Particle filtering in high-dimensional chaotic systems

Nishanth Lingala,[1, a)] N. Sri Namachchivaya,[1, b)] Nicolas Perkowski,[2, c)] and Hoong C. Yeong[1, d)]

[1)] *Department of Aerospace Engineering*
University of Illinois at Urbana-Champaign
306 Talbot Laboratory, MC-236
104 South Wright Street
Urbana, Illinois 61801, USA

[2)] *Institut für Mathematik*
Humboldt-Universität zu Berlin
Rudower Chaussee 25
12489 Berlin, Germany

We present an efficient particle filtering algorithm for multiscale systems, that is adapted for simple atmospheric dynamics models which are inherently chaotic. Particle filters represent the posterior conditional distribution of the state variables by a collection of particles, which evolves and adapts recursively as new information becomes available. The difference between the estimated state and the true state of the system constitutes the error in specifying or forecasting the state, which is amplified in chaotic systems that have a number of positive Lyapunov exponents. In this paper, we propose a reduced-order particle filtering algorithm based on the homogenized multiscale filtering framework developed in Imkeller *et al.*[1]. In order to adapt the proposed algorithm to chaotic signals, importance sampling and control theoretic methods are employed for the construction of the proposal density for the particle filter. Finally, we apply the general homogenized particle filtering algorithm developed here to the Lorenz'96 (Lorenz[2]) atmospheric model that mimics mid-latitude atmospheric dynamics with microscopic convective processes.

Keywords: particle filtering in high dimensions; chaotic systems; Lorenz'96; nonlinear filtering; dimensional reduction; homogenization

[a)] Electronic mail: lingala1@illinois.edu
[b)] Electronic mail: navam@illinois.edu
[c)] Electronic mail: perkowsk@math.hu-berlin.de
[d)] Electronic mail: hyeong2@illinois.edu

The future state of a dynamical system depends on the initial conditions, model errors, model parameters, and boundary conditions. Insufficient knowledge of these parameters leads to uncertainty in estimation and prediction of the system states. Data assimilation involves blending information from observations of the actual system states with information from dynamical models to estimate the current system states or certain model parameters. In this paper, we consider the data assimilation problem for multi-timescale systems, i.e. systems with inherent slow-fast dynamics with wide timescale separations. We present a particle filtering algorithm for multiscale systems, with an application to the Lorenz '96 model. The Lorenz '96 model is a system that mimics mid-latitude atmospheric dynamics and is regularly used as a testbed for data assimilation methods for high-dimensional chaotic systems. The particle filtering algorithm, the Homogenized Hybrid Particle Filter (HHPF), was developed based on the theoretical result that combined stochastic averaging with filtering theory to construct a reduced-dimension nonlinear filter for a high-dimensional multiscale system. In the context of filtering applications, the reduced-order filter indicates that estimation of the slow dynamics can be performed without requiring explicit knowledge of the fast dynamics, hence reducing computational complexities and information storage requirements. In the application to the Lorenz '96 model, the HHPF is shown to have the advantage of shorter computational time compared to existing nonlinear filtering methods with comparable level of estimation accuracy.

I. INTRODUCTION

The main goal of filtering is to obtain, recursively in time, the best statistical estimate of the state of a system based on noisy partial observations. The dynamics of the system is called the signal process. The best statistical estimate of the state (in mean square distance) is given by the conditional distribution of the signal given the observation process.

This paper deals with real time filtering of chaotic signals from atmospheric models involving many degrees of freedom. Since small errors in the estimate of the current state of a chaotic system can grow to have a major impact on the subsequent forecast, better estimate of the signal is needed to make accurate predictions of the future state. This is obviously a problem of significant importance in real time filtering and prediction applications, for example, for the weather and climate system, which involves coupled atmosphere-ocean systems, and for hazardous plumes or pollutants spread, which is governed by extremely complex flows.

Climate models can have millions of degrees of freedom. They contain noise terms that account for the interaction of resolved modes with unresolved modes. Hence, efficient algorithms to estimate the present and future state of the climate models based upon partial observations are required. While perfect determination of the state is impossible under these noisy observations, it is still desirable to obtain probabilistic estimates of the state conditioned on the information that is available.

The task of filtering such large-scale random dynamical systems is computationally overwhelming. Hence, we consider the Lorenz '96 model of type II (Lorenz[2]) with two time-scale simplified ordinary differential equation describing the advection, damping and forcing of some (slow) resolved atmospheric variables that are coupled to some (fast) sub-scale variables, as a nontrivial example of an atmospheric dynamics model. This model is a simple "toy model" of the atmosphere, exhibiting chaotic nature, and is an excellent testbed for schemes that will be developed in Section IV. Though simple, it still has a more realistic dimension compared to the multitude of "Lorenz 1963"-type studies (i.e., 360 dimensions rather than 3 dimensions). Even though this simple model is still a long way from the primitive equations in actual global circulation models, it is hoped that the filtering methods presented here can ultimately be adapted for filtering the realistic weather and climate prediction models.

Particle filtering method employs samples of particles to represent the posterior conditional distribution of state variables. The particle samples are evolved according to signal dynamics and updated as new information becomes available. Hence, particle filtering provides a recursive procedure for estimating an evolving state from a noisy observation process. Despite the general applicability and rigorous convergence results, particle filters have not been extensively used in high dimensional filtering problems, for example, weather prediction, as pointed out in Snyder *et al.*[3]. This is due to the fact that a large number of particles is required since the particle filters suffer from particle degeneracy (see, for example, Snyder *et al.*[3], Daum and Huang[4]) in high dimensional systems. In this paper, we combine our study of stochastic dimensional reduction and nonlinear filtering to provide a rigorous framework for developing an algorithm for computing lower dimensional particle filters that are specifically adapted to multiscale signal models. In such models, when the rates of change of variables differ by orders of magnitude, efficient data assimilation can be accomplished by constructing a particle filter for the nonlinear filtering equation for the coarse-grained (slow) signal. The approach described in this paper reduces the computational burden for real time applications and attempts to overcome the problem of particle degeneracy. The first objective is to predict the self-contained description of the coarse-grained dynamics without fully resolving the fast scale dynamics. The second objective is to adapt the filter for the coarse-grained prediction to problems with sensitivity to initial conditions.

We begin in Section II by presenting the general formulation of the multiscale nonlinear filtering problem. Here we introduce the homogenized equations that were derived in Imkeller *et al.*[1] for the reduced-dimension unnormalized filter. In Section III we present the model that was originally suggested by Lorenz (Lorenz'96) which possesses two distinct time scales. Section IV outlines a sequential particle filtering algorithm (the Sequential Importance Sampling method) and a numerical algorithm for dimensional reduction (the Heterogeneous Multiscale Method of Vanden-Eijnden[5], E, Liu, and Vanden-Eijnden[6]). These two methods are combined to give an efficient algorithm for particle filtering in a multiscale environment. Finally, we present in Section V, the results from several data assimilation experiments on the Lorenz '96 model.

II. FORMULATION OF MULTISCALE NONLINEAR FILTERING PROBLEMS

The results presented here are set within the context of slow-fast dynamical systems, where the rates of change of different variables differ by orders of magnitude. The effects of the multiscale signal and observation processes were presented in Park, Sowers, and Namachchivaya[7] and Imkeller *et al.*[1], where a lower dimensional Zakai equation was derived in the limit. The dimensional reduction for nonlinear filters presented in Park, Sowers, and Namachchivaya[7] for two-dimensional signal processes was generalized to the multidimensional case in Imkeller *et al.*[1], where also an explicit rate of convergence could be obtained. In general, these results provide the mathematical rigor that supports numerical algorithms based on the idea that stochastically averaged models are potentially helpful in developing inexpensive lower-dimensional filtering algorithms as demonstrated by Park, Namachchivaya, and Yeong[8] in the context of homogenized particle filters. But while the algorithm in Park, Namachchivaya, and Yeong[8] is efficient for filtering high-dimensional multiscale signals with only negative Lyapunov exponents, it performs rather poorly when applied to chaotic signals, as we will point out below. For this reason the algorithm has to be adapted by incorporating importance sampling and or optimal control techniques. This is the purpose of the present paper, where we show that after suitably adapting the HHPF, it is an efficient algorithm for filtering high-dimensional chaotic multiscale signals.

We will present the main result of Imkeller *et al.*[1] here. Note that these results are formulated in continuous time. This is the most difficult case, because then we need to deal with stochastic partial differential equations that describe the evolution of the conditional distributions. The discrete time observation case can be obtained as a simple corollary of the continuous time case, as it is well known that it is possible to discretize the observation in time and obtain the real conditional distribution in the limit as the mesh size of the time partition goes to zero (see, for example, Crisan[9]). Conversely we can always interpret a discrete time observation as discretized continuous time observation.

Let $(\Omega, \mathcal{F}, (\mathcal{F}_t), \mathbb{Q})$ be a filtered probability space that supports a standard Brownian motion $(V, W, B) \in \mathbb{R}^{k \times l \times d}$. We consider the signal process that is the solution of the two time scale stochastic differential equation (SDE) driven by (W, V):

$$dX_t^\varepsilon = b(X_t^\varepsilon, Z_t^\varepsilon)dt + \sigma(X_t^\varepsilon, Z_t^\varepsilon)dV_t, \quad X_0^\varepsilon = \xi \in \mathbb{R}^m \tag{1a}$$

$$dZ_t^\varepsilon = \frac{1}{\varepsilon} f(X_t^\varepsilon, Z_t^\varepsilon)dt + \frac{1}{\sqrt{\varepsilon}} g(X_t^\varepsilon, Z_t^\varepsilon)dW_t, \quad Z_0^\varepsilon = \eta \in \mathbb{R}^n. \tag{1b}$$

Here, $\varepsilon << 1$ is the timescale separation parameter, so X^ε is the slow component and Z^ε is the fast component. W, V and B are independent of each other as well as of the random initial conditions ξ and η.

Associated to the signal is the d-dimensional observation Y^ε that is perturbed by the Brownian motion B, given by

$$Y_t^\varepsilon = \int_0^t h(X_s^\varepsilon, Z_s^\varepsilon)ds + B_t.$$

Define the sigma algebra generated by the observation, $\mathcal{Y}_t^\varepsilon = \sigma(Y_s^\varepsilon : 0 \leq s \leq t) \vee \mathcal{N}$, where $\mathcal{N}$ are the $\mathbb{Q}$-negligible sets.

The optimal filter, which provides the best L^2-estimate of the signal, is a conditional expectation that satisfies a recursive equation driven by the observation (the Kushner-Stratonovich equation, Kushner[10]). It is given as the measure-valued process π^ε, defined by $\pi^\varepsilon_t(\varphi) \overset{\text{def}}{=} \mathbb{E}_{\mathbb{Q}}[\varphi(X^\varepsilon_t, Z^\varepsilon_t)|\mathcal{Y}^\varepsilon_t]$ for every bounded and measurable function φ on $\mathbb{R}^{m+n}$.

In order to calculate the specified conditional expectation, it is easier to work on a new probability space $\mathbb{P}^\varepsilon$, on which the observation Y^ε is a Brownian motion independent of the signal noise (W, V). $\mathbb{P}^\varepsilon$ is related to the original probability space $\mathbb{Q}$ by the Girsanov transform (Bain and Crisan[11])

$$D^\varepsilon_t \overset{\text{def}}{=} \left.\frac{d\mathbb{P}^\varepsilon}{d\mathbb{Q}}\right|_{\mathcal{F}_t} = \exp\left(-\int_0^t h(X^\varepsilon_s, Z^\varepsilon_s)^T dB_s - \frac{1}{2}\int_0^t |h(X^\varepsilon_s, Z^\varepsilon_s)|^2 ds\right), \tag{2}$$

which effectively removes the drift h from the observation equation. $\cdot^T$ denotes the transpose of a matrix or a vector. Define an unnormalized filter as $\rho^\varepsilon_t(\varphi) \overset{\text{def}}{=} \mathbb{E}_{\mathbb{P}^\varepsilon}[\varphi(X^\varepsilon_t, Z^\varepsilon_t)\,(D^\varepsilon_t)^{-1}\,|\mathcal{Y}^\varepsilon_t]$. The normalized and unnormalized filters are related using the Girsanov transform:

$$\pi^\varepsilon_t(\varphi) = \mathbb{E}_{\mathbb{Q}}[\varphi(X^\varepsilon_t, Z^\varepsilon_t)|\mathcal{Y}^\varepsilon_t] = \frac{\mathbb{E}_{\mathbb{P}^\varepsilon}[\varphi(X^\varepsilon_t, Z^\varepsilon_t)(D^\varepsilon_t)^{-1}|\mathcal{Y}^\varepsilon_t]}{\mathbb{E}_{\mathbb{P}^\varepsilon}[(D^\varepsilon_t)^{-1}|\mathcal{Y}^\varepsilon_t]} = \frac{\rho^\varepsilon_t(\varphi)}{\rho^\varepsilon(1)}.$$

The unnormalized filter ρ^ε satisfies the Zakai equation (see, for example, Zakai[12], Bain and Crisan[11]):

$$d\rho^\varepsilon_t(\varphi) = \rho^\varepsilon_t(\mathcal{L}^\varepsilon\varphi)dt + \rho^\varepsilon_t(h\varphi)dY^\varepsilon_t, \quad \rho^\varepsilon_0(\varphi) = \mathbb{E}_{\mathbb{Q}}[\varphi(X^\varepsilon_0, Z^\varepsilon_0)]$$

for sufficiently regular test functions φ. Here, $\mathcal{L}^\varepsilon = \frac{1}{\varepsilon}\mathcal{L}_F + \mathcal{L}_S$ is the differential operator associated to $(X^\varepsilon, Z^\varepsilon)$, with

$$\begin{aligned}\mathcal{L}_F &= \sum_{i=1}^n f_i(x,z)\frac{\partial}{\partial z_i} + \frac{1}{2}\sum_{i,j=1}^n (gg^T)_{ij}(x,z)\frac{\partial^2}{\partial z_i \partial z_j},\\ \mathcal{L}_S &= \sum_{i=1}^m b_i(x,z)\frac{\partial}{\partial x_i} + \frac{1}{2}\sum_{i,j=1}^m (\sigma\sigma^T)_{ij}(x,z)\frac{\partial^2}{\partial x_i \partial x_j}.\end{aligned}$$

We assume that for every $x \in \mathbb{R}^m$, the process Z^x which is the solution of (1b) with $X^\varepsilon = x$ fixed satisfies the Doeblin condition, i.e. Z^x is ergodic and converges exponentially fast to its unique stationary distribution $p_\infty(x, \cdot)$. In this case, it is well known that X^ε converges in distribution to a diffusion X^0 governed by an SDE

$$dX^0_t = \bar{b}(X^0_t)dt + \bar{\sigma}(X^0_t)dV_t$$

for appropriately averaged $\bar{b}$ and $\bar{\sigma}$. In other words, a stochastically averaged model provides a qualitatively useful approximation to the actual multiscale system. Hence, if we are only interested in estimating the *slow* process, or the "coarse-grained" dynamics, then we should make use of the homogenized diffusion X^0. Specifically, if we are only interested in the x-marginal, $\pi^{\varepsilon,x}$, of the optimal π^ε, then this X^0 can be used to construct a homogenized filter π^0 that can appropriately replace $\pi^{\varepsilon,x}$. In high-dimensional problems, π^0 would be preferable to $\pi^{\varepsilon,x}$ since calculation of π^0 using X^0 does not directly involve calculations of the fast process.

