Journal of Computational Physics 396 (2019) 400–415

Contents lists available at ScienceDirect

Journal of Computational Physics

www.elsevier.com/locate/jcp

Sequential Monte Carlo with kernel embedded mappings: The mapping particle filter

Manuel Pulido [a,b,*], Peter Jan van Leeuwen [a,c]

[a] Data Assimilation Research Centre, Department of Meteorology, University of Reading, UK
[b] Department of Physics, FaCENA, Universidad Nacional del Nordeste and CONICET, Argentina
[c] National Centre for Earth Observation, Department of Meteorology, University of Reading, UK

ARTICLE INFO

Article history:
Received 18 October 2018
Received in revised form 15 May 2019
Accepted 24 June 2019
Available online 27 June 2019

Keywords:
Optimal transport
Particle flows
Kernel embedding
Stein Variational Gradient Descent

ABSTRACT

In this work, a novel sequential Monte Carlo filter is introduced which aims at an efficient sampling of the state space. Particles are pushed forward from the prediction to the posterior density using a sequence of mappings that minimizes the Kullback-Leibler divergence between the posterior and the sequence of intermediate densities. The sequence of mappings represents a gradient flow based on the principles of local optimal transport. A key ingredient of the mappings is that they are embedded in a reproducing kernel Hilbert space, which allows for a practical and efficient Monte Carlo algorithm. The kernel embedding provides a direct means to calculate the gradient of the Kullback-Leibler divergence leading to quick convergence using well-known gradient-based stochastic optimization algorithms. Evaluation of the method is conducted in the chaotic Lorenz-63 system, the Lorenz-96 system, which is a coarse prototype of atmospheric dynamics, and an epidemic model that describes cholera dynamics. No resampling is required in the mapping particle filter even for long recursive sequences. The number of effective particles remains close to the total number of particles in all the sequence. Hence, the mapping particle filter does not suffer from sample impoverishment.

1. Introduction

Several applications ranging from meteorology, oceanography, and hydrology to biological systems need to estimate the state of a system from an imperfect numerical model and partial noisy observations. The combination of a dynamical model which represents the nonlinear evolution of the (hidden) state with sparse noisy observations is usually represented through a hidden Markov model, also known as a state-space model [9,5]. The aim of this work is to develop a methodology for sequential Bayesian inference of state-space models composed of nonlinear chaotic dynamical systems with non-Gaussian uncertainty in the state-space variables using a small number of samples.

Particle filters are Monte Carlo techniques that sample the space through so-called particles–state realizations– that represent the probability density of the state conditioned on the partial noisy observations, i.e. the posterior density. The challenge for particle filters is to represent recursively the high probability regions of the posterior density with a limited number of particles. In general, particle filters suffer from filter degeneracy [9]. After a few time iterations, they tend to

* Corresponding author.
E-mail address: pulido@exa.unne.edu.ar (M. Pulido).

https://doi.org/10.1016/j.jcp.2019.06.060

give all the weight to a single particle. In the importance sampling framework, a proposal distribution of the transition density conditioned to the current observations can be chosen to improve the sampling [9,2]. Although, a good choice of a proposal density may alleviate the filter degeneracy, if the filter is applied recursively a resampling step is still required. The application of resampling produces another issue, particle impoverishment, since only a few particles with large weights are kept and copied, the particle diversity is decreased in the resampling step. This is an important contention point for hidden-Markov models in high-dimensional state spaces.

Chorin et al., [7], proposed an implicit sampling scheme in which the particles from a Gaussian proposal density are mapped to the high probability regions of the posterior density through the solution of an algebraic equation for each particle. The latter relates the mode of the proposal density with the mode of the posterior density for each particle. To find the mode of the posterior density for each particle, a minimization is required. To reduce the variance of the weights, Van Leeuwen, [26], proposes a mapping that enforces equal weights by scaling the deterministic movement of particles in the optimal proposal step. Since the high probability region of the proposal density is chosen smaller than the model transition density, the filter can be overly optimistic about its performance [27]. Recently, a two-step mapping has been proposed to alleviate these issues and remove these biases. However, a mathematical understanding of the convergence of the filter in the limit for large number of particles is still missing. Reich, [23], applied optimal transport principles to filters. A linear ensemble transform is proposed that minimizes the Euclidean distance with optimal coupling in one mapping step. The degeneracy problem is still present in this method. It is, however, well suited for localization procedures for high-dimensional applications [6], in which an observation only affects the region of the state space that is close to it in Euclidean distance.

In this work, we propose a novel particle filter which is also based on a mapping from the prior density to the posterior density as the implicit sampling filter [7,3], the implicit equal weight particle filter [26,31] and the nonparametric ensemble transform method [23]. The main difference is that the mapping is done iteratively via a gradient flow. A sampling methodology based on optimal transport is derived to drive the particles from the prediction density to the posterior density. The transport of particles aims to minimize the Kullback-Leibler divergence that measures the differences between the intermediate densities and the posterior density. The mapping is embedded in a reproducing kernel Hilbert space which allows for an analytical expression for the gradient of the Kullback-Leibler divergence. The proposed sampling method for the particle filter is based in the Stein variational gradient descent introduced in [14]. They found a connection between the gradient Kullback-Leibler divergence and the Stein discrepancy. In this work, we develop a sequential Monte Carlo filter based on the variational mapping, evaluate it in three hidden Markov models and discuss its potential applications.

2. Methodology

2.1. Sequential Bayesian estimation

We assume the estimation problem is encompassed of a dynamical model $\mathcal{M}$, which predicts the state $\mathbf{x}$ from a previous state, and an observational model $\mathcal{H}$, which transforms the state from the (hidden) state space to the observational space. The set of equations that defines the estimation problem is known as a state-space model or a hidden Markov model. These are

$$\mathbf{x}_k = \mathcal{M}(\mathbf{x}_{k-1}, \boldsymbol{\eta}_k), \tag{1}$$

$$\mathbf{y}_k = \mathcal{H}(\mathbf{x}_k, \boldsymbol{\nu}_k), \tag{2}$$

where $\mathbf{x}_k \in \mathbb{R}^{N_x}$ is the state at time k, $\mathbf{y}_k \in \mathbb{R}^{N_y}$ are the observations, $\boldsymbol{\eta}_k \sim p(\boldsymbol{\eta})$ is the random model error, and $\boldsymbol{\nu}_k \sim p(\boldsymbol{\nu})$ is the observational error. The method developed here is general and does not rely on the additive or Gaussian assumption in model nor observational errors.

Observations $\mathbf{y}_k$ are assumed to be measured at discrete times. Some apriori knowledge of the state at $k = 0$ is assumed, the initial prior state density $p(\mathbf{x}_0)$. The sequential Bayesian state inference is given in two stages:

1. Firstly, in the evolution stage the prediction density is determined as

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

   where we denote $\{\mathbf{y}_1, \cdots, \mathbf{y}_{k-1}\}$ by $\mathbf{y}_{1:k-1}$. At $k = 1$, $\mathbf{y}_{1:0} = \varnothing$ so that the prediction density is $p(\mathbf{x}_1) = \int p(\mathbf{x}_1|\mathbf{x}_0) p(\mathbf{x}_0) \mathrm{d}\mathbf{x}_0$.
2. Secondly, in the assimilation stage Bayes rule is used to express the inference as a sequential process

$$p(\mathbf{x}_k|\mathbf{y}_{1:k}) = \frac{p(\mathbf{y}_k|\mathbf{x}_k) p(\mathbf{x}_k|\mathbf{y}_{1:k-1})}{p(\mathbf{y}_k|\mathbf{y}_{1:k-1})}, \tag{4}$$

   where $p(\mathbf{y}_k|\mathbf{x}_k)$ is the observation likelihood and $p(\mathbf{y}_k|\mathbf{y}_{1:k-1}) = \int p(\mathbf{y}_k|\mathbf{x}_k) p(\mathbf{x}_k|\mathbf{y}_{1:k-1}) \mathrm{d}\mathbf{x}_k$ is the marginalized likelihood.

Note that the marginalized posterior density for the current state is inferred in Eq. (4) instead of the posterior density for the whole trajectory $\mathbf{x}_{0:K}$. This inference of the current state is useful for systems which routinely receive observations and therefore the estimation is produced only at the current state without keeping information of the whole path of the particles. This approach is followed by ensemble Kalman filters, particle flow filters (e.g. [8]) and the implicit equal-weight particle filter [31,28].

2.2. Particle flows and optimal transport

A filter needs to update its prior knowledge, provided in the prediction density, with the observation likelihood, Eq. (4). The concept of homotopy can be used to smoothly transform the prior density $q(\mathbf{x})$ towards the posterior density $p(\mathbf{x})$. Continuous deformations through a parameter can be used to achieve this, such as $T(\mathbf{x},\lambda):\mathbb{R}^{N_x}\times[0,1]\to\mathbb{R}$, $T(\mathbf{x},\lambda)=p(\mathbf{x})^{\lambda}q(\mathbf{x})^{1-\lambda}$. (To simplify the notation we have left the conditioning to observations implicit in this and the following subsection.) The parameter λ is interpreted as a pseudo-time which is varied from 0 to 1 at a fixed real time. This concept was used in [8] for particle filters and is directly related to tempering [16]. The densities are represented through particles, so that the deformations are interpreted as particles moving in a flow according to a set of ordinary differential equations,

$$\frac{d\mathbf{x}_\lambda}{d\lambda}=\mathbf{v}_\lambda(\mathbf{x}_\lambda), \tag{5}$$

where $\mathbf{v}_\lambda$ is the drift or velocity field. We focus on deterministic flows so that diffusivity processes are not considered. The evolution of the density under this flow is then given by the Liouville equation,

$$\partial_t q_\lambda+\nabla\cdot(\mathbf{v}_\lambda q_\lambda). \tag{6}$$

In some recent works, the particles are pushed smoothly forward in pseudo-times using a Gaussian approximation to represent the velocity field (e.g. [4,13]). On the other hand, in the current work we avoid the use of that approximation by obtaining the gradient flow from optimal local transport.

The optimal transport is another promising approach for the sampling of complex posterior distributions [17]. Given the prior probability mass distribution $q(\mathbf{x})$ and the target probability mass distribution $p(\mathbf{z})$, we want to transport the probability mass from q to p using a mapping $T:\mathbf{x}\to\mathbf{z}$. The optimal transport problem seeks the transformation T that gives the minimal cost to transport the distribution of mass from q to p. This represents the classical Monge optimal transport problem. There is a rigorous proof of the existence of such optimal mapping under mild conditions [18]. Furthermore, [1] have formulated the Monge-Kantorovich problem as a steepest descent gradient optimization.

