Computers & Fluids 46 (2011) 52–58

Contents lists available at ScienceDirect

Computers & Fluids

journal homepage: www.elsevier.com/locate/compfluid

Efficient nonlinear data-assimilation in geophysical fluid dynamics

Peter Jan van Leeuwen

Dept. of Meteorology, Univ. of Reading, Earley Gate, Reading RG6 6BB, UK

ARTICLE INFO

Article history:
Received 28 April 2010
Received in revised form 8 November 2010
Accepted 9 November 2010
Available online 3 December 2010

Keywords:
Inverse modelling
Data assimilation
Inverse methods
High-dimensional constraint PDE's

ABSTRACT

New ways of combining observations with numerical models are discussed in which the size of the state space can be very large, and the model can be highly nonlinear. Also the observations of the system can be related to the model variables in highly nonlinear ways, making this data-assimilation (or inverse) problem highly nonlinear. First we discuss the connection between data-assimilation and inverse problems, including regularization. We explore the choice of proposal density in a Particle Filter and show how the 'curse of dimensionality' might be beaten. In the standard Particle Filter ensembles of model runs are propagated forward in time until observations are encountered, rendering it a pure Monte-Carlo method. In large-dimensional systems this is very inefficient and very large numbers of model runs are needed to solve the data-assimilation problem realistically. In our approach we steer all model runs towards the observations resulting in a much more efficient method. By further 'ensuring almost equal weight' we avoid performing model runs that are useless in the end. Results are shown for the 40 and 1000 dimensional Lorenz 1995 model.

1. Introduction

This paper does not discuss numerical schemes, but concentrates on a 'higher order' problem if you like, in which we try to improve the model simulations of nature by including observations of nature. The emphasis is on geophysical flows, but the application of the results is not limited to these flows.

When simulating actual geophysical flows, inaccuracies in initial conditions, forcing fields and in the model equations themselves, both numerical and physical, tend to lead to differences between the simulation and the actual behavior of the flow. One way to address this is to try to incorporate the uncertainties in the simulations, e.g. in the form of probability density functions (pdf's). This gives one the possibility to express the uncertainty in the simulations. A problem is that for large-dimensional simulations in e.g. numerical weather prediction, the state space is so large, typically 10^8 variables, that no computer is large enough to store these probability density functions. So, if we want to include these uncertainties we need an efficient representation of the pdf's.

Because the geophysical flows we have in mind are highly nonlinear the model pdf can have 'any' shape. To this end, we will represent the pdf by a number of points or particles in model space. So each particle represents a full model state.

However, just representing the uncertainties is not enough, we also want to reduce them. Direct observations of the system at hand is a possibility that we will explore here. Using Bayes Theorem on information transfer [7], we can update the pdf of the model with the pdf of the observations, in a procedure called data assimilation. Up to now, the use of Bayes Theorem has been simplified by assuming linear or linearized methods like the Kalman Filter and gradient-descent methods like 4DVAR in operational geophysical problems. In this contribution we will focus on efficient ways to do the fully nonlinear data-assimilation problem in an efficient way.

The Particle Filter will be introduced and its inefficiency in high dimensional systems is high-lighted. In a Particle Filter the particles that represent the model pdf at initial time are integrated forward until the next observations become available. Then each particle is 'weighted' with its closeness to all new observations, and the probability weights of the particles are changed accordingly. So, if initially all particles had equal probability weight (i.e. the truth could equally well be represented by each particle), these weights are now changed with particles close to all new observations obtaining a high probability weight, and particles not so getting low weight. The use of these weights becomes apparent when calculating e.g. the mean of all particles. Initially the mean is just the sum of all particles divided by the total number of particles. After confrontation with the observations this becomes a weighted mean. Clearly, particles with very low weight have no statistical meaning in determining the mean, or for that matter, any moment of the pdf. So, we can just ignore them.

The reason for the inefficiency of the Particle Filter is related to the fact that the change for a model run to end up close to a number of observations is very small in high-dimensional systems. Hence only a very small number of particles gets a high weight,

E-mail address: p.j.vanleeuwen@reading.ac.uk

0045-7930/$ - see front matter © 2010 Elsevier Ltd. All rights reserved.
doi:10.1016/j.compfluid.2010.11.011

while all others can be ignored. This is referred to in the literature as the 'curse of dimensionality' [12].

Part of the solution comes from exploring the so-called proposal density [4,14]. Bayes Theorem allows one to steer the model runs towards the observations as long as we properly correct the relative weights of the particles in the whole ensemble of particles. It is shown that this is not enough to solve the full problem, and a second ingredient is to ensure that all particles have almost equal weight in the posterior ensemble by adding small terms to each particle in the last time step towards the new observations.

The next section Bayes Theorem is introduced and it is shown how present-day data-assimilation methods for geophysical flows are derived from it. It discusses the relation with inverse methods and the different philosophies behind the two approaches. Then the particle filter is introduced and the curse of dimensionality is discussed. Section 4 introduces the two new ingredients for making the Particle Filter more efficient and section 5 discusses the application to the 40 dimensional Lorenz 1995 model [9]. The paper closes with conclusions and a short discussion on what we have achieved, what not, and where we might go.

