing/varPS (sampling the distribution $p(\mathbf{x}_{k-1}|\mathbf{y}_{1:k})$) is the time at which a Gaussian approximation is made. We argue that a Gaussian approximation of $p(\mathbf{x}_{k-1}|\mathbf{y}_{1:k-1})$ is more sensible than a Gaussian approximation of $p(\mathbf{x}_k|\mathbf{y}_{1:k-1})$. Numerical evidence for this statement may be the success of 4D-Var techniques and of IEnKF/IEnKS (Sakov *et al.*, 2012; Bocquet and Sakov, 2013; 2014; Bocquet, 2016), which rely on this approximation, for NWP. The varPS we propose makes the same Gaussian approximation of a "background", or prior distribution, as 4D-Var, the IEnKF and the IEnKS, but it can draw samples from a possibly non-Gaussian posterior distribution.

Finally, one may wonder how other PFs, such as the equal weights particle filter of Ades and van Leeuwen (2013) and van Leeuwen (2010), nudging techniques as described by Weare (2009) and Vanden-Eijnden and Weare (2012), some implicit PFs described by Morzfeld *et al.* (2012), Chorin and Tu (2009) and Chorin *et al.* (2010), PFs using an EnKF proposal as in Papadakis *et al.* (2010), or optimal PFs described by Arulampalam *et al.* (2002), Doucet *et al.* (2000; 2001), Zaritskii and Shimelevich (1975), Liu and Chen (1995) and Snyder *et al.* (2015) fit into this picture. Such filters are built for stochastic models, which are slightly different and discussed briefly in Appendix A. In particular, we show that even optimal PFs, optimally localized, cannot match the performance of the EnKF in a linear benchmark problem. Here it is important to note that optimality refers to optimality over a class of PFs, defined by a certain family of proposal distributions. Equivalent weights PFs, for example, make use of more general mechanisms for proposing an ensemble and are not members of this family of PFs. Therefore, the results we report in the Appendix do not apply to them.

4 | WEIGHT-LOCALIZATION OF THE VARIATIONAL PARTICLE SMOOTHER

We noted above that the varPS does not collapse on idealized linear/Gaussian problems. The reason is that the proposal distribution of varPS is a Gaussian, centred at the mode and with a covariance equal to the inverse Hessian of the 4D-Var cost function. Thus, the proposal distribution is equal to the posterior distribution of a linear problem. This implies that all weights are equal, which in turn implies that collapse does not occur. In practice however a problem is rarely linear and the Hessian is typically not known exactly. We show that the weights of varPS collapse in this situation, and that the collapse can be prevented by weight-localization. We first consider "diagonal" problems, for which weight-localization is straightforward. We then present a weight-localization strategy that can be used for more general, non-diagonal problems.

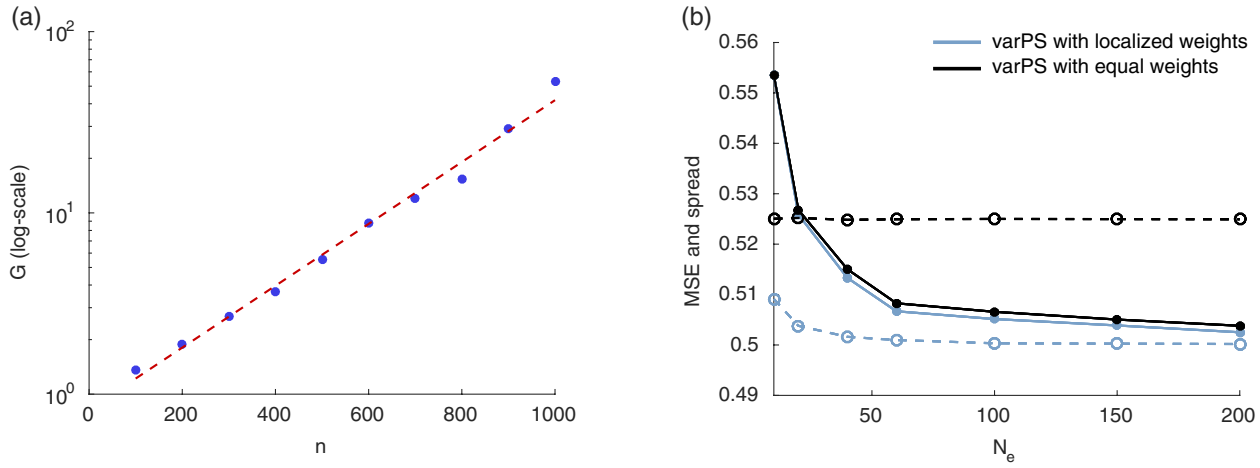

FIGURE 2 (a) G as a function of dimension computed for a given n using an ensemble size N_e=10^5 (blue dots). The dashed red line shows an exponential fit. (b) MSE (solid) and spread (dashed) as a function of the ensemble size for varPS with weight-localization (light blue) and varPS without weights (black) [Colour figure can be viewed at wileyonlinelibrary.com]

4.1 | Linear diagonal problems

We first neglect effects of nonlinearity and describe how the collapse of the varPS can be caused by a proposal distribution with larger covariance than the posterior distribution. In practice, this situation is likely to occur because of inflation or approximations used when generating 4D-covariances.

We first illustrate the collapse of the varPS by considering a Gaussian posterior distribution $\mathcal{N}(0, \mathbf{I})$ and Gaussian varPS proposal distribution with slightly larger covariance:

$$p(\mathbf{x}_{k-1}|\mathbf{y}_{1:k}) = \mathcal{N}(0, \mathbf{I}), \quad q(\mathbf{x}_{k-1}) = \mathcal{N}(0, (1+\beta)\,\mathbf{I}), \quad \beta > 0.$$

We can vary the dimension n of this problem and compute G in Equation 10 as a function of dimension (Snyder *et al.*, 2015 give the calculation):

$$G = \left(\frac{1+\beta}{\sqrt{1+2\beta}}\right)^n. \tag{18}$$

The exponential dependence of G on dimension implies that the ensemble size required by the varPS grows exponentially with dimension (since $N_{\text{eff}} = G{\cdot}N_e$).

In this diagonal problem, weight-localization is straightforward and can prevent collapse. As described in the context of PFs, weight-localization in a diagonal problem can be done by computing weights for varPS independently for each state component because there are no complex multivariate relationships. The weight-localization implies that the n factors of G in Equation 18 apply to each variable separately, i.e. for each variable we have $G_i = (1+\beta)/\sqrt{1+2\beta}$, which is "small", so that a moderate number of ensemble members is sufficient to solve this problem, independently of dimension n. Therefore, weight-localization breaks the exponential dependence of the required ensemble size on dimension and prevents the collapse of varPS.

We further illustrate the collapse of the varPS by revisiting the linear diagonal benchmark problem of section 3.4. We now relax the assumption that the varPS proposal covariance is exactly equal to the posterior covariance and consider a varPS with proposal distribution

$$q(\mathbf{x}) = \mathcal{N}(\mathbf{x}^*, \mathbf{C}), \qquad \mathbf{C} = (1+\beta)\,\mathbf{J}^{-1},$$

where $\mathbf{x}^*$ is the posterior mode and $\beta = 0.05$. As before, we vary dimension n for this problem from $n = 100$ to $n = 1,000$, and compute G of varPS as a function of n. To compute G we use an ensemble size $N_e = 10^5$. Such a large ensemble size is necessary here because the larger G is, the larger N_e is required to compute G accurately. The results are shown in Figure 2a. We note the exponential dependence of G on n, which causes the collapse of varPS.

When we fix dimension, $n = 100$, and apply weight-localization to the varPS, the collapse is prevented and we obtain small MSE and comparable spread even when N_e is small, as shown in Figure 2b. Indeed, this varPS (light blue) yields results comparable to what we obtained under idealized conditions in section 3.4.

The collapse of a localized varPS does not occur if G, in each variable, is small. However, if G is small, then the varPS proposal is a good approximation of the posterior distribution (locally, in that variable). The reason is that the varPS weights (Equation 13) account for differences between the varPS proposal and posterior distributions. G is only small if the weights are nearly constant, which means that the varPS proposal differs only minimally from the posterior distribution. Thus, one may question whether an accurate solution can be obtained by setting the weights of each varPS ensemble member to $w = 1/N_e$, rather than using the weights in Equation 13. The resulting ensemble is distributed according to the varPS proposal distribution in Equation 12, not the posterior distribution $\widehat{p}$ in Equation 5. Thus, posterior means and covariances are approximated by proposal means and covariances, and this approximation should be "good" if G is small. Naturally, replacing the varPS weights by $w = 1/N_e$ also prevents the collapse in high-dimensional problems (similar to weight-localization). For the reminder, we will call an

implementation of varPS with weights $w = 1/N_e$ the *equally weighted varPS* or *varPS with equal weights*. The equally weighted varPS is similar to some ensemble formulations of 4D-Var which are currently in practical or operational use (e.g. Zupanski, 2004; Sakov *et al.*, 2012; Bocquet and Sakov, 2013; 2014; Kuhl *et al.*, 2013; Auligné *et al.*, 2016; Bocquet, 2016). Indeed, one can view varPS as an importance sampling method that uses IEnKF/IEnKS as a proposal distribution, or, vice versa, one can view the IEnKF/IEnKS as varPS with equal weights. The reason is that the proposed, unweighted ensemble of varPS coincides with the ensemble used by IEnKF/IEnKS (setting numerical implementations of optimization or Hessian/Jacobian computations aside). Note that the use or "non-use" of weights has also been discussed in the context of RTO (section 3.3 and Bardsley *et al.*, 2014). For some problems, samples from the RTO proposal distribution, without any weights, lead to accurate estimates even if the sampling problem is not Gaussian (figures 1 and 3 in Bardsley *et al.*, 2014). In fact, the corrections induced by the weights often seem negligible. These ideas were also discussed in the context of pollutant source retrieval in Liu *et al.* (2017). We emphasize that varPS or RTO with equal weights are fundamentally different from the "equal weights particle filter" (EWPF) of Ades and van Leeuwen (2013) (Appendix A). All three methods, varPS and RTO with equal weights, and EWPF have equal weights "by construction" and, therefore, avoid filter collapse, but EWPF modifies samples so that all samples receive an equal weight, whereas varPS and RTO with equal weights simply neglect the weights, i.e. these methods accept the proposal distribution as the posterior distribution.