Our approach combines the ideas of optimal transport and particle flows. We take the local approach to optimal transport [29] in which a gradient flow needs to be obtained. Through a sequence of mappings given by the flow we seek to push forward the particles from the prior to the target density. The mappings in the sequence are required to be as smooth as possible. The particles behave as active Lagrangian tracers in a flow. The velocity field at each pseudo-time step of the mapping sequence is chosen following a local optimal transport criterion.

2.3. Variational mappings

At each iteration of the sequence, we propose a local transformation T that follows Eq. (5),

$$\mathbf{x}_{\lambda+\epsilon}=T(\mathbf{x}_\lambda)=\mathbf{x}_\lambda+\epsilon\mathbf{v}(\mathbf{x}_\lambda), \tag{7}$$

where $\epsilon=\delta\lambda$ is assumed to be small and $\mathbf{v}(\mathbf{x}_\lambda)$ is an arbitrary vectorial function "sufficiently" smooth, which represents the velocity of the flow defined in Eq. (5).

The Kullback-Leibler divergence is used as a measure of the difference between the intermediate density $q(\mathbf{x})$ and the posterior density $p(\mathbf{x})$. The Kullback-Leibler divergence between q and p after the transformation, T, in terms of the inverse mapping is

$$\mathcal{D}_{KL}(T)=\int q_X(\mathbf{x})\log\left(\frac{q_X(\mathbf{x})}{p_{T^{-1}}(\mathbf{x})}\right)\mathrm{d}\mathbf{x}, \tag{8}$$

where $p_{T^{-1}}(\mathbf{x})=p(T(\mathbf{x}))\det\nabla_x T(\mathbf{x})$ and $\det\nabla_x T(\mathbf{x})$ is the determinant Jacobian of the mapping and for brevity $\mathbf{x}=\mathbf{x}_\lambda$ (time index is left implicit).

Our goal is to find the velocity field $\mathbf{v}(\mathbf{x})$ of the transformation T that gives the minimum of $\mathcal{D}_{KL}(T)$ for each mapping of the sequence. In other words, we need to find the direction $\mathbf{v}(\mathbf{x})$, that gives the steepest descent of $\mathcal{D}_{KL}$.

We use the Gateaux derivative of a functional, which is a generalization of the directional derivative. Given the functional F, it is defined as

$$D_h F=\lim_{\epsilon\to0}\frac{F(\mathbf{x}+\epsilon\mathbf{h}(\mathbf{x}))-F(\mathbf{x})}{\epsilon}=\mathrm{d}_\epsilon F(\mathbf{x}+\epsilon\mathbf{h})|_{\epsilon=0}. \tag{9}$$

The Gateaux derivative of the Kullback-Leibler divergence, Eq. (8), in the direction $\mathbf{v}(\mathbf{x})$ is given by

$$D_v\mathcal{D}_{KL} = -\int q(\mathbf{x})\ \mathrm{d}_\epsilon \log p_{T^{-1}}(\mathbf{x})\big|_{\epsilon=0}\ \mathrm{d}\mathbf{x}. \tag{10}$$

The derivative of the transformed log-posterior density is

$$\mathrm{d}_\epsilon \log p_{T^{-1}}(\mathbf{x})\big|_{\epsilon=0} = \nabla_T \log p(T(\mathbf{x}))^\top\, \mathrm{d}_\epsilon T + \mathrm{Tr}(\nabla_x T^{-1} \mathrm{d}_\epsilon \nabla_x T)\big|_{\epsilon=0}. \tag{11}$$

Considering that $T(\mathbf{x}) = \mathbf{x} + \epsilon\mathbf{v}(\mathbf{x})$ in Eq. (11) and replacing in Eq. (10), the directional Gateaux derivative of $\mathcal{D}_{KL}$ results in

$$D_v\mathcal{D}_{KL} = -\int q(\mathbf{x}) \Big[\nabla_x \log p(\mathbf{x})^\top \mathbf{v}(\mathbf{x}) + \mathrm{Tr}(\nabla_x \mathbf{v})\Big]\ \mathrm{d}\mathbf{x}. \tag{12}$$

Equation (12) gives the $\mathcal{D}_{KL}$ derivative along $\mathbf{v}$. However, we require the negative gradient of $\mathcal{D}_{KL}$ in terms of the samples of q for the optimization of $\mathcal{D}_{KL}$ as a function of T. In general, the flow $\mathbf{v}$ belongs to an infinite dimensional Hilbert space. Hence, the full optimization problem is still intractable in practice since we do not have a way to determine the $-D_v\mathcal{D}_{KL}$ that gives the steepest descent direction.

One way to limit our functional optimization problem so that it becomes tractable is choosing as space of functions the unit ball of a reproducing kernel Hilbert space (RKHS), which we denote as $\mathcal{F}$. This was proposed by [14]. In this way, we constrain $\mathbf{v}$ to $\mathcal{F}$ and require that $\|\mathbf{v}\|_{\mathcal{F}} \leq 1$ to find the gradient of $\mathcal{D}_{KL}$. The optimization problem is then to find the $\mathbf{v} \in \mathcal{F}$ that gives the direction of steepest descent of $\mathcal{D}_{KL}$. The main properties of the RKHS are discussed in [24].

The functions to be represented in the RKHS are vector-valued, i.e. $\mathbf{v}(\mathbf{x}) \in \mathbb{R}^{N_x}$. The kernel is thus assumed to be diagonal and isotropic, $\mathbf{K} = K\mathbf{I}_{N_x}$, where $\mathbf{I}_{N_x}$ is the identity matrix and K is a scalar kernel. The reproducing property states that any function from $\mathcal{F}$ can be expressed as the dot product by the kernel that defines the RKHS,

$$\mathbf{v}(\mathbf{x}) = \langle \mathbf{K}(\cdot, \mathbf{x}), \mathbf{v}(\cdot)\rangle_{\mathcal{F}}. \tag{13}$$

Equation (13) is known as the reproducing property. The scalar kernel K defines $\mathcal{F}^1$.

Replacing the vector field, $\mathbf{v}(\mathbf{x})$, in the two terms in Eq. (12) by the expression given in Eq. (13) and then using dot product properties, we find that

$$D_v\mathcal{D}_{KL} = \langle -\int q(\mathbf{x}) \left[K(\mathbf{x}, \cdot)\nabla_x \log p(\mathbf{x}) + \nabla_x K(\mathbf{x}, \cdot)\right]\ \mathrm{d}\mathbf{x},\ \mathbf{v}(\cdot)\rangle_{\mathcal{F}^1}. \tag{14}$$

This is valid for any $\mathbf{v}$ such that $\|\mathbf{v}\|_{\mathcal{F}} \leq 1$. Therefore, the first term of the dot product in Eq. (14) is by definition the gradient of $\mathcal{D}_{KL}$ at $T_{\epsilon=0} = \mathbf{x}$, $D_v\mathcal{D}_{KL} = \langle \nabla\mathcal{D}_{KL}, \mathbf{v}\rangle_{\mathcal{F}^1}$ so that

$$\nabla\mathcal{D}_{KL}(\mathbf{x}) = -\mathcal{E}_{x'\sim q}\left[K(\mathbf{x}', \mathbf{x})\nabla_x \log p(\mathbf{x}') + \nabla_x K(\mathbf{x}', \mathbf{x})\right], \tag{15}$$

where $\mathcal{E}$ represents the expectation operator $\mathcal{E}_{x\sim p}[f(x)] \triangleq \int p(x) f(x)\mathrm{d}x$. (Note that by $D_v\mathcal{D}_{KL}(\mathbf{x})$ we denote the differences between $\mathcal{D}_{KL}$ with a mapping T at $\mathbf{x}$ in a direction $\mathbf{v}$ from the Kullback-Leibler divergence without the mapping, i.e. the Gateaux derivate, while $\nabla\mathcal{D}_{KL}(\mathbf{x})$ denotes the direction in which $\mathcal{D}_{KL}$ increases the most when mappings of the form Eq. (7) are conducted at $\mathbf{x}$.)

This expression for the gradient, Eq. (15), is particularly suitable for Monte-Carlo integration when q is only known through a set of sample points. Since we seek to minimize the Kullback-Leibler divergence we choose as direction of the transformation T the negative of its gradient, the steepest descent direction,

$$\mathbf{x}_{\lambda+\epsilon} = \mathbf{x} - \epsilon\nabla\mathcal{D}_{KL}(\mathbf{x}). \tag{16}$$

Using the kernel reproducing property and integration by parts in Eq. (15), the resulting gradient flow of the mapping is

$$\mathbf{v}_\lambda(\mathbf{x}) = -\nabla\mathcal{D}_{KL}(\mathbf{x}) = q_\lambda(\mathbf{x})\nabla \log p(\mathbf{x}) - \nabla q_\lambda(\mathbf{x}). \tag{17}$$

As shown in [25], the gradient flow, Eq. (17), has as stationary solution $q(\mathbf{x}) = p(\mathbf{x})$ so that the gradient flow converges toward the target density.

In [19], the Kullback-Leibler divergence is also used but with an extra regularization term to find a single global optimal map. Here, we apply a gradient flow via a sequence of local mappings. Under weak smoothness constrains [29], the minimum of the cost function as a function of the mappings, Eq. (7), is uniquely determined in Eq. (16), i.e. no regularization term is required.

2.4. Mapping particle filter

Consider we have a set of equal-weighted particles $\mathbf{x}_{k-1}^{(1:N_p)}$ which sample the posterior density at time $k-1$. The target density at time k which we aim to sample using the variational mapping is the posterior density $p(\mathbf{x}_k|\mathbf{y}_{1:k})$. The mapping approach only requires a set of samples of the prior density. It is started by the set of unweighted particles that are evolved from the previous estimate to the present assimilation time by the dynamical model, i.e. $\left\{\mathbf{x}_{k,0}^{(j)} = \mathcal{M}(\mathbf{x}_{k-1}^{(j)}, \boldsymbol{\eta}_k^{(j)})\right\}_{j=1}^{N_p}$, where the second subscript represents the mapping iteration. These sample points of the prediction density are then pushed towards the sequential posterior density by the mapping iterations.

