View metadata, citation and similar papers at core.ac.uk
brought to you by CORE
provided by Woods Hole Open Access Server

A Hybrid Particle–Ensemble Kalman Filter for Lagrangian Data Assimilation

LAURA SLIVINSKI*

Brown University, Providence, Rhode Island

ELAINE SPILLER

Marquette University, Milwaukee, Wisconsin

AMIT APTE

International Centre for Theoretical Sciences, Bangalore, India

BJÖRN SANDSTEDE

Brown University, Providence, Rhode Island

(Manuscript received 6 February 2014, in final form 15 September 2014)

ABSTRACT

Lagrangian measurements from passive ocean instruments provide a useful source of data for estimating and forecasting the ocean's state (velocity field, salinity field, etc.). However, trajectories from these instruments are often highly nonlinear, leading to difficulties with widely used data assimilation algorithms such as the ensemble Kalman filter (EnKF). Additionally, the velocity field is often modeled as a high-dimensional variable, which precludes the use of more accurate methods such as the particle filter (PF). Here, a hybrid particle–ensemble Kalman filter is developed that applies the EnKF update to the potentially high-dimensional velocity variables, and the PF update to the relatively low-dimensional, highly nonlinear drifter position variable. This algorithm is tested with twin experiments on the linear shallow water equations. In experiments with infrequent observations, the hybrid filter consistently outperformed the EnKF, both by better capturing the Bayesian posterior and by better tracking the truth.

1. Introduction

Lagrangian ocean instruments—drifters, floats, and gliders, which are advected by velocity fields while taking measurements—provide a significant and important source of the data collected on our oceans. However, the Lagrangian nature of the data makes the assimilation of it into models a formidable task (Kuznetsov et al. 2003; Salman et al. 2006; Mariano et al. 2002). *Assimilation* is a blending of data and a physical model with the aim of both accurately determining the state of the system and reflecting inherent uncertainties in that estimation due to the physical model and/or the available data. Assimilation methods can be broadly categorized either as sequential filtering or nonsequential smoothing methods. We will be focusing on filtering methods in this work, since this is quite natural within the context of using Lagrangian data for forecasting the state of the ocean.

Two primary challenges hindering sequential assimilation of data collected from Lagrangian instruments into ocean models are 1) the inherent nonlinearity of the Lagrangian paths and 2) the high-dimensionality of realistic ocean models. Different assimilation methods tackle these two aspects in different manners. Particle filtering (PF) methods are well suited for nonlinear problems but face difficulties with high-dimensional systems (cf. Bickel et al. 2008; Bengtsson et al. 2008; Snyder et al. 2008), although recent work has shown promise in addressing these issues (Morzfeld et al. 2012; Ades and van Leeuwen 2013). One commonly used method in the earth sciences

* Current affiliation: Woods Hole Oceanographic Institution, Woods Hole, Massachusetts.

Corresponding author address: Laura Slivinski, Woods Hole Oceanographic Institution, MS 21, Woods Hole, MA 02543.
E-mail: lslivinski@whoi.edu

DOI: 10.1175/MWR-D-14-00051.1

© 2015 American Meteorological Society

is the ensemble Kalman filter (EnKF), which can be modified to work well for high-dimensional systems (Houtekamer and Mitchell 1998; Hamill et al. 2001; Anderson, 2007; Hunt et al. 2007) but fails for problems with a high degree of nonlinearity (Apte et al. 2008).

In this paper, we propose a hybrid particle–ensemble Kalman filter method that overcomes both of these challenges. We consider two specific aspects of the Lagrangian data assimilation problem, namely (i) the low-dimensionality of the highly nonlinear Lagrangian observations and (ii) the less severe nonlinearity of the high-dimensional Eulerian model of the velocity flow. These two aspects and the discussion of the complementary challenges tackled by the PF and EnKF approaches motivate the primary idea behind our strategy, which consists of using a particle filter in the low-dimensional, highly nonlinear Lagrangian coordinate variables and an ensemble Kalman filter in the high-dimensional, relatively linear flow variables.

Previous work on hybrid schemes includes the combined ensemble Kalman–particle filter of Frei and Künsch (2012), mixture ensemble Kalman filters (Frei and Künsch 2013b), the ensemble Kalman–particle filter (EnKPF; Frei and Künsch 2013a), the weighted ensemble Kalman filter (WEnKF; Papadakis et al. 2010), the hybrid grid–particle filter (Salman 2008a,b), and many hybrid ensemble–variational schemes, such as that of Hamill and Snyder (2000). The EnKPF algorithm of Frei and Künsch provides a continuous interpolation between the EnKF and the particle filter, depending on the interpolation parameter. The WEnKF of Papadakis and Mémin (Papadakis et al. 2010) is primarily a particle filter in which the proposal distribution is motivated by the ensemble Kalman filter.

Doucet et al. (2000a) introduced the Rao–Blackwellised particle filter (RBPF), which involves splitting the state space into a Gaussian component and a non-Gaussian component. However, unlike the filter presented here, the RBPF relies on a decomposition in which the non-Gaussian variable is not coupled dynamically to the Gaussian variable. Bengtsson et al. (2003) also discuss an ensemble mixture filter that consists of a mixture of Gaussian ensembles, in which each component is updated with an EnKF analysis step and the component's associated weight is calculated using a Bayesian update. This filter requires choosing various parameters, including the number of Gaussian mixture components. In contrast, the hybrid filter presented here does not make any mixture assumptions; it consists of estimating the distribution on the drifters using a particle filter, and thus we do not need a priori knowledge of the necessary number of Gaussian components. Within the context of Lagrangian data assimilation, the idea of splitting the state space into two different parts (the Eulerian flow variables and the Lagrangian position variables) and of using different methods for the two parts first appeared in the work of Salman (2008a,b). The main differences between the work of Salman and the present paper are described in section 3a.

The remainder of the paper is organized as follows: in section 2, we review the traditional ensemble data assimilation methods of the particle filter and the ensemble Kalman filter. In section 3, we describe the algorithm for the hybrid particle–ensemble Kalman filter. In section 4, we provide numerical results of the hybrid filter applied to the linear shallow water equations. Section 5 provides a discussion and future directions.

2. Brief review of assimilation and filters

Lagrangian data assimilation (LaDA) is concerned with estimating the Eulerian flow field of a system (say, currents in the ocean) given Lagrangian observations of the positions of tracers (e.g., drifters or floaters). In most cases, the tracers can be approximated as being passive particles, whose motion is subject to the flow. In this case, the dynamical system of interest is

$$\dot{\mathbf{x}}^F = f_1(\mathbf{x}^F) \quad \text{and} \tag{1a}$$

$$\dot{\mathbf{x}}^D = f_2(\mathbf{x}^F, \mathbf{x}^D), \tag{1b}$$

where $\mathbf{x}^F$ denotes the Eulerian velocity field (generally, this is a solution to a PDE, which is discretized over a grid) and $\mathbf{x}^D$ denotes the position of the drifter(s) (Ide et al. 2002). Define the augmented state vector $\mathbf{x} = [\mathbf{x}^F, \mathbf{x}^D]^{\mathrm{T}}$. The observations $\mathbf{y}$ are then

$$\mathbf{y} = \mathbf{x}^D + \boldsymbol{\epsilon} = \mathbf{H}\mathbf{x} + \boldsymbol{\epsilon}, \quad \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{R}), \tag{2}$$

where ϵ represents the Gaussian observation noise with covariance $\mathbf{R}$ and $\mathbf{H} = [\mathbf{0}\ \mathbf{I}]$ is the observation operator mapping $\mathbf{x}$ into the observation space.

As we will see below, f_1 can be linear or nonlinear, but the evolution of $\mathbf{x}^D$ will always be nonlinear, typically to a very high degree (Apte et al. 2008). In addition, the flow field $\mathbf{x}^F$ is usually high-dimensional. These two defining characteristics lead to complications with traditional data assimilation algorithms, including the ensemble Kalman filter and the particle filter. However, we aim to show that a hybrid particle–ensemble Kalman filter avoids many of these issues.

One initial challenge of assimilating data from Lagrangian instruments is that models of velocity fields are almost always gridded, but the data collected are not on grid points. One approach was to interpolate the Lagrangian paths into velocity estimates at neighboring grid points and then assimilate (Molcard et al. 2003). The other approach, as described above, Lagrangian data assimilation, appends equations for advection of an instrument's coordinates to the model equations for the velocity field. LaDA was developed and applied successfully in several theoretical and methodological studies over the last decade (Ide et al. 2002; Kuznetsov et al. 2003; Salman et al. 2006; Spiller et al. 2008; Vernieres et al. 2011). In some ways there is a trade-off between an observation operator that is not local in time and could be nonlinear, which is the case with interpolation, versus the strongly nonlinear dynamics of modeled advected paths where the paths are observed directly. The latter approach demands an assimilation strategy that can deal with strong nonlinearities. This further motivates the primary idea behind the proposed hybrid assimilation scheme: use a particle filter in low-dimensional, highly nonlinear instrument coordinate variables and an ensemble Kalman filter in the high-dimensional flow variables. We describe the basics of each filtering method below.

From a Bayesian perspective, the goal of any sequential filtering algorithm is to approximate the posterior distribution $p(\mathbf{x}\,|\,\mathbf{y})$ of a state of interest $\mathbf{x}$ at some time t_k given a prior distribution $p(\mathbf{x})$ on $\mathbf{x}$ at t_k, based on our current knowledge of the state, and a likelihood distribution $p(\mathbf{y}\,|\,\mathbf{x})$ of the observations $\mathbf{y}$ given the state $\mathbf{x}$, based on our knowledge of the noise in the observations. Bayes's rule gives the true posterior distribution in this case:

$$p(\mathbf{x}\,|\,\mathbf{y}) = \frac{p(\mathbf{y}\,|\,\mathbf{x})p(\mathbf{x})}{p(\mathbf{y})}. \qquad (3)$$

In both the PF and EnKF methods, the prior and posterior distributions are approximated by a weighted ensemble of the state $\mathbf{x}$ given by $\{\mathbf{x}_i, w_i\}_{i=1}^{N_e}$, which implies the distribution

$$\sum_{i=1}^{N_e} w_i\delta(\mathbf{x}-\mathbf{x}_i) \quad \text{with} \quad \sum_{i=1}^{N_e} w_i = 1, \qquad (4)$$

where $\delta(\mathbf{x}-\mathbf{x}_i)$ is the Dirac delta centered at $\mathbf{x}_i$. Usually for the EnKF, $w_i = 1/N_e$. Between the observation times, the weights are kept fixed and the state variables are evolved according to the dynamics of the system. The main difference between the two methods comes at the time when observations are available. This is described in the next two subsections. We will use the notation that $\{\mathbf{x}_i^f, w_i^f\}_{i=1}^{N_e}$ is the ensemble from the prior distribution $p(\mathbf{x})$ whereas $\{\mathbf{x}_i^a, w_i^a\}_{i=1}^{N_e}$ is from the posterior distribution $p(\mathbf{x}\,|\,\mathbf{y})$.

a. Particle filter

The posterior is approximated by updating the weights but leaving the particle positions fixed (i.e., $\mathbf{x}_i^a = \mathbf{x}_i^f := \mathbf{x}_i$). The updated weights are obtained by applying Bayes's rule to the weights as follows:

$$w_i^a = \frac{p(\mathbf{y}\,|\,\mathbf{x}_i)w_i^f}{\sum_{j=1}^{N_e} p(\mathbf{y}\,|\,\mathbf{x}_j)w_j^f}, \qquad (5)$$

that is, the updated weights are found by multiplying the likelihood of that particle by the previous weight and normalizing to sum to 1. This is the simplest implementation of the particle filter, also known as sequential importance sampling (Gordon et al. 1993; Doucet et al. 2000b).