Results obtained by the equally weighted varPS are shown in black in Figure 2. In this example the equally weighted varPS can achieve MSE comparable to the weighted varPS, i.e. the weights do not have a large effect on MSE. However, the equally weighted varPS overestimates the spread. The reason is that the covariance matrix of the proposal is larger than the covariance of the posterior distribution. Nonetheless, one can obtain good results by an equally weighted varPS whenever the covariance of the proposal is a "good" approximation of the posterior covariance (inverse Hessian of the 4D-Var cost function). This also suggests that the equally weighted varPS can be an effective strategy in high-dimensional linear problems, as the equal weights prevent the collapse. We investigate if the "equal weights" strategy is applicable to (mildly) nonlinear problems in the next section.

4.2 | Diagonal nonlinear problems

The rate at which the collapse of varPS occurs in mildly nonlinear problems can be studied by "small noise theory", e.g. Goodman *et al.* (2015). For a small noise analysis, we assume that the approximate posterior $\widehat{p}$ in Equation 5 is "near" a Gaussian, e.g. because the model $\mathbf{f}$ is mildly nonlinear. This means that the 4D-Var cost function $\mathcal{J}$ is quadratic plus an order-γ perturbation, and possibly higher-order terms (HOTs):

$$\mathcal{J}(\mathbf{x}_{k-1}) = \frac{1}{2}\left(\mathbf{x}_{k-1} - \mathbf{x}^*\right)^{\mathrm{T}} \mathbf{J}\left(\mathbf{x}_{k-1} - \mathbf{x}^*\right) + O(\gamma) + \mathrm{HOTs},$$

where $\mathbf{J}$ is the Hessian of $\mathcal{J}$ evaluated at $\mathbf{x}^*$, and where $O(\gamma)$ denotes terms that are equal to some constant multiplied by γ. A Taylor expansion of G in Equation 10 can be written as

$$G = 1 + E\left[C_3(\mathbf{x}_{k-1} - \mathbf{x}^*)^2\right] \cdot O(\gamma^2) + \mathrm{HOTs}$$

where C_3 is the third coefficient of a Taylor expansion of $\mathcal{J}$. To leading order, and for a fixed dimension n, the required ensemble size of varPS thus scales quadratically in the perturbation parameter γ. In contrast, the standard PF has the property that $G \to \infty$ as $\gamma \to 0$, i.e. the required ensemble size blows up as the perturbation decreases in size, indicating that the collapse of the varPS happens "more slowly" than for the standard particle filter in near-Gaussian problems (also Weare, 2009 and Vanden-Eijnden and Weare, 2012).

Nonetheless, there is a "hidden" dependence of G on dimension, which we investigate by considering diagonal problems for which

$$G = \frac{E[w^2]}{E[w]^2} = \left(\frac{E_1}{(E_2)^2}\right)^n. \tag{19}$$

The derivation of this formula and expressions for the quantities E_1 and E_2 are in Appendix B. Since G is exponential in n, our calculation indicates that even in mildly nonlinear problems (small perturbation parameter γ), varPS collapses exponentially fast if dimension is large.

We illustrate the collapse of varPS in nonlinear problems, by a nonlinear test problem similar to the linear problem of section 3.4:

$$\mathbf{x}_k = \mathbf{x}_{k-1} + \beta\left(-\sqrt{3}\,\mathbf{x}_{k-1}^2 + \mathbf{x}_{k-1}^3\right),$$
$$\mathbf{y}_k = \mathbf{x}_k + \varepsilon_k,$$

where ε_k are iid Gaussians and where $\mathbf{x}_k^s$ is to be interpreted element-wise, i.e. $(\mathbf{x}^s)^i = (x^i)^s$. Note that the perturbation parameter β controls the nonlinearity and that we recover the linear benchmark problem of section 3.4 for $\beta = 0$.

We first fix $\beta = 0.1$ and vary dimension n and compute G as a function of dimension. Results obtained with an ensemble of size $N_e = 10^4$ are shown in Figure 3a.

We note the expected exponential scaling of G with dimension, leading to the collapse of varPS. Next, we fix dimension $n = 10$ and vary the perturbation parameter β between zero (linear problem) and one (nonlinear problem). This allows us to investigate the performance of varPS as the problem becomes "more nonlinear". In order to prevent the collapse of varPS for $\beta > 0$, we localize its weights by decoupling (as described above). Results are shown in Figure 3a for an ensemble size of $N_e = 100$. Since MSE and spread are "random" for each experiment (the true state and the observation are drawn at random), we show the average of MSE and spread of 10,000 experiments. We compare the results we obtain by varPS with "optimal" weight-localization (light

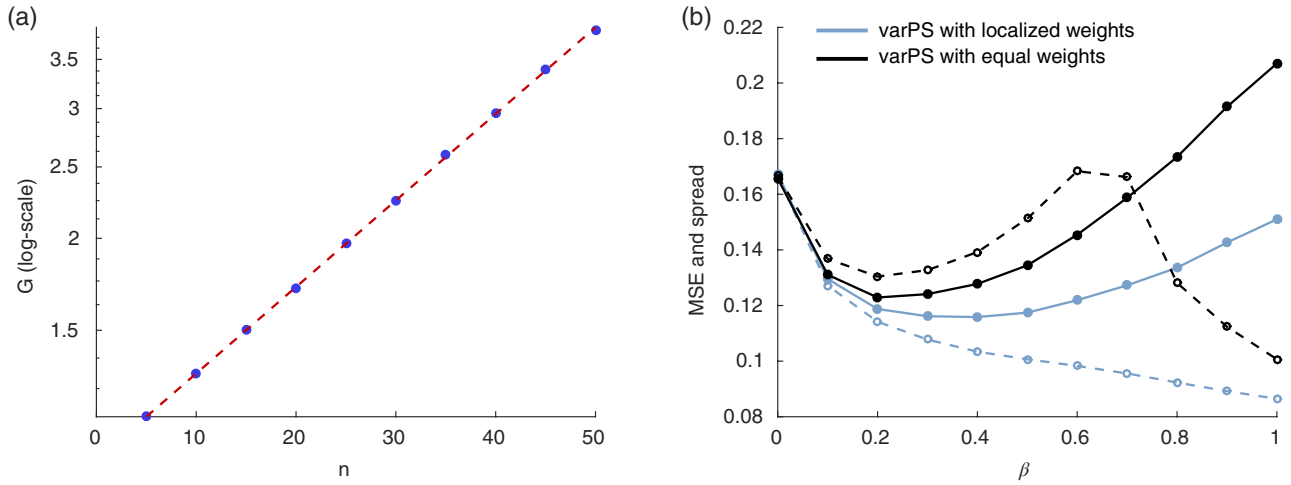

FIGURE 3 (a) G as a function of dimension computed for a given n using a ensemble size $N_e = 10^4$ (blue dots). The dashed red line shows an exponential fit. (b) MSE (solid lines) and spread (dashed lines) as a function of the perturbation parameter β for varPS with weight-localization (light blue) and equally weighted varPS (black) [Colour figure can be viewed at wileyonlinelibrary.com]

blue) to results we obtain by the equally weighted varPS, i.e. setting all weights to $w = 1/N_e$ (see above). Both methods give identical results when the perturbation parameter $\beta = 0$, since $\beta = 0$ corresponds to a linear problem, so that the varPS weights indeed are all equal $w = 1/N_e$. However, even for relatively large perturbation parameters $\beta \approx 0.4$, the varPS with equal weights yields acceptable results. Thus, even for moderately nonlinear problems, and even for relatively large ensemble sizes, the equally weighted varPS provides an effective means to obtain "useful" solutions of the nonlinear problem. The varPS weight become important only when the nonlinearity is substantial (large β). This result suggests that the weighted varPS solution only provides benefits over the unweighted solution for "highly" nonlinear/non-Gaussian problems. In near-Gaussian problems, using localized weights may not yield significant advantages over the equally weighted varPS, or other linearized solutions.

We illustrate the above statements by illustrating the posterior distributions $p(x_0|y)$ and $p(x_1|y)$ for the above nonlinear example with $\beta = 0.4$. In Figure 4, we plot the posterior distribution of one of the variables at time $t = 0$, and its Gaussian approximation, which is the proposal distribution of varPS, or equivalently, the approximation used by the equally weighted varPS.

We use $N_e = 10^6$, because ensemble size is not the issue here, and because we wish to study the errors this method makes in addition to any sampling error caused by small ensemble sizes. In Figure 4a, we show the posterior distribution $p(x_0|y)$ at time $t = 0$ and its approximation by the equally weighted varPS. Figure 4b shows the posterior distribution $p(x_1|y)$ at time $t = 1$, its approximation by the equally weighted varPS, and its approximation by EnKF, also with $N_e = 10^6$.

We note that there is significant error, both at time $t = 0$ and $t = 1$. However the modes of the distributions generated by the equally weighted varPS and the posterior distributions nearly coincide (at times $t = 0$ and at time $t = 1$). The good "match" between these modes leads to the small MSE we observe in our previous experiments. The EnKF approximation of the posterior distribution at time $t = 0$, $p(x_0|y)$, is the prior (standard Gaussian, in this example). The EnKF approximation of the posterior distribution at time $t = 1$, $p(x_1|y)$, is shown in Figure 4a and we note that the mean and mode of the EnKF approximation are far from their true values. Moreover, the EnKF overestimates posterior covariances even more than the equally weighted varPS. We wish to emphasize again that the equally weighted varPS operates in a way very similar to some current ensemble formulations of 4D-Var (e.g. Zupanski, 2004; Sakov *et al.*, 2012; Bocquet and Sakov, 2013; 2014; Kuhl *et al.*, 2013; Auligné *et al.*, 2016; Bocquet, 2016), none of which makes use of weights (which is equivalent to setting all weights equal to $w = 1/N_e$, as in the equally weighted varPS). This suggest that the equally weighted varPS may be successful in practical problems with moderate nonlinearity, and, perhaps more importantly, it suggest that the varPS proposal distribution is accurate and that weight calculation and possibly localization is straightforward.