In making use of X^0, we would like to find a homogenized (unnnormalized) filter ρ^0 that satisfies

$$d\rho_t^0(\varphi) = \rho_t^0(\bar{\mathcal{L}}\varphi)dt + \rho_t^0(\bar{h}\varphi)dY_t^\varepsilon, \quad \rho_0^0(\varphi) = \mathbb{E}_{\mathbb{Q}}[\varphi(X_0^0)],$$

such that for small ε, the x-marginal of ρ^ε, $\rho^{\varepsilon,x}$, is close to ρ^0. We let the generator $\bar{\mathcal{L}}$ of X^0 be defined as

$$\bar{\mathcal{L}} = \sum_{i=1}^m \bar{b}_i(x)\frac{\partial}{\partial x_i} + \frac{1}{2}\sum_{i,j=1}^m \bar{a}_{ij}(x,z)\frac{\partial^2}{\partial x_i \partial x_j},$$

where $\bar{b}(x) = \int b(x,z)p_\infty(x,dz)$ and $\bar{a} = \int(\sigma\sigma^T)(x,z)p_\infty(x,dz)$. Also, define $\bar{h}(x) = \int h(x,z)p_\infty(x,dz)$. Note that the homogenized filter is still driven by the real observation Y^ε and not by a "homogenized observation", which is practical for implementation of the homogenized filter in applications since such averaged observation is usually not available. However, even if such homogenized observation was available, using it would lead to loss of information for estimating the signal compared to using the actual observation.

Now define the measure-valued processes π^0 and $\pi^{\varepsilon,x}$ in terms of ρ^0 and $\rho^{\varepsilon,x}$ as π^ε was in terms of ρ^ε:

$$\pi_t^0(\varphi) = \frac{\rho_t^0(\varphi)}{\rho_t^0(1)} \qquad \text{and} \qquad \pi_t^{\varepsilon,x}(\varphi) = \frac{\rho_t^{\varepsilon,x}(\varphi)}{\rho_t^{\varepsilon,x}(1)}.$$

The main result of Imkeller *et al.*[1] is that under appropriate assumptions on the coefficients of (1a), (1b) and h, for every $T \geq 0$ there exists $C > 0$ such that

$$\mathbb{E}_{\mathbb{Q}}\left[d(\pi_T^{\varepsilon,x}, \pi_T^0)\right] \leq \sqrt{\varepsilon}C, \text{ i.e. } \lim_{\varepsilon\to 0}\mathbb{E}_{\mathbb{Q}}\left[d(\pi_T^{\varepsilon,x}, \pi_T^0)\right] = 0,$$

where d is a metric that generates the topology of weak convergence on the space of probability measures. In other words, the x-marginal of the optimal filter π^ε converges to the averaged filter π^0 in the space of probability measures as the separation parameter ε goes to 0. Hence, it is appropriate to use the homogenized filter in place of the actual $\pi^{\varepsilon,x}$ in estimating the "coarse-grained" dynamics X^ε in a setting with wide timescale separation. In terms of filtering applications, π^0 presents the advantage of not requiring exact knowledge of the fast dynamics for the purpose of estimating the "coarse-grained" dynamics. Only knowledge of the invariant measure of Z^x is required, hence by applying appropriate multiscale averaging numerical schemes, computation and information storage for the fast dynamics can be reduced.

An outline of the proof in Imkeller *et al.*[1] is presented in the appendix.

III. THE LORENZ '96 SYSTEM

In this paper, we consider a simple atmospheric model, which exhibits many of the difficulties that arise in realistic models, to gain insight into data assimilation for high-dimensional multiscale models. The Lorenz '96 model was originally introduced (in Lorenz[2]) to mimic multiscale mid-latitude weather, considering an unspecified scalar meteorological quantity X in K sectors (with K ranging from 4 to 36) along a latitude circle:

$$\dot{X}_t^k = -X_t^{k-1}(X_t^{k-2} - X_t^{k+1}) - X_t^k + F_x + \frac{h_x}{J}\sum_{j=1}^{J} Z_t^{k,j}, \quad k = 1, \ldots, K, \tag{3}$$

$$\dot{Z}_t^{k,j} = \frac{1}{\varepsilon}\left\{-Z_t^{k,j+1}(Z_t^{k,j+2} - Z_t^{k,j-1}) - Z_t^{k,j} + h_z X_t^k\right\}, \quad j = 1, \ldots, J. \tag{4}$$

X_t^k represents the value of X at time t, in the k^{th} sector. Each X_t^k is coupled to its neighbors X_t^{k+1}, X_t^{k-1}, and X_t^{k-2}. Equation (3) applies for all values of k by letting $X_t^{k+K} = X_t^{k-K} = X_t^k$, so, for example for $K = 36$, the $k = 1$ case is $X_t^{k+1} = X_t^2$, $X_t^{k-1} = X_t^{36}$, and $X_t^{k-2} = X_t^{35}$.

Each sector is divided into J subsectors ($J = 4$ to $J = 10$) and the fast variables, $\{Z_t^{k,j}\}_{j=1}^J$, given by (4), are associated with each subsector. Thus, each X_t^k represents a slowly-varying, large amplitude atmospheric quantity, with J fast-varying, low amplitude quantities, $\{Z_t^{k,j}\}_{j=1}^J$, associated with it. The coupling terms between neighbors model advection between sectors and subsectors, while the coupling between each sector and its subsectors model damping within the model. The model is subjected to linear external forcing, F_x, on the slow timescale.

The dynamics of the fast process can be modified to include nonlinear self-interaction effects by adding forcing in the form of stochastic terms (see, for example, Majda, Timofeyev, and Vanden-Eijnden[13,14]). This is appropriate if we are only interested in the coarse-grained dynamics occuring in the slow timescale. This is called stochastic consistency in Majda, Timofeyev, and Vanden-Eijnden[14]. Considering (4), where only a quadratic nonlinearity is present, the motivation behind adding stochastic forcing is thus to model higher order self-interaction effects. This can be done using a mean-zero Ornstein-Uhlenbeck process:

$$\dot{Z}_t^{k,j} = \frac{1}{\varepsilon}\left\{-Z_t^{k,j+1}(Z_t^{k,j+2} - Z_t^{k,j-1}) - Z_t^{k,j} + h_z X_t^k\right\} + \frac{1}{\sqrt{\varepsilon}}\zeta(t). \tag{5}$$

The two-scale Lorenz '96 model has been used extensively by several authors: Wilks[15] to study stochastic parametrization, Lorenz and Emanuel[16] for analyzing targeted observations, and Herrera *et al.*[17] and several others for analyzing the influence of large-scale spatial patterns on the growth of small perturbations. This model has also been used as a testbed for nonlinear filtering algorithms, for example, the reduced-order ensemble Kalman filters, in Harlim and Majda [18] and Kang and Harlim [19].

IV. HOMOGENIZED HYBRID PARTICLE FILTER (HHPF)

The results presented in Section II allow us to combine homogenization schemes with nonlinear filtering methods to develop efficient filters that are adapted to multiscale signals. The filtering algorithm proposed in this paper employs the Heterogeneous Multiscale Method (HMM) as a numerical scheme for homogenization, and Sequential Importance Sampling (SIS) using particles as a nonlinear filtering method. In the following two subsections we explain the HMM and SIS. These techniques are then combined in the Homogenized Hybrid Particle Filter (HHPF).

To illustrate the homogenization scheme, we consider a general system of stochastic differential equations (SDEs) that corresponds with the problem of Section III, with stochastic forcing in the fast component:

$$\begin{aligned} \dot{X}_t^\varepsilon &= b(X_t^\varepsilon, Z_t^\varepsilon), \quad X_0^\varepsilon \sim \mathcal{N}(\mu_0^x, \sigma_0^x) \\ \dot{Z}_t^\varepsilon &= \varepsilon^{-1} f(X_t^\varepsilon, Z_t^\varepsilon) + \varepsilon^{-1/2} g(X_t^\varepsilon, Z_t^\varepsilon)\dot{W}_t, \quad Z_0^\varepsilon \sim \mathcal{N}(\mu_0^z, \sigma_0^z). \end{aligned} \tag{6}$$

Following the results of Section II the homogenized system is

$$\dot{\bar{X}}_t = \bar{b}(\bar{X}_t), \tag{7}$$

where

$$\bar{b}(x) = \int b(x,z)\, p_\infty(x,dz). \tag{8}$$

It is well-known that, as $\varepsilon \to 0$, X^ε converges to $\bar{X}$ in probability. The difference $\zeta := (X^\varepsilon - \bar{X})/\sqrt{\varepsilon}$ converges to a Gaussian process satisfying

$$d\zeta_t = \frac{\partial \bar{b}}{\partial x}(\bar{X}_t)\zeta_t dt + \sigma(\bar{X}_t)dW_t, \qquad \zeta_0 = 0,$$

where $\sigma\sigma^T := \lim_{T\to\infty} \frac{1}{T}\int_0^T\int_0^T cov(b(x,Z_s), b(x,Z_t))dsdt$ (see Khas'minskii[20]).

Since we will apply importance sampling (see below) for the reasons mentioned in Section I, we need to add a noise term to the coarse-grained signal in order for the filtering algorithm to work. From the homogenization perspective, the addition of noise to the averaged slow signal is discussed in Hasselmann[21], which has been studied extensively in the literature. Instead of adding the term in the averaged equations, we add a Brownian forcing of the form $\dot{V}_t$, where V is independent of W, to the slow signal in the original model, in order to apply importance sampling scheme.

The system becomes

$$\begin{aligned} \dot{X}_t^\varepsilon &= b(X_t^\varepsilon, Z_t^\varepsilon) + \dot{V}_t, \quad X_0^\varepsilon \sim \mathcal{N}(\mu_0^x, \sigma_0^x) \\ \dot{Z}_t^\varepsilon &= \varepsilon^{-1} f(X_t^\varepsilon, Z_t^\varepsilon) + \varepsilon^{-1/2} g(X_t^\varepsilon, Z_t^\varepsilon)\dot{W}_t, \quad Z_0^\varepsilon \sim \mathcal{N}(\mu_0^z, \sigma_0^z). \end{aligned} \tag{9}$$

However, the algorithm proposed in this paper works for noise with any diffusion matrix, as long as it is uniformly elliptic. The homogenized system, after adding noise, becomes

$$\dot{\bar{X}}_t = \bar{b}(\bar{X}_t) + \dot{V}_t, \quad \bar{X}_0 \sim \mathcal{N}(\mu_0^x, \sigma_0^x). \tag{10}$$

For the applications we discretize and regularize the system and the discretized version of equation (10) is given by

$$\bar{X}_{t_k} = \bar{X}_{t_{k-1}} + \bar{b}\left(\bar{X}_{t_{k-1}}\right)(t_k - t_{k-1}) + (V_{t_k} - V_{t_{k-1}}) \tag{11}$$

with the same initial conditions. We also discretize the observation equation

$$Y_t^\varepsilon = \int_0^t h(X_s^\varepsilon, Z_s^\varepsilon)ds + \sigma_y B_t, \tag{12}$$

into discrete time steps:

$$\tilde{Y}_{t_k}^\varepsilon = Y_{t_k}^\varepsilon - Y_{t_{k-1}}^\varepsilon = \int_{t_{k-1}}^{t_k} h(X_s^\varepsilon, Z_s^\varepsilon)ds + \sigma_y(B_{t_k} - B_{t_{k-1}}). \tag{13}$$

Then clearly $\sigma(Y^\varepsilon_{t_0}, \ldots, Y^\varepsilon_{t_k}) = \sigma(\tilde{Y}^\varepsilon_{t_0}, \ldots, \tilde{Y}^\varepsilon_{t_k})$. The quantity $\sigma_y(B_{t_k} - B_{t_{k-1}})$ is a Gaussian random variable of variance $(\sigma_y \sigma_y^T)(t_k - t_{k-1})$. Because Z^ε satisfies the Doeblin condition, we can approximate

$$\int_{t_{k-1}}^{t_k} h(X^\varepsilon_s, Z^\varepsilon_s) ds \simeq \bar{h}(X^\varepsilon_{t_{k-1}})(t_k - t_{k-1}). \tag{14}$$

Therefore we reduced our continuous time model to a discrete time signal, discrete time observation model, with which we will work from now on.

A. Multiscale numerical integration

It is usually impossible to obtain the invariant distribution $p_\infty(x, dz)$ of the fast variable in (8) analytically. Hence, we adopt the HMM introduced in Vanden-Eijnden[5], which is a method of determining the effective dynamics (7) through numerical approximation of the invariant distribution of the fast process. It is based on the observation that the fast variable Z^ε reaches its invariant distribution (equilibrium) on a time-scale much smaller than the time-scale needed to evolve the slow variable X^ε (the Doeblin condition).