Given the set of particles $\mathbf{x}_{k,i-1}^{(1:N_p)}$ that are samples of the intermediate density at mapping iteration $i-1$, the gradient of the Kullback-Leibler divergence from Eq. (15) at a state space position $\mathbf{x}$ by Monte-Carlo integration is

$$\nabla\mathcal{D}_{KL}(\mathbf{x}) = -\frac{1}{N_p}\sum_{l=1}^{N_p}\left[K(\mathbf{x}_{k,i-1}^{(l)}, \mathbf{x})\nabla\log p(\mathbf{x}_{k,i-1}^{(l)}) + \nabla_x K(\mathbf{x}_{k,i-1}^{(l)}, \mathbf{x})\right]. \tag{18}$$

This expression is equivalent to the one obtained in [14] for a time independent inference. The negative of this gradient represents the velocity field of the gradient flow.

The first term of the gradient of the Kullback-Leibler divergence, Eq. (18), tends to drive the particles towards the peaks of the posterior density. It gives a weighted average of $\nabla\log p$ including the current particle and surrounding ones within the kernel scale of the current one. This is the typical behavior of a variational importance sampler so that it accumulates particles at the high probability regions of the posterior density. On the other hand, the second term of $\nabla\mathcal{D}_{KL}$ in Eq. (18) tends to separate the particles. If the current particle $\mathbf{x}_k^{(j)}$ is in the influence region of another particle say $\mathbf{x}_k^{(l)}$, i.e. within the kernel scale, the term $\nabla_{\mathbf{x}_{k,i-1}^{(l)}} K(\mathbf{x}_{k,i-1}^{(l)}, \mathbf{x}_{k,i-1}^{(j)})$ will tend to separate them acting as a repulsive force between particles.

At the mapping iteration i, the particle j is transformed, according to the mapping Eq. (16), by

$$\mathbf{x}_{k,i}^{(j)} = \mathbf{x}_{k,i-1}^{(j)} - \epsilon\,\nabla\mathcal{D}_{KL}(\mathbf{x}_{k,i-1}^{(j)}), \tag{19}$$

where $-\nabla\mathcal{D}_{KL}(\mathbf{x}_{k,i-1}^{(j)})$ is the steepest descent direction at the particle position. Eq. (19) may be interpreted as the movement of the particles along the streamlines of the flow assuming small ϵ.

An ingredient needed to evaluate Eq. (18) is an expression for the gradient of the posterior density. A problem in sequential Bayesian inference is that there is no exact expression for the posterior density. We do know the likelihood function, but we only have a set of particles that represent the prior density, not the density itself. The prior density using the particle representation of the posterior density at time $k-1$ in Eq. (3) results in

$$p(\mathbf{x}_k|\mathbf{y}_{1:k-1}) \approx \frac{1}{N_p}\sum_{j}^{N_p} p(\mathbf{x}_k|\mathbf{x}_{k-1}^{(j)}), \tag{20}$$

and the expression for the posterior density from Eq. (4) becomes

$$p(\mathbf{x}_k|\mathbf{y}_{1:k}) \propto \frac{1}{N_p}p(\mathbf{y}_k|\mathbf{x}_k)\sum_{j}^{N_p} p(\mathbf{x}_k|\mathbf{x}_{k-1}^{(j)}). \tag{21}$$

Because of the finite ensemble size at time $k-1$, this expression is an approximation of the posterior density. Equation (21) is the target density in the mapping sequence.

At each mapping iteration, all the particles are moved following Eq. (19). Successive applications of the transformation, through gradient descent, with the corresponding updates of $\nabla\mathcal{D}_{KL}$, will converge towards the minimum of the Kullback-Leibler divergence. The pseudo-code of the implemented algorithm which combines variational mapping with the sequential particle filter is shown in Algorithm 1.

The form in which the variational mapping is obtained, as steepest gradient descent, is suitable for the stochastic optimization algorithms used in the machine learning community. In this case, ϵ, known as the learning rate, can be determined adaptively using $\nabla\mathcal{D}_{KL}$ from previous iterations (e.g. [30]).

An important part of any iterative variational method is a stopping criterion. One possibility in Algorithm 1 is to check the value of $|\nabla\mathcal{D}_{KL}|$ for each particle, or averaged over all particles. Another option is to use importance sampling and to interpret the final density of the mapping sequence as a proposal density and calculate weights with respect to the posterior density, which will then automatically lead to an unbiased estimator. To use this, an expression for the final intermediate density q is required, however, we have only its particle representation. To implement this in a practical way, the density transformations can be used, that we do have explicitly, and the final density can be related to the prior density. In the Appendix, the implementation of importance sampling within the variational mapping particle filter is described.

Algorithm 1 Mapping particle filter algorithm.

Input: Given $\mathbf{x}_{k-1}^{(1:N_p)}$, $\mathbf{y}_k$, $\mathcal{M}(\cdot)$, $p(\mathbf{y}|\mathbf{x})$ and $p(\boldsymbol{\eta})$
  for $j = 1, N_p$ **do**
    $\mathbf{x}_{k,0}^{(j)} \leftarrow \mathcal{M}(\mathbf{x}_{k-1}^{(j)}, \boldsymbol{\eta}_k)$ ▷ Forecast stage
  end for
  repeat ▷ Mapping iterations
    for $j = 1, N_p$ **do**
      $\mathbf{x}_{k,i}^{(j)} \leftarrow \mathbf{x}_{k,i-1}^{(j)} - \epsilon \nabla \mathcal{D}_{KL}(\mathbf{x}_{k,i-1}^{(j)})$ ▷ $\nabla \mathcal{D}_{KL}$ from Eq. (18)
    end for
    i ← i+1
  until Stopping criterion met
Output: $\mathbf{x}_{k,i}^{(1:N_p)}$

To avoid the resampling in the importance sampling step, one can calculate the effective ensemble size as N_{eff} (see Appendix), and decide to continue the iterations in Algorithm 1 until N_{eff} reaches a certain threshold N_t. In the following assimilation cycle, the weights of the particles representing the final intermediate density has to be considered in the sequential posterior density, resulting in a weighted posterior density instead of Eq. (21). In this way, the mapping particle filter would readily extend to include importance sampling. One should keep in mind, however, that our estimate of the posterior density is rather poor in high-dimensional systems with a small number of particles, so the weights would not be very accurate. Thus, our standard implementation uses the magnitude of the gradient as the stopping criteria, without calculating the weights.

3. Numerical experiments

The sequential implementation of the mapping particle filter is evaluated using three chaotic nonlinear dynamical models with state spaces of up to 40 dimensions (see Section 3.1). As usual in these experiments, the observational and model errors are assumed Gaussian. A derivation of the hidden Markov model with Gaussian errors is given in Section 3.2. We conducted a set of stochastic twin experiments, in which the dynamical model used to produce the synthetic observations is the same as the one used in the state-space model, including the statistical parameters. Details of the experiments are given in Section 3.3.

3.1. Description of the systems

The Lorenz-63 system is given by

$$\begin{aligned} \frac{\mathrm{d}x}{\mathrm{d}t} &= \sigma(y - x), \\ \frac{\mathrm{d}y}{\mathrm{d}t} &= x(\rho - z) - y, \\ \frac{\mathrm{d}z}{\mathrm{d}t} &= xy - \beta z. \end{aligned} \tag{22}$$

The equations are solved using a fourth-order Runge-Kutta scheme. The parameters are chosen at their standard values $\sigma = 10$, $\rho = 28$ and $\beta = 8/3$. The integration time step is 0.001 and the assimilation cycles are every 0.01 (10 integration time steps). The observation operator $\mathcal{H}$ is taken as the identity matrix, for the full-observed state experiments in Section 4. The model error covariance $\mathbf{Q}$ is chosen diagonal. Its diagonal elements are set to 30% of the climatological variance of the variable, e.g. $Q_{11} = 0.3\sigma_x^2$, where σ_x^2 is the time-series variance of variable x. The observation error covariance is $\mathbf{R} = 0.5\mathbf{I}$.

The stochastic dynamical model for cholera is a 5-variable system allowing for susceptible, infected, and three classes of recovery individuals in which cholera mortality is the only observed variable. From the inference point of view, it presents some challenges [10]. It has partial noisy observations whose variance depends on time. Transmission is assumed stochastic and has a multiplicative model error. The compartment dynamical model for cholera used in this work is the one throughly described in [10]. A short description is given here. The individuals are classified as susceptible (S), infected individuals (I), while recovered individuals belong to three classes ($R^{1:3}$) to allow for different immune periods. The equations for the number of individuals for each category are given by

$$\begin{aligned} dS_t &= dN_t^{BS} - dN_t^{NI} - dN_t^{SD} + dN_t^{R^k S} \\ dI_t &= dN_t^{SI} - dN_t^{IR^1} - dN_t^{IC} + dN_t^{ID} \\ dR_t^1 &= dN_t^{IR^1} - dN_t^{R^1R^2} - dN_t^{R^1D} \\ dR_t^2 &= dN_t^{R^1R^2} - dN_t^{R^2S} - dN_t^{R^2D} \\ dR_t^3 &= dN_t^{R^2R^3} - dN_t^{R^3S} - dN_t^{R^3D} \end{aligned} \tag{23}$$

Transmission is given by a stochastic differential equation

$$dN_t^{SI} = \lambda_t S_t dt + \epsilon I_t S_t / P_t dW_t, \tag{24}$$

where dW_t is a Gaussian white noise. The transitions between categories are given by

$$\begin{aligned} dN_t^{IR^1} &= \gamma I_t dt, & dN_t^{R^{l-1}R^l} &= rkR_t^{l-1} dt, \\ dN_t^{R^l S} &= rkR_t^k dt, & dN_t^{SD} &= mS_t dt, \\ dN_t^{ID} &= mI_t dt, & dN_t^{R^l D} &= mR_t^l dt, \\ dN_t^{IC} &= m_c I_t dt, & dN_t^{BS} &= dP_t + mP_t dt, \end{aligned} \tag{25}$$

where $l = 2, 3$, B represents birth, C is cholera mortality and D denotes death from other causes. Equations (23) are integrated using an Euler-Maruyama scheme with time step of 1/20 month. The observations are cholera mortality data which are given by

$$y_t \sim \mathcal{N}\left[N_t^{IC} - N_{t-1}^{IC}, \tau^2 (N_t^{IC} - N_{t-1}^{IC})^2\right]. \tag{26}$$

While the mapping particle filter can handle non-additive model errors, we transform the system of equations above to allow for an additive model error implementation. We use an augmented state space defining a new state variable T as $dT_t = dW_t$, so that the augmented system has additive Gaussian model error whose gradient of the log-posterior density is represented by Eq. (30). Note that a non-Gaussian density for additive model error would give more realistic features. To add diversity to the particles in the filter a small additive noise term is added in all the equations [15]. The variance of this additive noise is set to 10% of the dW_t variance. This noise is not used to generate the true trajectory, it is only included in the particle filter.