Due to the finite nature of the approximation and the recursive updating of the weights, sequentially applying this algorithm eventually leads to one particle with very high weight, while the rest of the particles have almost zero weight (so-called filter divergence or weight collapse). To avoid this, various resampling methods may be used (van Leeuwen 2009). The basic idea behind each of these methods is to monitor when a predetermined threshold is hit [e.g., when the "effective sample size" becomes small; Kong et al. (1994)], and to then resample the particles from the discrete approximation of the posterior distribution and reset all weights to $1/N_e$. In the remainder of this paper, we will use two different resampling methods: the Metropolis–Hastings method of resampling (Dowd 2007; van Leeuwen 2009) and the Gaussian resampling method of Xiong et al. (2006). We approximate the effective sample size to be

$$N_{\text{eff}} \approx \frac{1}{\sum_{i=1}^{N_e} w_i^2}, \qquad (6)$$

as in Kong et al. (1994). We will apply resampling when $N_{\text{eff}} < N_{\text{eff}}^{\text{thresh}}$ for a predetermined threshold $N_{\text{eff}}^{\text{thresh}}$ that will typically be a small fraction of the total number of particles N_e.

One major drawback to applying the particle filter in the Lagrangian data assimilation setup is that it has been shown to fail in high dimensions (Snyder et al. 2008). Thus, in the case where the state of interest is a velocity field discretized over some domain, the particle filter is intractable. However, when the state dimension is small enough and the number of particles is large enough, the particle filter can provide an accurate approximation to the exact Bayesian posterior distribution. This is especially useful if this distribution is skewed or multimodal, which is often the case in Lagrangian data assimilation.

b. Ensemble Kalman filter

Like the particle filter, the ensemble Kalman filter (Evensen 1994, 2003) employs an ensemble of state vectors $\{\mathbf{x}_i\}_{i=1,\ldots,N_e}$ to represent the posterior distribution;

however, unlike the particle filter, the ensemble members are equally weighted for the entire assimilation window. Instead of updating the weights at analysis times, the members themselves are updated according to an ensemble approximation of the traditional Kalman filter update step, given here by the so-called perturbed observation EnKF (Burgers et al. 1998; Houtekamer and Mitchell 1998; Evensen 2003):

$$\mathbf{x}_i^a = \mathbf{x}_i^f + \mathbf{K}(\mathbf{y} - \mathbf{H}\mathbf{x}_i^f + \boldsymbol{\epsilon}_i), \quad \text{and} \tag{7}$$

$$\mathbf{K} = \mathbf{P}^f\mathbf{H}^{\mathrm{T}}(\mathbf{H}\mathbf{P}^f\mathbf{H}^{\mathrm{T}} + \mathbf{R})^{-1}, \tag{8}$$

where $\mathbf{x}_i^f$ is the forecast of the ith ensemble member, $\mathbf{x}_i^a$ is the ith updated (analysis) ensemble member, $\mathbf{K}$ is the Kalman gain matrix, $\mathbf{R}$ is the covariance of the observation error, and $\epsilon_i \sim \mathcal{N}(\mathbf{0}, \mathbf{R})$ are the observation perturbations. In addition, $\mathbf{P}^f$ is the forecast ensemble covariance given by

$$\mathbf{P}^f = \frac{1}{N_e - 1}\sum_{i=1}^{N_e}(\mathbf{x}_i^f - \overline{\mathbf{x}}^f)(\mathbf{x}_i^f - \overline{\mathbf{x}}^f)^{\mathrm{T}},$$

$$\overline{\mathbf{x}}^f = \frac{1}{N_e}\sum_{i=1}^{N_e}\mathbf{x}_i^f. \tag{9}$$

However, the ensemble Kalman filter has its drawbacks as well: when the true posterior distribution is non-Gaussian, the ensemble Kalman filter will often result in a posterior distribution that is close to Gaussian (cf. Lawson and Hansen 2004).

There has been a lot of work toward improving the EnKF in nonlinear, high-dimensional systems; for example, covariance inflation and localization have provided significant improvement of the performance of the EnKF (see Hamill et al. 2001; Houtekamer and Mitchell 1998, 2001; Anderson and Anderson 1999). However, these methods do not overcome the basic shortcoming of the EnKF, which is its inability to capture highly non-Gaussian distributions. In addition, the performance of the EnKF in Lagrangian data assimilation is limited by the observation period. It has been shown that the EnKF fails when the time between drifter observations becomes long, even with improvements such as localization (Kuznetsov et al. 2003; Salman et al. 2006), due to the nonlinearity that becomes strong over these time periods (Apte et al. 2008; Apte and Jones 2013).

3. Hybrid particle-ensemble Kalman filter

a. The proposed filter

As mentioned above, neither the particle filter nor the ensemble Kalman filter is ideal (either theoretically or practically) in the case of Lagrangian data assimilation. The aim of the hybrid particle–ensemble Kalman filter proposed here is to exploit the advantages of each filter by splitting the drifter coordinates away from the flow variables. The high-dimensional, relatively linear Gaussian flow component is estimated via the ensemble Kalman filter, and the low-dimensional, highly nonlinear, and possibly non-Gaussian drifter variables are estimated via a particle filter.

A similar splitting was achieved by the hybrid grid–particle filter of Salman (2008a,b). The main difference is that Salman uses an advection-diffusion equation to solve the Fokker–Planck equation associated with Eq. (1b) in order to propagate the probability density function of the drifter variables $\mathbf{x}^D$ and then updates that density using Bayes's rule. This process effectively gives a weighted ensemble of the flow variables $\mathbf{x}^F$, which is resampled to get an ensemble with equal weights. In contrast, we use a Monte Carlo approximation of the Fokker–Planck equation, by constructing an ensemble of drifter positions, each of which is propagated using Eq. (1b). Additionally, instead of applying the particle filter update to the flow variables in each update step, we use a version of the EnKF (explained in detail in appendix A) for weighted ensembles.

There are two main reasons for choosing a combined particle–ensemble Kalman filter strategy instead of the hybrid grid/particle filter strategy of Salman. 1) The flow is usually high-dimensional, so a traditional particle filter approximation of the Eulerian variables (as used in the work of Salman) will require an intractable ensemble size. Since these variables usually do not behave very nonlinearly on time scales of instrument deployment, we choose to work with an EnKF approximation for the updates of these variables. 2) Solving a Fokker–Planck equation for the drifter distribution function (as done in the work of Salman) can by itself be quite computationally challenging and, in the case of multiple drifters, may not be feasible at all. Hence, we choose to work with a Monte Carlo approximation given by a weighted ensemble of drifters, with the weights updated in a manner similar to a particle filter described in section 2a. Thus, we expect the method that we propose to work well even for realistic models of the ocean flow, augmented by equations for drifter dynamics.

1) Setup

Let $\mathbf{x}^F \in \mathbb{R}^{N_F}$ denote the (potentially high-dimensional) flow variable and let $\mathbf{x}^D$ denote the drifter position variable, which consists of the x and y components of each of the N_D drifters, so that $\mathbf{x}^D \in \mathbb{R}^{2N_D}$. We assume a planar fluid flow in which we can only observe the position of the drifter on the surface, and not the height of the fluid at its location. These variables evolve under Eq. (1). At discrete times t_k, we have observations of the drifter available: $\mathbf{y}^k = \mathbf{x}^{D,k} + \boldsymbol{\epsilon}^k$, $\boldsymbol{\epsilon}^k \sim \mathcal{N}(\mathbf{0}, \mathbf{R})$, where $\mathbf{R}$ is the observation error covariance as above. At time t_k, the joint distribution of the flow and drifter variables is $p(\mathbf{x}^{F,k}, \mathbf{x}^{D,k}) = p(\mathbf{x}^{D,k} \,|\, \mathbf{x}^{F,k})p(\mathbf{x}^{F,k})$. We discretely approximate the marginal distribution on the flow $p(\mathbf{x}^{F,k}) = \sum_{i=1}^{N_e} \tilde{w}_i^k \delta(\mathbf{x}^{F,k} - \mathbf{x}_i^{F,k})$ with an ensemble of N_e weighted states $\{\mathbf{x}_i^{F,k}, \tilde{w}_i^k\}_{i=1}^{N_e}$. Initially, we set $\tilde{w}_i^k = 1/N_e$, so that the joint distribution is approximated by $p(\mathbf{x}^{F,k}, \mathbf{x}^{D,k}) \approx (1/N_e)\sum_{i=1}^{N_e} p(\mathbf{x}^{D,k} \,|\, \mathbf{x}_i^{F,k})\delta(\mathbf{x}^{F,k} - \mathbf{x}_i^{F,k})$. Next, we approximate the conditional distribution of the drifters given each flow ensemble member with a weighted ensemble of M states: $p(\mathbf{x}^{D,k} \,|\, \mathbf{x}_i^{F,k}) \approx \sum_{j=1}^{M} w_{i,j}^k \delta(\mathbf{x}^{D,k} - \mathbf{x}_{i,j}^{D,k})$, where $\{\mathbf{x}_{i,j}^{D,k}\}_{j=1,\ldots,M}$ is the ensemble of drifter states associated with (and subject to) the flow $\mathbf{x}_i^{F,k}$ and $\{w_{i,j}^k\}_{j=1,\ldots,M}$ are the associated weights. Thus, the full joint distribution is approximated discretely as

$$p(\mathbf{x}^{F,k}, \mathbf{x}^{D,k}) \approx \sum_{i=1}^{N_e} \sum_{j=1}^{M} w_{i,j}^k \delta(\mathbf{x}^{D,k} - \mathbf{x}_{i,j}^{D,k})\delta(\mathbf{x}^{F,k} - \mathbf{x}_i^{F,k}), \tag{10}$$

where $\sum_{i=1}^{N_e}\sum_{j=1}^{M} w_{i,j}^k = 1$. [For simplicity, we have absorbed the factor $1/N_e$ into the weights in Eq. (10).] We denote this ensemble by $\{\mathbf{x}_i^{F,k}, \mathbf{x}_{i,j}^{D,k}, w_{i,j}^k\}_{i=1,\ldots,N_e}^{j=1,\ldots,M}$. We also define $\tilde{w}_i^k$ in terms of $w_{i,j}^k$ for general times t_k by

$$\tilde{w}_i^k = \sum_j w_{i,j}^k, \tag{11}$$

so that the weighted ensemble representing the marginal distribution of the flow is given by $\{\mathbf{x}_i^{F,k}, \tilde{w}_i^k\}_{i=1}^{N_e}$. As with the typical particle filter, we will assume that, at time 0, $w_{i,j}^0 = 1/(MN_e)$ and that the ensemble members have all been drawn independently from their respective prior distributions. Finally, we define the following quantities at time t_k:

$$\tilde{\mathbf{x}}_i^{D,k} = \frac{1}{\tilde{w}_i^k}\sum_j \mathbf{x}_{i,j}^{D,k} w_{i,j}^k, \tag{12}$$

$$\overline{\mathbf{x}}^{F,k} = \sum_i \mathbf{x}_i^{F,k}\tilde{w}_i^k, \quad \overline{\mathbf{x}}^{D,k} = \sum_{i,j} \mathbf{x}_{i,j}^{D,k} w_{i,j}^k = \sum_i \tilde{\mathbf{x}}_i^{D,k}\tilde{w}_i^k. \tag{13}$$

Thus, $\tilde{\mathbf{x}}_i^{D,k}$ denotes the mean of the drifter particles associated with flow member i, while $\overline{\mathbf{x}}^{D,k}$ is the mean over all the drifter particles, and $\overline{\mathbf{x}}^{F,k}$ denotes the mean of the flow variables. Figure 1 provides a schematic of the setup for the hybrid filter.