Following the procedure presented in Vanden-Eijnden[5], we describe the HMM. The evolution of the averaged equation (7) is approximated numerically using a forward integration scheme. For simplicity, we will use the Euler and Euler-Maruyama schemes in the following description, although higher order schemes are also applicable (see, for example, Kloeden and Platen[23]). Consider the interval $[0, T]$ discretized into timesteps of size $\Delta t = \lfloor \frac{T}{N} \rfloor$. Let $t_k \overset{\text{def}}{=} k\Delta t$ and write X_{t_k} as X_k. The timestep Δt is called the *macro-timestep* (*Note:* subscript k here indicates time index, different from the spatial index superscript k in the Lorenz '96 model in Section III). Each macro-timestep interval $[k\Delta t, (k+1)\Delta t]$, $k = 0, 1, \ldots$ is further divided into micro-timestep intervals $[j\delta t, (j+1)\delta t]$, $j = 0, 1, \ldots, N_\delta := \lfloor \frac{\Delta t}{\delta t} \rfloor$. For a fixed k, the fast process is evolved on micro-timestep intervals according to the *micro-solver*

$$Z^\varepsilon_{k,j+1} = Z^\varepsilon_{k,j} + \frac{1}{\varepsilon} f(\bar{X}_k, Z^\varepsilon_{k,j})\delta t + \frac{1}{\sqrt{\varepsilon}} g(\bar{X}_k, Z^\varepsilon_{k,j})\delta W. \tag{15}$$

The averaged coefficient $\bar{b}$ in (8) and the averaged sensor function $\bar{h}$ in (14) can be approximated by $\tilde{b}$ and $\tilde{h}$ as

$$\tilde{b}(\bar{X}_k) = \frac{1}{MN_m} \sum_{r=1}^{M} \sum_{j=n_T}^{n_T+N_m} b(\bar{X}_k, Z^{\varepsilon,r}_{k,j}), \tag{16}$$

$$\tilde{h}(X_k) = \frac{\Delta t}{M N_m} \sum_{r=1}^{M} \sum_{j=n_T}^{n_T+N_m} h(\bar{X}_k, Z^{\varepsilon}_{k,j,r}), \tag{17}$$

where M is the number of replicas of the fast process Z for spatial averaging, N_m is the number of *micro-timesteps* δt for time averaging, and n_T is the number of micro-timesteps skipped to eliminate transient effects. The homogenized process is integrated forward using (11) with $\tilde{b}$ in place of $\bar{b}$.

The advantage of the HMM method is that, due to the Doeblin condition on the fast process, $n_T + N_m$ can be selected much smaller than N_δ. Note that a combination of spatial and temporal averaging is used in (16) but it is shown in Vanden-Eijnden[5] that the combinations of M, N_m, and δt can be chosen based on error analysis such that no spatial averaging or no spatial and temporal averaging is required. For more detailed explanation and error analysis, we refer the reader to the references Vanden-Eijnden[5], Fatkullin and Vanden-Eijnden[24].

B. Homogenized Hybrid Particle Filter (HHPF)

The algorithm for the continuous-time HHPF is presented in Park, Namachchivaya, and Yeong[8]. In high dimensional chaotic systems, the HHPF algorithm as presented in Park, Namachchivaya, and Yeong[8] would encounter the problem of particle collapse, i.e. very few particles receive large fraction of the weight. In order to avoid this, the particles should be directed towards regions of high likelihood. One method to do this is to apply a continuous time control on the particles and compensate for the control by multiplying the weights with the appropriate Radon-Nikodym derivative. The other alternative is to discretize the continuous-time equations as in (11) and perform filtering on the resulting discrete-time models employing importance sampling. The proposal density involved in the importance sampling can be obtained by applying a control on the particles. In this paper, we follow the second approach.

Here, we present the discrete-time version of the HHPF using the SIS algorithm (see, for example, Arulampalam *et al.*[25], Gordon, Salmond, and Smith[26]). We will first provide a brief overview of the idea behind importance sampling and how it is applied sequentially in particle filters. Then we present how the HHPF uses the SIS algorithm.

1. Importance Sampling

The simplest form of particle filtering is the following: Approximate the posterior density at t_{k-1} by a weighted collection of particles $\pi_{t_{k-1}} = \sum_i w_i \delta_{x^i_{k-1}}(x)$. Simulate the particles according to the system equations and arrive at the new locations x^i_k. Then, update the weights according to the Bayes' rule: $w^i_k \propto p\left(y_k|\, x^i_k\right) p\left(x^i_k|\, x^i_{k-1}\right) w^i_{k-1}$ where $p\left(y_k|\, x^i_k\right)$ is called likelihood and is given by the sensor dynamics and $p\left(x^i_k|\, x^i_{k-1}\right)$ is called prior and is given by the system dynamics.

The above procedure faces the well known problem of particle collapse i.e. very few of the particles end up close to the actual location, and hence very few receive large fractions of the total weight. A remedy for this problem is importance sampling.

Importance sampling is a technique for approximating integrals with respect to one probability distribution using a collection of samples from another. Let p be the target distribution of interest over space $\mathbb{X}$ and $q \gg p$ (p is absolutely continuous with respect to q) be the distribution from which sampling is done (q is also called the proposal distribution). Denote by $\mathbb{E}_p[.]$ and $\mathbb{E}_q[.]$ the expectation with respect to the distributions p and q, respectively. For any integrable function $\varphi : \mathbb{X} \to \mathbb{R}$, we have

$$\begin{aligned} \mathbb{E}_p\left[\varphi(X)\right] &= \int_{\mathbb{X}} \varphi(x) p(dx) = \int_{\mathbb{X}} \varphi(x) \frac{dp}{dq}(x) q(dx) \\ &= \int_{\mathbb{X}} \varphi(x) w(x) q(dx) = \mathbb{E}_q\left[w(X)\varphi(X)\right], \end{aligned} \tag{18}$$

where $w \overset{\text{def}}{=} \frac{dp}{dq}$. A collection $\{x^i\}_{i=1}^{N_s}$ of N_s particles can be sampled from q and the particles can be weighted according to $w^i \propto \frac{dp}{dq}(x^i)$ to represent the target distribution p.

Incorporating a proposal density q into Bayes' rule, we have for the weight update equation (see, for example, Arulampalam *et al.*[25])

$$w^i_k \propto \frac{p\left(y_k|\, x^i_k\right) p\left(x^i_k|\, x^i_{k-1}\right)}{q\left(x^i_k|\, x^i_{k-1}, y_k\right)} w^i_{k-1}. \tag{19}$$

So, the SIS algorithm is this: At step $k-1$ the location of particles $\{x^i_{k-1}\}_{i=1}^{N_s}$ is known. New observation y_k is recorded. Choose a form for the proposal $q(\cdot|x^i_{k-1}, y_k)$. Sample a particle according to this proposal density and get the new location x^i_k. Then update the weights according to equation (19).

In filtering with a fixed number of particles, it is crucial to keep the variance of the weights to a minimum: if a collection of particles is such that the weight is concentrated in a small number of particles, this collection represents the distribution poorly. As can be seen from the weight update equation (19), the particle weights depend crucially on the choice of the proposal distribution q. So, q should be chosen with an aim of minimizing the variance of particle weights.

It can be shown (see, for example, Arulampalam *et al.*[25]) that, conditioned on the previous location of the particle, the proposal density which keeps the variance of the weights to a minimum is

$$q^{\mathrm{opt}}(x_k|x_{k-1},y_k) = \frac{p\left(y_k|\,x_k\right)p\left(x_k|\,x_{k-1}\right)}{\int p\left(y_k|\,x_k\right)p\left(x_k|\,x_{k-1}\right)dx_k}. \tag{20}$$

Employing this in weight update equation (19) we have that, if we choose the optimal proposal density, then

$$w_k^i \propto w_{k-1}^i \int p\left(y_k|\,x_k\right)p\left(x_k|\,x_{k-1}^i\right)dx_k. \tag{21}$$

This choice of importance density is optimal in the sense that given x_{k-1}^i, w_k^i takes the same value whatever sample is drawn from $q^{\mathrm{opt}}(x_k|x_{k-1}^i,y_k)$. In general it is difficult to sample from q^{opt}. However, it is easy when both the likelihood $p\left(y_k|\,x_k\right)$ and the conditional prior $p\left(x_k|\,x_{k-1}\right)$ are Gaussians (see, for example, Arulampalam *et al.*[25]).

Consider the system dynamics and observation equation:

$$X_{k+1} = F(X_k) + \sigma_X W_{k+1} \tag{22}$$

$$Y_{k+1} = HX_{k+1} + \sigma_Y V_{k+1} \tag{23}$$

where W_k and V_k are independent centered Gaussian increments with variance I and $Q \overset{\mathrm{def}}{=} \sigma_X\sigma_X^T$, $R \overset{\mathrm{def}}{=} \sigma_Y\sigma_Y^T$ are strictly positive definite. Since this subsection on SIS and the next one on optimal control discuss a general method and we are working with discretized models, we have used F to denote the vector field instead of b and f as in the previous sections. We then have that the likelihood $p(y_k|x_k) = \mathcal{N}(Hx_k, R)$ and the conditional prior $p\left(x_k|\,x_{k-1}\right) = \mathcal{N}(F(X_{k-1}), Q)$. Using (20), we have

$$\begin{aligned} q^{\mathrm{opt}}(x_k|x_{k-1},y_k) &= \mathcal{N}(F(x_{k-1}) + \alpha(x_{k-1},y_k), \hat{Q}), \\ \hat{Q} &= \left(Q^{-1} + H^T R^{-1} H\right)^{-1}, \\ \alpha(x_{k-1},y_k) &= \hat{Q}H^T R^{-1}(y_k - HF(x_{k-1})). \end{aligned} \tag{24}$$

This q^{opt} is a Gaussian. Once we have particle locations $\{x^i_{k-1}\}_{i=1}^{N_s}$ representing the posterior at time $k-1$, and the observation y_k is recorded, the particles can be evolved according to

$$X_{k+1} = F(X_k) + \alpha(X_k, y_{k+1}) + \hat{\sigma}_X W_{k+1} \tag{25}$$

where $\hat{\sigma}_X$ is such that $\hat{\sigma}_X \hat{\sigma}_X^T = \hat{Q}$. Then X^i_k behaves like a particle sampled from $q^{\mathrm{opt}}(\cdot | x^i_{k-1}, y_k)$. The weights are updated according to (21): we will have

$$\begin{aligned} w^i_k \;&\propto\; w^i_{k-1} \exp\left\{ -\frac{1}{2}(y_k - HF(x^i_{k-1}))^T \hat{R}^{-1}(y_k - HF(x^i_{k-1})) \right\}, \\ \hat{R} &= HQH^T + R. \end{aligned} \tag{26}$$

It should be emphasized that this scheme minimizes variance conditioned on the previous location of the particle. There are importance sampling schemes which minimize the weight variance over the entire sample: see, for example, van Leeuwen[27]. These can also be employed in the HHPF.

2. *Stochastic optimal control approach*

The purpose of the present section is to show that control methods can be used as a basic and flexible tool for the construction of the proposal density inherent in particle filtering.

We consider the same discrete-time nonlinear system with linear observation (22), (23). The particle method presented in this section consists of control terms in the "prognostic" equations, that nudge the particles toward the observations. We nudge the particles by applying a control $u_k(y, x)$ according to

$$X_{k+1} = F(X_k) + u_k(y_{k+1}, X_k) + \sigma_X W_{k+1}. \tag{27}$$

The technique for determining the nudging term $u_k(y_{k+1}, x)$ is by method of stochastic optimal control by minimizing a quadratic cost:

$$J := \mathbb{E}_{k,x} \frac{1}{2} \left[u_k^T(y,x) Q^{-1} u_k(y,x) + (y - HX^{(k,x)}_{k+1})^T R^{-1} (y - HX^{(k,x)}_{k+1}) \right], \tag{28}$$

where Q, R are the signal and observation noise covariance matrices and by $X^{(k,x)}$ we mean that the process X started at time k at the value x.

The first term in (28) represents the control energy and if we allow u to become too big, then heuristically all the particles will coincide with the observation. Then the particles will be a sample from a Dirac distribution, whereas the conditional distribution that we try to simulate is absolutely continuous. The second term in (28) represents the distance between HX_{k+1} and the observation that we want minimized. Covariance matrices Q and R in the quadratic terms indicate that dimensions of the signal and observation that have larger noise variance are penalized less by the control. This means that in directions where noise amplitude is large, we allow for more correction by taking Q^{-1}, which puts less penalty on the size of the control. Similarly, the cost given by the second term in (28) incurs a penalty for being far away from the actual signal based on observation, but in directions where the quality of the observation is not very good, we allow our particle to be further away from the observation, hence R^{-1}.

The solution to this linear-quadratic optimal control problem is well known (see, for example, Chap. 12 in Bryson and Ho[28]), and it is given by

$$u_k^{opt} = (Q^{-1} + H^T R^{-1} H)^{-1} H^T R^{-1} (y - HF(x)),$$

which is similar to the solution from the previous approach, with the difference being the noise term, $\hat{\sigma}_X$ in (25) and σ_X in (27).

This section provided the results related to the control design of the particles (prior to updating the weights) that is needed to nudge the particle solutions toward the observations. This procedure consists of adding forcing terms to the "prognostic" equations for the construction of the proposal density inherent in particle filtering.

The stochastic optimal control approach presented in this section is similar to the derivation of the 4D-VAR method that is used in geophysical data assimilation (see, for example, Kalnay[29]). The 4D-VAR method considers the problem of determining the best initial condition at time t_0 for the forward integration of the model PDEs based on discrete observations collected, up to a finite time t_K, in the future of t_0. In the 4D-VAR method, the cost function to be minimized with respect to the initial condition $x(t_0)$ is

$$\begin{aligned} J(x(t_0)) = {} & \frac{1}{2}[x(t_0) - x^b(t_0)]^T B_0^{-1} [x(t_0) - x^b(t_0)] \\ & + \frac{1}{2}[H(x(t_K)) - y(t_K)]^T R^{-1} [H(x(t_K)) - y(t_K)], \end{aligned}$$

where $x^b(t_0)$ was predicted using the model equations from time before t_0 and $x(t_K)$ is obtained by integration of the model PDEs using $x(t_0)$ as initial condition. From this point of view, the stochastic optimal control approach presented here can be viewed as determining the optimal initial condition at every discrete time t_k using the next available observation at t_{k+1}. The optimal control u_k^{opt} is the correction made to the state x_k predicted from t_{k-1}.