For the 40-variable Lorenz-96 system experiments, the set of equations is

$$\frac{\mathrm{d}x_i}{\mathrm{d}t} = (x_{i+1} - x_{i-2})x_{i-1} - x_i + F, \tag{27}$$

where $i = 1, \cdots, 40$ and cyclic boundary conditions are imposed, $x_0 = x_{40}$, $x_{-1} = x_{39}$ and $x_{41} = x_1$. The equations were integrated using a fourth-order Runge-Kutta scheme, with an integration step of 0.001. The forcing is $F = 8$ which results in chaotic dynamics. The observational time resolution is 0.05. The observational and model error covariances are $\mathbf{R} = 0.5\mathbf{I}$ and $\mathbf{Q} = 0.3\mathbf{I}$, respectively. This amplitude of the model error noise is representative for the two-scale Lorenz 96 system which was estimated to be 0.3 using information measures by [20] and 0.3-0.5 using an Expectation-Maximization algorithm coupled to an ETKF [21]. The initial ensemble particles are states taken randomly from a climatology.

3.2. Hidden Markov models with Gaussian errors

Next, we derive the expression of the posterior density in the sequential framework as a function of the particles at $k-1$. For the numerical experiments, model and observational errors are assumed Gaussian in the three dynamical systems used in this work (Section 3.1). These are taken as an example, but the framework is general. In particular, it is suitable for non-additive and non-Gaussian errors. The prediction probability density can be obtained analytically under the additive Gaussian model error assumption, $\boldsymbol{\eta}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_k)$ in Eq. (1). The evolution of the particles from Eq. (3) gives

$$p(\mathbf{x}_k|\mathbf{y}_{1:k-1}) \propto \sum_{m=1}^{N_p} \exp\left[-\frac{1}{2}\|\mathbf{x}_k - \mathcal{M}(\mathbf{x}_{k-1}^{(m)})\|_{\mathbf{Q}_k}^2\right], \tag{28}$$

where $\left\|\mathbf{x}_k - \mathcal{M}(\mathbf{x}_{k-1}^{(m)})\right\|_{\mathbf{Q}_k}^2 \triangleq (\mathbf{x}_k - \mathcal{M}(\mathbf{x}_{k-1}^{(m)}))^\top \mathbf{Q}_k^{-1} (\mathbf{x}_k - \mathcal{M}(\mathbf{x}_{k-1}^{(m)}))$. We assume equal-weighted particles here; it is straightforward to assume weighted particles at time $k-1$ if so desired.

The observational error is also assumed additive Gaussian, $\boldsymbol{\nu}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_k)$ in Eq. (2), so that the observational likelihood at k is

$$p(\mathbf{y}_k|\mathbf{x}_k) \propto \exp\left[-\frac{1}{2}\left\|\mathbf{y}_k - \mathcal{H}(\mathbf{x}_k)\right\|_{\mathbf{R}_k}^2\right]. \tag{29}$$

The Bayes rule, Eq. (4), is used to obtain the sequential posterior density combining the prediction density, Eq. (28), and the observation likelihood, Eq. (29). The sequential posterior density given the particles at the previous time step is proportional to

$$p(\mathbf{x}_k|\mathbf{y}_{1:k}) \propto \sum_{m=1}^{N_p} \exp\left[-\frac{1}{2}\left\|\mathbf{x}_k - \mathcal{M}(\mathbf{x}_{k-1}^{(m)})\right\|_{\mathbf{Q}_k}^2\right] \cdot \exp\left[-\frac{1}{2}\left\|\mathbf{y}_k - \mathcal{H}(\mathbf{x}_k)\right\|_{\mathbf{R}_k}^2\right]. \tag{30}$$

This is the target density that the particles $\mathbf{x}_k^{(1:N)}$ as Monte Carlo samples should be representing in the filter.

The gradient of the logarithm of the sequential posterior density, Eq. (30), at $\mathbf{x}_{k,i-1}^{(l)}$ is given by

$$\nabla_{\mathbf{x}} \log p(\mathbf{x}_{k,i-1}^{(l)}) = Cp^{-1}(\mathbf{x}_{k,i-1}^{(l)}) \sum_{m=1}^{N_p} \psi_{l,m}^{Q} \psi_l^{R} \left\{ \mathbf{H}^{\top} \mathbf{R}_k^{-1} \left[\mathbf{y}_k - \mathcal{H}(\mathbf{x}_{k,i-1}^{(l)}) \right] - \mathbf{Q}_k^{-1} \left[\mathbf{x}_{k,i-1}^{(l)} - \mathcal{M}(\mathbf{x}_{k-1}^{(m)}) \right] \right\} \tag{31}$$

with $\psi_{l,m}^{Q} = \exp\left[-\frac{1}{2} \| x_{k,i-1}^{(l)} - \mathcal{M}(\mathbf{x}_{k-1}^{(m)}) \|_{\mathbf{Q}_k}^2 \right]$, $\psi_l^{R} = \exp\left[-\frac{1}{2} \| \mathbf{y}_k - \mathcal{H}(\mathbf{x}_k^{l}) \|_{\mathbf{R}_k}^2 \right]$ and $p(\mathbf{x}_{k,i-1}^{(l)}) = C \sum_{m=1}^{N_p} \psi_{l,m}^{Q} \psi_l^{R}$. Reducing we obtain

$$\nabla_{\mathbf{x}} \log p(\mathbf{x}_{k,i-1}^{(l)}) = \mathbf{H}^{\top} \mathbf{R}_k^{-1} \left(\mathbf{y}_k - \mathcal{H}(\mathbf{x}_{k,i-1}^{(l)}) \right) - \mathbf{Q}_k^{-1} \left[\mathbf{x}_{k,i-1}^{(l)} - \frac{\sum_{m=1}^{N_p} \psi_{l,m}^{Q} \mathcal{M}(\mathbf{x}_{k-1}^{(m)})}{\sum_{m=1}^{N_p} \psi_{l,m}^{Q}} \right]. \tag{32}$$

This gradient of the log-posterior, Eq. (32), does not depend on the unknown normalization constant of the sequential posterior density. Hence, the numerical evaluation of the gradient of the Kullback-Leibler divergence is feasible.

3.3. Experiment details

A Gaussian kernel is chosen for the experiments,

$$K(\mathbf{x}, \mathbf{x}') = \exp\left[-\frac{1}{2} (\mathbf{x} - \mathbf{x}')^{\top} \mathbf{A}^{-1} (\mathbf{x} - \mathbf{x}') \right]. \tag{33}$$

The kernel covariance $\mathbf{A}$ is taken proportional to the model error covariance matrix, $\mathbf{Q}$, i.e. $\mathbf{A} = \alpha \mathbf{Q}$. This appears to be a convenient choice since the model uncertainty, $\mathbf{Q}$, already includes the physics so that it is expected to represent the scaling between the variables and also the correlations between variables. Under this choice the only parameter of the kernel that requires to be defined is α. With growing dimension of the state vector and the number of particles small, the distance between the particles grows larger. To account for this, α should be chosen larger to increase the distance between particles. In the experiments, we have found that α should be chosen of the order of the dimension of the state vector.

3.4. The stochastic optimization methods and their convergence

An optimization of the Kullback-Leibler divergence is conducted in the mapping particle filter to push the prior particles to samples from the posterior density. The particles are driven in a purely deterministic transformation by the variational mapping from the prior density to the posterior. The intermediate densities are represented through particles. Each particle is then moved along the direction of the steepest descent of the Kullback-Leibler divergence, which considers all the particle positions.

Several stochastic gradient-based optimization methods have been recently developed in the machine learning community [30,11] which are directly applicable to the mapping particle filter. They are focused on efficient first-order convergence in high-dimensional control spaces when the cost function is noisy. The source of this noise is subsampling. These optimization methods, and their successful convergence rates, have been instrumental for the success of deep learning applications (e.g. [12]). The variational mapping in this work is based on stochastic gradient-based optimization methods.

The gradient descent method has a fixed learning rate for all the iterations and all the control variables. This is well-known to cause some inconveniences, in particular along extended valleys where zig-zag convergence arises. As the gradient decreases in some directions, the convergence along those variables is very slow. The adaptative schemes search for a dynamic learning rate for each control variable. Ideally, a Newton method would be the optimal learning rate if the optimization problem is quadratic, but it requires Hessian evaluations which are computationally prohibitive in high-dimensional spaces. The learning rate for each control variable in the stochastic optimization method, adadelta [30], is based on a global learning rate over a running average of the past absolute directional derivatives along the direction of the control variable. Therefore, the learning rate increases for directions with small directional derivatives. The method improves substantially but still some stagnation after the first iterations has been reported. To overcome this weakness, the algorithm for stochastic optimization adam [11] estimates the first and second moments of the gradients with bias correction. These two stochastic optimization algorithms, adadelta and adam, together with a descent gradient method were evaluated in the experiments. The chosen initial learning rate of the optimization methods is a tradeoff between convergence speed and conserving the smoothness of the flow. In this sense, considering that each particle is following a streamline, the particles should not cross the streamlines of other particles. The values recommended in [30] exhibited a good performance, so that no manual tuning of the learning rate was required.

These first-order adadelta and adam optimization methods were chosen because the cost function was expected in principle to be noisy. These methods only account for the diagonal of the Hessian using an adaptative step in each variable. For higher-dimensional optimization problems, the information of off-diagonal elements of the Hessian may be essential for relatively smooth cost functions. In that case, quasi-Newton methods may be a feasible option since the required number of

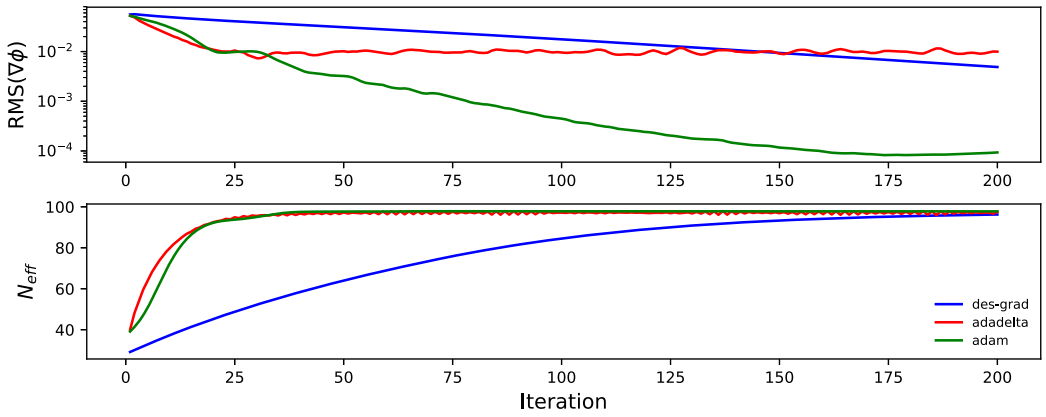

Fig. 1. The root-mean-square of the $\mathcal{D}_{KL}$ gradient as a function of the mapping iteration for the descent gradient (blue line), adadelta (red line) and adam (green line) optimization algorithms (upper panel). This corresponds to the first assimilation cycle of the Lorenz-63 experiment in which the three optimization algorithms share the same prior density. The number of effective particles as a function of the mapping iteration for $N_p = 100$ (lower panel). (For interpretation of the colors in the figure(s), the reader is referred to the web version of this article.)