2) Between updates

Suppose, at time t_{k-1}, we have the ensemble $\{\mathbf{x}_i^{F,k-1}, \mathbf{x}_{i,j}^{D,k-1}, w_{i,j}^{k-1}\}_{i=1,\ldots,N_e}^{j=1,\ldots,M}$ and our next observation is at time t_k. Before assimilating the observation, we must obtain an ensemble at time t_k. This will generally be performed by first numerically integrating each flow member $\mathbf{x}_i^{F,k-1}$ according to the model given by Eq. (1a). The drifter particles $\mathbf{x}_{i,j}^{D,k-1}$ are then numerically advected according to Eq. (1b) subject to the ith flow member:

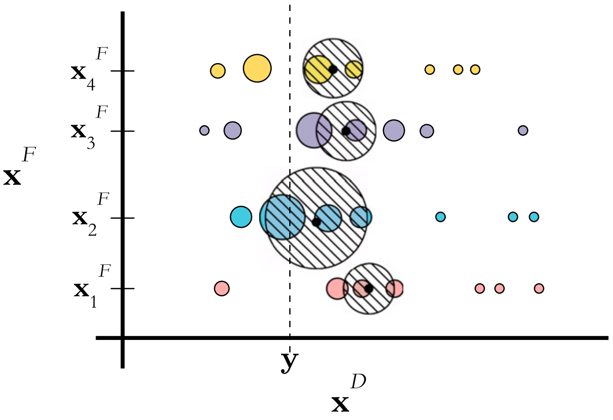

FIG. 1. Schematic of the hybrid filter setup: the flow variable $\mathbf{x}^F$ is projected onto the y axis while the drifter variable $\mathbf{x}^D$ is projected onto the x axis. The solid colored circles represent the ensemble: each of the four realizations of the flow has seven realizations of the drifter position affiliated with it, and the area of the circle represents the weight $w_{i,j}$. Striped circles represent within-flow averages $\tilde{\mathbf{x}}_i^D$ with sizes representing the weight $\tilde{w}_i$.

$$\dot{\mathbf{x}}_{i,j}^D = f_2(\mathbf{x}_i^F, \mathbf{x}_{i,j}^D). \tag{14}$$

The weights $w_{i,j}^{k-1}$ are then tested to determine whether the resampling condition is met or not. That is, at time t_k, the prior weights are used to calculate the effective dimension and determine whether $N_{\text{eff}} < N_{\text{eff}}^{\text{thresh}}$, with N_{eff} defined in Eq. (6). (This is done to avoid the computational effort of saving two sets of weights at every step, since the prior weights are necessary for the update step with resampling.) If $N_{\text{eff}} < N_{\text{eff}}^{\text{thresh}}$, the update with resampling is performed; otherwise, the update with no resampling is performed, as described immediately below.

3) Update: No resampling

The time integration described above yields the prior ensemble of state values at time t_k, $\{\mathbf{x}_i^{F,k}, \mathbf{x}_{i,j}^{D,k}\}$, which, along with the weights $\{w_{i,j}^{k-1}\}$, describe the prior distribution at time t_k. Suppose an observation $\mathbf{y}^k$ is available at time t_k, and that $N_{\text{eff}} \geq N_{\text{eff}}^{\text{thresh}}$: in this case, we update the weights $w_{i,j}^{k-1}$ to $w_{i,j}^k$, but leave the ensemble members $\{\mathbf{x}_i^{F,k}\}$ and the particles $\{\mathbf{x}_{i,j}^{D,k}\}$ unchanged. Following Eq. (5), the weight update equation is given by

$$w_{i,j}^k = \frac{p(\mathbf{y}^k \,|\, \mathbf{x}_{i,j}^{D,k})w_{i,j}^{k-1}}{\sum_{l,m} p(\mathbf{y}^k \,|\, \mathbf{x}_{l,m}^{D,k})w_{l,m}^{k-1}}; \tag{15}$$

this may be viewed as a standard particle filter update on the specific ensemble: $\{(\mathbf{x}_1^{F,k};\mathbf{x}_{1,1}^{D,k}), (\mathbf{x}_1^{F,k};\mathbf{x}_{1,2}^{D,k}), \ldots, (\mathbf{x}_1^{F,k};\mathbf{x}_{1,M}^{D,k}), (\mathbf{x}_2^{F,k};\mathbf{x}_{2,1}^{D,k}), \ldots, (\mathbf{x}_{N_e}^{F,k};\mathbf{x}_{N_e,M}^{D,k})\}$. The discrete approximation of the joint posterior distribution of the flow and drifters conditioned on all the observations up to and including the observation at time t_k is then given by

$$p(\mathbf{x}^{F,k}, \mathbf{x}^{D,k} \mid \mathbf{y}^k) \approx \sum_{i=1}^{N_e} \sum_{j=1}^{M} w_{i,j}^k \delta(\mathbf{x}^{D,k} - \mathbf{x}_{i,j}^{D,k}) \delta(\mathbf{x}^{F,k} - \mathbf{x}_i^{F,k}). \quad (16)$$

4) Update: With resampling

In this section, we discuss how to update the full ensemble when the weights cross the resampling threshold so that $N_{\text{eff}} < N_{\text{eff}}^{\text{thresh}}$. Since this update will occur entirely at time t_k, we drop the time dependence. Instead, we indicate whether a variable has been updated using the observation $\mathbf{y}^k =: \mathbf{y}$ or not: the superscript f (*forecast*) will denote variables that have not yet been updated, and the superscript a (*analysis*) will denote variables that have been updated with the observation $\mathbf{y}$. In particular, note that $w_{i,j}^f$ will denote $w_{i,j}^{k-1}$ since the weights do not change when the particles themselves are evolved forward in time, and $w_{i,j}^a$ will denote the weights at time t_k after they are updated according to the observation.

Traditionally, when applying the particle filter, one would resample $\{\mathbf{x}_i\}_{i=1}^{N_e}$ from $\sum_{i=1}^{N_e} w_i \delta(\mathbf{x} - \mathbf{x}_i)$ when some predetermined threshold of the effective sample size is hit; that is, the particles are resampled from the approximate full distribution on $\mathbf{x}$. In the proposed hybrid filter algorithm, the flow variables will be resampled from the EnKF posterior distribution, while the drifter variables will be resampled using the updated weights.

Generally, our hybrid filter update consists of three main steps, which we will briefly overview here before discussing specific details below. First, we change the values of the flow ensemble members using a version of the EnKF update, yielding a (weighted) ensemble approximation of the EnKF posterior distribution. Second, we update the weights using the current observation $\mathbf{y}$. Finally, we resample the flow members and the drifter particles from their respective distributions. More precisely, the flow variables will be resampled from the EnKF approximation of the joint distribution between the flow and the averaged drifters $\tilde{\mathbf{x}}^D$, marginalized over $\tilde{\mathbf{x}}^D$: $p(\mathbf{x}^F \mid \mathbf{y}) = \int p(\mathbf{x}^F, \tilde{\mathbf{x}}^D \mid \mathbf{y})\, d\tilde{\mathbf{x}}^D$. The drifter variables will be resampled from the approximation of the marginal distribution of the drifters conditioned on their respective flow members: $(\mathbf{x}_{i,j}^{a,D})_{i=1,\ldots,N_e}^{j=1,\ldots,M}$ are sampled from $\sum_{i=1}^{N_e}\sum_{j=1}^{M} w_{i,j}^a \delta(\mathbf{x}^D - \mathbf{x}_{i,j}^{f,D})$.

We will see that the resampling of the flow variables only uses information from the first and second moments of the distribution on $(\mathbf{x}^F, \mathbf{x}^D)$, as the EnKF does. This will produce a reasonable approximation of $p(\mathbf{x}^F \mid \mathbf{y})$ under the assumption that the marginal distribution on the flow variables is approximately Gaussian. We now describe the update process in more detail.

Let $\mathbf{A}_F^f$ be the $N_F \times N_e$ matrix with the ith column given by $\mathbf{x}_i^{f,F}$, and let $\tilde{\mathbf{A}}_D^f$ be the $2N_D \times N_e$ matrix with the ith column given by the average $\tilde{\mathbf{x}}_i^{f,D}$ of the drifters associated with $\mathbf{x}_i^{f,F}$, defined in Eq. (12). Recall that N_D denotes the number of drifters in the flow. We will use the perturbed-observation formulation of the EnKF and therefore define the $2N_D \times N_e$ matrix $\mathbf{Y}$,

$$\mathbf{Y} = [\mathbf{y} + \boldsymbol{\epsilon}_1, \mathbf{y} + \boldsymbol{\epsilon}_2, \ldots, \mathbf{y} + \boldsymbol{\epsilon}_{N_e}], \quad (17)$$

of perturbed observations; the distribution of each $\boldsymbol{\epsilon}_i$ must account for the fact that the ensembles of flow members and drifter particles have associated weights, and we therefore discuss them below.

As usual in the context of Lagrangian data assimilation, we divide the covariance $\mathbf{P}$ into four blocks:

$$\mathbf{P} = \begin{bmatrix} \mathbf{P}_{FF} & \mathbf{P}_{FD} \\ \mathbf{P}_{FD}^{\mathrm{T}} & \mathbf{P}_{DD} \end{bmatrix}, \quad (18)$$

which correspond to the covariance of the flow, the covariance of the drifters, and the cross covariances between the flow and drifters. Since this algorithm uses a different ensemble size for the flow members $\mathbf{x}_i^F$ and the drifter particles $\mathbf{x}_{i,j}^D$, the calculation of these covariance matrices is not as straightforward as with the traditional EnKF, and we therefore discuss methods for calculating these matrices for the hybrid filter in detail below.

Other than the differences in $\mathbf{Y}$ and $\mathbf{P}^f$, which will be described in the following two paragraphs, the update step on the flow members for the hybrid filter has the same formulation as the traditional EnKF update in the Lagrangian case:

$$\mathbf{A}_F^a = \mathbf{A}_F^f + \mathbf{P}_{FD}^f(\mathbf{P}_{DD}^f + \mathbf{R})^{-1}(\mathbf{Y} - \tilde{\mathbf{A}}_D^f). \quad (19)$$

In particular, note that $\mathbf{P}_{FD}^f(\mathbf{P}_{DD}^f + \mathbf{R})^{-1}$ defines the upper block of the Kalman gain matrix; since the drifter variables will be updated separately, the lower block is not needed here.

In the Gaussian case with a linear model, the traditional ensemble Kalman filter can be shown to give the correct Bayesian posterior mean and covariance in the limit as $N_e \to \infty$ (Mandel et al. 2011); additional relevant results on the EnKF are given in Furrer and Bengtsson (2007). In order for this to hold in the weighted case, the observation perturbations must have the correct distribution when considered as weighted samples. Specifically, let $\mathbf{\Upsilon}$ be as defined in Eq. (17). In the traditional EnKF, $\boldsymbol{\epsilon}_i \sim \mathcal{N}(\mathbf{0}, \mathbf{R})$; however, in our case, we need the weighted ensemble $\{\boldsymbol{\epsilon}_i, \tilde{w}_i^f\}$ to be a discrete approximation of the continuous normal distribution with mean $\mathbf{0}$ and covariance $\mathbf{R}$. Details on how to produce this ensemble are given in appendix A.

There are essentially two choices for calculating $\mathbf{P}_{FD}^f$ and $\mathbf{P}_{DD}^f$: they can be calculated using the "full" ensemble $\{\mathbf{x}_i^{f,F}, \mathbf{x}_{i,j}^{f,D}, w_{i,j}^f\}$ or the "averaged" ensemble $\{\mathbf{x}_i^{f,F}, \tilde{\mathbf{x}}_i^{f,D}, \tilde{w}_i^f\}$. These two ensembles will have the same means but different covariances, denoted with a subscript full or avg (see appendix B for details). We emphasize that both the full and averaged covariances will result in an approximation when used to update the flow members, due to the nested ensemble setup. In the experiments presented in the following section, we implemented both methods and found very little difference in the results (see section 4c). Using the averaged ensemble has the advantage that, in the linear Gaussian case, the resulting posterior (analysis) covariance of the flow is consistent with that of the traditional EnKF. Indeed, note that the posterior mean and covariance of the flow variables (after the EnKF update) will depend on which ensemble (full or averaged) is used to calculate $\mathbf{P}_{DD}^f$ and $\mathbf{P}_{FD}^f$. We define $\mathbf{K}^{(1)} = \mathbf{P}_{FD}^f(\mathbf{P}_{DD}^f + \mathbf{R})^{-1}$, the upper block of the Kalman gain matrix. Then, the posterior mean of the flow is given by $\overline{\mathbf{x}}^{a,F} = \overline{\mathbf{x}}^{f,F} + \mathbf{K}^{(1)}(\mathbf{y} - \overline{\mathbf{x}}^{f,D})$, and the posterior covariance can be shown to be $\mathbf{P}_{FF}^a = \mathbf{P}_{FF}^f - \mathbf{K}^{(1)}(\mathbf{P}_{FD,\mathrm{avg}}^f)^{\mathrm{T}} - \mathbf{P}_{FD,\mathrm{avg}}^f\mathbf{K}^{(1)\mathrm{T}} + \mathbf{K}^{(1)}(\mathbf{P}_{DD,\mathrm{avg}}^f + \mathbf{R})\mathbf{K}^{(1)\mathrm{T}}$. Now, if the covariances from the full ensemble are used in $\mathbf{K}^{(1)}$, this expression cannot be simplified further. However, if the covariances from the averaged ensemble are used, this can be further simplified to $\mathbf{P}_{FF}^a = \mathbf{P}_{FF}^f - \mathbf{K}^{(1)}(\mathbf{P}_{FD,\mathrm{avg}}^f)^{\mathrm{T}}$, which is the same form given by the traditional EnKF for Lagrangian data assimilation. Thus, since the prior statistics on the flow use the averaged ensemble, the update step on the flow should also use the averaged ensemble; in the linear Gaussian case, this will lead to posterior statistics that are consistent with those of the traditional EnKF. In particular, the innovations $(\mathbf{y} - \tilde{\mathbf{x}}_i^{f,D})$ depend on the averaged statistics, which prevents further simplifications of the posterior covariance if the full statistics are used in $\mathbf{K}^{(1)}$.