Chorin *et al.*[30] use a different approach which gives the same results for this case. Let $e^{-G(x_k^i)} = p(x_k^i|x_{k-1}^i)p(y_k|x_k^i)$. They try to find a map $\xi \to x$, such that $G(x) - \min G(x) = \frac{1}{2}\xi^T\xi$. Value of ξ is chosen according to a standard Gaussian distribution, and the corresponding x from the map is chosen as the new location of the particle. As ξ is a Gaussian, the highly likely values of ξ are in the neighborhood of 0, and these ξ produce x_k^i near the minimum of G, hence a high probability position for the particle. It is shown in Chorin *et al.*[30] that this leads to the optimal importance sampling density of the previous section.

The SIS presented in Section IV B 1 allows the modification of the drift terms as well as the stochastic coefficients in the "prognostic" equations and SIS is an easy approach to implement numerically, therefore we will use SIS in conjunction with HHPF throughout this paper. However, when dealing with sparse data, a proposal density based on stochastic control theory presented in this section is essential, as can be shown in Lingala *et al.*[35].

The HHPF described next is applicable even for nonlinear sensor functions, for a chosen proposal density. Note that in deriving the optimal proposal, we assumed linearity on the sensor function. Linearity is assumed only because it is difficult to obtain optimal proposal density explicitly when the sensor function is nonlinear.

3. HHPF

In the following we describe how the HHPF uses the SIS algorithm. For now, we would choose the proposal density $q\left(\cdot|\, x_{k-1}^i, y_k\right) = p(\cdot|x_{k-1}^i)$ i.e. the conditional prior which can be obtained from the signal dynamics. A particle can be sampled from this q by propagating the location of the particle at $k-1$ using the signal dynamics. We note that by choosing such a q we are being blind to the observation y_k. We address a better choice in Section IV B 4.

The HHPF is developed based on the results presented in Section II. It incorporates the HMM described in Section IV A in the particle evolution step of a particle filtering algorithm.

Recall that we are working with the discretized version of the homogenized equation (11). Recall the HMM macro-timestep $\Delta t = t_{k+1} - t_k$, $k = 0, 1, \ldots$, and the micro-timestep

$\delta t << \Delta t$, where δt is chosen small enough compared to ε for numerical stability. The evolution of the particle system via the HHPF is given by the following steps with a graphical illustration of each step shown in Figure 1. For notational consistency, we will denote the particles representing the averaged $\bar{X}_k$ by $\bar{x}_k^i$.

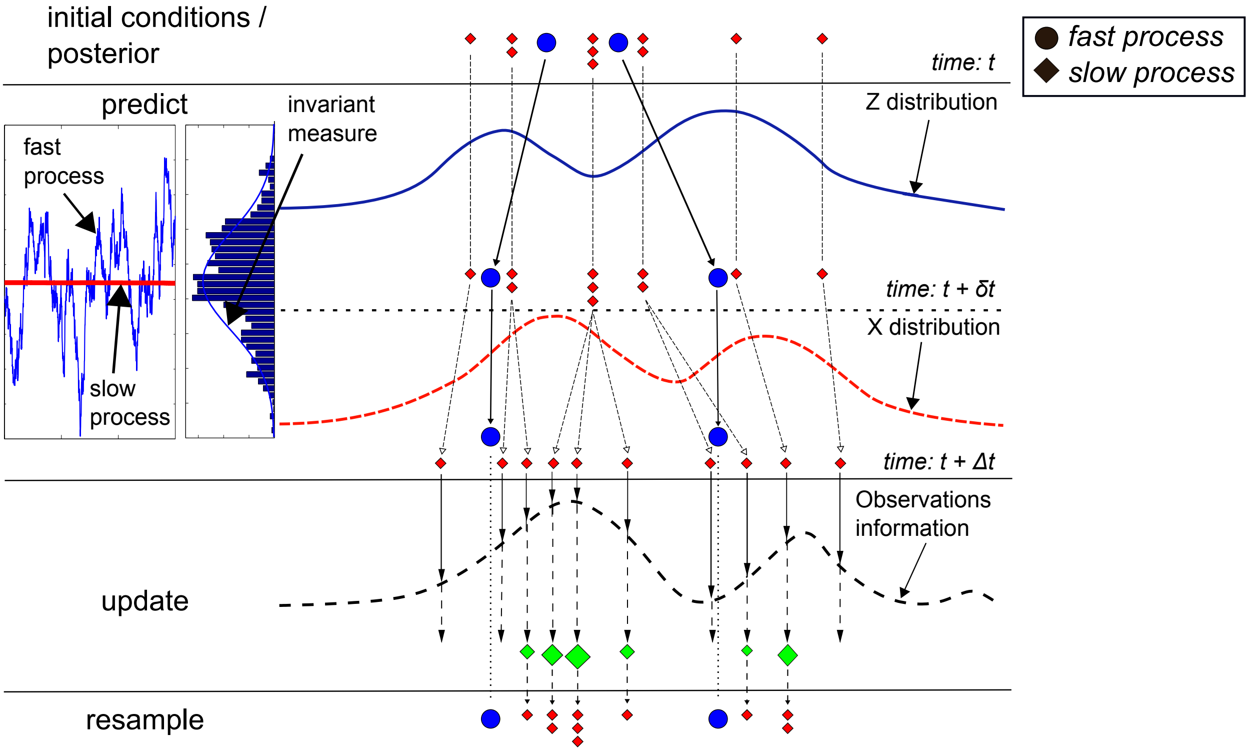

FIG. 1. HHPF illustrative scheme

(a) **initial condition**: The initial distribution of the slow process, π_0^0, is approximated by a sample of N_s particles $\{\bar{x}_0^i\}_{i=1}^{N_s}$ drawn from π_0^0, each of mass $\frac{1}{N_s}$, i.e.

$$\pi_0^0 \approx \frac{1}{N_s} \sum_{i=1}^{N_s} \delta(x - \bar{x}_0^i).$$

For each slow particle $\bar{x}_0^i$, we assign a collection of M fast particles $\left\{z_0^{\varepsilon,i,r}\right\}_{r=1}^{M}$ where $z_0^{\varepsilon,i,r} \in \mathbb{R}^n$ is the position of the r^{th} particle. Note that we are only interested in the coarse-grained dynamics $\dot{\bar{X}}_t$ of the system. The fast process, assumed to be exponentially mixing, is spatially and temporally averaged through the implementation of the HMM. Thus, under the assumption of ergodicity, we can set $M << N_s$ or even $M = 1$ without loss of accuracy in approximating the distribution of $\bar{X}_k$.

(b) **prediction and update**: The prediction and update step is where HHPF differs from regular particle filters through the incorporation of dimensional reduction and homogenization techniques.

In a regular particle filter, particles evolve independently according to the discretized signal equations which govern the behaviour over a micro-step δt. Recall that δt is small compared to Δt. So, the regular filters simulate the slow process every δt as well, even though it does not change much in such a small time, thus wasting computation resources.

The HHPF utilizes the multiscale scheme of the HMM to propagate the sample forward in time. Particles $\{\bar{x}_k^i\}_{i=1}^{N_s}$ are propagated independently over macro-timesteps Δt using the macro-solver (11), where the averaged drift is approximated using (16), by employing the fast particles $\left\{z_0^{\varepsilon,i,r}\right\}_{r=1}^{M}$ (see Section IV A).

Particle propagations in a macro-timestep interval are illustrated in the "predict" segment of Figure 1.

The particles are updated at time-steps when observations are available, as illustrated in the update segment of Figure 1. Only the sample of the slow process $\{\bar{x}_k^i\}_{i=1}^{N_s}$ is updated since we are interested only in the slow process. The SIS algorithm is used for updating the sample. Propagating the particles $\{\bar{x}_k^i\}$ according to the homogenized signal dynamics over Δt gives us new particle locations at $\{\bar{x}_{k+1}^i\}$. Each new location $\bar{x}_{k+1}^i$ can be thought of as a sample drawn from the proposal density $q(\cdot|x_k^i)$, which is same as the prior $p(\cdot|x_k^i)$ of the homogenized dynamics. Particle weights are calculated sequentially according to (19) and because of our assumption of the proposal, we have

$$w_k^i \propto p\left(y_k^\varepsilon \middle| \bar{x}_k^i\right) w_{k-1}^i.$$

Consider the observation equation (13). Because of the ergodic nature of Z^ε, the integral term $\int_{t_{k-1}}^{t_k} h(X_s^\varepsilon = x, Z_s^\varepsilon)ds$ converges to

$$\Delta t \int_{\mathbb{R}^n} h(x,z)p_\infty(x,dz) =: \bar{h}(x), \tag{29}$$

and this $\bar{h}(x)$ is approximated using HMM as in (17). Hence

$$p\left(y_k^\varepsilon \middle| \bar{x}_k\right) \propto \exp\left\{-\frac{1}{2}\left(y_k^\varepsilon - \bar{h}\left(\bar{x}_k\right)\right)^T \left(\sigma_y\sigma_y^T\right)^{-1}\left(y_k^\varepsilon - \bar{h}\left(\bar{x}_k\right)\right)\right\}. \tag{30}$$

It is worthwhile to note that the actual available observation Y_k^ε is used instead of a fictitious averaged $\bar{Y}_k$ in calculating the weights. Thus the homogenized system (7) is combined with the actual observation Y^ε, from which the name "homogenized hybrid" was derived.

(c) **resampling:** The nature of the sequential importance sampling algorithm is such that the variance of the unnormalized weights increases with each iteration (see, for example, Prop. 3, p. 7 in Doucet[31], Theorem, p. 285 in Kong, Liu, and Wong[32]). Due to the gradual increase in weights variance over time, weights will tend to concentrate on a few particles, causing the issue of sample degeneracy, which decreases the ability of the weighted sample to properly represent the target posterior distribution. Additionally, it incurs the cost of propagating and storing particles with insignificant weights, which effectively do not contribute to representing the target distribution. One method of addressing this issue is to perform occasional resampling when the sample degeneracy level reaches a certain threshold. Resampling does not overcome the issue of weights degeneracy; it only serves to rejuvenate the sample by eliminating particles with insignificant weights and multiplying those with weights that significantly contribute to approximating the posterior.

The measure of sample degeneracy can be determined by the effective sample size N^{eff} (see, for example, Arulampalam *et al.*[25]):

$$N_k^{\text{eff}} \overset{\text{def}}{=} \frac{N_s}{1+\operatorname{Var}(\bar{w^*}_k)}, \qquad \bar{w^*}_k^i \overset{\text{def}}{=} \frac{p\left(\bar{x}_k^i \middle| y_{0:k}^\varepsilon\right)}{q\left(\bar{x}_k^i \middle| \bar{x}_{k-1}^i, y_k^\varepsilon\right)},$$

where $\bar{w^*}^i$ is the "true" weight. The exact values of $\bar{w^*}^i$ cannot be evaluated since $p(\bar{x}_k^i|y_{0:k}^\varepsilon)$, the actual posterior, is not known, so the effective sample size is approximated numerically using the normalized weights by Arulampalam *et al.*[25]

$$\tilde{N}_k^{\text{eff}} = \frac{1}{\sum_{i=1}^M (\bar{w}_k^i)^2}.$$

$\tilde{N}_k^{\text{eff}} \le N_s$ represents the effective sample size at timestep k, and a resampling procedure is carried out whenever $\tilde{N}_k^{\text{eff}}$ falls below a set threshold.

One resampling procedure is the systematic resampling procedure, as described in Arulampalam *et al.*[25], Doucet[31]. The resampling procedure involves sampling with replacement on the current sample, multiplying particles with significant weights and discarding those with insignificant weights. The new sample is reinitialized with uniform weights $\frac{1}{N_s}$ for each particle.

In addition to resampling, the issue of sample degeneracy can also be addressed by choosing a good importance sampling density q. As described so far, the importance density $q\left(x_k \middle| x_{k-1}\right)$ is based on the propagation of the sample from the previous estimation step. i.e. the observation Y_k was not involved in the importance density at time k. In Section IV B 4 below, we propose a method of specifying an importance density $q\left(x_k \middle| x_{k-1}, y_k\right)$ at time k that makes use of the observation at time k .

Algorithm 1 shows a pseudo code for the HHPF. Overall, the advantages of Algorithm 1 are:

(a) The number of fast samples evaluations is greatly reduced (if the number of fast sample replicas is set as $M = 1$ by choosing n_T, N_m appropriately), since $n_T + N_m < \lfloor \Delta t/\delta t \rfloor$.

(b) The total number of timesteps is decreased due to the relatively large macro-timesteps.

(c) The number of function evaluations are also decreased accordingly.

Even though there are additional function evaluations (as in equation (16)) in incorporating the HMM, these are negligible compared to the simulation and weight calculations of the regular particle method.