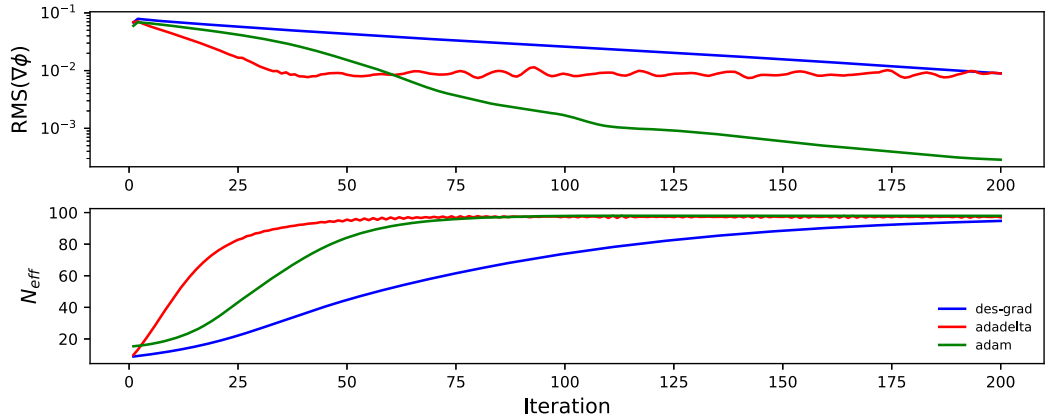

Fig. 2. As Fig. 1 but for the 10-th time iteration, so that the prior density for the optimization algorithms is not the same.

particles in the filter is expected to be small (< 200). In terms of the convergence rate, note that the optimization step in the mapping is constrained. It should be relatively small to avoid that particle paths intersect.

The convergence of the variational mapping is shown in Fig. 1 as a function of the mapping iteration at the first assimilation cycle. At this cycle, the three optimization methods, descent gradient, adadelta and adam, have the same initial density. Upper panels of Fig. 1 show the root-mean-square of the Kullback-Leibler gradient, and lower panels the number of effective particles. The adadelta converges fastest, close to adam. For a larger number of iterations, the gradient of Kullback-Leibler divergence is still diminishing for adam but it shows almost no change after 25 iterations for adadelta. However, the number of effective samples does not differ between them, 40 iterations are required to achieve the maximum–98% of the total number of particles. The descent gradient requires about 200 iterations to achieve this number of effective particles.

Fig. 2 shows the convergence at the 10th assimilation cycle of the optimization methods for evaluating the recursive effects. Notably, adadelta has the fastest convergence with the minimum root-mean-square gradient similar to the one obtained in the first iteration. On the other hand, adam converges slowly in the first mapping iterations and starts to increase the convergence rate after 20 iterations (an effect that is likely caused by the momentum equations in adam which require some iterations to "warm up"). Adadelta has reached the maximum number of effective particles in 50 iterations. Because of the fast convergence and because the effective particle numbers is the most relevant parameter for the particle filter we took adadelta with 50 iterations and an initial learning rate of 0.03 as the default setting for the experiments.

There is a clear correlation between the decrease of the root-mean-square of the Kullback-Leibler gradient and the increase in the number of effective particles in Fig. 2 for the three stochastic optimization methods. This implies as mentioned that both measures could be used to determine the convergence in the optimization and the required number of iterations for a given threshold. Along this line, we note that the Kullback-Leibler divergence can be directly expressed in terms of the weights.

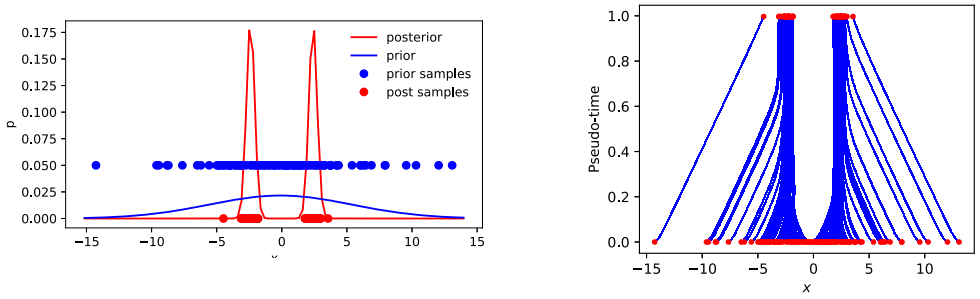

Fig. 3. The Gaussian prior density (blue line) and the bimodal posterior density (red line), blue dots are the samples from the prior density (for visibility has been located at $p = 0.05$) and red dots show the result of applying the mapping particle filter in a single observation cycle (left panel). The path of the particles as a function of pseudo time λ (right panel), at $\lambda = 0$ they are the samples of the prior density and at $\lambda = 1$ is the result of the filter when the convergence criteria are satisfied.

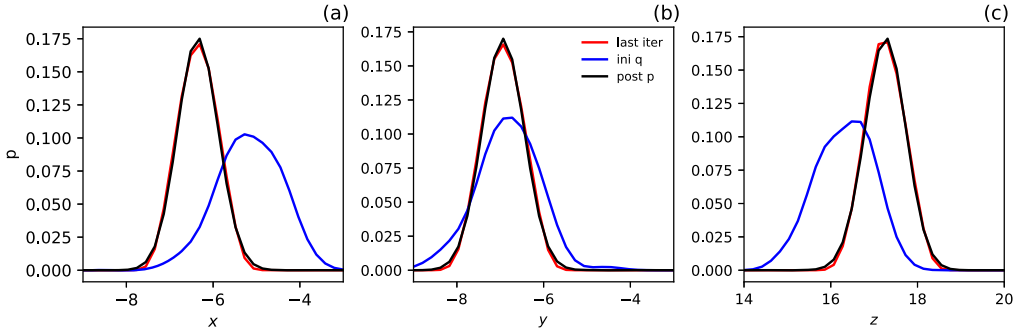

Fig. 4. Marginalized prior density for each variable of Lorenz-63 system determined at the 115-th time iteration, $k = 115$ (blue line), the final density after the variational mappings (red line) and the marginalized sequential posterior "target" distribution (black).

Using Monte Carlo integration of the Kullback-Leibler divergence,

$$\mathcal{D}_{KL}(q\|p) = \frac{1}{N_p}\sum_{j=1}^{N_p} \log\left[\frac{q(\mathbf{x}_k^{(j)})}{p(\mathbf{x}_k^{(j)}|\mathbf{y}_{1:k})}\right], \tag{34}$$

from Eq. (A.1), we find

$$\mathcal{D}_{KL}(q\|p) = -\frac{1}{N_p}\sum_{j=1}^{N_p} \log\left(w_k^{(j)} N_p\right). \tag{35}$$

As expected if the weigths are equally distributed in the particles, the Kullback-Leibler divergence is zero, then $q(\mathbf{x}_k) \approx p(\mathbf{x}_k|\mathbf{y}_{1:k})$. In other words, as the number of effective particles N_{eff} tends to N_p, the Kullback-Leibler divergence tends to zero, its minimum.

4. Results

A first simple experiment illustrates the potential of the method in a non-Gaussian inverse problem. Suppose a one-dimensional space with Gaussian a priori and Gaussian observational error. The observational operator is the module of the state, i.e. $\mathcal{H}(x) = |x|$. In this case, the resulting posterior density is a bimodal distribution. We apply the mapping particle filter to assimilate a single observation. The experiment could represent an instrument that measures the speed and the state variable is the velocity. Fig. 3a shows the prior and posterior distribution. Blue dots corresponds to the prior density samples while red dots represent the dots after application of the mapping particle filter. Fig. 3b shows the paths of the particles as a function of pseudo-time λ. The particles following the paths of minimum transport cost flow nicely toward the bimodal distribution.

Fig. 4 shows the marginalized prior density in the experiment with the Lorenz-63 system as a function of the three state variables at the 115-th time assimilation cycle, $k = 115$ (each panel exhibits a variable). For plotting, the marginalized density is determined with kernel density estimation. The cycle was chosen so that the differences between the mapping particle filter (MPF) and the sampling importance resampling (SIR) filter are emphasized (i.e. a cycle for which the a priori density is not a good representation of the posterior density). The experiment uses 100 particles and the full state is

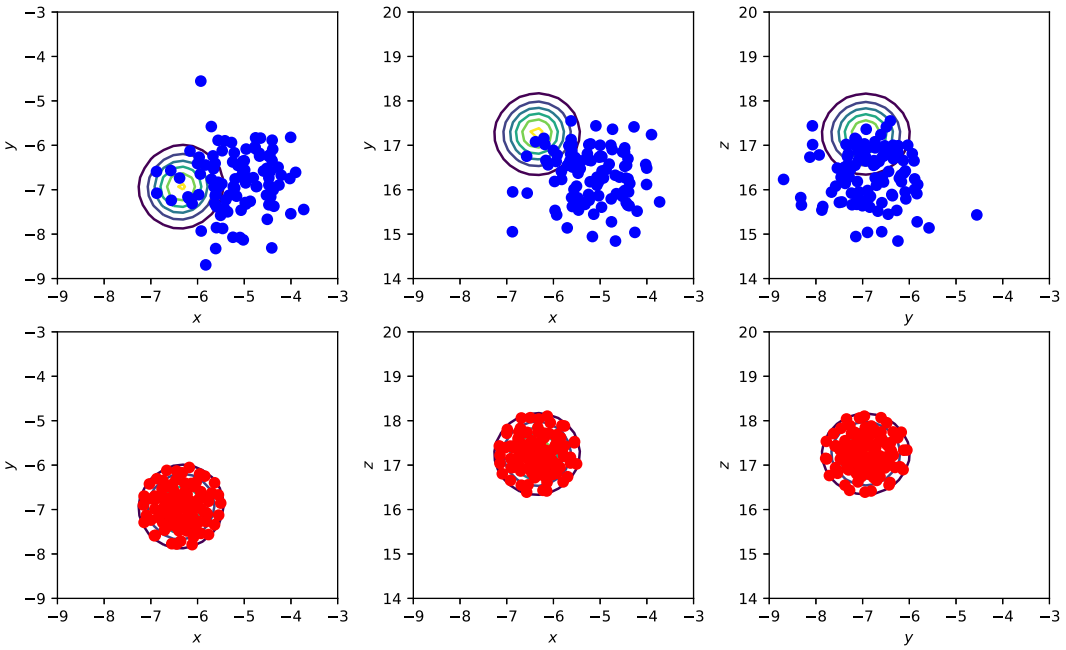

Fig. 5. Marginalized posterior density (contours) and the particles sampling the prior density at the 115-th time iteration for the three sections at the mode location (upper panels). The particles sampling the final density after the variational mapping with 50 iterations (lower panels).

observed. Because particles are spread in and out the high observation likelihood region, the proposal density is broader, asymmetric compared with the marginalized sequential posterior density, Eq. (30). After the variational mapping, the resulting density is a close representation of the sequential posterior density, the target density for the mapping, in the three variables (see the three panels in Fig. 4).