We now present a concise description of the implementation of the update with resampling. We consider the prior–forecast ensemble to be two ensembles: one for the flow variables, $\{\mathbf{x}_i^{f,F}, \tilde{w}_i^f\}$ [with $\tilde{w}_i^f$ as defined in Eq. (11)], and one for the drifter variables, $\{\mathbf{x}_{i,j}^{f,D}, w_{i,j}^f\}$. The PF–EnKF hybrid updating–resampling algorithm proceeds as follows (with additional explanations provided subsequently):

(i) Change the state values of the flow ensemble members using the observation $\mathbf{y}$ with the EnKF update given in Eq. (19), where covariances $\mathbf{P}_{FD}^f$ and $\mathbf{P}_{DD}^f$ are obtained using the averaged drifter ensemble as described above. We obtain $\{\mathbf{x}_i^{a,F}, \tilde{w}_i^f\}$.

(ii) Find $w_{i,j}^a$ using $\mathbf{y}$ and $w_{i,j}^f$ with the standard particle filter update described in Eq. (15). This gives $\{\mathbf{x}_{i,j}^{f,D}, w_{i,j}^a\}$.

(iii) Now $\{\mathbf{x}_i^{a,F}, \tilde{w}_i^f\}$ and $\{\mathbf{x}_{i,j}^{f,D}, w_{i,j}^a\}$ together represent the posterior distribution at time t_k, which has incorporated the observation $\mathbf{y}$. The forecast weights are used in the representation of the posterior distribution on the flow, since the flow ensemble members have already been updated to represent the observations; this sample then has the same properties as the traditional EnKF, except that each ensemble member has an associated weight. (See appendix A for more details.)

(iv) Resample the flow variables from $\{\mathbf{x}_i^{a,F}, \tilde{w}_i^f\}$ and the drifter variables from $\{\mathbf{x}_{i,j}^{f,D}, w_{i,j}^a\}$ using standard methods. Call these $\{\check{\mathbf{x}}_i^{a,F}\}$ and $\{\check{\mathbf{x}}_i^{a,D}\}$, respectively. Note, for a specific flow member $i = m$, if $\mathbf{y}$ falls far from the support of $\{\mathbf{x}_{m,j}^{f,D}\}$, we recommend resampling the drifter variables for the mth flow around the observation using the observation error statistics.

(v) Set $w_{i,j}^a = 1/(MN_e)$, $\mathbf{x}_i^{a,F} = \check{\mathbf{x}}_i^{a,F}$, and $\mathbf{x}_{i,j}^{a,D} = \check{\mathbf{x}}_i^{a,D}$. Then, the posterior is represented by $\{\mathbf{x}_i^{a,F}, \mathbf{x}_{i,j}^{a,D}, w_{i,j}^a\}$ and sequential filtering proceeds as normal.

In particular, note that the EnKF update on the flow variables uses the prior weights, as updating both the weights and the flow members would lead to the observations being incorporated into the flow update twice. However, since the weights must all be equal at the end of the full update, the final flow members must be resampled from $\{\mathbf{x}_i^{a,F}, \tilde{w}_i^f\}$, so that the ensemble with equal weights approximates the same distribution.

At any point in time, the ensemble $\{\mathbf{x}_i^F, \mathbf{x}_{i,j}^D, w_{i,j}\}_{i=1,\ldots,N_e}^{j=1,\ldots,M}$ may be used to calculate statistics of interest such as the mean or covariance, as with a typical particle filter. The mean flow state is given by $\overline{\mathbf{x}}^F = \sum_{i=1}^{N_e}\mathbf{x}_i^F\tilde{w}_i$, and the mean drifter state is given by $\overline{\mathbf{x}}^D = \sum_{j=1}^{M}\sum_{i=1}^{N_e}\mathbf{x}_{i,j}^D w_{i,j}$. The covariance matrices will be subject to the same complications described above: they can be calculated either using the full ensemble of drifter particles or the averaged ensemble. Define $\mathbf{x}_{i,j} = [\mathbf{x}_i^F, \mathbf{x}_{i,j}^D]^{\mathrm{T}}$; then, the covariance matrix of the full ensemble is $\mathbf{P}_{\mathrm{full}} = \sum_{j=1}^{M}\sum_{i=1}^{N_e}(\mathbf{x}_{i,j} - \overline{\mathbf{x}})(\mathbf{x}_{i,j} - \overline{\mathbf{x}})^{\mathrm{T}}w_{i,j}$. Next, let $\tilde{\mathbf{x}}_i = [\mathbf{x}_i^F, \tilde{\mathbf{x}}_i^D]^{\mathrm{T}}$;

then, the covariance matrix of the averaged ensemble is $\mathbf{P}_{\text{avg}} = \sum_{i=1}^{N_e}(\tilde{\mathbf{x}}_i - \overline{\mathbf{x}})(\tilde{\mathbf{x}}_i - \overline{\mathbf{x}})^{\mathrm{T}}\tilde{w}_i$.

b. *Expected benefits and outcomes*

The hybrid filter presented above is developed mainly for the case of Lagrangian data assimilation, when the flow field has been discretized into a high-dimensional vector and the drifter trajectories are highly nonlinear. In this case, the particle filter is completely intractable, whereas the ensemble Kalman filter breaks down when the prior distribution is highly non-Gaussian. The latter case arises when the drifter dynamics, either near a saddle or a center, leads to non-Gaussian distributions (Apte and Jones 2013). In these cases, we expect the hybrid filter to outperform the ensemble Kalman filter, since the hybrid filter employs a particle filter on the drifters, in order to effectively approximate such non-Gaussian distributions.

When the flow leads to highly non-Gaussian distributions on the drifter trajectory, the ensemble Kalman filter may break down. In these cases, although the hybrid filter may take more machine time to run due to the large number of drifter particles, it will produce much more accurate results than the ensemble Kalman filter, as shown in numerical examples in section 4. In any case, the increase in computation will be largely nominal, since running many more evolutions of the drifter particles will be significantly cheaper than evolving more realizations of the flow. In addition, each drifter particle is independent given the flow field ensemble, so these advections can be easily parallelized. We also note that the EnKF update of the flow field in the high-dimensional case can take advantage of common EnKF methods such as localization, so that local observations do not overly affect the flow at large spatial distances in this update step. Localization can simply be applied to the weighted covariances given in Eq. (18). Finally, we expect that this general methodology, of splitting the state space into two parts and applying different assimilation techniques to them, will be useful in other contexts as well.

4. Numerical results

In this section, we test the hybrid filter on a model with a low-dimensional flow variable. This was chosen because the particle filter is tractable for Lagrangian data assimilation when the flow is low-dimensional, and we are particularly interested in comparing the hybrid and EnKF posterior distributions to the particle filter posterior. Since Lagrangian data assimilation leads to non-Gaussian distributions, we need a method that can handle non-Gaussianity as a benchmark to which we can compare both the traditional method of the EnKF and the new method of the hybrid filter. Unless otherwise noted, the resampling method used will be a Metropolis–Hastings (MH) scheme based on the work of Dowd (2007) and van Leeuwen (2009). However, in Table 1, we also include an experiment in which the flow variables are resampled using the Gaussian resampling (GR) scheme of Xiong et al. (2006), while the drifter variables are still resampled using the MH scheme.

TABLE 1. Time-averaged errors (as described in the text) of each filter over the assimilation window, averaged over 20 trials, with 95% confidence intervals: scenario 2. PF (lg.) provides a baseline, using $N_e = 2 \times 10^6$.

Obs frequency	Method	Drifter error	Flow error
High	PF (lg.)	0.504	0.387
	PF	0.504 ± 0.007	0.473 ± 0.023
	Hybrid	0.475 ± 0.012	0.478 ± 0.041
	Hybrid with GR	0.467 ± 0.008	0.392 ± 0.037
	EnKF	0.325 ± 0.010	0.268 ± 0.023
	EnKF (lg.)	0.295 ± 0.0002	0.220 ± 0.0004
Low	PF (lg.)	0.793	0.635
	PF	0.801 ± 0.010	0.643 ± 0.021
	Hybrid	0.802 ± 0.031	0.778 ± 0.060
	Hybrid with GR	0.809 ± 0.020	0.743 ± 0.040
	EnKF	1.119 ± 0.102	0.787 ± 0.130
	EnKF (lg.)	1.016 ± 0.001	0.822 ± 0.003

a. *Model*

As a proof of concept, we apply the particle filter, ensemble Kalman filter, and hybrid filter to the linear shallow water equations with a single drifter. This model, and the decomposed solution given below, are based on (Pedlosky 1986) and were used as a test problem in (Apte et al. 2008). Derived from the Navier–Stokes equations under certain assumptions and approximations, the linear shallow water equations describe the time evolution of the horizontal velocity u, the meridional velocity v, and the offset from the mean height field h, and are given by

$$\begin{aligned}
\frac{\partial u}{\partial t} &= v - \frac{\partial h}{\partial x},\\
\frac{\partial v}{\partial t} &= -u - \frac{\partial h}{\partial y},\\
\frac{\partial h}{\partial t} &= -\frac{\partial u}{\partial x} - \frac{\partial v}{\partial y}.
\end{aligned} \tag{20}$$

For simplicity, we use periodic boundary conditions so that explicit solutions to this model can be found as sums of Fourier modes:

$$\begin{aligned}
u(x,y,t) &= -l\sin(kx)\cos(ly)u_0 + \cos(my)u_1(t),\\
v(x,y,t) &= k\cos(kx)\sin(ly)u_0 + \cos(my)v_1(t),\\
h(x,y,t) &= \sin(kx)\sin(ly)u_0 + \sin(my)h_1(t),
\end{aligned} \tag{21}$$

where the Fourier amplitudes solve linear ordinary differential equations. We will consider a noisy version of this system, where the noise is given by $\boldsymbol{\eta} \sim \mathcal{N}(\mathbf{0}, \mathbf{Q})$ (independent across the three variables) and $\boldsymbol{\eta} = [\eta_{(1)}, \eta_{(2)}, \eta_{(3)}]$:

$$\begin{aligned}\dot{u}_0 &= 0,\\ \dot{u}_1 &= v_1 + \eta_{(1)},\\ \dot{v}_1 &= -u_1 - mh_1 + \eta_{(2)},\\ \dot{h}_1 &= mv_1 + \eta_{(3)}.\end{aligned} \qquad (22)$$

Note that, even though the original model is symmetric in u and v, the system governing the amplitudes is not symmetric in u_1 and v_1 unless $m = 0$. Next, the position of the drifter $\mathbf{x}^D = (x, y)$ solves

$$\begin{aligned}\dot{x} &= u(x,y,t),\\ \dot{y} &= v(x,y,t).\end{aligned} \qquad (23)$$

In particular, even if the flow evolution Eqs. (20) are linear in (u, v, h), the drifter evolution Eqs. (23) will be nonlinear in (x, y) unless (u, v, h) is constant. In this model, the Eulerian variables of interest are $\mathbf{x}^F = (u_0, u_1, v_1, h_1) := (x_1^F, x_2^F, x_3^F, x_4^F)$, whereas the drifter variables are $\mathbf{x}^D = (x, y)$. We observe a noisy measurement of (x, y), for which the covariance is assumed to be

$$\mathbf{R} = \sigma_{\mathbf{R}}^2 \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \qquad (24)$$

for some scalar $\sigma_{\mathbf{R}}^2$.

In the following experiments, we will estimate the Fourier amplitudes as the flow variables. Therefore, since we are only estimating a relatively small number of variables, the particle filter is tractable. With enough particles, it can also be assumed to provide an approximation to the true Bayesian posterior distribution, since it captures all non-Gaussian behavior. In particular, this choice of system provides the ability to easily compare the marginal distributions of the flow from each filter graphically.