4.3 | Weight-localization for general problems

Thus far, we have addressed weight-localization of varPS for diagonal problems (where the optimal localization is trivial to implement), and investigated the validity of neglecting the weights altogether (as in the equally weighted varPS). We now present a weight-localization for more general applications, but assuming that $\mathbf{R}$ is a diagonal matrix and that $[\mathbf{y}]_j = [\mathbf{h}]_j([\mathbf{f}(\mathbf{x}_{k-1})]_j)$, i.e. each component $[\mathbf{y}]_j$ of an observation $\mathbf{y}$ depends on only one component of $\mathbf{f}(\mathbf{x}_{k-1})$. We define a "localization function" which deceases exponentially with distance from the observation $[\mathbf{y}]_j$:

$$\rho_j(\Delta x) = \exp\left[-(\Delta x/2L)^2\right],$$

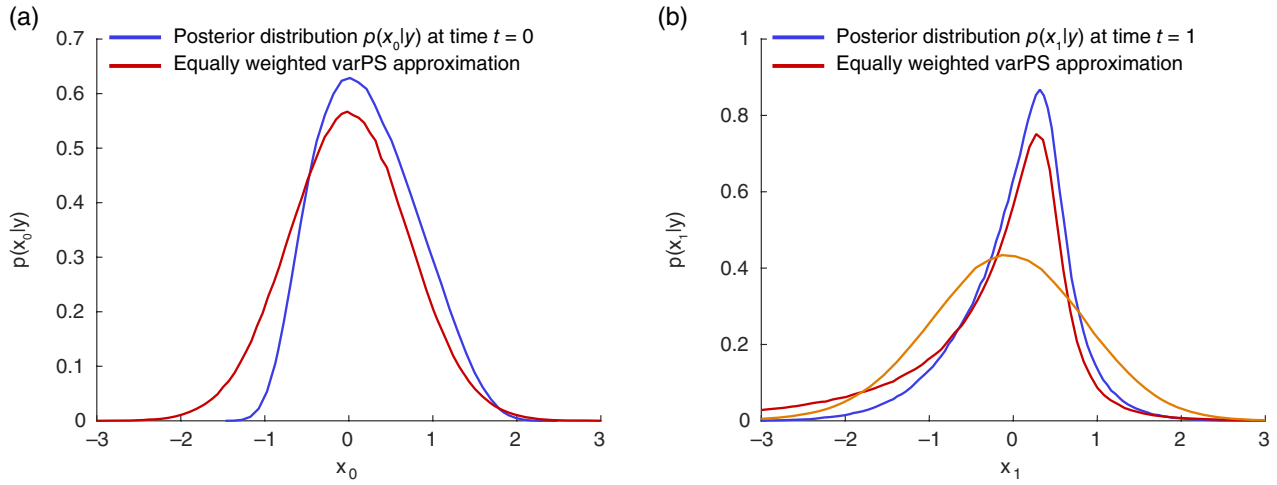

FIGURE 4 (a) Posterior distribution, $p(x_0|y)$, at time $t = 0$ (blue) and equally weighted varPS approximation (red). (b) Posterior distribution, $p(x_1|y)$, at time $t = 1$ (blue), equally weighted varPS approximation (red), and EnKF approximation (orange) [Colour figure can be viewed at wileyonlinelibrary.com]

where Δx measures the distance to the observation $[\mathbf{y}]_j$ and L is a tuning parameter. Under our assumptions, the weights in Equation 13 can be written as

$$w \propto \frac{\exp\left[-\frac{1}{2}(\mathbf{x}_{k-1}-\mu)\mathbf{B}^{-1}(\mathbf{x}_{k-1}-\mu)\right]}{\exp\left[-\frac{1}{2}(\mathbf{x}_{k-1}-\mathbf{x}^*)\mathbf{J}^{-1}(\mathbf{x}_{k-1}-\mathbf{x}^*\right]}\prod_{j=1}^{k} \exp\left[-\frac{1}{2}\frac{\{[\mathbf{y}]_j-[\mathbf{h}]_j([\mathbf{f}(\mathbf{x}_{k-1})]_j)\}^2}{[R]j,j}\right].$$

Taking the negative logarithm simplifies this equation to

$$-\log w = \frac{1}{2}(\mathbf{x}-\mu)\mathbf{B}^{-1}(\mathbf{x}-\mu) - \frac{1}{2}(\mathbf{x}-\mathbf{x}^*)\mathbf{J}(\mathbf{x}-\mathbf{x}^*) + \sum_{j=1}^{k}\frac{1}{2}\frac{\{[\mathbf{y}]_j-[\mathbf{h}]_j([\mathbf{f}(\mathbf{x})]_j)\}^2}{[R]j,j}.$$

This above expression suggests that we can define a weight at an observation location by

$$-\log w_j = \frac{1}{2}\frac{\{[\mathbf{y}]_j - [\mathbf{h}]_j([\mathbf{f}(\mathbf{x})]_j)\}^2}{[R]j,j} + \frac{1}{2}||\rho_j \circ \left\{\mathbf{B}^{-1/2}\,(\mu - \mathbf{x}_{k-1})\right\}||^2 - \frac{1}{2}||\rho_j \circ \left\{\mathbf{J}^{1/2}\,(x^* - \mathbf{x}_{k-1})\right\}||^2,$$

where the open circle denotes element-wise vector–vector multiplication. Weights near the observation locations are computed by interpolating between weights at observation locations. Taking the exponential and normalizing the weights so that their sum *over the ensemble members and at every location* is one results in a $n \times N_e$ matrix of weights, $\mathbf{W}$. This weight matrix contains the weights of the N_e ensemble members, and vary over the (spatial) domain.

With the spatially varying weights, we can compute a weighted mean and weighted covariance matrix using essentially the same methods as in Tödter and Ahrens (2015). Computing only weighted means and covariances is sufficient for updating the "background" mean and covariance and avoids difficulties that arise from localized resampling. In fact, we have experimented extensively with resampling strategies based on localized weights, but none of the methods we tried lead to results that are comparable to EnKF or ensemble formulations of 4D-Var. For that reason, we are here satisfied with updating the background mean and covariance using the localized weights, however accurate computation of higher moments requires more sophisticated techniques; also van Leeuwen (2009) gives a discussion of the difficulties of resampling and localization for PFs.

Specifically, let $\mathbf{w}^j$ be the jth column of the weight matrix, which contains the spatially varying weights for the jth sample at time k, $\mathbf{x}_k^j = \mathbf{f}_k(\mathbf{x}_{k-1}^j)$. Then

$$\bar{\mathbf{x}}_k = \sum_{j=1}^{N_e} \mathbf{w}^j \circ \mathbf{x}_k^j$$

is the weighted sample mean. We define the $n \times N_e$ matrix $\mathbf{U}$ with colums $\mathbf{u}^j = \sqrt{\mathbf{w}^j} \circ (\mathbf{x}_k{}^j - \bar{\mathbf{x}}_k)$, where the square root of a vector is to be understood as taking the square root of each of its elements. Following Tödter and Ahrens (2015), the background matrix for the next assimilation cycle is computed as $\mathbf{B} = \mathbf{U}\mathbf{U}^{\mathrm{T}}$. Note that an "infinite" localization radius implies that

$$\mathbf{w}^j = w^j\,\mathbf{1},$$

where $\mathbf{1}$ is an n-dimensional vector whose elements are all equal to one. In this case, one obtains the usual formulae for weighted covariance

$$\mathbf{B} = \sum_{j=1}^{N_e} w^j\,(\mathbf{x}^j{}_k - \bar{\mathbf{x}}_k)(\mathbf{x}_k{}^j - \bar{\mathbf{x}}_k)^{\mathrm{T}}, \qquad \sum_{j=1}^{N_e} w^j = 1, \tag{20}$$

i.e. the varPS with weight-localization in the limit of large localization radius is equal to a varPS without weight localization. When all weights are equal to $w = 1/N_e$, the above formula reduces to

$$\mathbf{B} = \frac{1}{N_e}\sum_{j=1}^{N_e}(\mathbf{x}^j{}_k - \bar{\mathbf{x}}_k)(\mathbf{x}_k{}^j - \bar{\mathbf{x}}_k)^{\mathrm{T}},$$

which is not an unbiased estimator for the covariance. This suggest that one replaces $\mathbf{B}$ in Equation 20 by

$$\mathbf{B} = \frac{N_e}{N_e - 1} \sum_{j=1}^{N_e} w^j \, (\mathbf{x}_k{}^j - \bar{\mathbf{x}}_k)(\mathbf{x}_k{}^j - \bar{\mathbf{x}}_k)^{\mathrm{T}}, \qquad \sum_{j=1}^{N_e} w^j = 1.$$

Note that the varPS requires localization in two stages: weight-localization is done by computing weights locally to produce a weighted covariance estimate. In addition, for small ensemble sizes, there is significant sampling error, regardless of how effective the weight-localization is and, hence, one must localize the resulting weighted covariance matrix $\mathbf{B}$ to reduce effects of spurious correlations. This second localization, and a required inflation, can be done using the usual techniques, e.g. by setting

$$\mathbf{B}_{\mathrm{loc}} = \alpha \, (\mathbf{L} \circ \mathbf{B}),$$

where $\mathbf{L}$ is a suitable localization matrix and $\alpha > 1$ is an inflation parameter, as in EnKF or ensemble formulations of 4D-Var.

4.4 | Summary of varPS, its localization and the equally weighted varPS

We summarize our discussion of varPS so far:

1. the varPS exploits near-Gaussian problem structure by merging ideas from 4D-Var with the particle approach;
2. varPS can exploit sparse/banded problem structure by weight localization, which prevents its collapse in high-dimensional problems;
3. in near-Gaussian problems, the equally weighted varPS generates ensembles that are as appropriate as weighted ensembles, while also avoiding collapse.

Items (1) and (2) are essential for obtaining useful results with small ensemble sizes in high-dimensional problems. In contrast, PFs make use of sparse/banded structure by weight-localization, which makes them applicable to high-dimensional problems because the required ensemble size is moderate (at least not exponential in dimension). However, our benchmark tests suggest that localized PFs that do not exploit Gaussian structure in near-Gaussian problems are not as effective as techniques that do.