The distribution of the particles can be seen in Fig. 5. Upper panels of Fig. 5 show the particles belonging to the prior density for each variable, the initial density for the mapping. Contours show the posterior density, marginalized to the 2-dimensional plane. Lower panels show the final locations of the samples after 50 mapping iterations. The mapping has pushed most particles towards the high probability region of the posterior density. Note that the algorithm is not collapsing the particles toward the mode of the posterior, but representing the density with the samples as a whole. The diversity of the particles produced by the mapping is notable.

The same experiment conducted with the standard SIR, or bootstrap filter using resampling when $N_{eff} < N_p/2$ is shown in Fig. 6. The sampling deficiencies in the SIR filter are visible in Fig. 6 where the particles of the prior density (upper panels) and the resampled density (lower panels) are shown with dots. Because the resampling conserves and replicates statistically only the particles with high likelihood, there is an evident sample impoverishment even in this low-dimensional experiment.

Because of the good sampling, the MPF exhibits a very weak sensitivity to the number of particles in the root-mean-square error (RMSE) metrics. Fig. 7 shows the RMSE of the analysis mean with respect to the true state, for MPF and SIR filters for 100, 20 and 5 particles (Fig. 7a, b and c respectively). Even for a large number of particles (with respect to the state dimension), the MPF outperforms the mean estimation with respect to the SIR filter. SIR filter diverges for 5 particles. Fig. 7a, b and c shows that the time-mean RMSE for the MPF practically does not change between 100 particles with a time-mean RMSE of 0.482 to 5 particles whose time-mean RMSE is 0.489. Thus, the performance of the MPF is quite robust, only small changes in the RMSE are found when decreasing the number of particles.

Note the complexity of Algorithm 1 is proportional to N_p^2 while it is proportional to N_p for the SIR filter, so that the computational cost of the assimilation stage is higher for the MPF with the same number of particles. However, it requires a smaller number of particles so that it represents a promising venue for computationally expensive dynamical models. The SIR filter could use more particles at the same computational cost. However a generally applicable comparison of the filters and conclusion of this kind is not possible because it depends on the computational cost of the dynamical model $\mathcal{M}$ that integrates the state from time $k-1$ to k. In particular, one of the potential applications for which we envisage the MPF, geophysical systems, typically use extremely computationally demanding models. Thus, only a small number (< 100) of model integrations–particles – are affordable. The overall computational cost of the MPF for such a system would be similar to 3D variational assimilation (i.e. MAP estimation) for each particle.

A third set of experiments was conducted using a compartment stochastic dynamical model for cholera dynamics [10]. Fig. 8a shows the true mortality time series and the one estimated by the MPF. The experiment started with an initial

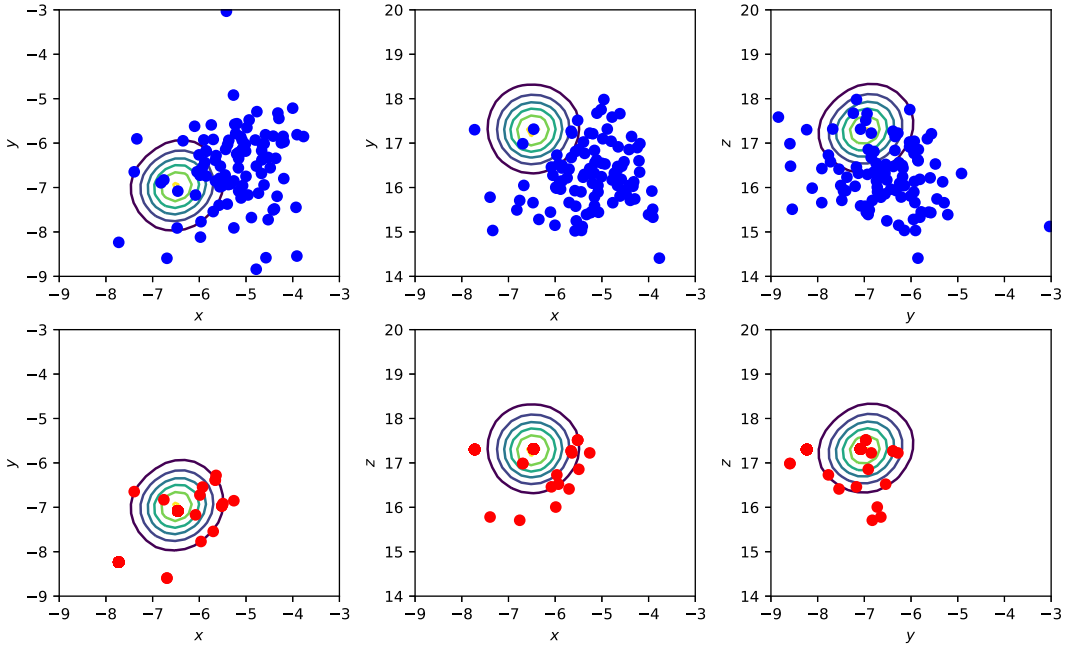

Fig. 6. As in Fig. 5 for the SIR particle filter.

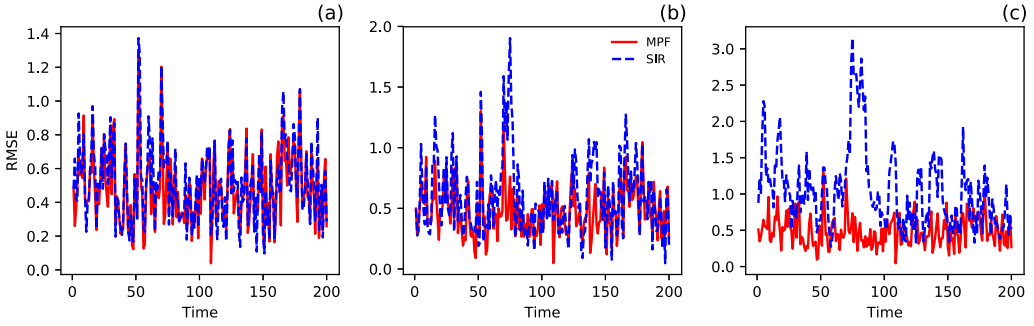

Fig. 7. RMSE for the MPF and the SIR filters as a function of the number of particles. Panels (a) 100, (b) 20, and (c) 5 particles.

mean state which underestimates the number of true susceptible individuals by 20%. A good performance is found for the MPF (total RMSE is 0.0031). The SIR filter follows the cholera outbreaks but the estimate is much noisier when the same level of noise is used as for the MPF (total RMSE is 0.0039). By decreasing the level of noise in the SIR filter, the noise diminishes but at the expense of a strong underestimation of the peaks (total RMSE is 0.0077). Overall the performance of the mapping particle filter is excellent, it can handle very well this partial observation configuration, i.e. 5 state variables and one observed variable, and the nonadditive model error.

To evaluate the performance of the MPF for a larger state-space model, an experiment using the 40-variable Lorenz-96 dynamical system was conducted. An ensemble of 20 particles was used in the MPF experiment. Scalable sampling methods are expected to produce useful results for these experiments in which $N_p < N_x$. Fig. 9a compares the resulting RMSE as a function of time given by the MPF experiment and two stochastic EnKF experiments one using 20 members (inflation factor 1.1) and one with 100 members (inflation factor 1.05). The SIR filter is not shown since it exhibits degeneracy in the Lorenz-96 system because of the high dimensional state and observation spaces. The MPF takes a longer time to converge (20 cycles), but the EnKF with the same number of members (20) shows a larger RMSE. On the other hand, the EnKF with 100 members presents rather similar RMSE to the MPF experiment with 20 particles. Furthermore, the mapping particle filter estimates are rather stable in time, which is a result of its deterministic mappings. The spread is rather stable and comparable to the RMSE in both filters (Fig. 9b).

An experiment to evaluate the performance of MPF for a partially observed Lorenz-96 system (40 variables and 20 observations) was conducted. The settings are similar to the previous experiment but now the observations are sparse. Fig. 10a shows the RMSE given by the MPF and the stochastic EnKF. The MPF outperforms the EnKF in terms of the RMSE metrics. Fig. 10b shows the spread of the particles for both filters. No localization is used in the filters. This is not by far an

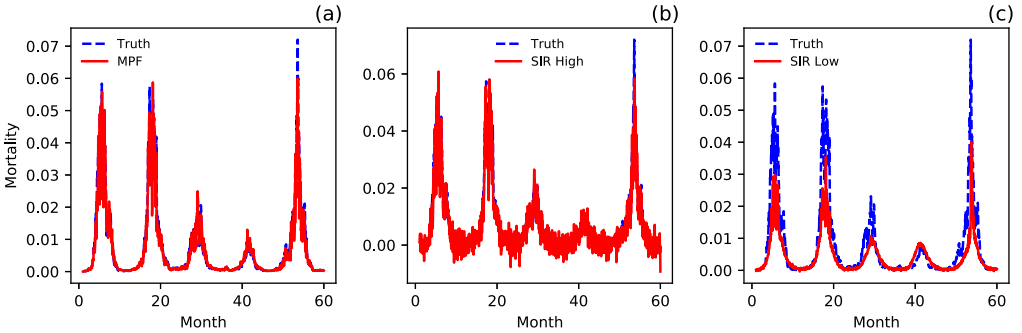

Fig. 8. Mortality time series for a model of cholera dynamics, true mortality and the one estimated with the MPF (a), with the SIR filter with high additive noise (b) and with the SIR filter with low additive noise (c).