In some experiments we will also compare the errors between the mean of each filter (denoted by an overbar) and the truth (denoted by the superscript "true") as a function of time, for the flow and drifter variables separately. At a given assimilation step, these errors are calculated according to

$$\text{flow error} = \left[\sum_{m=1}^{N_F} (\overline{x}_m^F - x_m^{F,\text{true}})^2\right]^{1/2} \quad \text{and} \qquad (25)$$

$$\text{drifter error} = \frac{1}{\sigma_{\mathbf{R}}}[(\overline{x} - x^{\text{true}})^2 + (\overline{y} - y^{\text{true}})^2]^{1/2}, \qquad (26)$$

where N_F may be 3 or 4 depending on the scenario (described in the following subsections) and whether or not we estimate u_0 in that case. In particular, note also that the error on the drifter is normalized by the observation error standard deviation.

In the remainder of this section, we explore two scenarios: first, in section 4b, a single-step update in which a bimodal prior distribution is enforced; second, in section 4c, a long trajectory in which the drifter crosses through several cells. In this second scenario, we consider cases where the observations are available at both a high and low frequency. In scenario 1, no noise is added to the system. Figure 2 (left) shows a snapshot in time of the flow field in this case. (Exact parameters for each scenario will be given in the subsections below.) In scenario 2, nonzero noise is added to the evolution of the flow, and the drifter crosses between several cells. The true trajectory for this case is given in Fig. 2 (right). Black circles represent how often drifter position was assimilated for the high-frequency case, and red asterisks represent the low-frequency case. As demonstrated in Fig. 2, in the low-frequency case, observations are available about 4 times per drifter orbit, whereas observations are available about 40 times per orbit in the high-frequency case. However, this depends heavily on how close the drifter is to a saddle point in the flow, which affects how quickly the drifter is moving at that point in time.

b. Scenario 1: Single step, bimodal prior

In this simple case, we consider the marginal posterior distributions on the four flow variables u_0, u_1, v_1, h_1 and the drifter coordinates x and y after a single forecast-update step of each assimilation algorithm. The particle filter update step includes Metropolis–Hastings resampling and the hybrid filter update step includes the EnKF update on the flow variables described in section 3a. In this case, $k = l = m = 1$ and no noise is added to the system: $\mathbf{Q} = \mathbf{0}$. We let the EnKF ensemble size and the number of particles for the particle filter both be $N_e = 10^4$. The ensemble of flow members for the hybrid filter is $N_e = 1000$ and the number of drifter particles for each flow member is $M = 100$, so that the total number of particles in the hybrid filter is $MN_e = 10^5$. Since the dimension of the estimated state is relatively low and only one update step is performed, the particle filter distribution is taken to be an approximation to the true Bayesian posterior.

The prior distributions on each of the flow variables u_0, u_1, v_1, and h_1 are Gaussian. The prior distribution on x is also Gaussian, while the prior distribution on y is bimodal to simulate a saddle case. Based on previous applications of the EnKF to a bimodal distribution, we

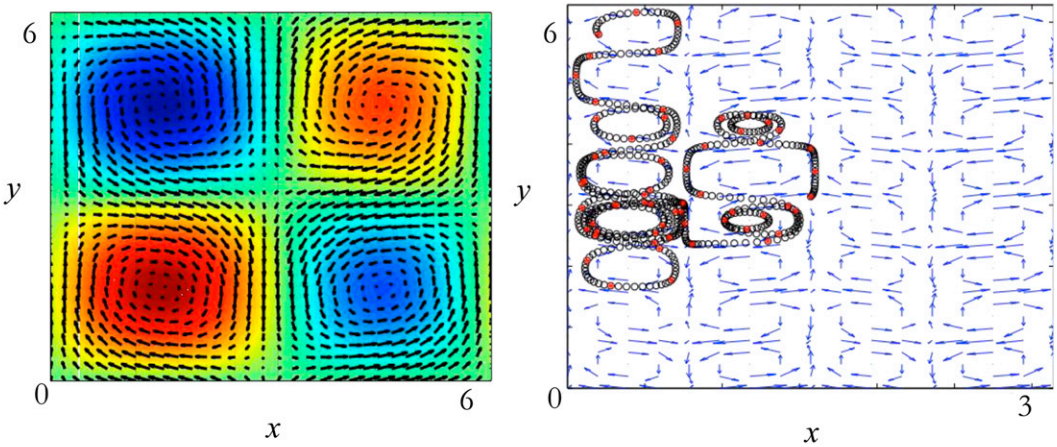

FIG. 2. Setup for each scenario. (left) Snapshot in time of the flow field (u, v) (arrows) and height field h (shading); scenario 1, no noise. (right) True drifter trajectory (high observation frequency, black circles; low observation frequency, red asterisks) and snapshot of the flow field (u, v) (blue arrows) for scenario 2.

expect the EnKF to fail to capture the true distribution of the y coordinate, but we expect the hybrid filter to capture this distribution more accurately (since the algorithm uses a particle filter on the drifter variables). Indeed, in Fig. 3, the particle filter posterior on the y coordinate is highly non-Gaussian, and while the hybrid filter captures this shape, the EnKF posterior is much closer to Gaussian. The particle filter posterior on the x coordinate is much closer to Gaussian, and while the EnKF posterior is more accurate than for the y coordinate, it still does not quite capture the covariance of the particle filter, while the hybrid filter does. The hybrid filter and EnKF are equivalent on the flow variables, since the hybrid filter employs the EnKF update on these variables. In this case, since the flow variables evolve linearly, the EnKF posterior and particle filter posterior distributions are fairly close to each other.

c. Scenario 2: Long trajectory

Next, we test the performance of each filter in the case where a drifter passes through many cells in the flow. Within this scenario, we run experiments for two sets of observations: high and low frequency. Here, we only estimate three flow variables (u_1, v_1, h_1) using Eqs. (22) and the drifter (x, y) using Eqs. (23) with wavenumbers $k = l = m = 4$, and model noise covariance

$$\mathbf{Q} = \begin{bmatrix} 0.05 & 0 & 0 \\ 0 & 0.1 & 0 \\ 0 & 0 & 0.1 \end{bmatrix}.$$

The observation error covariance is $\mathbf{R} = 0.01\mathbf{I}$. The high-frequency case uses 600 observations with $T_{\text{final}} = 10$, and the low-frequency case uses 60 observations for the same time window. The true initial conditions are $[u_0^{\text{true}}(0), u_1^{\text{true}}(0), v_1^{\text{true}}(0), h_1^{\text{true}}(0), x^{\text{true}}(0), y^{\text{true}}(0)] = (1, 0.5, 0.9, 1, \pi/2, \pi)$. In each case, the initial ensembles for the filters are drawn from Gaussian distributions, which are centered away from the truth, in order to judge whether the filters are able to recover from this initial error. The initial ensembles for the flow variables are drawn from distributions with mean $[u_1^{\text{true}}(0) + 0.2, v_1^{\text{true}}(0) + 0.5, h_1^{\text{true}}(0) + 0.5]$ and covariance $\mathbf{I}$. The initial ensembles for the drifter variables are drawn from distributions with mean $[x^{\text{true}}(0) + 0.1, y^{\text{true}}(0) + 0.1]$ and covariance $0.1\mathbf{I}$. The particle filter uses ensemble size $N_e = 9 \times 10^4$ and the EnKF uses ensemble size $N_e = 50$.

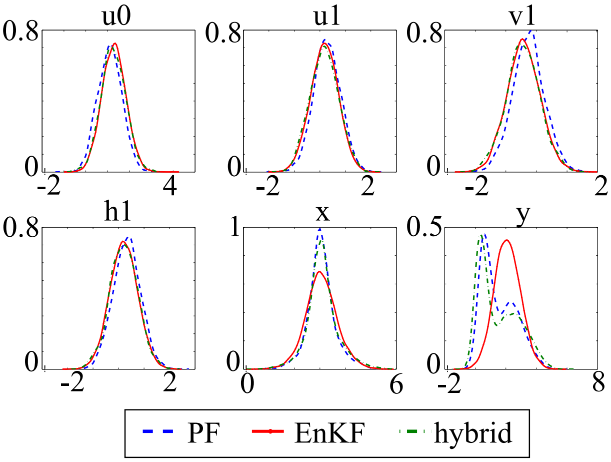

FIG. 3. Comparison of posterior distributions of particle filter (blue dashed curve), ensemble Kalman filter (red solid curve), and hybrid filter (green dashed–dotted curve): single forecast and update step of stationary linear shallow water equations. Bimodal prior on y; Gaussian priors on u_0, u_1, v_1, and h_1: scenario 1.

The hybrid filter uses $N_e = 50$ and $M = 2000$; that is, each of the 50 flow members has 2000 drifter particles associated with it.

The time-averaged errors for the flow variables and the drifter position are given in Table 1. We have included a single experiment of the particle filter with $N_e = 2 \times 10^6$ [denoted PF (lg.)] to provide a baseline for the errors. For the other filters, the values are averaged over 20 assimilation trials, where each trial uses the same observations but different realizations of the initial filter ensemble and the system noise. For each trial, the errors are averaged in time over the assimilation window. The 95% confidence intervals are calculated over the 20 trials, using a Student's t distribution with $p = 0.025$ and 19 degrees of freedom.

For the case with a high frequency of observations, all filters perform well. The hybrid filter and particle filter perform similarly, while the EnKF errors are the lowest. This is likely due to the fact that the flow is evolving linearly and the observations are close enough to each other than the drifter trajectory between observations is also fairly linear. Additionally, since the EnKF errors are lower than even the large-sample PF results, this may be due to the EnKF overfitting to the observations; this is discussed further below.

On the other hand, in the case of a low frequency of observations, the particle filter and hybrid filter generally outperform the EnKF with $N_e = 50$. The hybrid filter estimates the drifter position about as well as the particle filter, though it does not estimate the flow variables quite as well. However, the hybrid filter estimates the flow variables slightly better than does the EnKF on average. This is likely due to the fact that the EnKF does not estimate the drifter position very well in this case, which affects its estimate of the flow. Additionally, note that the confidence intervals for the EnKF are much larger than those for the other filters. This is because the EnKF is more likely to fail (diverge) in this case than the hybrid or particle filter, due to the nonlinearity of the drifter trajectory.

Figures 4 and 5 each show an example of one of these divergent cases. The hybrid and particle filter trials chosen here display representative behavior. Figure 4 includes the errors between each filter mean from the truth as a function of time for the flow variables and the drifter position. Figure 5 shows the evolution of the mean of the EnKF, hybrid, and PF, as well as the truth (black) of the flow variables u_1, v_1, h_1 and the drifter trajectory. Note that the EnKF fails to estimate the drifter position at a saddle point, near the coordinate (0.7, 2.5). The true drifter trajectory moves southwest into the left cell, while the EnKF estimate moves southeast into the right cell for several observations, until it is eventually pulled back. However, this affects the estimate of the flow for the rest of the assimilation window.

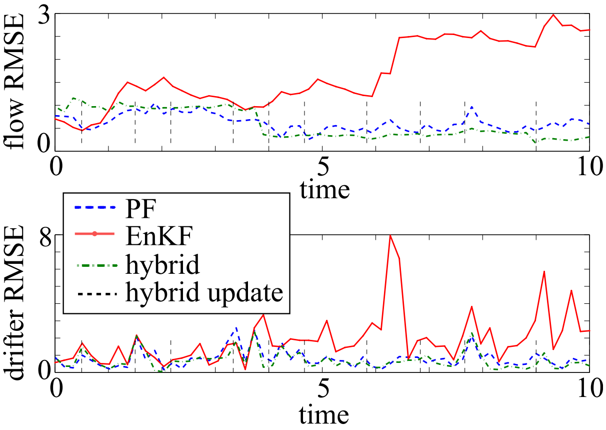

FIG. 4. Scenario 2b: long drifter trajectory, low observation frequency. Errors of means of particle filter (blue dashed), EnKF (red solid), and hybrid filter (green dashed–dotted) from truth as functions of assimilation step, for the (top) flow variables and (bottom) drifter position. Vertical dashed lines represent steps at which the hybrid filter performed the EnKF update, according to the resampling threshold described in the text.