The equally weighted varPS effectively represents the posterior distribution by the Gaussian varPS proposal. The weights (localized, if necessary) morph the varPS proposal into the posterior distribution. However, our preliminary tests suggest that these weights have a significant effect only if the nonlinearity is substantial. Even in moderately nonlinear problems, using equal weights can be effective, especially if small MSE and spread are the main concern, and if one is limited to small ensemble size.

It is important to re-iterate connections of varPS with equal weights and IEnKF/IEnKS (Sakov *et al.*, 2012; Bocquet and Sakov, 2013; 2014; Bocquet, 2016). We explained above that the ensemble of IEnKF/IEnKS coincides with the unweighted (proposal) ensemble of varPS. Thus, varPS with equal weights can be viewed as an implementation of an IEnKF/IEnKS. For that reason, we do not compare IEnKF/IEnKS with varPS or varPS with equal weights in our numerical experiments below; comparisons with varPS with equal weights are direct indications of what to expect from IEnKF/IEnKS.

5 | NUMERICAL EXPERIMENTS WITH THE LORENZ'96 MODEL

We test the varPS on the L96 model (Lorenz, 1996). Our goal is to test if the ideas we developed above can hold true for a simple test problem that is popular for testing algorithms in NWP. More specifically, we use numerical simulations to examine whether the varPS is better than standard PFs at preventing weight collapse, and whether the proposed method is an effective data assimilation technique for high-dimensional nonlinear problems. To that extent, we compare the varPS to the localized PF, EnKF (square root and stochastic) and ensemble formulations of 4D-Var.

5.1 | Results for 40-dimensional problems

We first consider a model with $n = 40$ variables. The function $\mathbf{f}$ in Equation 1 is given by a fourth-order Runge–Kutta discretization of the L96 dynamics with time step $\Delta t = 0.05$ (as in Poterjoy, 2015). We collect observations of every other state variable, every fourth time step ($\Delta T = 0.2$ between observations). We vary the accuracy of the observations and consider the noise covariances $\mathbf{R} = \mathbf{I}$ and $\mathbf{R} = 0.1\,\mathbf{I}$. For each observation-error covariance we perform data assimilation by the following algorithms:

1. EnKF (stochastic) with inflation and localization;
2. EnKF (square root) with inflation and localization;
3. PF with localization by Poterjoy's method;
4. varPS with inflation and weight-localization;
5. varPS with inflation but *without* weight-localization;
6. varPS with inflation and *equal* weights (similar to the IEnKF);
7. E4D-Var with inflation and localization;
8. EDA with inflation and localization.

Localization of the standard PF is done by Poterjoy's method described in Poterjoy (2015) and Poterjoy and Anderson (2016), with squared exponential localization function. The method also requires setting a minimum weight, which has effects similar to that of covariance inflation in EnKF, and this parameter is also tuned. We make use of an additional particle adjustment step based on kernel density estimation (KDDM). However, we ran some of the numerical experiments without the KDDM step and noticed similar performance.

Our E4D-Var method is as follows. Given a background covariance and a set of observations, we minimize the associated 4D-cost function by a Gauss–Newton method. The background covariances are updated by an EnKF (stochastic) which we run in parallel to our 4D-Var system. Information is exchanged between the 4D-Var and EnKF systems in the sense that the background covariance in 4D-Var is updated by the EnKF analysis covariance of the previous assimilation window, and the EnKF ensemble is re-centred around the 4D-Var state estimate. Our EDA method amounts to solving the 4D-Var optimization problem repeatedly using perturbed observations and perturbed states.

We initialize all algorithms with an initial ensemble drawn from an EnKF run with a large ensemble (and tuned localization and inflation), so that the methods start with a spun-up ensemble. We then perform 10^3 data assimilation cycles. Localization and inflation are tuned for each method and ensemble size by evaluating time-averaged MSE over a matrix of localization and inflation parameters. We declare the localization/inflation parameters that lead to a minimum time averaged MSE as "optimal". MSE and spread, at time k, are defined as described in section 3.4, Equations 16 and 17. Note that MSE and spread defined in this way are posterior quantities (computed using the analysis, rather than forecast ensemble), and that we compute MSE and spread, for all methods, at observation time. For E4DVar, EDA and varPS, this means that we propagate the ensemble to time k (when observation y_k is received) using the model (1). The time-averaged MSE is defined as the average MSE over 800 assimilation cycles (disregarding the first 200 cycles as additional spin-up):

$$\mathrm{MSE}_{\mathrm{avg}} = \frac{1}{800}\sum_{j=1}^{800} \mathrm{MSE}_{200+j}.$$

Results are shown in Figure 5, where we plot time-averaged MSE and time-averaged spread (defined in the same way as time-averaged MSE) as a function of the ensemble size N_e. Note that our simulation and assimilation runs are relatively short. For that reason the average statistics of MSE and spread may not be accurate to more than a few digits, however our numerical experiments reliably indicate the methods' performances.

Both EnKF implementations (stochastic and square root) yield comparable results and the EnKFs and localized PF yield a larger MSE and spread than the variational methods or varPS. Moreover, the localized PF yields the largest MSE and its performance degrades when the observation-error covariance is small (consistent with previously reported results). The variational methods (E4D-Var and EDA) yield the smallest MSE.

We note that the varPS can "beat" EnKF in this mildly nonlinear problem, but varPS does not perform better than ensemble formulations of 4D-Var (also Bocquet and Sakov, 2013). It is also remarkable that the varPS does not require weight localization in this 40-dimensional problem, for which the standard PF without weight-localization collapses. Indeed, the results we obtain by weight-localization are comparable to those obtained without weight-localization. One can argue that we did not "tune" the weight-localization sufficiently, since the localized and unlocalized implementations are equal if the localization radius is large enough (infinite). However, we obtained the results shown in the figure by tuning the weight-localization and inflation of varPS over a number of finite choices. Thus, our experiments confirm that the weight-localization strategy of section 4 does not introduce large additional errors, even if the localization radius is not chosen "optimally" (which is likely the case in practice).

The varPS does not collapse in this problem because the weights are well-distributed (small G), which indicates that the varPS proposal distribution is a good approximation of the posterior distribution in Equation 5. For this reason, the equally weighted varPS produces results which are almost identical to the weighted varPS.

5.2 | Results for 400-dimensional problems

We repeat some of the calculations for a L96 problem of dimension $n = 400$, where we observe every other variable every four time steps. Our results are shown in Figure 6.

We obtain qualitatively and, to a large extent *quantitatively*, the same results as in the experiments with $n = 40$ dimensions. The reason is that the L96 problem has the anticipated "sparse structure" we exploit during localization, so that the overall dimension is irrelevant. What defines performance of localized data assimilation algorithms is the structure of each loosely coupled block, not the overall number of blocks.

These numerical experiments are important for testing the localization of varPS. Only if an unlocalized varPS "fails", but a localized varPS leads to useful results can one claim that the localization is successful. And indeed, $n = 400$ is large enough to make the varPS without weight localization collapse when the observation noise is large ($\mathbf{R} = \mathbf{I}$). Weight-localization prevents this collapse, and yields results comparable to the variational methods, but MSE is slightly larger for varPS. The equally weighted varPS is also effective and leads to MSE as small as those obtained by the variational methods. As before, we note that the localized PF performs poorly when the observation errors are small ($\mathbf{R} = 0.1\,\mathbf{I}$), and that the local PF causes larger MSE than EnKF. Moreover, varPS and the variational methods lead to smaller MSE than EnKF or localized PF, and the variational methods give the smallest MSE.

5.3 | Results for a 2,000-dimensional problem

We repeat some of our computations on a problem of dimension $n = 2,000$. Here we do not tune localization/inflation for the various algorithms we consider, but re-use the localization and inflation parameters we obtained when tuning the $n = 400$ dimensional problem. All algorithms use $N_e = 40$