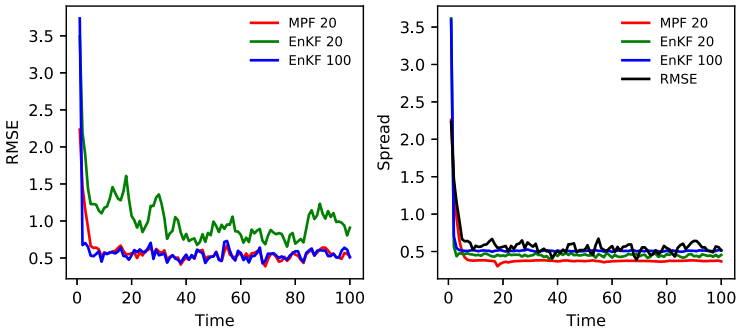

Fig. 9. RMSE (a) and spread (b) of the state as a function of the assimilation cycles produced by EnKF and MPF in the 40-variables Lorenz-96 system.

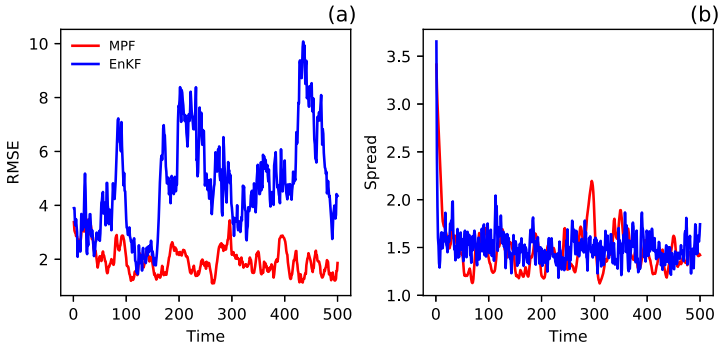

Fig. 10. RMSE (a) and spread (b) of the state as a function of the assimilation cycles produced by EnKF and MPF in the 40-variables Lorenz-96 system with 20 observations. Both filters use 20 particles.

exhaustive comparison, the only purpose is to show that the MPF can give promising results in relatively large dimensional spaces in which the SIR filter does not work. Further experiments and more extensive comparisons are required to evaluate the overall performance of MPF in median and higher-dimensional systems (> 100 state space dimensions), these go beyond the present work which is focused on the introduction and development of the technique.

The radial basis functions used as kernel require to set the bandwidth, which in this work was chosen as $\mathbf{A} = \alpha\mathbf{Q}$. The uknown parameter α depends on the number of particles and the dimensions of the problem so that it requires some tuning. The optimal parameter will be the one that produces the sample spread equal to the spread of the sequential posterior density. The bandwidth parameter α used in the Lorenz-63 experiment with 20 particles is $\alpha = 1$. For the experiment with 100 particles we use $\alpha = 0.5$. Fig. 11 shows experiments for the 40 variables Lorenz-96 system in which we vary the bandwidth from $\alpha = 2$ to $\alpha = 100$. The RMSE shows only slight changes with similar values in the three experiments (Fig. 11a). The RMSE measure for the MPF appears rather robust to the kernel bandwidth. Fig. 11b shows the spread of the experiments for different α values. A choice of $\alpha = 20$ appears to give an optimal spread of the particles for this experiment.

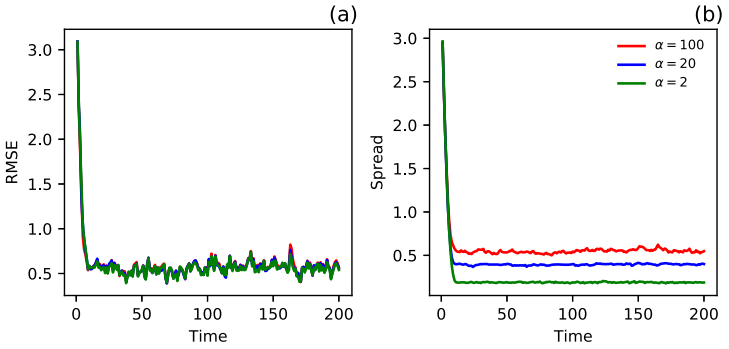

Fig. 11. (a) RMSE for the MPF in the Lorenz-96 experiments using a bandwidth of $\alpha = 2, 20, 100$. (b) The corresponding spread of the particles of the posterior density.

5. Discussion and conclusions

The proposed mapping particle filter is based on a deterministic gradient flow. It is able to keep a large effective sample size, avoiding the resampling step, even for long recursions subject to the convergence of the mapping sequences. Furthermore, it shows a robust behavior in terms of the RMSE with a consistent ensemble spread in the conducted experiments, using only a small number of particles.

In the limit of a single particle, $N_p = 1$, the gradient of the Kullback-Leibler divergence is equal to the negative of the gradient of the log-posterior density (see Eq. (18)). In that case, the optimization via gradient descent determines the mode of the posterior density. Therefore, the method for $N_p = 1$ is equivalent to three-dimensional variational data assimilation (3D-Var) with $\mathbf{Q}$ in the role of $\mathbf{B}$.

The MPF requires the adjoint of the observational operator to evaluate the gradient of the posterior density (see Eq. (32)). An extension of the MPF was proposed in [22] which represents the observational operator in the RKHS, and in this way it avoids the need of the Jacobian of the observational operator. The experiments show that this extension of the MPF retains the main non-Gaussian properties of the posterior density.

The mapping of samples in the variational mapping particle filter is produced by moving the particles where they maximize the information gain of the proposal density with respect to the sequential posterior density, via the Kullback-Leibler divergence. In these terms, the proposed mapping particle filter seeks to maximize the amount of information available in a complex high-dimensional posterior density using a stochastic optimization method and given a limited number of particles.

Acknowledgements

This work has been founded by the European Research Council via the CUNDA project number 694509 under the European Union Horizon 2020 programme.

Appendix A. Importance sampling and the mapping particle filter

The variational mapping sampling scheme is suitable to be coupled with importance sampling. The last intermediate density obtained with the variational mapping sampling scheme can be used as a proposal density of the posterior density. Suppose we denote the particles obtained in the last mapping iteration by $\mathbf{x}_k^{(1:N)} \doteq \mathbf{x}_{k,I}^{(1:N)}$. The values of the proposal and posterior densities at the particle positions are used to estimate the weights

$$\tilde{w}_k^{(j)} = \frac{p(\mathbf{y}_k|\mathbf{x}_k^{(j)})\, p(\mathbf{x}_k^{(j)}|\mathbf{y}_{1:k-1})}{q(\mathbf{x}_k^{(j)})}, \tag{A.1}$$

and then they are normalized with $w_k^{(j)} = \tilde{w}_k^{(j)} / \sum_j^{N_p} \tilde{w}_k^{(j)}$. The effective ensemble size is given by $N_{eff} = 1/\sum_j (w_k^{(j)})^2$.

Since the output of the filter is a set of weighted samples, the sequential posterior density (e.g. Eq. (30)) has to include the weights of the previous cycle,

$$p(\mathbf{x}_k^{(j)}|\mathbf{y}_{1:k}) \propto p(\mathbf{y}_k|\mathbf{x}_k^{(j)}) \sum_l w_{k-1}^{(l)}\, p(\mathbf{x}_k^{(j)}|\mathbf{x}_{k-1}^{(l)}). \tag{A.2}$$

To compute the weights between the posterior density and the proposal density –final density of the MPF–, we also need to evaluate the proposal density at the particle locations. We require the proposal density to be known (apart from

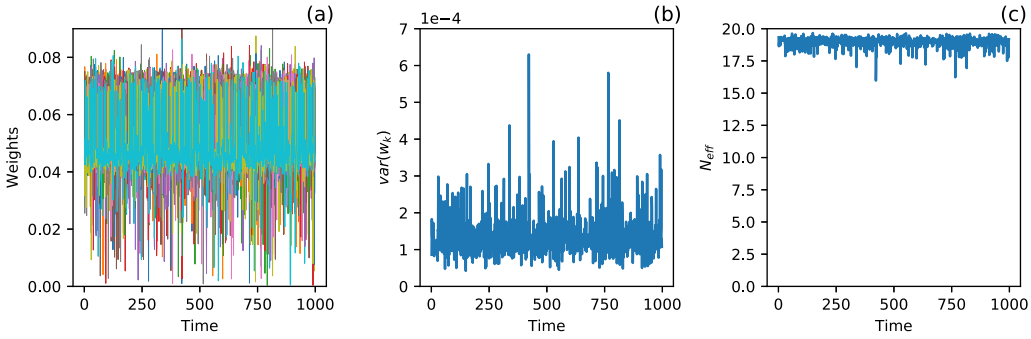

Fig. A.12. (a) Evolution of the weights of all the particles for the Lorenz-63 system in a long time sequence for $I = 50$ and without threshold. Because the weights of all the particles are plotted, the last particle plotted is the most visible one. (b) The variance of the weights as a function of time. (c) Number of effective particles.

the set of samples that we have from that density). There are two ways to obtain it. One way is to use kernel density estimation with the set of samples, $\mathbf{x}_k^{(1:N)}$, using the same kernels as in the mapping. The second option is to determine at each mapping iteration how the density is transformed with the mapping (e.g. [19]).