Figure 6 shows the distributions of the x and y coordinates of the drifter at $t = 6$ for each of the filters, as well as the true value of about (0.6, 2.3). In the x coordinate, the particle filter distribution is somewhat skewed at this time. The hybrid filter captures this behavior well, while the EnKF does not. In the y coordinate, although the particle filter distribution is close to Gaussian, the EnKF fails to capture this distribution at all, while the hybrid filter captures it well. In fact, the true value does not even fall within the support of the EnKF distribution.

Table 1 also includes the errors of the hybrid filter with the Gaussian resampling method of Xiong et al. (2006) applied to the flow variables at the update step. These errors are generally comparable to those of the hybrid filter with MH resampling on both the flow and drifter. This is likely due to the linear behavior of the flow variables in this example, which suggests that the distributions on the flow variables should be close to Gaussian. Additionally, since this method of resampling tends to spread the ensemble out more than the MH method, it may be useful for deterministic models or systems with low noise levels.

As discussed earlier, these experiments used the version of the hybrid filter with a drifter covariance calculated using the averaged ensemble. We also performed these experiments using the full ensemble drifter covariance in the update step, and the errors were indistinguishable.

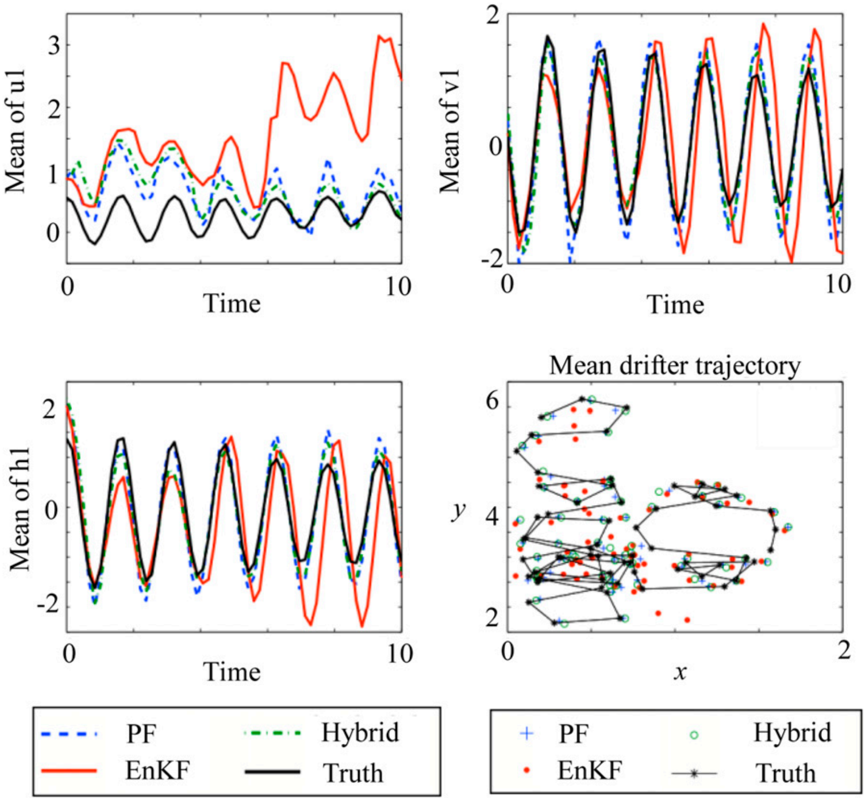

FIG. 5. Scenario 2b: long drifter trajectory, low observation frequency. Evolution of the particle filter (blue dashed), EnKF (red solid), and hybrid filter (green dashed–dotted) means of flow variables (top left) u_1, (top right) v_1, and (bottom left) h_1 as a function of assimilation step, and (bottom right) trajectory of drifter over entire assimilation window. True evolutions are given in black.

This is likely due to the small difference between these two covariances. In particular, after evaluating Eq. (B15) at each update step for each trial, the average norm of the difference ($\mathbf{P}_{DD,\mathrm{full}} - \mathbf{P}_{DD,\mathrm{avg}}$) is 4.50×10^{-4} for the high-frequency case, and 0.02 for the low-frequency case. The average norms of $\mathbf{P}_{DD,\mathrm{full}}$ are 1.66 and 3.48 for the high- and low-frequency cases, respectively. These differences are relatively constant in time, so taking the time average does not lose information.

We anticipate that the hybrid filter will prove most beneficial when the flow field is high-dimensional, in which case the ensemble size for the EnKF and for the flow part of the hybrid filter will be limited. On the other hand, the number of drifter particles in the hybrid filter is only limited by the number of drifters, not by the dimension of the flow. For this reason, the EnKF with $N_e = 50$ was compared to the hybrid filter with $N_e = 50$ flow members, and $M = 1000$. However, we also ran the EnKF with $N_e = 9 \times 10^4$, the same ensemble size as the particle filter, to avoid conflation with the effects of sampling error as much as possible. These results are also included in Table 1, as EnKF (lg).

The large-sample EnKF errors for the high-frequency case are significantly lower than for the other filters. However, this information is limited: it only contains the error of the mean estimate from the truth and does not contain any information about the underlying distribution. Additionally, the EnKF ensemble may be overtightened around the mean. The results from the large-sample

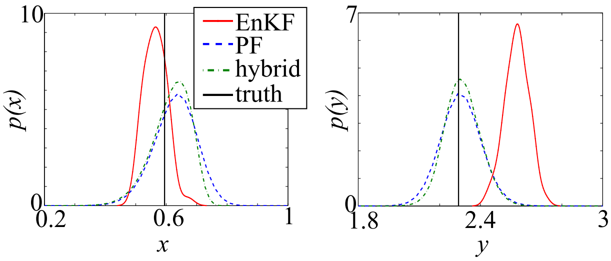

FIG. 6. Distributions of (left) x and (right) y drifter variables for particle filter (blue dashed), EnKF (red solid), and hybrid filter (green dashed–dotted), with true value given by the vertical black lines; scenario 2b: long drifter trajectory, low observation frequency.

particle filter support this, since the errors do not decrease drastically when the ensemble size is increased. While this does not have a detrimental effect on the high-frequency case, we see that the EnKF (lg) errors for the low-frequency case are still larger than for the other filters, though the confidence intervals are much smaller. This suggests that the drastic failure that occurs in the low-frequency, small-ensemble EnKF is much less likely to occur with a large ensemble, but that in general, the large-ensemble EnKF still does not perform as well as the hybrid filter in the case of low-frequency observations.

Finally, we emphasize that the large ensemble size used for the particle filter is meant to allow this method to be considered as a benchmark to which the hybrid and ensemble Kalman filters can be compared. In systems with a high-dimensional flow, the particle filter will fail due to the limited ensemble size. On the other hand, the hybrid filter only needs a large ensemble to capture the behavior of the drifters, which we assume will be of a relatively low dimension. Here, we have also included results with smaller ensemble sizes, to hint at the behavior of these filters in systems with high dimension. Table 2 includes the errors for the same scenarios described above, but the particle filter ensemble size is $N_e = 100$ and the hybrid filter ensemble sizes are $N_e = 50$, $M = 100$. The particle filter has much worse performance with this small sample size, especially in the low-frequency case. However, the hybrid filter errors are indistinguishable from those with the larger ensemble size.

TABLE 2. Time-averaged errors (as described in the text) of the filters over the assimilation window, averaged over 20 trials, with 95% confidence intervals: scenario 2, small ensemble sizes.

Obs frequency	Method	Drifter error	Flow error
High	PF	0.734 ± 0.081	1.536 ± 0.348
	Hybrid	0.476 ± 0.010	0.443 ± 0.061
Low	PF	1.330 ± 0.168	1.493 ± 0.395
	Hybrid	0.846 ± 0.038	0.787 ± 0.092

5. Discussion and outlook

We have introduced a hybrid particle–ensemble Kalman filter for assimilating Lagrangian data into ocean models. The two primary challenges when performing Lagrangian data assimilation are 1) the strong nonlinearity of Lagrangian paths taken by instruments that are advected through the ocean and 2) the high-dimensionality of realistic ocean models. We have devised a hybrid filter to exploit the strengths of the two individual filtering methods—handling nonlinearity for particle filters and handling high-dimensional systems for ensemble Kalman filters—by decomposing the underlying model as suggested by the challenges. As such, we take a small number of ensemble members to represent the flow field (ocean dynamics). We think of updating these via an ensemble Kalman filter. However, we represent the drifter advected by each flow member with a large number of particles akin to a particle-filtering scheme. Due to the sometimes chaotic nature of a drifter's path, representing its path by many particles enables accurate approximation of multimodal prior distributions, which can arise when a drifter travels near a saddle point between observations. Thus, the forecast for the hybrid filter has an excellent chance of placing high-weight particles near the observation because the drifter space is very well sampled. In contrast, an EnKF would only have *one* drifter sample per ensemble member. One could easily imagine that the drifter of a ''very accurate'' flow field ensemble member happens to follow the other natural path away from the saddle and thus away from the observation. In this case, lack of sampling in the nonlinear dimension will strongly degrade estimates provided by the EnKF posterior.

The numerical experiments presented in this paper demonstrate that the hybrid filter outperforms the ensemble Kalman filter and often performs on par with posterior densities estimated by the particle filter. In the linear flow case, the hybrid filter estimated the full posterior distribution much more accurately than did the EnKF. Many applications involve sampling from this posterior in order to get a sense of different possible outcomes as well as variability among them. Thus, an incorrect posterior distribution would result in incorrect samples (even if the distribution has the correct mean and covariance, after inflation). Therefore, in cases where the true posterior distribution is highly non-Gaussian, the EnKF will likely give poor results regardless of algorithmic improvements such as covariance inflation. In the cases shown here, the hybrid filter overcame this problem and yielded posterior distributions that more closely represented those of the particle filter. In addition, when the time between observations became long, the EnKF failed more often than did the hybrid filter, while the mean of the hybrid filter consistently provided accurate estimations of the truth. This is precisely the case that motivated the hybrid filter, as drifter path nonlinearity is hard to avoid when the time between observations is long.

A practical strength of the proposed hybrid filter, which we believe will make it very attractive to ocean scientists, is that it ''feels like'' an EnKF, but can easily deal with the nonlinear nature of the data at a relatively nominal added computational expense. However, we foresee some remaining challenges. For instance, one may want to assimilate multiple drifter tracks into an ocean model. If these drifters are well separated, then treating them independently would be natural as typical EnKF schemes employ some kind of localization to avoid (spurious) long-range correlations. So, one could use the proposed hybrid filter to update part of the ocean with drifter *a* and another part with drifter *b*. However, if these drifters are in the same region of the ocean, we would expect some correlations between drifter paths, and the effects of correctly accounting for those correlations within the hybrid algorithm would need to be carefully thought through. Another potential challenge involves resampling of the flow field after an EnKF update of the flow. Recall, as opposed to an EnKF, the hybrid scheme update yields weighted flow field ensemble members. But, one purpose of updating–resampling is to generate empirical samples of the posterior distribution. To do so, one would have to resample the high-dimensional flow posterior. General ideas for high-dimensional resampling have recently been proposed in N. Kantas, A. Beskos, and A. Jasra (2013, personal communication), and we imagine something similar would need to be employed when using the hybrid filter for realistic ocean models. The hybrid update with Gaussian resampling on the flow variables may be a promising approach for high-dimensional resampling; this would yield a spread of flow states from the posterior where EnKF localization could be applied. The use of Gaussian resampling for both the flow and drifter variables may also affect the performance of the hybrid filter adversely. Both of these points will need further investigation.

Another issue may arise if one relaxes the assumption that the flow field dynamics are close to linear. Under this assumption, we have shown that if flow variables are updated with the Kalman gain matrix, then their prior weights remain unchanged and become each ensemble member's respective posterior weight (details are explained in appendix A). It remains unclear that this is the correct approach if the prior distribution is far from Gaussian. Potential reweighting methods for the case of a non-Gaussian prior for the flow field are currently being investigated and will be the subject of future work.