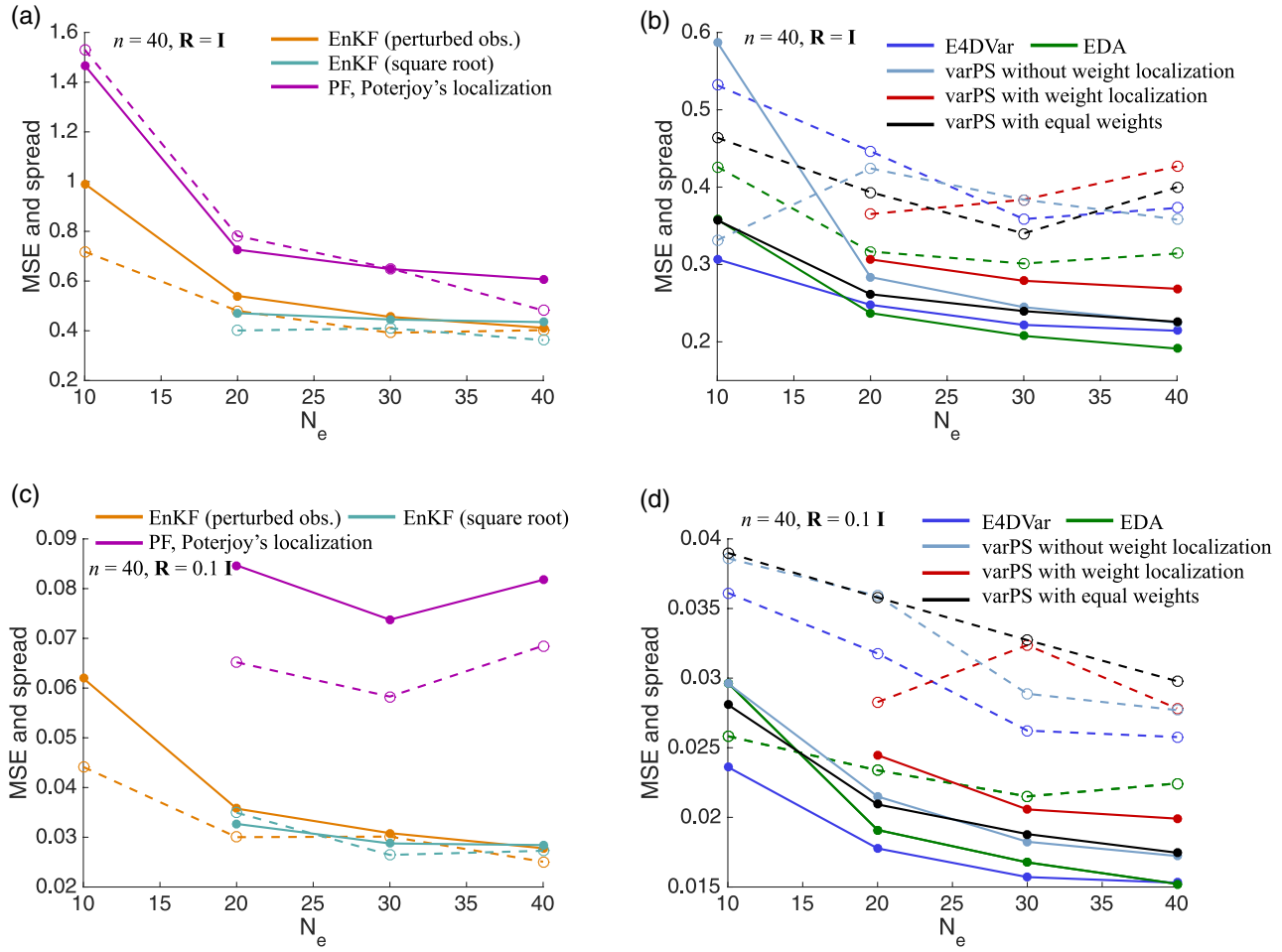

FIGURE 5 MSE (solid lines) and spread (dashed lines) as a function of the ensemble size for several data assimilation algorithms and a Lorenz model of dimension $n = 40$. (a, c) Localized PF and EnKFs, (b, d) variational methods and varPS, for (a, b) $\mathbf{R} = \mathbf{I}$, and (c, d) $\mathbf{R} = 0.1\,\mathbf{I}$ [Colour figure can be viewed at wileyonlinelibrary.com]

ensemble members and the observation-error covariance is $\mathbf{R} = \mathbf{I}$. Our results are summarized in Table 1.

The localized standard PF struggles in this high-dimensional case and gives larger MSE, but keeps the spread comparable to that of EnKF. The varPS *without* weight-localization collapsed in this problem. Weight-localization prevents this collapse and leads to MSE and spread as in the $n = 40$ or $n = 400$ dimensional problems. As before, we also obtain small MSE by the equally weighted varPS. Moreover, as before, varPS with weight-localization or the equally weighted varPS produce MSE and spread comparable to what we obtain by E4D-Var, and "beats" the EnKF, which yields larger MSE and spread. This numerical experiment further suggests that varPS (with weight-localization or with equal weights) can perform well with ensemble sizes that are significantly smaller than the dimension.

5.4 | Discussion of numerical experiments

We draw the following conclusions from our numerical experiments.

1. We reminds readers that localization of the data assimilation algorithms exploits banded problem structure of L96. This is the reason why localized algorithms perform identically on L96 problems of dimensions $n = 40$, $n = 400$, and $n = 2,000$. Unlocalized methods do not exploit (or know of) the banded problem structure, and this is what makes the unlocalized algorithms fail.
2. Localized particle methods with small ensemble sizes do not collapse on any of the problems we considered, and yield small MSE and comparable spread. The standard PF yields larger MSE than EnKF. The varPS (in its various implementations) yields smaller MSE than EnKF, but slightly larger MSE than E4D-Var or EDA.
3. The varPS can perform robustly with small ensembles and *without* localization in problems where other unlocalized particle filters collapse. This property follows from the varPS exploiting Gaussian assumptions for posterior densities. While the varPS without weight-localization would work flawlessly in linear/Gaussian problems, even small deviations from linearity/Gaussianity, or, equivalently, small errors in approximating covariances will lead to the collapse of the varPS if the dimension becomes large (as

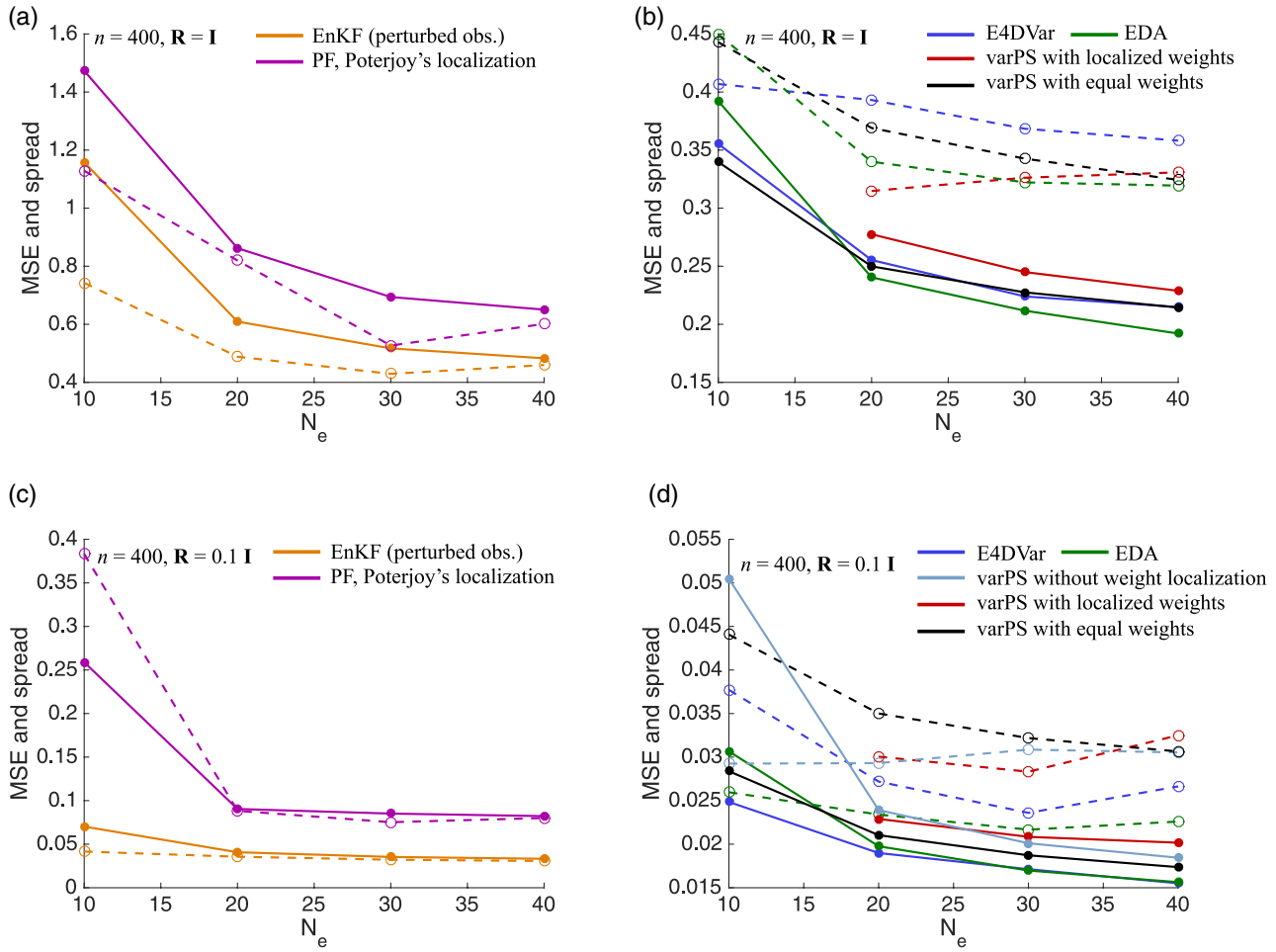

FIGURE 6 As Figure 5, but for a Lorenz model of dimension $n = 400$ [Colour figure can be viewed at wileyonlinelibrary.com]

TABLE 1 Results for a 2,000-dimensional L96 problem

Algorithm	MSE	Spread
EnKF (stochastic)	0.54	0.50
PF, Poterjoy's localization	0.81	0.51
E4D-Var	0.27	0.37
varPS *with* weight localization	0.27	0.32
varPS *without* weight localization	13.5	0.09
varPS with *equal* weights (IEnKF)	0.28	0.36

is the case for the L96 problem of dimension $n = 400$ and $n = 2,000$).

4. Weight-localization makes the varPS applicable to problems of high dimension ($n = 2,000$) where the varPS without weight-localization collapses. The varPS with weight-localization thus exploits linear *and* sparse/banded problem structure. The fact that we obtain similar results with varPS and ensemble formulations of 4D-Var suggests that the weight-localization strategy is applicable in the sense that the additional errors due to localization are small.
5. The equally weighted varPS performs similarly to or better than the weighted varPS (even when the weights are localized). Thus, one should be careful when using particle methods and localized weights in near-Gaussian problems: using localized weights may not lead to significant advantages over unweighted or linearized solutions, especially when the ensemble size is small (in which case sampling error is large, perhaps dominant over errors due to Gaussian approximations). However, in "more" nonlinear/non-Gaussian problems, this results cannot be expected to hold. Our numerical experiments do not allow us to draw conclusions about strongly nonlinear/non-Gaussian problems because we focused on a nearly Gaussian problem class in our theory, algorithm design, and numerical experiments.

6 | SUMMARY AND CONCLUSIONS

We have benchmarked localized PFs against EnKF on diagonal linear problems for which an "optimal" localization strategy is available. We found that localized PFs cannot reach the performance of EnKF with small ensemble sizes on these linear test problems. Motivated by our benchmarks, we revisited a variational particle smoother (varPS) that exploits Gaussian problem structure by merging 4D-Var methods with the particle approach. We studied how weight localization can prevent the collapse of varPS and what role the weights play

in mildly nonlinear/non-Gaussian problems. We found that the performance of varPS is comparable to that of EnKF on our linear test problems, and discussed connections of varPS with ensemble formulations of 4D-Var, in particular with the IEnKF and the IEnKS.