Although the HHPF has been adapted for multiscale computations, another issue arises in its application on complex problems with inherent chaotic nature such as that described in Section III. As shown in Section V, we are unable to perform state estimation for a chaotic system using the HHPF as described so far; the procedure can be made more accurate via a good choice of the sampling density q. In the following section, we present a method of constructing a good sampling density by introducing a forcing term in the particle evolution of (11).

```
Draw samples from initial distribution: $\{\bar{x}^i_{k=1}\}^{N_s}_{i=1}$,$\{z^{\varepsilon,i,r}_{k=1,j=1}\}^{N_s,M}_{i,r=1}$
for k = 1:number of macro-timesteps ($K$) do
    for r = 1:number of replicas ($M$) do
        for j = 1:number of micro-timesteps ($N_m$) do
            Solve micro-solver (15): $z^{\varepsilon,i,r}_{k,j+1}$
        end for
    end for
    Perform averaging (16) and (17): $\{\tilde{f}(\bar{x}^i_k), \tilde{h}(\bar{x}^i_k)\}^{N_s}_{i=1}$
    Solve macro-solver (11): $\{\bar{x}^i_{k+1}\}^{N_s}_{i=1}$
    Compute weights: $\{\bar{w}^i_{k+1}\}^{N_s}_{i=1}$
    Compute effective sample size: $\tilde{N}^{k+1}_{\text{eff}}$
    Resample using choice of resampling algorithm if $\tilde{N}^{k+1}_{\text{eff}} < N_{\text{thres}}$
    Reinitialize: $z^{\varepsilon,i,r}_{k+1,j=1} = z^{\varepsilon,i,r}_{k,N_m}$
end for
```

Algorithm 1: HHPF (Park, Namachchivaya, and Yeong[8])

4. HHPF using the optimal proposal

In the event that the averaged sensor function (29) is expressible as $\bar{h}_k(x) = H_k x + C_k$, where C_k is a vector which can change with time k, then one can use the optimal proposal, as explained in IV B 1. Note that $\bar{h}$ is of this form if the observation function h is linear in the slow components. Only the following changes need to be made to the algorithm:

- Instead of propagating the particles $\bar{x}^i_{k+1}$ by using (11), we propagate according to

$$X_k = X_{k-1} + \tilde{b}\,(X_{k-1})\,\Delta t + \alpha(X_k, y_k - C_k) + \hat{\sigma}_x \Delta W_k, \tag{31}$$

  where $\hat{\sigma}_x$ is such that $\hat{Q} \overset{\text{def}}{=} (\hat{\sigma}_x \hat{\sigma}_x^T) = ((\sigma_x \sigma_x^T)^{-1} + H^T(\sigma_y \sigma_y^T)^{-1} H)^{-1}$, and $\alpha(X_k, y_k) = \hat{Q} H^T R^{-1}(y_k - Hf(X_{k-1}))$, where $f(x) = (x + \tilde{b}\,(x)\,\Delta t)$.

- The weights are updated according to

$$\begin{aligned} w^i_k &\propto w^i_{k-1} \exp\left\{-\frac{1}{2}(y_k - C_k - Hf(x^i_{k-1}))^T \hat{R}^{-1}(y_k - C_k - Hf(x^i_{k-1}))\right\}, \\ \hat{R} &= R + H(\sigma_x \sigma_x^T) H^T. \end{aligned} \tag{32}$$

V. APPLICATION

Based on the results of homogenization and optimal importance sampling, we have developed a new lower-dimensional particle filter, the HHPF, for state estimation in nonlinear multiscale systems. In this section, we illustrate the HHPF's potential for state estimation in a high-dimensional complex problem by applying the HHPF algorithm to the Lorenz '96 (Lorenz[2]) atmospheric model (see Section III) to estimate the *slow* variables. The HHPF algorithm is implemented using SIS, as presented in Section IV B. The model parameters for application of the HHPF are as follows: $K = 36$ $J = 10$, $F_x = 10$, $h_x = -0.8$, $h_z = 1$, $\varepsilon = 1/128$.

A realization of the slow-fast system is obtained by simulating the equations (9), for 40960 timesteps of size 2^{-11} starting from a random initial condition. The slow process of this realization is fixed as "true signal" that is to be estimated. The timestep value of 2^{-11} was picked such that it is small enough compared to the separation parameter ε to ensure numerical stability. The Lorenz '96 model given by (3), (5) was integrated using a split timestepping scheme: fourth order Runge-Kutta scheme for the deterministic drift and the Euler-Maruyama scheme for the stochastic parts. These schemes were selected for simplicity of implementation in the numerical experiments, but of course, coarser timesteps may be used by implementing higher order integration schemes. Observations generated from the true states were of the form assumed in Section IV B 4: $Y^{\varepsilon}_t = HX^{\varepsilon}_t + B_t$, i.e. Y^{ε} depends linearly on X^{ε}, perturbed by a standard Gaussian noise. Observations were generated every 128 timesteps, i.e. at every timestep of size 2^{-4}. This is the size of the macrotimestep Δt that we select for the HMM integration scheme. This means that we are considering the case where observations are available sequentially (at every timestep) at the timescale that we had chosen for the numerical integration of the slow process. In Lorenz[2], the timestep chosen for numerical integration of the multiscale system with $\varepsilon = 0.01$ was 0.05, which corresponds to 36 minutes in real time. We follow Fatkullin and Vanden-Eijnden[24] in selecting $\Delta t = 2^{-4}$ and $\delta t = 2^{-11}$ for the HMM scheme for $\varepsilon = 2^{-7}$. Based on Lorenz[2]'s scale, these timesteps approximately correspond to 45 minutes and 35 seconds, respectively, in real time and the duration of the model simulation and data assimilation experiments correspond to 10 days (14400 minutes). On the real time scale, the assumption of sequentially available observation data corresponds to observation data collection every 45 minutes, which is not an unrealistic assumption.

We consider two cases for the linear observation matrix H. The first is $H = I_{K\times K}$, where I is the identity matrix, i.e. all slow states are observed (non-sparse observations). The second is $H_{ij} = 1$ if $i = j$ and i odd, and otherwise $H_{ij} = 0$, i.e. only the odd-indexed slow dimensions are observed (sparse observations). Observation noise covariance matrix is assumed to be the identity matrix, i.e. observation noise in each dimension is independent of the rest. We chose the slow and fast signal noise covariance matrices to be 1 on the diagonals and 0.5 on the sub- and super-diagonals as in van Leeuwen[27], with the fast noise scaled by $\varepsilon^{-1/2}$. The filtering objective is to estimate the *slow* states using sequentially available observations. We discuss the numerical experiments in the following. The numerical experiments were performed using MATLAB v. 7.11.0.584 (R2010b), without explicitly employing parallel processing capabilities, on an Intel Xeon DP Hexacore X5675 3.07 GHz processor with 12×4 Gb RAM.

5. *Optimized HHPF with linear observations*

In the first set of numerical experiments, we consider the case of non-sparse observations and consider two variants of the HHPF to study the effects of the optimal particle propagation for linear observation functions discussed in Section IV B 4. The SIS algorithm is used in both variants of the HHPF. In one variant, the proposal density used for the importance sampling procedure is generated using particles propagated using the optimal drift and diffusion as given in (24), with weights updated accordingly. In the other variant, proposal density is the prior density generated by propagating particles directly according to (11). We call the first one the *optimized HHPF* and the second one the *direct HHPF*. For the HMM scheme, the HMM window was set at $N_m = 64$ microtimesteps, and number of microtimesteps skipped to ignore transient effects is $n_T = 32$. The number of replicas is set at $M = 1$.

Numerical experiments were performed using both HHPFs with varying sample sizes starting from $N_s = 2$ to $N_s = 400$. Both filters achieved better accuracy with increasing N_s, as they should, and Figure 2 shows the comparison of one dimension of the true state with its estimates using the HHPFs and the corresponding estimation errors. In the upper figure, blue curve represents the true state, broken red curve represents the estimate using the optimized HHPF, and green curve represents that of the direct HHPF. In the lower figure, blue curve represents the observation error, broken red curve and blue curves represent the estimation error of the optimized and direct HHPFs, respectively. The error shown is the absolute error over all 36 slow dimensions, i.e. $\text{error}_t = \sqrt{\sum_{k=1}^{36}(\hat{X}_t^k - X_t^{k,\varepsilon})^2}$.

We see in the error plot of Figure 2 that the optimized HHPF was able to provide a better estimate of the truth than what was known directly from the observations. The run time for the experiment presented in the figure was 135 seconds, which is the typical time recorded for the optimized HHPF with $N_s = 100$. The estimate from the direct HHPF improved but was at best as good as the observation with $N_s = 100$. Increasing N_s up to 400 improved both filters' estimates but not significantly for the optimized HHPF. The direct HHPF was seen to be able to match the optimized HHPF with $N_s = 100$ by using 600 particles but the run time required was 639 seconds. The second and third columns of Table I show the run times for different sample sizes for a typical experiment using the HHPFs. Occasional drastic variation in run times between the optimized and direct HHPFs, for example for $N_s = 100$ and $N_s = 400$, can be explained based on the number of times the resampling procedure were required to be performed in each filter. Resampling, however, will not drastically increase for the optimized HHPF because particles are simultanesously driven towards the truth based on the observations through the optimal propagation procedure.

The same numerical experiments were also performed for the case of sparse observations (i.e. observing only the dimensions with odd index), to study the performance of the optimized HHPF in estimating hidden states. The same trend as for the case of non-sparse observations was observed in comparing the estimates using the optimized and direct HHPFs. For fixed N_s, the optimized HHPF achieved better accuracy than the direct version, although both estimates displayed higher estimation errors compared to the non-sparse observation case at the same N_s due to the presence of hidden states. Figure 3 shows the comparisons of the estimates from the HHPFs, using 100 particles for the optimized and 400 particles for the direct, with the truth. The upper plot shows the estimates of an observed state and the lower shows those of an unobserved state. The optimized HHPF (broken red curve) provided good estimates of all the states, including the unobserved ones and the corresponding error plot in Figure 4 shows that the estimation was as good as the observation (of course, observations only observed half the states and the error shown only compares error in the observed dimensions; but it is important to note that the HHPF estimates the

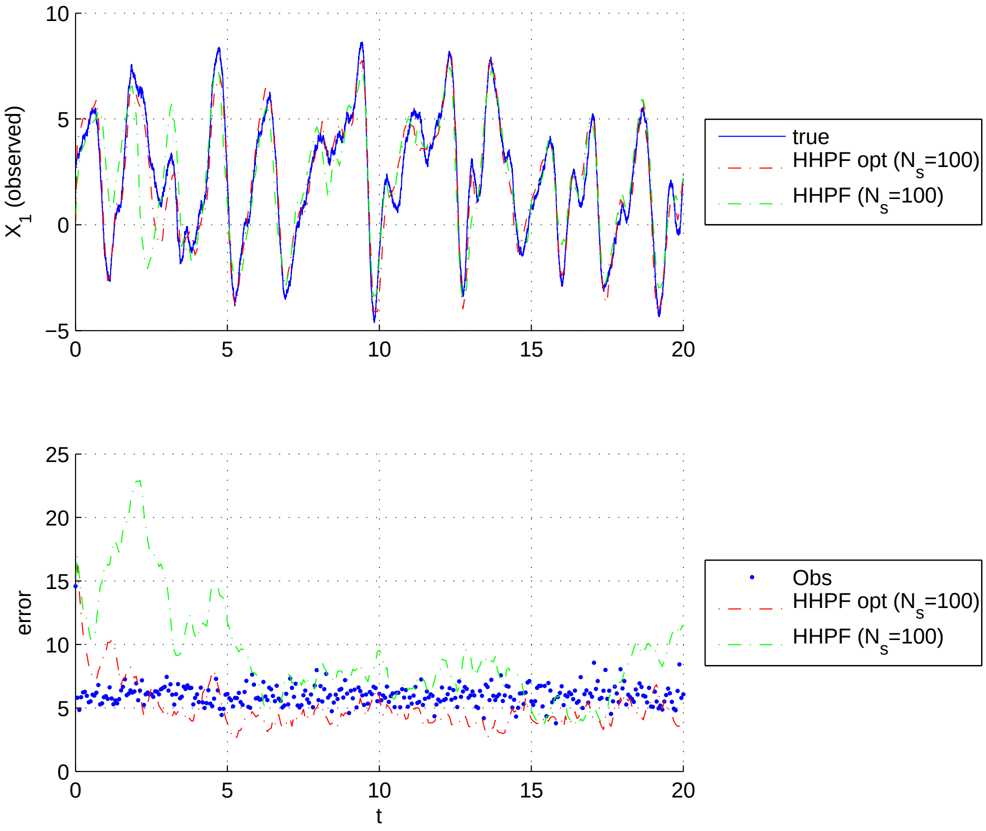

FIG. 2. Non-sparse observations, optimized and direct HHPFs, $N_s = 100$. The optimized HHPF estimate was better than observation data but the direct HHPF was still not as good. The direct HHPF was able to match the optmized one with $N_s = 600$ at the cost of longer computation time.

unobserved dimensions also). The direct HHPF, was also able to capture the shape of the truth in the observed and unobserved dimensions, but in the observed dimensions, the estimates were poorer than the observations, even with 400 particles. Increasing sample size further up to $N_s = 800$ did not lead to significant improvement. The run times and the trend of increase in run time with sample size for the sparse observations experiments were the same as that of the non-sparse observations experiment shown in Table I.

The homogenized filters comparison experiments showed that the optimized HHPF displayed significant estimation performance over the direct version for a fixed sample size. This indirectly led to improvement in computation time in comparison with the direct HHPF, in the sense that for the same or even better level of estimation error, the optimized HHPF could be implemented using smaller sample size than the direct version, hense reducing computational costs. In the discussion here and in Table I, we have only considered the comparisons of the HHPFs with N_s up to 400, beyond which the run time advantage of the HHPF over other unhomogenized nonlinear filters, for the same level of accuracy, is lost. In the next set of experiments, we compare the optimized HHPF with two other nonlinear filters.

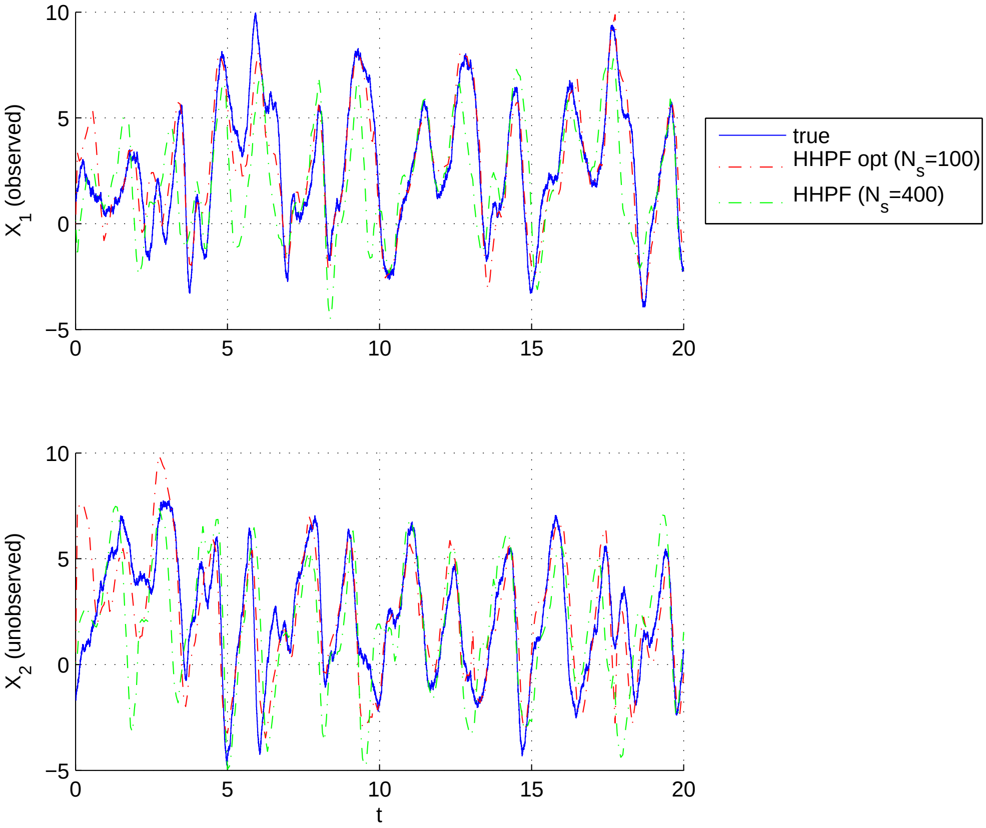

FIG. 3. Sparse observations, optimized ($N_s = 100$) and direct ($N_s = 400$) HHPFs. The upper plot shows the estimates of an observed state, the lower plot shows that of an unobserved state. The unobserved state was estimated well by the optimized HHPF with $N_s = 100$. Even with $N_s = 400$, The direct HHPF captures the fluctuations in the truth but did not follow the trajectory well.