Acknowledgments. The work of Apte benefited from the support of the AIRBUS Group Corporate Foundation Chair in Mathematics of Complex Systems established in ICTS-TIFR. Spiller would like to acknowledge support by NSF Grant DMS-1228265 and ONR Grant N00014-11-1-0087. Sandstede gratefully acknowledges support by the NSF through Grant DMS-0907904. Slivinski was supported by the NSF through Grants DMS-0907904 and DMS-1148284, and would also like to acknowledge support for a visit to TIFR Centre for Applicable Mathematics (TIFR-CAM) by the Virtual Institute for Mathematical and Statistical Sciences (VIMSS) under SAVI and NSF, as well as local support from TIFR-CAM. The authors thank Chris Jones for initially suggesting this collaboration, and the Mathematics and Climate Research Network (NSF Grant DMS-0940363) for enabling this joint effort. Finally, the authors thank the three anonymous referees for their helpful comments and suggestions.

APPENDIX A

EnKF with Weights

The EnKF update on a weighted ensemble has the same posterior mean and covariance as the traditional EnKF in the Gaussian case, and thus the correct Bayes's posterior mean and covariance, provided the observation perturbations are correctly defined. Here, we derive the correct observation perturbations in this case and show consistency with the traditional EnKF theory.

Suppose, at some time t_k, the true state of the system is $\mathbf{x}$ and we have an observation available given by $\mathbf{y} = \mathbf{H}\mathbf{x} + \boldsymbol{\epsilon}$, where $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{R})$. To represent the initial uncertainty in the true state, we have a weighted ensemble of states $\{\mathbf{x}_i^f, w_i\}_{i=1,\ldots,N_e}$, which represents a normal distribution with mean $\mathbf{m}_0$, covariance $\mathbf{C}_0$. That is,

$$\sum_{i=1}^{N_e} \mathbf{x}_i^f w_i = \mathbf{m}_0, \quad \sum_{i=1}^{N_e} (\mathbf{x}_i^f - \mathbf{m}_0)(\mathbf{x}_i^f - \mathbf{m}_0)^{\mathrm{T}} w_i = \mathbf{C}_0. \tag{A1}$$

The goal will be to show that, after updating the ensemble members themselves (but not the weights) via the Kalman update step, the posterior mean and covariance of the updated ensemble will be equivalent to the true Bayes's posterior mean and covariance. That is, we want the updated ensemble $\{\mathbf{x}_i^a, w_i\}_{i=1,\ldots,N_e}$ to have mean and covariance:

$$\mathbf{m}_1 = \mathbf{m}_0 + \mathbf{K}(\mathbf{y} - \mathbf{H}\mathbf{m}_0), \quad \mathbf{C}_1 = (\mathbf{I} - \mathbf{K}\mathbf{H})\mathbf{C}_0, \tag{A2}$$

where the Kalman gain matrix is, as usual, $\mathbf{K} = \mathbf{C}_0\mathbf{H}^{\mathrm{T}}(\mathbf{H}\mathbf{C}_0\mathbf{H}^{\mathrm{T}} + \mathbf{R})^{-1}$. The ensemble will be updated according to the ensemble Kalman update, in the perturbed observation format:

$$\mathbf{x}_i^a = \mathbf{x}_i^f + \mathbf{K}(\mathbf{y} - \mathbf{H}\mathbf{x}_i^f + \boldsymbol{\epsilon}_i). \tag{A3}$$

Then, the updated mean is

$$\mathbf{m}_1 = \sum_{i=1}^{N_e} \mathbf{x}_i^a w_i \tag{A4}$$

$$= \sum_{i=1}^{N_e} w_i[\mathbf{x}_i^f + \mathbf{K}(\mathbf{y} - \mathbf{H}\mathbf{x}_i^f + \boldsymbol{\epsilon}_i)] \tag{A5}$$

$$= \mathbf{m}_0 + \mathbf{K}(\mathbf{y} - \mathbf{H}\mathbf{m}_0) + \mathbf{K}\sum_{i=1}^{N_e} w_i\boldsymbol{\epsilon}_i . \tag{A6}$$

Thus, for the updated mean to coincide with the correct Bayesian posterior mean, we need $\sum_i w_i\boldsymbol{\epsilon}_i = 0$.

The updated covariance is now

$$\mathbf{C}_1 = \sum_{i=1}^{N_e} (\mathbf{x}_i^a - \mathbf{m}_1)(\mathbf{x}_i^a - \mathbf{m}_1)^{\mathrm{T}} w_i \tag{A7}$$

$$= \sum_{i=1}^{N_e} w_i[\mathbf{x}_i^f + \mathbf{K}(\mathbf{y} - \mathbf{H}\mathbf{x}_i^f + \boldsymbol{\epsilon}_i) - \mathbf{m}_0 - \mathbf{K}(\mathbf{y} - \mathbf{H}\mathbf{m}_0)](\cdot)^{\mathrm{T}} \tag{A8}$$

$$= \sum_{i=1}^{N_e} w_i[(\mathbf{I} - \mathbf{K}\mathbf{H})(\mathbf{x}_i^f - \mathbf{m}_0)](\cdot)^{\mathrm{T}} + \sum_{i=1}^{N_e} (\mathbf{I} - \mathbf{K}\mathbf{H})(\mathbf{x}_i^f - \mathbf{m}_0)(\mathbf{K}\boldsymbol{\epsilon}_i w_i)^{\mathrm{T}} + \sum_{i=1}^{N_e} [(\mathbf{I} - \mathbf{K}\mathbf{H})(\mathbf{x}_i^f - \mathbf{m}_0)]^{\mathrm{T}}\mathbf{K}\boldsymbol{\epsilon}_i w_i + \sum_{i=1}^{N_e} \mathbf{K}\boldsymbol{\epsilon}_i\boldsymbol{\epsilon}_i^{\mathrm{T}}\mathbf{K}^{\mathrm{T}} w_i , \tag{A9}$$

where $(\cdot)^{\mathrm{T}}$ represents the transpose of the same term in parentheses immediately preceding it. In this expression, the first term is equivalent to $(\mathbf{I} - \mathbf{KH})\mathbf{C}_0(\mathbf{I} - \mathbf{KH})^{\mathrm{T}}$ and the second and third terms are 0, as long as we assume independence between the noise terms $\boldsymbol{\epsilon}_i$ and the ensemble members $\mathbf{x}_i$. Now, if $\sum_i \boldsymbol{\epsilon}_i\boldsymbol{\epsilon}_i^{\mathrm{T}} w_i = \mathbf{R}$, then the final term reduces to $\mathbf{KRK}^{\mathrm{T}}$. Thus, we have

$$\mathbf{C}_1 = (\mathbf{I} - \mathbf{KH})\mathbf{C}_0(\mathbf{I} - \mathbf{KH})^{\mathrm{T}} + \mathbf{KRK}^{\mathrm{T}} = (\mathbf{I} - \mathbf{KH})\mathbf{C}_0 , \tag{A10}$$

as desired. Therefore, the weighted EnKF update step gives the correct posterior mean and covariance in the Gaussian case provided that the perturbations $\boldsymbol{\epsilon}_i$ satisfy

$$\sum_{i=1}^{N_e} \boldsymbol{\epsilon}_i w_i = \mathbf{0}, \quad \sum_{i=1}^{N_e} \boldsymbol{\epsilon}_i\boldsymbol{\epsilon}_i^{\mathrm{T}} w_i = \mathbf{R}. \tag{A11}$$

Essentially, the weighted ensemble $\{\boldsymbol{\epsilon}_i, w_i\}$ must approximate the Gaussian distribution with mean 0 and covariance $\mathbf{R}$.

We now briefly describe how we generate such an ensemble. First, draw a large (say 10^5) unweighted sample $\{\mathbf{z}_j\}$ from the target distribution; in our case, this is $\mathcal{N}(\mathbf{0}, \mathbf{R})$. Define g to be the probability density function of this distribution, and let $w_{\max}$ be the maximum weight over $\{w_i\}$. For each weight w_i, find $\mathbf{z}_j$ such that $|g(\mathbf{z}_j)/w_{\max} - w_i|$ is small. (This normalization allows the peak of the distribution function to have the same value as the maximum weight, and the rest of the distribution function is changed accordingly.) Let $\boldsymbol{\epsilon}_i = \mathbf{z}_j$, and repeat for $i = 1, \ldots, N_e$ (using the same unweighted sample).

APPENDIX B

Statistics of the Full and Averaged Ensembles

In this section we derive and compare the statistics for the full ensemble $\{\mathbf{x}_i^F, \mathbf{x}_{i,j}^D, w_{i,j}\}$ and for the averaged ensemble $\{\mathbf{x}_i^F, \tilde{\mathbf{x}}_i^D, \tilde{w}_i\}$. Let $\mathbf{x} = [\mathbf{x}^F, \mathbf{x}^D]^{\mathrm{T}}$, and consider the following decomposition of the covariance matrix into the flow–flow covariance, drifter–drifter covariance, and flow–drifter cross covariance:

$$\mathbf{P} = \begin{bmatrix} \mathbf{P}_{FF} & \mathbf{P}_{FD} \\ \mathbf{P}_{FD}^{\mathrm{T}} & \mathbf{P}_{DD} \end{bmatrix} . \tag{B1}$$

The full ensemble $\{\mathbf{x}_i^F, \mathbf{x}_{i,j}^D, w_{i,j}\}$ has mean and covariance

$$\overline{\mathbf{x}}_{\mathrm{full}}^F = \sum_i \mathbf{x}_i^F \tilde{w}_i , \tag{B2}$$

$$\overline{\mathbf{x}}_{\mathrm{full}}^D = \sum_{i,j} \mathbf{x}_{i,j}^D w_{i,j}, \quad \text{and} \tag{B3}$$

$$\mathbf{P}_{\mathrm{full}} = \sum_{i,j} w_{i,j}(\mathbf{x}_{i,j} - \overline{\mathbf{x}})(\mathbf{x}_{i,j} - \overline{\mathbf{x}})^{\mathrm{T}} . \tag{B4}$$

In particular,

$$\mathbf{P}_{FF,\mathrm{full}} = \sum_i \tilde{w}_i(\mathbf{x}_i^F - \overline{\mathbf{x}}^F)(\mathbf{x}_i^F - \overline{\mathbf{x}}^F)^{\mathrm{T}} , \tag{B5}$$

$$\mathbf{P}_{FD,\mathrm{full}} = \sum_{i,j} w_{i,j}(\mathbf{x}_i^F - \overline{\mathbf{x}}^F)(\mathbf{x}_{i,j}^D - \overline{\mathbf{x}}^D)^{\mathrm{T}}, \quad \text{and} \tag{B6}$$

$$\mathbf{P}_{DD,\mathrm{full}} = \sum_{i,j} w_{i,j}(\mathbf{x}_{i,j}^D - \overline{\mathbf{x}}^D)(\mathbf{x}_{i,j}^D - \overline{\mathbf{x}}^D)^{\mathrm{T}} . \tag{B7}$$

The averaged ensemble $\{\mathbf{x}_i^F, \tilde{\mathbf{x}}_i^D, \tilde{w}_i\}$ has mean

$$\overline{\mathbf{x}}_{\mathrm{avg}}^F = \sum_i \mathbf{x}_i^F \tilde{w}_i, \quad \overline{\mathbf{x}}_{\mathrm{avg}}^D = \sum_i \tilde{\mathbf{x}}_i^D \tilde{w}_i = \sum_{i,j} \mathbf{x}_{i,j}^D w_{i,j} , \tag{B8}$$

which is equivalent to the mean of the full ensemble, and the covariances are

$$\mathbf{P}_{FF,\mathrm{avg}} = \sum_i \tilde{w}_i(\mathbf{x}_i^F - \overline{\mathbf{x}}^F)(\mathbf{x}_i^F - \overline{\mathbf{x}}^F)^{\mathrm{T}}, \tag{B9}$$

$$\mathbf{P}_{FD,\mathrm{avg}} = \sum_i \tilde{w}_i(\mathbf{x}_i^F - \overline{\mathbf{x}}^F)(\tilde{\mathbf{x}}_i^D - \overline{\mathbf{x}}^D)^{\mathrm{T}}, \quad \text{and} \tag{B10}$$

$$\mathbf{P}_{DD,\mathrm{avg}} = \sum_i \tilde{w}_i(\tilde{\mathbf{x}}_i^D - \overline{\mathbf{x}}^D)(\tilde{\mathbf{x}}_i^D - \overline{\mathbf{x}}^D)^{\mathrm{T}}. \tag{B11}$$

Clearly, $\mathbf{P}_{FF,\mathrm{full}} = \mathbf{P}_{FF,\mathrm{avg}}$. We will show that $\mathbf{P}_{FD,\mathrm{full}} = \mathbf{P}_{FD,\mathrm{avg}}$ as well, but that $\mathbf{P}_{DD,\mathrm{full}} \neq \mathbf{P}_{DD,\mathrm{avg}}$. Indeed,

$$\mathbf{P}_{FD,\mathrm{full}} = \sum_{i,j} w_{i,j}(\mathbf{x}_i^F - \overline{\mathbf{x}}^F)(\mathbf{x}_{i,j}^D - \overline{\mathbf{x}}^D)^{\mathrm{T}} \tag{B12}$$

$$= \sum_i \left[(\mathbf{x}_i^F - \overline{\mathbf{x}}^F)\sum_j w_{i,j}(\mathbf{x}_{i,j}^D - \overline{\mathbf{x}}^D)^{\mathrm{T}}\right] \tag{B13}$$

$$\begin{aligned} &= \sum_i \left\{(\mathbf{x}_i^F - \overline{\mathbf{x}}^F)\sum_j [w_{i,j}(\mathbf{x}_{i,j}^D)^{\mathrm{T}}] - (\overline{\mathbf{x}}^D)^{\mathrm{T}}\tilde{w}_i\right\} \\ &= \mathbf{P}_{FD,\mathrm{avg}}, \end{aligned} \tag{B14}$$

as claimed.