Before the mapping, we choose as initial intermediate density, for instance, an equally weighted Gaussian mixture centered at $\mathcal{M}(\mathbf{x}_{k-1}^{(i)})$ and with covariances $\mathbf{Q}$,

$$q(\mathbf{x}_{k,0}|\mathbf{y}_{1:k}) \propto \sum_{i=1}^{N_p} \exp\left[-\frac{1}{2}\left\|\mathbf{x}_{k,0} - \mathcal{M}(\mathbf{x}_{k-1}^{(i)})\right\|_{\mathbf{Q}}^2\right]. \tag{A.3}$$

In the following mapping iterations $i > 0$, the density changes according to the mapping T,

$$q_T(\mathbf{x}_{k,i}^{(j)}) = \frac{q(\mathbf{x}_{k,i-1}^{(j)})}{\det \nabla_x T} = \frac{q(\mathbf{x}_{k,i-1}^{(j)})}{|\mathbf{I} - \epsilon\mathbb{H}|}, \tag{A.4}$$

where $|\mathbf{I} - \epsilon\mathbb{H}|$ is the determinant Jacobian of the transformation and $\mathbb{H}$ is the Hessian of the Kullback-Leibler divergence. From Eq. (18), the Hessian is given by

$$\mathbb{H} = \frac{1}{N_p}\sum_{l=1}^{N_p}\left\{\nabla_{\mathbf{x}_{k,i-1}^{(j)}} K(\mathbf{x}_{k,i-1}^{(l)}, \mathbf{x}_{k,i-1}^{(j)})\left[\nabla_{\mathbf{x}} \log p(\mathbf{x}_{k,i-1}^{(l)})\right]^\top + \nabla_{\mathbf{x}_{k,i-1}^{(j)}} \nabla_{\mathbf{x}_{k,i-1}^{(l)}} K(\mathbf{x}_{k,i-1}^{(l)}, \mathbf{x}_{k,i-1}^{(j)})\right\}. \tag{A.5}$$

The calculation of the Hessian of the Kullback-Leibler divergence, Eq. (A.5) is computationally demanding. Although note that the second derivatives are known analytically for standard kernel functions and the symmetry of the kernel can be used to reduce the calculations. The Hessian calculation is required at each mapping iteration to evolve $q_{k,i}^{(j)}$ from the previous intermediate density $q_{k,i-1}^{(j)}$.

The weights account for the bias introduced in the filter when the set of samples obtained by the sequence of mappings is not exactly distributed according to the target density. However, it should be kept in mind that our estimate of the posterior density Eq. (A.2) is not exact, so it is unclear what the calculated weights actually mean. It would be interesting to determine the accuracy of this estimate, but that is beyond the scope of this work.

The weights are also useful as a *diagnostic* of the quality of the convergence of the optimization method. In that case, an evaluation of the weights for the current intermediate density could be conducted in each mapping iteration. This allows to check in the algorithm if the given weights give an effective sample size which is over the threshold no further mapping iterations are required. In the conducted experiments of this work, in general, the effective sample size was over 98% after 50 mapping iterations during the whole recursion. In other words, if the number of mapping iterations is large enough ≥ 50, the weights will be almost equal. In high-dimensional systems, a convergence criterion based on the module of the gradient of Kulback-Leibler divergence can be used instead of the effective sample size. This avoids the calculations of the Hessian or kernel density estimations. The convergence of the technique is evaluated through experiments with both measures in Section 3.4.

The performance of the mapping particle filter can be evaluated through the weights of the particles. As a function of the mapping iterations of the filter, the sequence of mappings is expected to start with a set of particles with most of the particles with almost zero weights and a few particles with very large weights. Note from Eq. (35) that particles with almost zero weights produce a large Kullback-Leibler divergence. Then, as the particles are pushed toward the posterior density, their weights will tend to be equally distributed for most of the particles. If the variational mappings work effectively, only a few particles should remain with low weights. In other words, the variance of the weights should

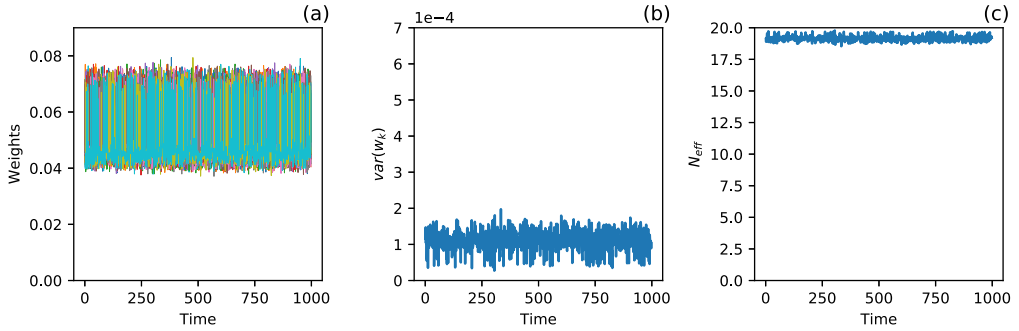

Fig. A.13. Idem Fig. A.12 for $I = 100$ and without threshold.

be small for all the cycles. Fig. A.12a shows the evolution of the weights of all the particles as a function of time for $I = 50$ and without threshold for the Lorenz-63 system. Most of the particle weights are concentrated around 0.05, in a range between 0.04 and 0.07 ($N_p = 20$ particles). There are some cycles in which one particle obtains a low weight but in the following cycle the weight of that particle is again within the equally-weighted range. Note that we did not perform any resampling. The variance of the weights is around 1.25 10^{-4} (Fig. A.12b). The number of effective particles is in general about 19 (from a total of 20 particles), only in a few cycles it descends down to 16 (Fig. A.12c). A further experiment was conducted in which the maximum number of iterations was increased to 100. In that case, all the particles remain with weights larger than 0.04 during the whole time sequence and the peaks of variance of the weights found in Fig. A.12b are not present (see Fig. A.13). The number of effective particles is in this case over 18 for all the cycles (Fig. A.13c).

References

[1] S. Angenent, S. Haker, A. Tannenbaum, Minimizing flows for the Monge–Kantorovich problem, SIAM J. Math. Anal. 35 (2003) 61–97.
[2] M.S. Arulampalam, S. Maskell, N. Gordon, T. Clapp, A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking, IEEE Trans. Signal Process. 50 (2002) 174–188.
[3] E. Atkins, M. Morzfeld, A.J. Chorin, Implicit particle methods and their connection with variational data assimilation, Mon. Weather Rev. 141 (2013) 1786–1803.
[4] P. Bunch, S. Godsill, Approximations of the optimal importance density using Gaussian particle flow importance sampling, J. Am. Stat. Assoc. 111 (2016) 748–762.
[5] O. Cappé, E. Moulines, T. Rydon, Inference in Hidden Markov Models, Springer Science+Business Media, New York, NY, 2005.
[6] Y. Cheng, S. Reich, Assimilating data into scientific models: an optimal coupling perspective, in: Nonlinear Data Assimilation, Springer, 2015, pp. 75–118.
[7] A.J. Chorin, X. Tu, Implicit sampling for particle filters, Proc. Natl. Acad. Sci. USA 106 (2009) 17249–17254.
[8] F. Daum, J. Huang, Nonlinear filters with log-homotopy, in: Signal and Data Processing of Small Targets 2007, vol. 6699, 2007, p. 669918.
[9] A. Doucet, S. Godsill, C. Andrieu, On sequential Monte Carlo sampling methods for Bayesian filtering, Stat. Comput. 10 (2000) 197–208.
[10] E.L. Ionides, C. Bretó, A.A. King, Inference for nonlinear dynamical systems, Proc. Natl. Acad. Sci. USA 103 (2006) 18438–18443.
[11] D. Kingma, J. Ba, Adam: a method for stochastic optimization, in: Int. Conf. on Learning Repres (ICLR), 2015, arXiv preprint arXiv:1412.6980.
[12] Y. LeCun, Y. Bengio, G. Hinton, Deep learning, Nature 521 (2015) 436–444.
[13] Y. Li, M. Coates, Particle filtering with invertible particle flow, IEEE Trans. Signal Process. 65 (2017) 4102–4116.
[14] Q. Liu, D. Wang, Stein variational gradient descent: a general purpose Bayesian inference algorithm, in: Advances in Neural Information Processing Systems, 2016, pp. 2378–2386.
[15] J. Liu, M. West, Combined parameter and state estimation in simulation-based filtering, in: Sequential Monte Carlo Methods in Practice, Springer, New York, 2001, pp. 197–223.
[16] R.M. Neal, Sampling from multimodal distributions using tempered transitions, Stat. Comput. 6 (1996) 353–366.
[17] Y. Marzouk, T. Moselhy, M. Parno, A. Spantini, An introduction to sampling via measure transport, in: R. Ghanem, D. Higdon, H. Owhadi (Eds.), Handbook of Uncertainty Quantification, Springer, 2017, in press, arXiv:1602.05023.
[18] R. McCann, Existence and uniqueness of monotone measure-preserving maps, Duke Math. J. 80 (1995) 309–323.
[19] T.A. Moselhy, Y.M. Marzouk, Bayesian inference with optimal maps, J. Comput. Phys. 231 (2012) 7815–7850.
[20] M. Pulido, O. Rosso, Model selection: using information measures from ordinal symbolic analysis to select model sub-grid scale parameterizations, J. Atmos. Sci. 74 (2017) 3253–3269, https://doi.org/10.1175/JAS-D-16-0340.1.
[21] M. Pulido, P. Tandeo, M. Bocquet, A. Carrassi, M. Lucini, Parameter estimation in stochastic multi-scale dynamical systems using maximum likelihood methods, Tellus 70 (2018) 1442099.
[22] M. Pulido, P.J. vanLeewen, D.J. Posselt, Kernel embedded nonlinear observational mappings in the variational mapping particle filter, in: Computational Science-ICCS 2019, in: Lecture Notes in Computer Science, vol. 11539, 2019, arXiv:1901.10426, 2019.
[23] S. Reich, A nonparametric ensemble transform method for Bayesian inference, SIAM J. Sci. Comput. 35 (2013) A2013–A2024.
[24] B. Scholkopf, A.J. Smola, Learning With Kernels: Support Vector Machines, Regularization, Optimization, and Beyond, MIT Press, 2002.
[25] E.G. Tabak, E. Vanden-Eijnden, Density estimation by dual ascent of the log-likelihood, Commun. Math. Sci. 8 (2010) 217–233.
[26] P.J. vanLeeuwen, Nonlinear data assimilation in geosciences: an extremely efficient particle filter, Q. J. R. Meteorol. Soc. 136 (2010) 1991–1999.
[27] P.J. vanLeeuwen, Nonlinear data assimilation for high-dimensional systems, in: Nonlinear Data Assimilation, Springer, 2015, pp. 1–73.
[28] P.J. vanLeeuwen, H.R. Künsch, L. Nerger, R. Potthast, S. Reich, Particle filters for high-dimensional geoscience applications: a review, Q. J. R. Meteorol. Soc. (2019), in press, https://doi.org/10.1002/qj.3551.
[29] C. Villani, Optimal Transport: Old and New, vol. 338, Springer Science & Business Media, 2008.
[30] M.D. Zeiler, ADADELTA: an adaptive learning rate method, arXiv:1212.5701 [cs.LG], 2012.
[31] M. Zhu, P.J. Van Leeuwen, J. Amezcua, Implicit equal weights particle filter, Q. J. R. Meteorol. Soc. 142 (2016) 1904–1919.