We obtained good results in simple nonlinear benchmark problems and also found that the performance of varPS is comparable to ensemble formulations of 4D-Var in numerical experiments with a L96 model of dimension $n = 40$, $n = 400$, and $n = 2,000$. Since ensemble formulations of 4D-Var and the varPS yield comparable performance in this mildly nonlinear problem, computational cost may ultimately decide which algorithm should be used. Both varPS and E4D-Var require an optimization but varPS does not require running an EnKF in addition to a variational system. The varPS may be more efficient also than EDA because it only requires one optimization, rather than one optimization per ensemble member. The computational cost of varPS and IEnKF/IEnKS is comparable, since IEnKF/IEnKS and varPS essentially only differ by the use, or non-use, of weights. Additional improvements due to the weights may determine which of these methods is most applicable.

We discussed in detail how the varPS collapses in high-dimensional problems and show that weight-localization can prevent this collapse. The varPS with weight localization exploits linear as well as sparse/banded problem structure, which may be important for solving NWP problems with small ensemble sizes. We recall that even a localized particle method may lead to poor results, or may collapse, when the number of observations is large. Our numerical experiments or theory do not allow us to draw conclusions about the applicability of varPS in practice, because we have not analyzed what happens when the number of observations is large (larger than the system dimension). Our numerical experiments suggest that an equally weighted varPS, which is equivalent to an implementation of IEnKF/IEnKS, can be effective if the nonlinearity is not too strong. In this case, localized weights may not lead to significant improvements over unweighted or linearized solutions. In strongly nonlinear problems, varPS may lead to improvements compared to varPS with equal weights, or IEnKF/IEnKS, but the required ensemble size is likely to increase as well. We hope to investigate such problems in future work; in particular we wish to investigate how the required ensemble size may scale with the degree of nonlinearity.

ACKNOWLEDGEMENTS

M. Morzfeld gratefully acknowledges support by the Office of Naval Research (grant number N00173-17-2-C003), the National Science Foundation under grant DMS-1619630, and by the Alfred P. Sloan Foundation. D. Hodyss gratefully acknowledges support from the Office of Naval Research PE-0601153N. J. Poterjoy acknowledges support from a National Research Council Research Associateship Program fellowship.

APPENDICES

A: BENCHMARKING LOCALIZED PARTICLE FILTERS FOR LINEAR, DIAGONAL, STOCHASTIC PROBLEMS

We consider data assimilation problems with a stochastic model, defined by

$$\mathbf{x}_k = \mathbf{f}_k(\mathbf{x}_{k-1}) + \eta_k,$$
$$\mathbf{y}_k = \mathbf{h}_k(\mathbf{x}_k) + \varepsilon_k,$$

where η_k are iid Gaussian random variables with means $E[\eta_k] = \mathbf{0}$, and covariance matrices $\mathbf{Q_k} = E\left[\eta_k \eta_k^{\mathrm{T}}\right]$; all other definitions are as in Equations 1–2. The posterior distribution typically used in particle filtering for such problems is

$$p(\mathbf{x}_{0:k}|\mathbf{y}_{1:k}) \propto p(\mathbf{x}_{0:k-1}|\mathbf{y}_{1:k-1}) p(\mathbf{x}_k|\mathbf{x}_{k-1}) p(\mathbf{y}_k|\mathbf{x}_k), \quad \text{(A1)}$$

(e.g. Doucet *et al.*, 2001). Note that this posterior distribution is defined over *trajectories* $\mathbf{x}_{0:k}$, rather than a state at a given time. Moreover, the factorization of the posterior distribution in Equation A1 implies that one can update the posterior distribution at time $k-1$, $p(\mathbf{x}_{0:k-1}|\mathbf{y}_{1:k-1})$, to time k, $p(\mathbf{x}_{0:k}|\mathbf{y}_{1:k})$, by sampling $p(\mathbf{x}_k|\mathbf{x}_{k-1})p(\mathbf{y}_k|\mathbf{x}_k)$. This can be done sequentially in time by using proposal distributions of the form

$$q(\mathbf{x}_{0:k}; \mathbf{y}_{1:k}) \propto q(\mathbf{x}_0) \prod_{j=1}^{k} q_j(\mathbf{x}_j; \mathbf{y}_{1:j}, \mathbf{x}_{1:j-1}). \quad \text{(A2)}$$

At each step in time, particle filtering thus amounts to importance sampling of the "update term", $p(\mathbf{x}_k|\mathbf{x}_{k-1})p(\mathbf{y}_k|\mathbf{x}_k)$, using the proposal distribution, $q_k(\mathbf{x}_k; \mathbf{y}_{1:k}, \mathbf{x}_{1:k-1})$. The weights are the ratio of posterior and proposal distributions

$$w_k \propto \frac{p(\mathbf{x}_{0:k}|\mathbf{y}_{1:k})}{q(\mathbf{x}_{0:k}; \mathbf{y}_{1:k})} \propto w_{k-1} \frac{p(\mathbf{x}_k|\mathbf{x}_{k-1}) p(\mathbf{y}_k|\mathbf{x}_k)}{q_k(\mathbf{x}_k; \mathbf{y}_{1:k}, \mathbf{x}_{1:k-1})}.$$

It is possible to evaluate these weights (without approximations) because the update term can be evaluated up to a multiplicative constant.

The "standard" particle filter (SPF) for stochastic problems uses the stochastic model to define the proposal distribution and weights

$$q_k(\mathbf{x}_k; \mathbf{y}_{1:k}, \mathbf{x}_{1:k-1}) = p(\mathbf{x}_k|\mathbf{x}_{k-1}), \qquad w_k \propto w_{k-1} p(\mathbf{y}_k|\mathbf{x}_k).$$

The "optimal particle filter" (OPF; e.g. Arulampalam *et al.*, 2002; Liu and Chen, 1995; Snyder *et al.*, 2015), uses a proposal distribution and weights given by

$$q_k(\mathbf{x}_k; \mathbf{y}_{1:k}, \mathbf{x}_{1:k-1}) = p(\mathbf{x}_k|\mathbf{x}_{k-1}, \mathbf{y}_k), \qquad w_k \propto w_{k-1} p(\mathbf{y}_k|\mathbf{x}_{k-1}).$$

It was shown by Snyder *et al.* (2015) that this choice of q is "optimal" in the sense that the variance of the weights is minimized over proposal distributions of the form (A2).

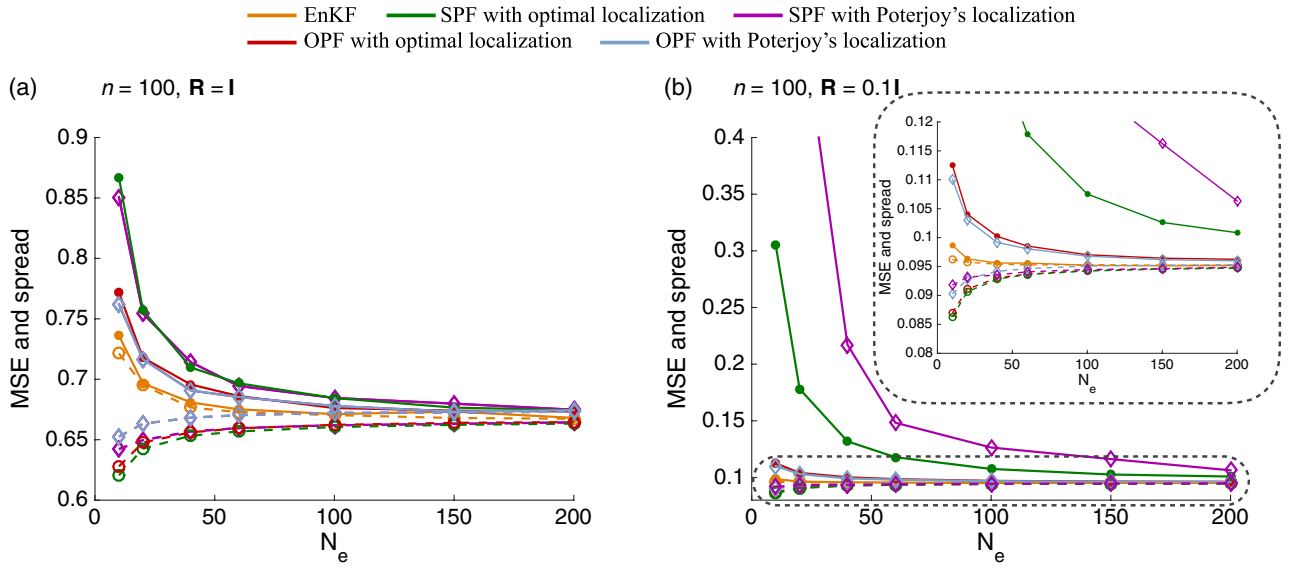

FIGURE A1 MSE (solid lines) and spread (dashed lines) as a function of the ensemble size for localized EnKF and particle filters: (a) larger noise in the observation with $\mathbf{R} = \mathbf{I}$, and (b) smaller noise in the observation with $\mathbf{R} = 0.1\,\mathbf{I}$ [Colour figure can be viewed at wileyonlinelibrary.com]

It is important to realize that "optimality" here is over the class of PFs defined by Equation A2. There are other PFs, e.g. equivalent weight PFs Ades and van Leeuwen (2013) and van Leeuwen (2010) that do not belong to this class and our results do not apply to these filters.

We now apply the SPF and OPF, as well as an EnKF to a stochastic version of the linear, diagonal test problem in Equations (14)–(15):

$$\mathbf{x}_k = \mathbf{x}_{k-1} + \eta_k,$$
$$\mathbf{y}_k = \mathbf{x}_k + \varepsilon_k.$$

Here $\mathbf{Q}_k = \mathbf{R}_k = \mathbf{I}$. We localize the PFs by decoupling and compare this optimal localization to the localization method of Poterjoy. The EnKF (stochastic) is localized by the identity matrix. As in section 3.4, we fix the dimension $n = 100$ and consider the cases $\mathbf{R}_k = \mathbf{I}$ and $\mathbf{R}_k = 0.1\,\mathbf{I}$. Our results are shown in Figure A1.

We note qualitatively similar results as in our experiments with a deterministic model in section 3.4: with small ensemble sizes, EnKF yields smaller MSE than both PFs, which underestimate the spread when the ensemble size is small.