6. Comparison of the optimized HHPF with other nonlinear filtering schemes

In the second set of numerical experiments, we compared the optimized HHPF with two other nonlinear filtering schemes: the ensemble Kalman filter (enKF) and a particle filter without homogenization. The aim was to compare the run time reduction in the HHPF,

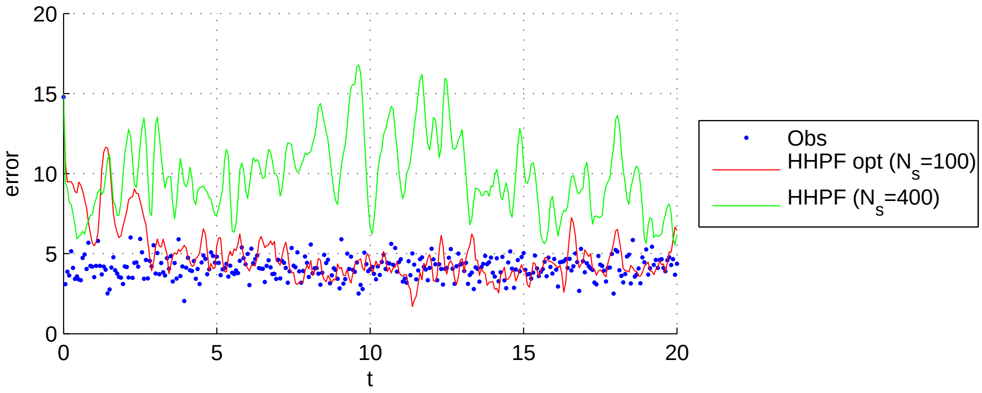

FIG. 4. Sparse observations, estimation error of the observed states for the optimized ($N_s = 100$) and direct ($N_s = 400$) HHPFs compared with observation error. The optimized HHPF estimates were as good as observations (but the optimized HHPF also provided estimates of the unobserved states so more information is gained by using the filter instead of just observation).

due to the implementation of a homogenization scheme, relative to unhomogenized nonlinear filters and assess the trade-off in estimation accuracy due to homogenization. We do not expect the optimized HHPF to outperform an unhomogenized filter in terms of estimation accuracy, but the optimized HHPF should possess comparable estimation capabilities to unhomogenized filters, with the advantage of the HHPF being that it requires shorter run times.

Since observations were generated at macrotimesteps of size 2^{-4}, we have observations being sparse in time as well for the unhomogenized filters, which use the same timestep size of 2^{-11} as the model simulation. The enKF Evensen and van Leeuwen[33] was implemented directly without modifications, by propagating the ensemble forward in time according to the model dynamics and performing information updates at timesteps when observations are available. For the particle filter, we implement a modified SIS particle filter algorithm developed by van Leeuwen[27] that is designed to accommodate for observations that are sparse in time. This particle filter is similar to the optimized HHPF presented here, in the sense that it uses the presently available observation to construct a better proposal density at a present time by driving particles in between observation timesteps using a time exponential function that is proportional to the model noise covariance and the distance of the intermediate particle locations from the observed state. We have used the same filtering parameters as van Leeuwen[27]. For details and better insight to this particle filter, see van

Leeuwen[27], and from here on, we will denote this particle filter as just PF, but it is implied that it is the particle filter adapted for sparse-in-time observations.

Similar numerical experiments as for the comparison of the optimized and direct HHPFs were performed for the enKF and the PF and the estimation results were compared with those of the HHPFs. We will first discuss the case of non-sparse (spatially) observations, i.e. the case $H = I_{K\times K}$.

1. The PF was able to provide good estimates of the truth with a sample size of just 2. Estimation error decreased as N_s was increased to 20 and was not seen to further decrease significantly as N_s was increased from 20. However, the EnKF provides better accuracy as particles are increased. Figure 5 shows the estimates of one dimension of the true state using the EnKF, the optimized HHPF, and the PF with $N_s = 20$, and their corresponding estimation errors. The blue curve is for the truth, the black for the EnKF, red for the optimized HHPF, and green for the PF.

2. With 20 particles, the optimized HHPF displayed the highest estimation error, its estimate being as good as the observation. The EnKF and PF estimates are better than the observation, with estimation errors of equal magnitude. However, when considering the run times, the optimized HHPF took 30 seconds while the EnKF and PF took 540 and 1757 seconds, respectively. (Even with $N_s = 2$, the PF took 1727 seconds due to the timestep size and the functional evaluations required for particle weight calculations, as well as the resampling procedures).

3. With $N_s = 100$, the optimized HHPF was almost as good as the PF, as shown in Figure 6, with the EnKF being slightly better than both particle filters. The key point is that using 100 particles, the optimized HHPF took 134 seconds. The EnKF and PF with $N_s = 20$ achieved the same accuracy, but they took 540 and 1757 seconds, respectively (the EnKF and PF took 920 and 2169 seconds, respectively, when 100 particles were employed; but this is redundant because they already achieved high accuracy with 20 particles). Considering that the levels of estimation errors are almost equal, the optimized HHPF provided significant advantage in terms of computation time.

4. To achieve better accuracy, we increased the sample size to 400. When increasing the particle number, the optimized HHPF loses its computation time advantage, for

the EnKF could be implemented using $N_s = 50$ or 100 with about the same level of estimation accuracy and computation time.

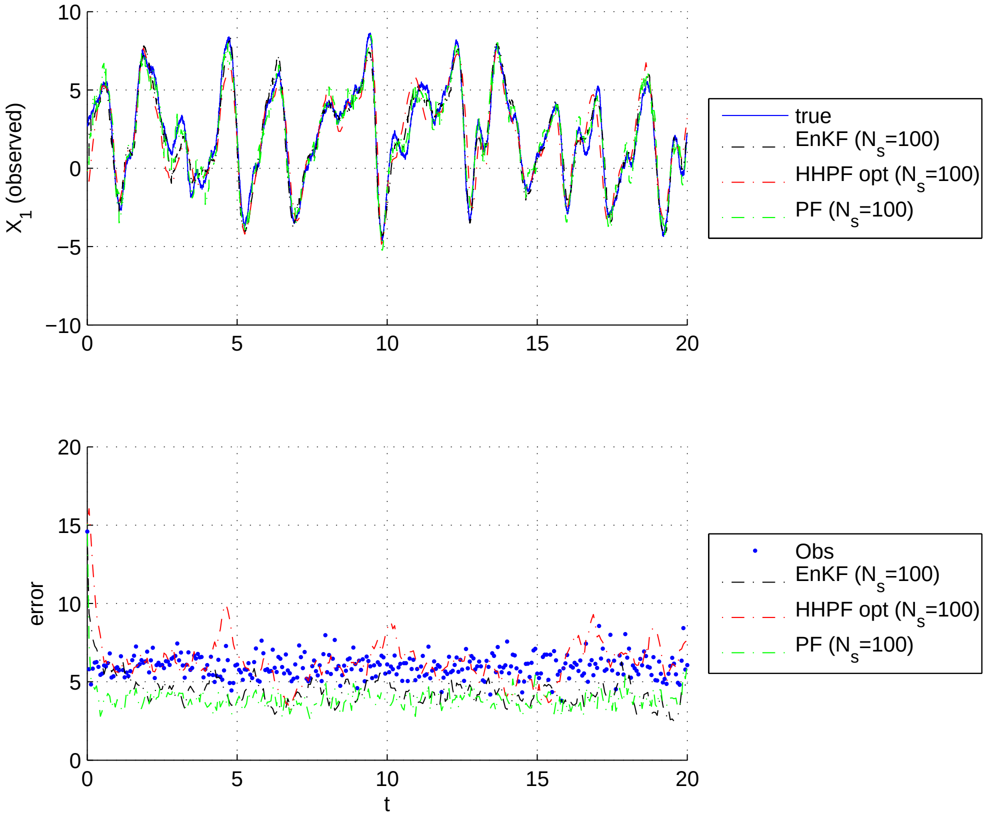

FIG. 5. Non-sparse observations, EnKF, PF and optimized HHPF comparison at fixed $N_s = 20$. The estimate of the optimized HHPF is comparable to those of the EnKF and the PF, but is obtained in a shorter run time.

Similar numerical experiments were again performed for the case of sparse observations (i.e. observing only dimensions with odd index) and Figures 7 and 8 show the comparisons of the estimates of the observed and unobserved states from the filters using 20 and 100 particles, respectively. With 20 particles, the EnKF and PF provided slightly better estimates of the observed states than the observation, but the optimized HHPF performed rather poorly in comparison. However, at $N_s = 100$, the optimized HHPF's performance improved significantly, performing as well as the PF. Figure 9 shows the estimation error over the observed dimensions and, to compare the filter estimate over all slow dimensions, Figure 10 shows the estimation error over all slow dimensions for each filter. The PF estimate converged to the truth faster but the optimized HHPF estimate eventually became

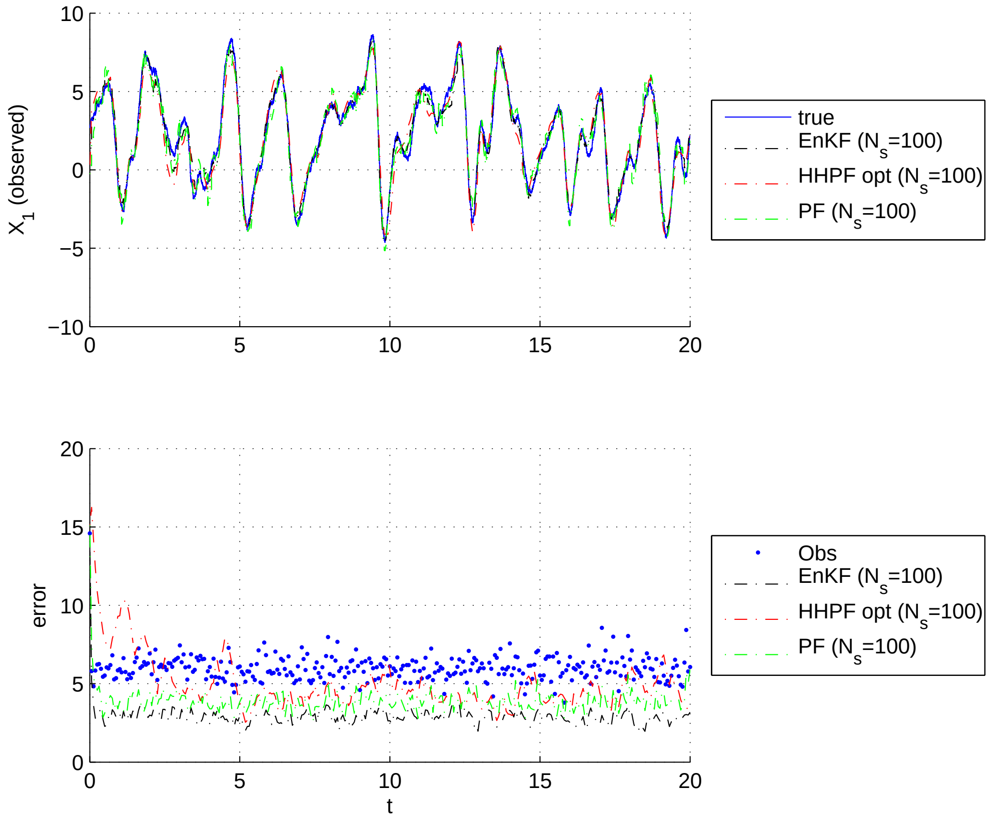

FIG. 6. Non-sparse observations, estimation error comparison the EnKF, PF and optimized HHPF comparison at fixed $N_s = 100$. The optimized HHPF could match the PF, with the EnKF being slightly better, but the optimized HHPF required the least computation time.

as good as the PF's, with the EnKF's being the best. The result 3 of the preceding discussion on non-sparse (spatially) observations is repeated in this case also. So, even in the sparse observations case, the optimized HHPF could be implemented in shorter time with estimation error comparable or equal to the EnKF and the PF.