After expanding Eqs. (B7) and (B11), only one term differs between the full distribution and the averaged distribution:

$$\begin{aligned} &|\mathbf{P}_{DD,\mathrm{full}} - \mathbf{P}_{DD,\mathrm{avg}}| \\ &= \left|\sum_{i,j} w_{i,j}(\mathbf{x}_{i,j}^D)(\mathbf{x}_{i,j}^D)^{\mathrm{T}} - \sum_i \tilde{w}_i(\tilde{\mathbf{x}}_i^D)(\tilde{\mathbf{x}}_i^D)^{\mathrm{T}}\right|. \end{aligned} \tag{B15}$$

Thus, this term determines how close the prior of the full distribution is to the prior of the averaged distribution.

REFERENCES

Ades, M., and P. van Leeuwen, 2013: An exploration of the equivalent weights particle filter. *Quart. J. Roy. Meteor. Soc.,* **139,** 820–840, doi:10.1002/qj.1995.

Anderson, J., 2007: An adaptive covariance inflation error correction algorithm for ensemble filters. *Tellus,* **59A,** 210–224, doi:10.1111/j.1600-0870.2006.00216.x.

——, and S. Anderson, 1999: A Monte Carlo implementation of the nonlinear filtering problem to produce ensemble assimilations and forecasts. *Mon. Wea. Rev.,* **127,** 2741–2758, doi:10.1175/1520-0493(1999)127<2741:AMCIOT>2.0.CO;2.

Apte, A., and C. Jones, 2013: The impact of nonlinearity in Lagrangian data assimilation. *Nonlinear Processes Geophys.,* **20,** 329–341, doi:10.5194/npg-20-329-2013.

——, ——, and A. Stuart, 2008: A Bayesian approach to Lagrangian data assimilation. *Tellus,* **60A,** 336–347, doi:10.1111/j.1600-0870.2007.00295.x.

Bengtsson, T., C. Snyder, and D. Nychka, 2003: Toward a nonlinear ensemble filter for high-dimensional systems. *J. Geophys. Res.,* **108,** 8775, doi:10.1029/2002JD002900.

——, P. Bickel, and B. Li, 2008: Curse of dimensionality revisted: Collapse of the particle filter in very large scale systems. *Probability and Statistics: Essays in Honor of David A. Freedman,* Vol. 2, D. Nolan and T. Speed, Eds., Institute of Mathematical Statistics, 316–334.

Bickel, P., B. Li, and T. Bengtsson, 2008: Sharp failure rates for the bootstrap particle filter in high dimensions. *Pushing the Limits of Contemporary Statistics: Contributions in Honor of Jayanta K. Ghosh,* Vol. 3, B. Clarke and S. Ghosal, Eds., Institute of Mathematical Statistics, 318–329.

Burgers, G., P. van Leeuwen, and G. Evensen, 1998: Analysis scheme in the ensemble Kalman filter. *Mon. Wea. Rev.,* **126,** 1719–1724, doi:10.1175/1520-0493(1998)126<1719:ASITEK>2.0.CO;2.

Doucet, A., N. de Freitas, K. Murphy, and S. Russell, 2000a: Rao–Blackwellised particle filtering for dynamic Bayesian networks. *Proc. 16th Conf. on Uncertainty in Artificial Intelligence,* San Francisco, CA, Association for Computing Machinery, 176–183. [Available online at http://dl.acm.org/citation.cfm?id=2073946.2073968.]

——, S. Godsill, and C. Andrieu, 2000b: On sequential Monte Carlo sampling methods for Bayesian filtering. *Stat. Comput.,* **10,** 197–208, doi:10.1023/A:1008935410038.

Dowd, M., 2007: Bayesian statistical data assimilation for ecosystem models using Markov chain Monte-Carlo. *J. Mar. Syst.,* **68,** 439–456, doi:10.1016/j.jmarsys.2007.01.007.

Evensen, G., 1994: Sequential data assimilation with a nonlinear quasi-geostrophic model using Monte Carlo methods to forecast error statistics. *J. Geophys. Res.,* **99,** 10 143–10 162, doi:10.1029/94JC00572.

——, 2003: The ensemble Kalman filter: Theoretical formulation and practical implementation. *Ocean Dyn.,* **53,** 343–367, doi:10.1007/s10236-003-0036-9.

Frei, M., and H. Künsch, 2012: Sequential state and observation noise covariance estimation using combined ensemble Kalman and particle filters. *Mon. Wea. Rev.,* **140,** 1476–1495, doi:10.1175/MWR-D-10-05088.1.

——, and ——, 2013a: Bridging the ensemble Kalman and particle filters. *Biometrika,* **100,** 781–800, doi:10.1093/biomet/ast020.

——, and ——, 2013b: Mixture ensemble Kalman filters. *Comput. Stat. Data Anal.,* **58,** 127–148, doi:10.1016/j.csda.2011.04.013.

Furrer, R., and T. Bengtsson, 2007: Estimation of high-dimensional prior and posterior covariance matrices in Kalman filter variants. *J. Multivariate Anal.,* **98,** 227–255, doi:10.1016/j.jmva.2006.08.003.

Gordon, N., D. Salmond, and A. Smith, 1993: Novel approach to nonlinear-non-Gaussian Bayesian state estimation. *IEE Proc. F,* **140,** 107–113, doi:10.1049/ip-f-2.1993.0015.

Hamill, T. M., and C. Snyder, 2000: A hybrid ensemble Kalman filter–3D variational analysis scheme. *Mon. Wea. Rev.,* **128,** 2905–2919, doi:10.1175/1520-0493(2000)128<2905:AHEKFV>2.0.CO;2.

——, J. S. Whitaker, and C. Snyder, 2001: Distance-dependent filtering of background error covariance estimates in an ensemble Kalman filter. *Mon. Wea. Rev.,* **129,** 2776–2790, doi:10.1175/1520-0493(2001)129<2776:DDFOBE>2.0.CO;2.

Houtekamer, P. L., and H. L. Mitchell, 1998: Data assimilation using an ensemble Kalman filter technique. *Mon. Wea. Rev.,* **126,** 796–811, doi:10.1175/1520-0493(1998)126<0796:DAUAEK>2.0.CO;2.

——, and ——, 2001: A sequential ensemble Kalman filter for atmospheric data assimilation. *Mon. Wea. Rev.,* **129,** 123–137, doi:10.1175/1520-0493(2001)129<0123:ASEKFF>2.0.CO;2.

Hunt, B., E. Kostelich, and I. Szunyogh, 2007: Efficient data assimilation for spatiotemporal chaos: A local ensemble transform Kalman filter. *Physica D,* **230,** 112–126, doi:10.1016/j.physd.2006.11.008.

Ide, K., L. Kuznetsov, and C. Jones, 2002: Lagrangian data assimilation for point vortex systems. *J. Turbul.,* **3,** doi:10.1088/1468-5248/3/1/053.

Kong, A., J. Liu, and W. Wong, 1994: Sequential imputations and Bayesian missing data problems. *J. Amer. Stat. Assoc.,* **89,** 278–288, doi:10.1080/01621459.1994.10476469.

Kuznetsov, L., K. Ide, and C. Jones, 2003: A method for assimilation of Lagrangian data. *Mon. Wea. Rev.,* **131,** 2247–2260, doi:10.1175/1520-0493(2003)131<2247:AMFAOL>2.0.CO;2.

Lawson, W., and J. Hansen, 2004: Implications of stochastic and deterministic filters as ensemble-based data assimilation methods in varying regimes of error growth. *Mon. Wea. Rev.,* **132,** 1966–1981, doi:10.1175/1520-0493(2004)132<1966:IOSADF>2.0.CO;2.

Mandel, J., L. Cobb, and J. Beezley, 2011: On the convergence of the ensemble Kalman filter. *Appl. Math.,* **56,** 533–541, doi:10.1007/s10492-011-0031-2.

Mariano, A., A. Griffa, T. Özgökmen, and E. Zambianchi, 2002: Lagrangian Analysis and Predictability of Coastal and Ocean Dynamics 2000. *J. Atmos. Oceanic Technol.,* **19,** 1114–1126, doi:10.1175/1520-0426(2002)019<1114:LAAPOC>2.0.CO;2.

Molcard, A., L. I. Piterbarg, A. Griffa, T. Özgökmen, and A. J. Mariano, 2003: Assimilation of drifter observations for the reconstruction of the Eulerian circulation field. *J. Geophys. Res.,* **108,** 3056, doi:10.1029/2001JC001240.

Morzfeld, M., X. Tu, E. Atkins, and A. J. Chorin, 2012: A random map implementation of implicit filters. *J. Comput. Phys.,* **231,** 2049–2066, doi:10.1016/j.jcp.2011.11.022.

Papadakis, N., E. Mémin, A. Cuzol, and N. Gengembre, 2010: Data assimilation with the weighted ensemble Kalman filter. *Tellus,* **62A,** 673–697, doi:10.1111/j.1600-0870.2010.00461.x.

Pedlosky, J., 1986: *Geophysical Fluid Dynamics*. 2nd ed. Springer, 710 pp.

Salman, H., 2008a: A hybrid grid/particle filter for Lagrangian data assimilation. I: Formulating the passive scalar approximation. *Quart. J. Roy. Meteor. Soc.,* **134,** 1539–1550, doi:10.1002/qj.270.

——, 2008b: A hybrid grid/particle filter for Lagrangian data assimilation. II: Application to a model vortex flow. *Quart. J. Roy. Meteor. Soc.,* **134B,** 1551–1565, doi:10.1002/qj.279.

——, L. Kuznetsov, C. Jones, and K. Ide, 2006: A method for assimilating Lagrangian data into a shallow-water-equation ocean model. *Mon. Wea. Rev.,* **134,** 1081–1100, doi:10.1175/MWR3104.1.

Snyder, C., T. Bengtsson, P. Bickel, and J. Anderson, 2008: Obstacles to high-dimensional particle filtering. *Mon. Wea. Rev.,* **136,** 4629–4640, doi:10.1175/2008MWR2529.1.

Spiller, E., A. Budhiraja, K. Ide, and C. Jones, 2008: Modified particle filter methods for assimilating Lagrangian data into a point-vortex model. *Physica D,* **237,** 1498–1506, doi:10.1016/j.physd.2008.03.023.

van Leeuwen, P., 2009: Particle filtering in geophysical systems. *Mon. Wea. Rev.,* **137,** 4089–4114, doi:10.1175/2009MWR2835.1.

Vernieres, G., C. Jones, and K. Ide, 2011: Capturing eddy shedding in the Gulf of Mexico from Lagrangian observations. *Physica D,* **240,** 166–179, doi:10.1016/j.physd.2010.06.008.

Xiong, X., I. Navon, and B. Uzunoglu, 2006: A note on the particle filter with posterior Gaussian resampling. *Tellus,* **58A,** 456–460, doi:10.1111/j.1600-0870.2006.00185.x.