When the observations are "more accurate", $\mathbf{R} = 0.1\mathbf{I}$, the choice of the proposal distribution becomes important. We note that with the larger observation noise ($\mathbf{R} = \mathbf{I}$), not much is gained by using OPF over SPF, however when the noise is small ($\mathbf{R} = 0.1\,\mathbf{I}$), then OPF yields smaller MSE at smaller ensemble sizes than SPF. Nonetheless, no PF – not even the optimal particle filter with "optimal" localization scheme – can come close to performance of EnKF with small ensembles (significantly smaller than the dimension, $N_e \ll n_x$).

In addition, we note that Poterjoy's localization method gives results almost identical to what can be achieved by the "idealized" localization when the optimal proposal is used, but leads to large MSE when the standard proposal is used (Figure A1b). Small observation-error covariances have been one shortcoming of the local PF (with standard proposal) in the past, e.g. Lee and Majda (2016), and our computations with linear models suggest that they can be overcome by using optimal rather than standard proposals.

B: DERIVATION OF EQUATION 19

We derive Equation 19. For a diagonal, nonlinear "small noise" problem, the cost function can be written as

$$\mathcal{J}(\mathbf{x}_k) = \sum_{i=1}^{n} \mathcal{J}_i(x^i),$$
$$\mathcal{J}_i(x^i) = \frac{1}{2}\mathcal{J}_{x^i,x^i} \cdot (x^i - x^{i,*})^2 + \gamma\left[\frac{1}{6}\mathcal{J}_{x^i,x^i,x^i} \cdot (x^i - x^{i,*})^3\right] + \text{HOT}.$$

Here x^i are the n components of $\mathbf{x}_k$, $x^{i,*}$ are the n components of the posterior mode $\mathbf{x}_k^*$, and $\mathcal{J}_{x^i,x^i}$ $\mathcal{J}_{x^i,x^i,x^i}$ are the second and third derivatives of the 4D-Var cost function evaluated at the posterior mode. Note that the approximate posterior distribution (Equation 5) is

$$\widehat{p}(\mathbf{x}_k|\mathbf{y}_{1:k}) \propto \exp\left[-\mathcal{J}(\mathbf{x}_k)\right] \propto \prod_{i=1}^{n} \exp\left[-\mathcal{J}_i(x^i)\right]$$

We define

$$\mathcal{J}^0(\mathbf{x}_k) = \sum_{i=1}^{n} \mathcal{J}_i^0(x^i), \qquad \mathcal{J}_i^0(x^i) = \frac{1}{2}\mathcal{J}_{x^i,x^i} \cdot (x^i - x^{i,*})^2$$

so that the proposal distribution of varPS can be written as

$$q(\mathbf{x}_k) \propto \exp\left(-\mathcal{J}^0(\mathbf{x}_k)\right) \propto \prod_{i=1}^{n} \exp\left[-\mathcal{J}_i^0(x^i)\right].$$

The weights are the ratio of posterior and proposal distribution

$$w \propto \exp\left\{-\left[\mathcal{J}(\mathbf{x}_k) - \mathcal{J}^0(\mathbf{x}_k)\right]\right\} \propto \prod_{i=1}^{n} \exp\left\{-\left[\mathcal{J}_i(x^i) - \mathcal{J}_i^0(x^i)\right]\right\},$$

and we compute

$$E[w^2] = \int \dots \int \prod_{i=1}^{n} \exp\left\{-2\left[\mathcal{J}_i(x^i) - \mathcal{J}_i^0(x^i)\right]\right\} \frac{\exp\left[-J_i^0(x^i)\right]}{\sqrt{2\pi/\mathcal{J}_{x^i,x^i}}} \mathrm{d}x^1 \dots \mathrm{d}x^n,$$

$$E[w] = \int \dots \int \prod_{i=1}^{n} \exp\left\{-\left[\mathcal{J}_i(x^i) - \mathcal{J}_i^0(x^i)\right]\right\} \frac{\exp\left[-J_i^0(x^i)\right]}{\sqrt{2\pi/\mathcal{J}_{x^i,x^i}}} \mathrm{d}x^1 \dots \mathrm{d}x^n.$$

Under our assumptions of identically and independently distributed variables x^i, we have that

$$E_1 = \int \exp\left\{-2\left[\mathcal{J}_i(x^i) - \mathcal{J}_i^0(x^i)\right]\right\} \frac{\exp\left[-J_i^0(x^i)\right]}{\sqrt{2\pi/\mathcal{J}_{x^i,x^i}}} \mathrm{d}x^i,$$

$$E_2 = \int \exp\left\{-\left[\mathcal{J}_i(x^i) - \mathcal{J}_i^0(x^i)\right]\right\} \frac{\exp\left[-J_i^0(x^i)\right]}{\sqrt{2\pi/\mathcal{J}_{x^i,x^i}}} \mathrm{d}x^i,$$

are independent of the variable index i, so that

$$E[w^2] = E_1 \cdot E_1 \cdot \dots \cdot E_1 = (E_1)^n,$$
$$E[w] = E_2 \cdot E_2 \cdot \dots \cdot E_2 = (E_2)^n,$$

which yields Equation 19.

REFERENCES

Ades, M. and van Leeuwen, P. (2013) An exploration of the equivalent weights particle filter. *Quarterly Journal of the Royal Meteorological Society*, 139, 820–840.

Anderson, J.L. (2007) Exploring the need for localization in ensemble data assimilation using a hierarchical ensemble filter. *Physica D: Nonlinear Phenomena*, 230, 99–111.

Anderson, J.L. (2012) Localization and sampling error correction in ensemble Kalman filter data assimilation. *Monthly Weather Review*, 140, 2359–2371. https://doi.org/10.1175/MWR-D-11-00013.1.

Arulampalam, M., Maskell, S., Gordon, N. and Clapp, T. (2002) A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking. *IEEE Transactions on Signal Processing*, 50, 174–188.

Atkins, E., Morzfeld, M. and Chorin, A. (2013) Implicit particle methods and their connection with variational data assimilation. *Monthly Weather Review*, 141, 1786–1803.

Auligné, T., Ménétrier, B., Lorenc, A.C. and Buehner, M. (2016) Ensemble–variational integrated localized data assimilation. *Monthly Weather Review*, 144, 3677–3696.

Bardsley, J., Solonen, A., Haario, H. and Laine, M. (2014) Randomize-then-optimize: a method for sampling from posterior distributions in nonlinear inverse problems. *SIAM Journal on Scientific Computing*, 36, A1895–A1910.

Bengtsson, T., Bickel, P. and Li, B. (2008) Curse of dimensionality revisited: the collapse of importance sampling in very large scale systems. *IMS Collections: Probability and Statistics: Essays in Honor of David A. Freedman*, 2, 316–334.

Bengtsson, T., Snyder, C. and Nychka, D. (2003) Toward a nonlinear ensemble filter for high-dimensional systems. *Journal of Geophysical Research – Atmospheres*, 108(D24), 8775. https://doi.org/10.1029/2002JD002900.

Bennet, A., Leslie, L., Hagelberg, C. and Powers, P. (1993) A cyclone prediction using a barotropic model initialized by a general inverse method. *Monthly Weather Review*, 121, 1714–1728.

Bickel, P., Bengtsson, T. and Anderson, J. (2008) Sharp failure rates for the bootstrap particle filter in high dimensions. In: Clarke, B.S. and Ghosal, S. (Eds.) *Pushing the Limits of Contemporary Statistics Contributions in Honor of Jayanta K. Ghosh*. Bethesda, MD: Institute of Mathematical Statistics, pp. 318–329.

Bickel, P. and Levina, E. (2008) Regularized estimation of large covariance matrices. *The Annals of Statistics*, 36, 199–227.

Bocquet, M. (2016) Localization and the iterative ensemble Kalman smoother. *Quarterly Journal of the Royal Meteorological Society*, 142, 1075–1089.

Bocquet, M., Pires, C. and Wu, L. (2010) Beyond Gaussian statistical modeling in geophysical data assimilation. *Monthly Weather Review*, 138, 2997–3023.

Bocquet, M. and Sakov, P. (2013) Joint state and parameter estimation with an iterative ensemble Kalman smoother. *Nonlinear Processes in Geophysics*, 20, 803–818.

Bocquet, M. and Sakov, P. (2014) An iterative ensemble Kalman smoother. *Quarterly Journal of the Royal Meteorological Society*, 140, 1521–1535.

Bonavita, M., Isaksen, L. and Hólm, E.V. (2012) On the use of EDA background-error variances in the ECMWF 4D-Var. *Quarterly Journal of the Royal Meteorological Society*, 138, 1540–1559.

Buehner, M. (2005) Ensemble-derived stationary and flow-dependent background-error covariances: evaluation in a quasi-operational NWP setting. *Quarterly Journal of the Royal Meteorological Society*, 131, 1013–1043.

Chorin, A. and Morzfeld, M. (2013) Conditions for successful data assimilation. *Journal of Geophysical Research – Atmospheres*, 118, 11522–11533.

Chorin, A., Morzfeld, M. and Tu, X. (2010) Implicit particle filters for data assimilation. *Communications in Applied Mathematics and Computational Science*, 5, 221–240.

Chorin, A. and Tu, X. (2009) Implicit sampling for particle filters. *Proceedings of the National Academy of Sciences of the United States of America*, 106, 17249–17254.

Doucet, A., de Freitas, N. and Gordon, N. (Eds.) (2001) *Sequential Monte Carlo Methods in Practice*. Berlin: Springer.

Doucet, A., Godsill, S. and Andrieu, C. (2000) On sequential Monte Carlo sampling methods for Bayesian filtering. *Statistics and Computing*, 10, 197–208.

Evensen, G. (2006) *Data Assimilation: The Ensemble Kalman Filter*. Berlin: Springer.

Fournier, A., Hulot, G., Jault, D., Kuang, W., Tangborn, W., Gillet, N., Canet, E., Aubert, J. and Lhuillier, F. (2010) An introduction to data assimilation and predictability in geomagnetism. *Space Science Review*, 155, 247–291.

Gaspari, G. and Cohn, S. (1999) Construction of correlation functions in two and three dimensions. *Quarterly Journal of the Royal Meteorological Society*, 125, 723–757.