We do not claim that the HHPF is better than the EnKF or the PF; as shown, the unhomogenized filters provided lower estimation error than the homogenized filter at low fixed N_s. As mentioned in the discussion of the previous set of experiments, the accuracy of the optimized HHPF could be increased by increasing N_s, but beyond $N_s = 400$, it loses computation time advantage over the EnKF. However, in terms of computation time and cost of storage, the optimized HHPF displayed advantage over the unhomogenized filters, with comparable level of accuracy. The EnKF would have been the better choice of filter if the lowest possible estimation error was required over computational time. Additionally, the

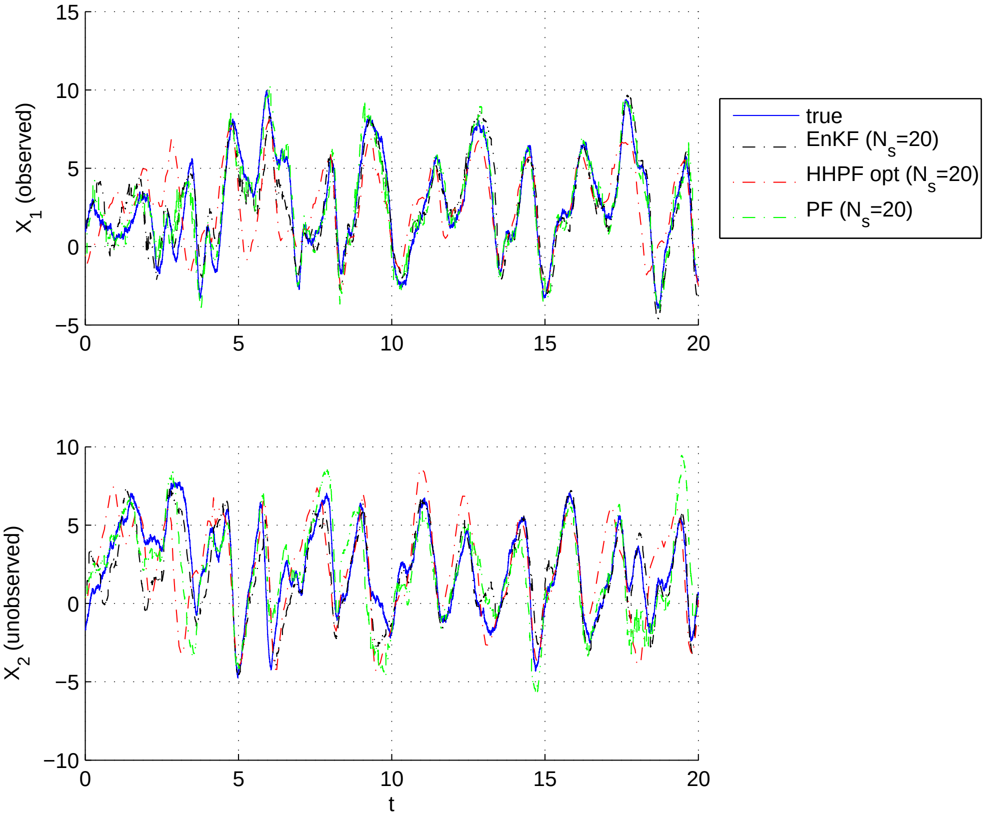

FIG. 7. Sparse observations, EnKF, PF and optimized HHPF comparison at fixed $N_s = 20$. The optimized HHPF did not perform as well as the EnKF or PF.

PF was designed to accommodate temporally sparse observations, as it has been shown to do in van Leeuwen[27] and here. This capability still needs to be incorporated in the HHPF as presented here, because in most real time applications, the time interval for availability of observation data can be greater than the 45 minute-interval assumed here.

In comparing the computational times of the numerical experiments using the EnKF, the PF and the optimized HHPF so far, we have not taken into account the cost of computing the homogenized observation function $\bar{H}$. In addition to H being a constant matrix, we also assumed that the sensor observed only the slow states, hence the observation function was independent of the fast process. So, the homogenized observation function is the same as the unhomogenized one. However, we do not expect the cost of evaluating the homogenized observation function to drastically affect the computational time advantage of the optimized HHPF over other unhomogenized nonlinear filters. In fact observing also the fast variables can even result in a further advantage for the HHPF. In experiments with simple linear

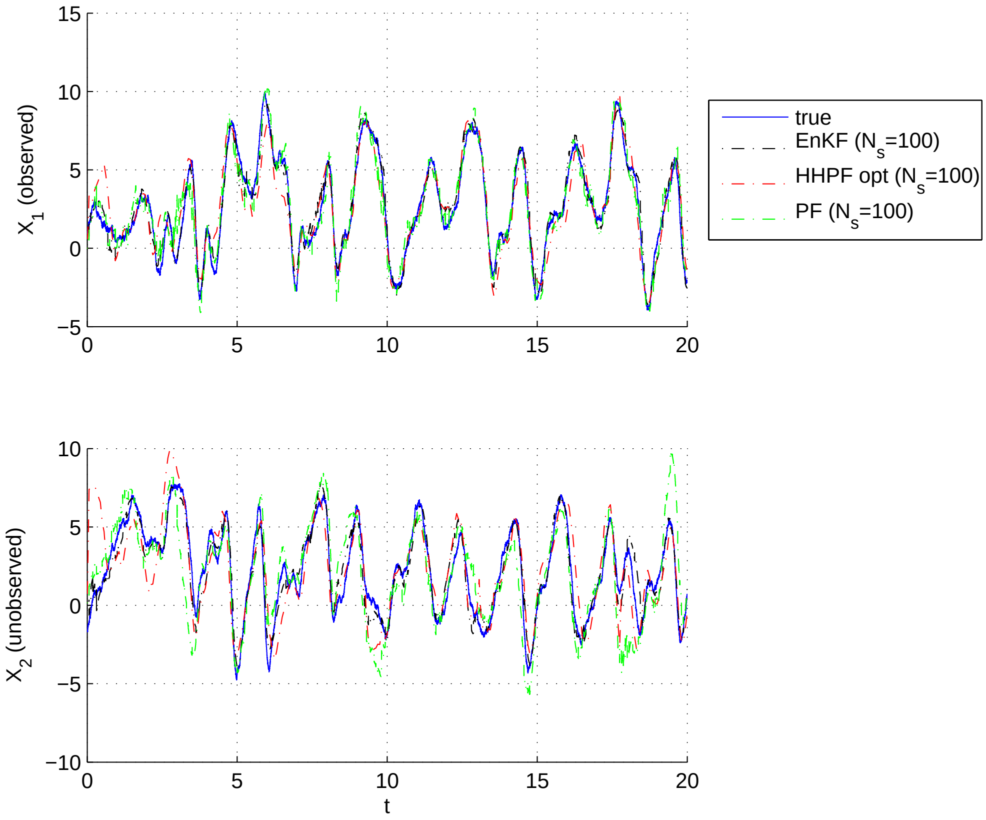

FIG. 8. Sparse observations, EnKF, PF and optimized HHPF comparison at fixed $N_s = 100$. The optimized HHPF performed as well as the PF, but in shorter time than both the EnKF and PF.

models we noticed that if every variable, slow and fast, was observed, then the HHPF was not only faster, but also it gave better estimates than the unhomogenized particle filter. Heuristically this can be explained as follows: For the unhomogenized particle filter, a particle consists of slow and fast dimensions, and its weight is calculated by how well it approximates the signal - for this the value of both the fast and the slow dimensions is taken into account. Therefore a particle which approximates the signal very well in the slow dimensions, but rather poorly in the fast dimensions, will receive a low weight. The HHPF does not suffer from this problem.

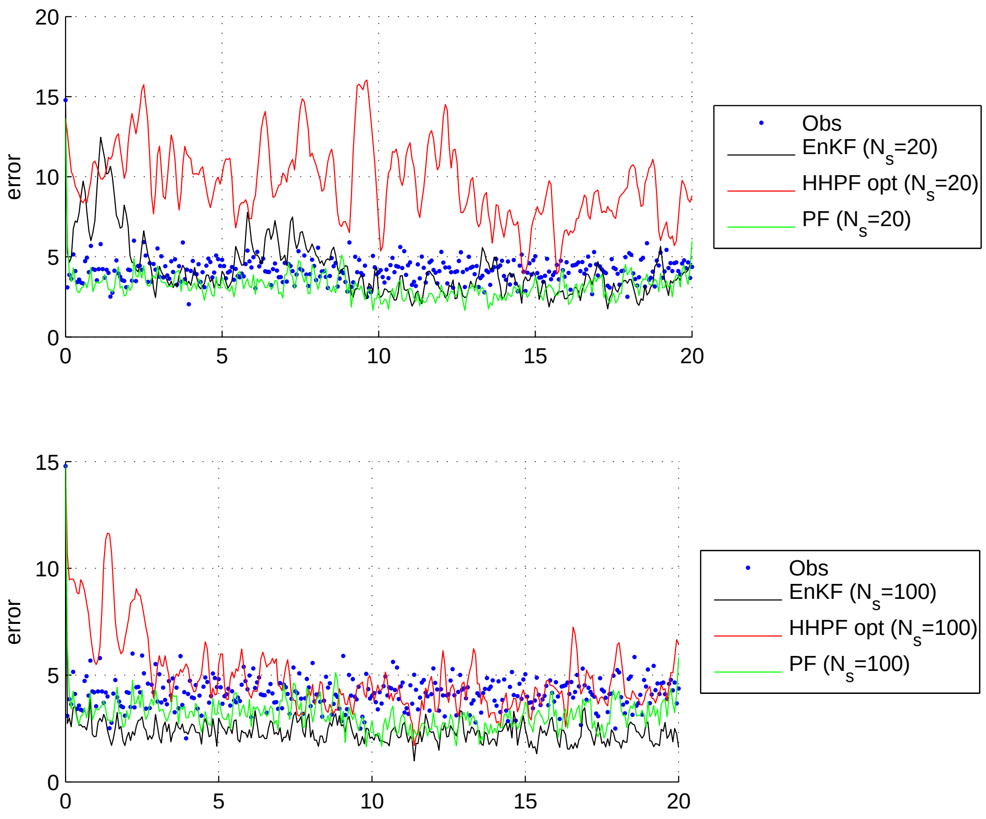

FIG. 9. Sparse observations, comparison of estimation errors of the EnKF, PF and optimized HHPF with observation error.

VI. FUTURE DIRECTIONS

In chaotic systems, such as the one studied in this paper, the transients become irrelevant from the dynamical systems point of view and the motion of the solution settles typically near a subset of the state space, called an attractor. However, in the data assimilation problem that is of interest in this paper, we are interested in the transients and, in particular, in directions that are stretched by the transient dynamics. This sensitivity to initial conditions is characterized by the finite time Lyapunov exponents, which are determined by the behavior of two neighboring orbits or the two point motion of the nonlinear system. We are mostly interested in filtering *deterministic* chaotic systems, and the particle filtering methods developed above will not work without the addition of noise, because knowing the initial conditions X_0, the distribution of X_t is a Dirac measure. Therefore we add Gaussian noise *artificially*. Hence, the finite time Lyapunov exponents may be handy in deciding how we choose the magnitude of the noise.

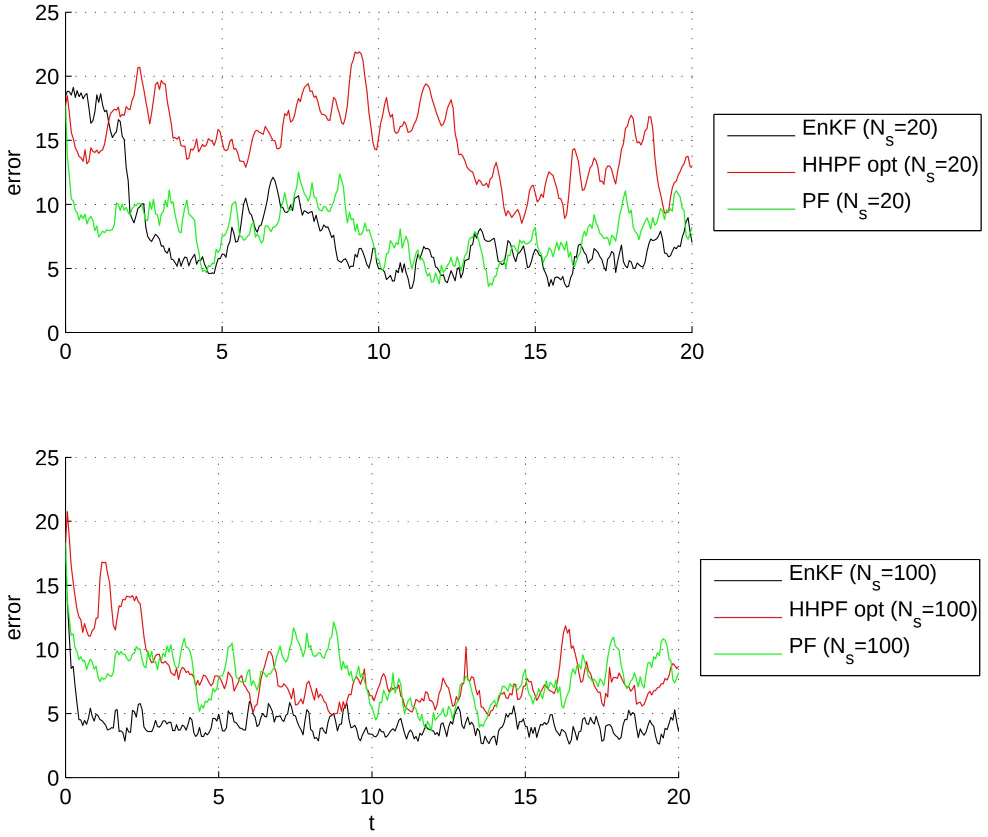

FIG. 10. Sparse observations, comparison of estimation errors of the EnKF, PF and optimized HHPF with observation error. Estimation error of the optimized HHPF and the PF are of the same magnitude at $N_s = 100$. The optimized HHPF still required less computation time than the EnKF and PF even with the EnKF and PF being implemented using minimum N_s possible.

The second question then arises as to the property of the sensor function, in the linear case, the span of the observation matrix. As pointed out by Lorenz and Emanuel[16] and Palmer *et al.*[34], who have coined the word "adaptive" or "targeted" observations, the sensors should be deployed at any given time, if the data that they gather are to be most effective in improving the analysis and forecasts.