Goodman, J., Lin, K.K. and Morzfeld, M. (2015) Small-noise analysis and symmetrization of implicit Monte Carlo samplers. *Communications on Pure and Applied Mathematics*, 69, 1924–1951. https://doi.org/doi:10.1002/cpa.21592.

Gordon, N., Salmond, D. and Smith, A. (1993) Novel approach to nonlinear/non-Gaussian Bayesian state estimation. *IEEE Proceedings F Radar and Signal Processing*, 140, 107–113.

Greybush, S., Kalnay, E., Miyoshi, T., Ide, K. and Hunt, B. (2011) Balance and ensemble Kalman filter localization techniques. *Monthly Weather Review*, 139, 511–522.

Hamill, T.M., Whitaker, J. and Snyder, C. (2001) Distance-dependent filtering of background covariance estimates in an ensemble Kalman filter. *Monthly Weather Review*, 129, 2776–2790.

Hodyss, D., Bishop, C. and Morzfeld, M. (2016) To what extent is your data assimilation scheme designed to find the posterior mean, the posterior mode or something else? *Tellus A*, 68. https://doi.org/10.3402/tellusa.v68.30625.

Houtekamer, P.L. and Mitchell, H.L. (2001) A sequential ensemble Kalman filter for atmospheric data assimilation. *Monthly Weather Review*, 129, 123–136.

Houtekamer, P.L., Mitchell, H.L., Pellerin, G., Buehner, M., Charron, M., Spacek, L. and Hansen, B. (2005) Atmospheric data assimilation with an ensemble Kalman filter: results with real observations. *Monthly Weather Review*, 133, 604–620.

Kalnay, E. (2003) *Atmospheric Modeling, Data Assimilation, and Predictability*. Cambridge: Cambridge University Press.

Kepert, J. (2009) Covariance localization and balance in an ensemble Kalman filter. *Quarterly Journal of the Royal Meteorological Society*, 135, 1157–1176.

Klaas, M., de Freitas, N. and Doucet, A. (2005) Towards practical N2 Monte Carlo: the marginal particle filter. In: *Proceedings of the 21st Annual Conference on Uncertainty in Artificial Intelligence (UAI-05)*, Arlington, VA, pp. 308–315.

Kuhl, D., Rosmond, T., Bishop, C., McLay, J. and Baker, N. (2013) Comparison of hybrid ensemble/4DVar and 4DVar within the NAVDAS-AR data assimilation framework. *Monthly Weather Review*, 141, 2740–2758.

Le Gland, F., Monbet, V. and Tran, V.D. (2011) *Large Sample Asymptotics for the Ensemble Kalman Filter*. Oxford: Oxford University Press.

Lee, Y. and Majda, A. (2016) State estimation and prediction using clustered particle filters. *Proceedings of the National Academy of Sciences of the United States of America*, 113, 14609–14614.

van Leeuwen, P. (2003) Nonlinear ensemble data assimilation for the ocean. In: *Seminar on Recent Developments in Data Assimilation for Atmosphere and Ocean*, 8–12 September 2003, Reading, ECMWF, pp. 265–286.

van Leeuwen, P. (2009) Particle filtering in geophysical systems. *Monthly Weather Review*, 137, 4089–4144.

van Leeuwen, P. (2010) Nonlinear data assimilation in geosciences: an extremely efficient particle filter. *Quarterly Journal of the Royal Meteorological Society*, 136(653), 1991–1999.

Lei, J. and Bickel, P. (2011) A moment matching ensemble filter for nonlinear non-Gaussian data assimilation. *Monthly Weather Review*, 139, 3964–3973.

Liu, C., Xiao, Q. and Wang, B. (2008) An ensemble-based four-dimensional variational data assimilation scheme. Part I: technical formulation and preliminary test. *Monthly Weather Review*, 136, 3363–3373.

Liu, J. and Chen, R. (1995) Blind deconvolution via sequential imputations. *Journal of the American Statistical Association*, 90, 567–576.

Liu, Y., Haussaire, J.M., Bocquet, M., Roustan, Y., Saunier, O. and Mathieu, A. (2017) Uncertainty quantification of pollutant source retrieval: comparison of bayesian methods with application to the Chernobyl and Fukushima-Daiichi accidental releases of radionuclides. *Quarterly Journal of the Royal Meteorological Society*, 143, 2886–2901.

Lorenc, A.C. (2003) The potential of the ensemble Kalman filter for NWP – A comparison with 4D-Var. *Quarterly Journal of the Royal Meteorological Society*, 129, 3183–3203.

Lorenc, A.C., Bowler, N., Clayton, A., Pring, S. and Fairbairn, D. (2015) Comparison of hybrid-4DEnVar and hybrid-4DVar data assimilation methods for global NWP. *Monthly Weather Review*, 143, 212–229.

Lorenz, E.N. (1996) Predictability: a problem partly solved. In: *Proceedings of Seminar on Predictability*, Vol. 1, Reading, ECMWF, pp. 1–18.

Mitchell, H. and Houtekamer, P. (2002) Ensemble size, balance, and model-error representation in an ensemble Kalman filter. *Monthly Weather Review*, 130, 2791–2808.

Morzfeld, M., Fournier, A. and Hulot, G. (2017) Coarse predictions of dipole reversals by low-dimensional modeling and data assimilation. *Physics of the Earth and Planetary Interiors*, 262, 8–27.

Morzfeld, M., Hodyss, D. and Snyder, C. (2017) What the collapse of the ensemble Kalman filter tells us about particle filters. *Tellus A*, 69. https://doi.org/10.1080/16000870.2017.1283809.

Morzfeld, M., Tu, X., Atkins, E. and Chorin, A. (2012) A random map implementation of implicit filters. *Journal of Computational Physics*, 231, 2049–2066.

Morzfeld, M., Tu, X., Wilkening, J. and Chorin, A. (2015) Parameter estimation by implicit sampling. *Communications in Applied Mathematics and Computational Science*, 10, 205–225.

Oliver, D., Reynolds, A. and Liu, N. (2008) *Inverse Theory for Petroleum Reservoir Characterization and History Matching*. Cambridge: Cambridge University Press.

Papadakis, N., Memin, E., Cuzol, A. and Gengembre, N. (2010) Data assimilation with the weighted ensemble Kalman filter. *Tellus A*, 62, 673–697.

Penny, S. and Miyoshi, T. (2015) A local particle filter for high-dimensional geophysical systems. *Nonlinear Processes in Geophysics*, 2, 1631–1658.

Poterjoy, J. (2015) A localized particle filter for high-dimensional nonlinear systems. *Monthly Weather Review*, 144, 59–76.

Poterjoy, J. and Anderson, J. (2016) Efficient assimilation of simulated observations in a high-dimensional geophysical system using a localized particle filter. *Monthly Weather Review*, 144, 2007–2020.

Poterjoy, J., Sobash, R. and Anderson, J. (2017) Convective-scale data assimilation for the weather research and forecasting model using the local particle filter. *Monthly Weather Review*, 145, 1897–1918.

Poterjoy, J. and Zhang, F. (2015) Systematic comparison of four-dimensional data assimilation methods with and without the tangent linear model using hybrid background error covariance: E4DVar versus 4DEnVar. *Monthly Weather Review*, 143, 1601–1621.

Rebeschini, P. and van Handel, R. (2015) Can local particle filters beat the curse of dimensionality? *The Annals of Applied Probability*, 25, 2809–2866.

Reich, S. (2013) A nonparametric ensemble transform method for Bayesian inference. *Monthly Weather Review*, 35, 1337–1367.

Robert, S., Leuenberger, D. and Künsch, H.R. (2017) A local ensemble transform Kalman particle filter for convective-scale data assimilationn. *Quarterly Journal of the Royal Meteorological Society*. https://doi.org/10.1002/qj.3116.

Sakov, P., Oliver, D.S. and Bertino, L. (2012) An iterative EnKF for strongly nonlinear systems. *Monthly Weather Review*, 140, 1988–2004.

Snyder, C. (2011) Particle filters, the 'optimal' proposal and high-dimensional systems. In: *Proceedings of Seminar on Data Assimilation for Atmosphere and Ocean*, Reading, ECMWF.

Snyder, C., Bengtsson, T., Bickel, P. and Anderson, J. (2008) Obstacles to high-dimensional particle filtering. *Monthly Weather Review*, 136, 4629–4640.

Snyder, C., Bengtsson, T. and Morzfeld, M. (2015) Performance bounds for particle filters using the optimal proposal. *Monthly Weather Review*, 143, 4750–4761.

Talagrand, O. and Courtier, P. (1987) Variational assimilation of meteorological observations with the adjoint vorticity equation. I: theory. *Quarterly Journal of the Royal Meteorological Society*, 113, 1311–1328.

Tippet, M., Anderson, J., Bishop, C., Hamill, T. and Whitaker, J. (2003) Ensemble square root filters. *Monthly Weather Review*, 131, 1485–1490.

Tödter, J. and Ahrens, B. (2015) A second-order exact ensemble square root filter for nonlinear data assimilation. *Monthly Weather Review*, 143, 1337–1367.

Vanden-Eijnden, E. and Weare, J. (2012) Data assimilation in the low noise regime with application to the Kuroshio. *Monthly Weather Review*, 141, 1822–1841.

Weare, J. (2009) Particle filtering with path sampling and an application to a bimodal ocean current model. *Journal of Computational Physics*, 228, 4312–4331.

Weir, B., Miller, R.N. and Spitz, Y.H. (2013) A potential implicit particle method for high-dimensional systems. *Nonlinear Processes in Geophysics*, 20, 1047–1060.

Zaritskii, V. and Shimelevich, L. (1975) Monte Carlo technique in problems of optimal data processing. *Automation and Remote Control*, 12, 95–103.

Zupanski, M. (2004) Maximum likelihood ensemble filter: theoretical aspects. *Monthly Weather Review*, 133, 1710–1726.

How to cite this article: Morzfeld M, Hodyss D, Poterjoy J. Variational particle smoothers and their localization. *Q J R Meteorol Soc.* 2018;144:806–825. https://doi.org/10.1002/qj.3256