The results presented in this paper assume that the observations are available every 45 minutes. The extension of this work that deals with temporal sparsities, performing intermittent in time sparse data assimilation that mimics a global weather model is presented in Lingala *et al.*[35]. The particle method presented in Lingala *et al.*[35], for a lower dimensional chaotic system, consists of control terms in the "prognostic" equations, that nudge the particles toward the observations, with observations recorded every 24 hours, and shows

N_s	Opt. HHPF	Direct HHPF	PF	EnKF
2	5	5	1727	N/A
10	16	15	1716	N/A
20	30	20	1757	540
50	67	46	1936	651
100	134	86	2169	920
200	177	176	3002	1415
400	401	539	3591	3152

TABLE I. Typical computation times (in sec.) for different sample sizes for the different nonlinear filters, in the case of non-sparse observations. We see that for a fixed N_s, the HHPF required less computational time. Circled are the computation times corresponding to the sample sizes that led to the same levels of estimation accuracy for the three different filters compared in Section V 6. For the same level of accuracy, the optimized HHPF required a larger sample size but still performed in less time. Times for the case of sparse observations are of the same magnitudes and display the same trend.

that control methods can be used as a basic and flexible tool for the construction of the proposal density inherent in particle filtering.

ACKNOWLEDGMENTS

Nishanth Lingala, N. Sri Namachchivaya, and Hoong C. Yeong are supported by the National Science Foundation under grant number EFRI 10-24772 and by AFOSR under grant number FA9550-12-1-0390. Nicolas Perkowski is supported by a Ph.D. scholarship of the Berlin Mathematical School. Part of this research was carried out while Nicolas Perkowski was visiting the Department of Aerospace Engineering of University of Illinois at Urbana-Champaign. He is grateful for the hospitality at UIUC. The visit of Nicolas Perkowski was funded by NSF grant number EFRI 10-24772 and by the Berlin Mathematical School. Any opinions, findings, and conclusions or recommendations expressed in this paper are those of the authors and do not necessarily reflect the views of the National Science

Foundation.

Appendix A: An outline of the proof in Imkeller *et al.*[1]

The convergence of the normalized filters, $\pi^{\varepsilon,x} \to \pi^0$, was shown by first obtaining the convergence of the unnormalized filters, $\rho^{\varepsilon,x}$ to ρ^0. For this purpose we worked with the dual processes of $\rho^{\varepsilon,x}$ and ρ^0. Those have the advantage of being function valued rather than measure valued, and therefore they are easier to handle. The dual representations of $\rho^{\varepsilon,x}_T(\varphi)$ and $\rho^0_T(\varphi)$ were introduced as in Pardoux[36]:

$$v_t^{\varepsilon,T,\varphi}(x,z) \stackrel{\text{def}}{=} \mathbb{E}_{\mathbb{P}^\varepsilon_{t,x,z}}[\varphi(X^\varepsilon_T)\left(D^\varepsilon_{t,T}\right)^{-1}|\mathcal{Y}^\varepsilon_{t,T}], \quad \text{and} \quad v_t^{0,T,\varphi}(x) = \mathbb{E}_{\mathbb{P}^0_{t,x}}[\varphi(X^0_T)\left(D^0_{t,T}\right)^{-1}|\mathcal{Y}^\varepsilon_{t,T}].$$

$\mathbb{P}^\varepsilon_{t,x,z}$ and $\mathbb{P}^0_{t,x}$ are the respective measures under which $(X^\varepsilon, Z^\varepsilon)$ and X^0 are governed by the same dynamics as under $\mathbb{P}^\varepsilon$ and P^0, but $(X^\varepsilon, Z^\varepsilon)$ and X^0 stays in (x,z) and x until time t. $D^\varepsilon_{t,T}$ and $D^0_{t,T}$ were defined as the Girsanov transform (2), but with moving limit of integration t,

$$D^\varepsilon_t \stackrel{\text{def}}{=} \left.\frac{d\mathbb{P}^\varepsilon}{d\mathbb{Q}}\right|_{\mathcal{F}_t} = \exp\left(-\int_t^T h(X^\varepsilon_s, Z^\varepsilon_s)^T dB_s - \frac{1}{2}\int_t^T |h(X^\varepsilon_s, Z^\varepsilon_s)|^2 ds\right),$$

$$D^0_{t,T} = \exp\left(-\int_t^T \bar{h}(X^0_r)^T dY^\varepsilon_r + \frac{1}{2}\int_t^T |\bar{h}(X^0_r)|^2 dr\right),$$

and $\mathcal{Y}^\varepsilon_{t,T} = \sigma(Y^\varepsilon_r - Y^\varepsilon_t : t \le r \le T) \vee \mathcal{N}$ is the filtration generated by the observation over $[t,T]$, minus observation history from 0 to t. The Markov property of $(X^\varepsilon, Z^\varepsilon, X^0)$ gives the relations between the unnormalized filters and the respective duals:

$$\rho^{\varepsilon,x}_T(\varphi) = \int v_0^{\varepsilon,T,\varphi}(x,z)\mathbb{Q}_{(X^\varepsilon_0,Z^\varepsilon_0)}(dx,dz) \quad \text{and} \quad \rho^0_T(\varphi) = \int v_0^{0,T,\varphi}(x)\mathbb{Q}_{X^0_0}(dx). \tag{A1}$$

Note that $\mathbb{Q}_{X^0_0} = \mathbb{Q}_{X^\varepsilon_0}$, because the homogenized process has the same starting distribution as the un-homogenized one.

For fixed T and $\varphi \in C^2_b(\mathbb{R}^m, \mathbb{R})$, we will write $v^\varepsilon_t = v_t^{\varepsilon,T,\varphi}$ and $v^0_t = v_t^{0,T,\varphi}$. The dual process v^ε_t essentially represents the conditional expectation $\rho^{\varepsilon,x}_T$ by an alternate conditional expectation that is run backwards in time from T to 0. For fixed starting point (x,z) at $t=0$, this backward in time conditional expectation is constructed using processes $(X^\varepsilon_T, Z^\varepsilon_T)$ that started at (x,z) and ran backwards to $t=0$. By integrating v^ε_0 over $\mathbb{Q}_{(X^\varepsilon_0,Z^\varepsilon_0)}$, we are integrating over all possible starting points (x,z), hence giving ρ^ε_T. This interpretation is similar for v^0 of ρ^0.

From (A1), we have

$$\mathbb{E}[|\rho_T^{\varepsilon;x}(\varphi) - \rho_T^0(\varphi)|^p] \leq \int \mathbb{E}[|v_0^\varepsilon(x,z) - v_0^0(x)|^p]\mathbb{Q}_{(X_0^\varepsilon, Z_0^\varepsilon)}(dx, dz). \tag{A2}$$

So, if $|v_0^\varepsilon(x,z) - v_0^0(x)|$ is small, then $|\rho_T^{\varepsilon;x}(\varphi) - \rho_T^0(\varphi)|$ will also be small as long as $\mathbb{Q}_{(X_0^\varepsilon, Z_0^\varepsilon)}$ is well behaved and (A2) will lead to the normalized filter convergence result. Therefore, the aim is to show that for nice test functions φ, $\mathbb{E}[|v_0^\varepsilon(x,z) - v_0^0(x)|^p]$ is small.

The key point is that v^ε and v^0 solve backward SPDEs. We formally expand v^ε as

$$v_t^\varepsilon(x,z) = \underbrace{u_t^0(x,z)}_{v^0(t,x)} + \underbrace{\varepsilon u^1_{t/\varepsilon}(x,z)}_{\psi(t,x,z)} + \underbrace{\varepsilon^2 u^2_{t/\varepsilon}(x,z)}_{R(t,x,z)}.$$

Then, v^0, ψ and R satisfy linear partial differential equations with appropriate terminal conditions. By existence and uniqueness of the solutions to these *linear* equations, we can apply superposition to obtain that indeed

$$v_t^\varepsilon(x,z) = v_t^0(x) + \psi_t(x,z) + R_t(x,z),$$

where ψ and R are the corrector and remainder terms, respectively. Based on the expansion, the problem of showing L^p-convergence of v^ε to v^0 reduces to showing L^p-convergence of ψ and R to 0. Details are provided in Imkeller *et al.*[1]. The outline of the method of proof is as follows: The backwards SPDEs were converted to their respective probabilistic representations, which are backward doubly stochastic differential equations (BDSDEs). The diffusion operators were replaced by the associated diffusions and explicit estimates for the finite dimensional BDSDEs in terms of the transition density function of the fast process were obtained. Pardoux and Veretennikov[37] proved very precise estimates for this transition function, and these estimates were used to obtain the desired bounds on ψ and R.

REFERENCES

[1] P. Imkeller, N. S. Namachchivaya, N. Perkowski, and H. Yeong, "Dimensional reduction in nonlinear filtering: A homogenization approach," (2012), submitted.

[2] E. N. Lorenz, "Predictability: A problem partly solved," in *Predictability of Weather and Climate*, (ECMWF, 2006) pp. 40–58.

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

[4] F. Daum and J. Huang, "Curse of dimensionality and particle filters," in *Aerospace Conference, 2003. Proceedings. 2003 IEEE*, Vol. 4 (IEEE, 2003) pp. 1979–1993.

[5] E. Vanden-Eijnden, "Numerical techniques for multi-scale dynamical systems with stochastic effects," Communications in Mathematical Sciences **1**, 385–391 (2003).

[6] W. E, D. Liu, and E. Vanden-Eijnden, "Analysis of multiscale methods for stochastic differential equations," Commun. Pur. Appl. Math. **58**, 1544–1585 (2005).

[7] J. Park, R. B. Sowers, and N. S. Namachchivaya, "Dimensional reduction in nonlinear filtering," Nonlinearity **23**, 305–324 (2010).

[8] J. Park, N. S. Namachchivaya, and H. C. Yeong, "Particle Filters in a Multiscale Environment: Homogenized Hybrid Particle Filter," J. Appl. Mech. **78**, 1–10 (2011).

[9] D. Crisan, "Particle Filters in a Continuous Time Framework," Nonlinear Statistical Signal Processing Workshop, 2006 IEEE **11**, 73–78 (2006).

[10] H. J. Kushner, "Nonlinear Filtering: The Exact Dynamical Equations Satisfied by the Conditional Mode," IEEE Transactions on Automatic Control **AC-12**, 262–267 (1967).

[11] A. Bain and D. Crisan, *Fundamentals of stochastic filtering* (Springer Verlag, 2009).

[12] M. Zakai, "On the Optimal Filtering of Diffusion Processes," Z. Wahrscheinlichkeitstheorie verw. Geb. **11**, 230–243 (1969).

[13] A. J. Majda, I. Timofeyev, and E. Vanden-Eijnden, "A Mathematical Framework for Stochastic Climate Models," Communications on Pure and Applied Mathematics **LIV**, 891–974 (2001).

[14] A. J. Majda, I. Timofeyev, and E. Vanden-Eijnden, "Systematic Strategies for Stochastic Mode Reduction in Climate," Journal of Atmospheric Sciences **60**, 1705–1722 (2003).

[15] S. Wilks, Daniel, "Effects of stochastic parametrizations in the lorenz '96 model," Quarterly Journal of the Royal Meteorological Society **131**, 389–407 (2005).

[16] E. Lorenz and K. Emanuel, "Optimal sites for supplementary weather observations: Simulation with a small model," Journal of Atmospheric Sciences **55**, 399–414 (1998).

[17] S. Herrera, D. Paz, J. Fernandez, and M. A. Rodriguez, "The role of large-scale spatial patterns in the chaotic amplification of perturbations in a lorenz'96 model," Tellus **63**,

978–990 (2011).

[18] J. Harlim, and A. J. Majda, "Filtering nonlinear dynamical systems with linear stochastic models," Nonlinearity **21**, 1281–1306 (2008).

[19] E. L. Kang, and J. Harlim, "Filtering partially observed multiscale systems with heterogeneous multiscale methods-based reduced climate models," Monthly Weather Review **140**, 860–873 (2008).

[20] R. Z. Khas'minskii, "A limit theorem for the solutions of differential equations with random right-hand sides," Theory of Probability and its Applications **11**, 390–406 (1966).

[21] K. Hasselmann, "Stochastic climate models: Part I. Theory," Tellus **28**, 473–485 (1969).

[22] Y. Kifer, "L^2 diffusion approximation for slow motion in averaging," Stochastics and Dynamics **3**, 213–246 (2003).

[23] P. E. Kloeden and E. Platen, *Numerical Solution of Stochastic Differential Equations* (Springer Verlag, Berlin, Heidelberg, 1992).

[24] I. Fatkullin and E. Vanden-Eijnden, "A computational strategy for multiscale systems,," Journal of Computational Physics **200**, 605–638 (2004).

[25] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A Tutorial on Particle Filters for Online Nonlinear/Non-Gaussian Bayesian Tracking," IEEE Transactions on Signal Processing **50**, 174–188 (2002).

[26] N. J. Gordon, D. J. Salmond, and A. F. M. Smith, "Novel approach to nonlinear/non-gaussian Bayesian state estimation," in *IEEE Proceedings-F*, Vol. 120 (1993) pp. 107–113.

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

[28] A. E. Bryson, Jr. and Y.-C. Ho, *Applied Optimal Control: Optimization, Estimation, and Control* (Taylor & Francis, New York, New York, 1975).

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

[30] A. Chorin, M. Morzfeld, and X. Tu, "Implicit Particle Filters for Data Assimilation," Communications in Applied Mathematics and Computational Science **5**, 221–240 (2010).

[31] A. Doucet, "On sequential simulation-based methods for Bayesian filtering," Tech. Rep. (Cambridge University, 1998).

[32] A. Kong, J. S. Liu, and W. H. Wong, "Sequential Imputations and Bayesian Missing Data

Problems," Journal of the American Statistical Association **89**, 278–288 (1994).

[33] G. Evensen and P. J. van Leeuwen, "Assimilation of Geosat Altimeter Data for the Agulhas Current Using the Ensemble Kalman Filter with a Quasigeostrophic Model," Monthly Weather Review **124**, 85–96 (1996).

[34] T. N. Palmer, R. Gelaro, J. Barkmeijer, and R. Buizza, "Singular vectors, metrics, and adaptive observations," Journal of Atmospheric Sciences **55**, 633–653 (1998).

[35] N. Lingala, N. S. Namachchivaya, N. Perkowski, and H. Yeong, "Optimal nudging in particle filters," (2012), submitted for the Proceedings of the IUTAM Symposium on Multiscale Problems in Stochastic Mechanics.

[36] E. Pardoux, "Stochastic partial differential equations and filtering of diffusion processes," Stochastics **3**, 127–167 (1979).

[37] E. Pardoux and A. Y. Veretennikov, "On Poisson equation and diffusion approximation 2," Ann. Probab. **31**, 1166–1192 (2003).