2. Bayes Theorem, data-assimilation, inverse methods and regularization

Arguably the most complete way to describe uncertainty is by means of probability densities (pdf's), and that is what we will do here. We adopt the Bayesian viewpoint and explore Bayes Theorem to update the pdf of the model by observations described by their own pdf. In fact, Bayes Theorem is just exploring the definition of conditional pdf's. We have the pdf of the model, $p(\psi)$, in which ψ is the state vector that contains all model variables, and we want to obtain the pdf of the model given the new observations d, so $p(\psi|d)$. This so-called posterior pdf is defined as:

$$p(\psi|d) = \frac{p(\psi,d)}{p(d)} = \frac{p(d|\psi)p(\psi)}{\int p(d|\psi)p(\psi)d\psi} \tag{1}$$

It states that the pdf of the model with state vector ψ given the observations d is found by the multiplication of the pdf of the observations given this model state, the so-called likelihood, and the pdf of the model before observations are taken into account. Read in this way, it tells us how to update the model pdf.

It is important to realize the full data-assimilation solution is the posterior pdf, which can be obtained by a multiplication of the known (in principle at least) pdf's. In this sense no inversion is involved, and one can view data-assimilation as a direct instead of an inverse problem. Data assimilation is just an update of information in that view. However, one can also hold the view that the observations are used to change a model pdf that was wrong initially. Since the observations are functions of the model variables we use their information to correct the model pdf. This suggests that data-assimilation is an inverse problem. This is especially apparent when considering the estimation of model parameters in a dynamical model, with observations that relate to the model state, and not to the model parameters. In that case the model has to be run first before the correct model parameters can be determined, and then the model has to be run again with the new parameter values.

Still, this author likes the first interpretation better. There is nothing wrong with the original model pdf (or model parameter pdf), it just did not include the latest information present in the observations. Data assimilation tells us that new information can be included by a 'simple' multiplication to find the model pdf with all latest information included. In my view, as soon as one realizes that the solution to the problem is a full pdf, which is our objective representation of our information on the system the directness of the problem becomes apparent, even for the parameter-estimation problem. Formulated as a Bayesian problem, the objective in parameter estimation is to obtain a new pdf of the parameters, not a new pdf of the model evolution. The model evolution pdf can be seen as resulting from having a complex observation operator working on the parameter pdf.

One of the reasons for the inverse problem terminology is that one is often only interested in the 'best' estimate. Best is usually meaning the maximum of the pdf, the highest mode. The connection with the inverse problem formulation is easy to see when one looks for the minimum of minus the logarithm of the posterior pdf:

$$J(\psi) = -log\left(\frac{p(\psi|d)}{\mu(\psi)}\right) \tag{2}$$

in which $\mu(\psi)$ is a 'non-informative prior', i.e. a very flat pdf. It has to be included because the posterior pdf is not dimensionless. Usually it is assumed that this non-informative prior does not change the position of the maximum of the posterior pdf. When we assume a Gaussian distributed model pdf and Gaussian distributed observations we find:

$$J(\psi) = \frac{1}{2}(\psi-\psi_0)^T B^{-1}(\psi-\psi_0) + \frac{1}{2}(d-H(\psi))^T R^{-1}(d-H(\psi)) \tag{3}$$

which is the familiar L2 norm costfunction or penalty function that is minimized in inverse problems. Here ψ_0 is a first guess value for the state vector (sometimes taken as zero), B is the prior error covariance (sometimes called regularization matix), R is the error covariance of the observations, and $H(\psi)$ is the operator that projects the model state vector to the observation space.

If one applies a coordinate transformation $\psi = \sigma_0 B^{1/2}\,\phi$ and defines $\widehat{R} = \sigma_d R$ we recover the standard inverse problem with L2 regularization:

$$J(\psi) = \lambda|\phi-\phi_0|_2 + (d-H(\sigma_0 B^{1/2}\phi))^T \widehat{R}^{-1}(d-H(\sigma_0 B^{1/2}\phi)) \tag{4}$$

with $\lambda = \sigma_0/\sigma_d$. In inverse problems with regularization term $\lambda|\phi-\phi_0|_2$ one tries to find the value of λ that gives the smallest value for both regularization and observation terms simultaneously. The term is introduced in the first place to remedy the ill-posedness of the minimization procedure when only the observation term is present. The problem is called ill-posed because it does not have a unique solution. This ill-posedness comes from the fact that the number of independent observations is generally smaller than the number of unknowns (otherwise the traditional least-squares solution would be sufficient). From the inverse-problem point of view this corresponds to a non-trivial null space, which has to be eliminated by modifying λ.

From a more general point of view this ill-posedness does not exist. If no prior information is present the solution is a hyperplane in the high-dimensional space in which the state vector lives. That is the best solution given the information we have. This hyperplane might be difficult to find, but that is another matter.

From a data-assimilation point of view the regularization term arises from our prior knowledge of the system, and as such λ is given. It follows from our knowledge of the system before the new observations are taken into account providing the relation between the mathematical problem and the physical (or chemical or ...) problem that one wants to solve. As such, it is not something that needs optimizing, so λ should be given. It is still possible that no unique solution exists given all our prior knowledge. That points to ill-posedness for inverse modelers, but to a hyperplane solution for a Bayesian.

Actually, to a Bayesian the inverse problem can be viewed as a Bayes problem on a higher level. The search is not for a state vector, but for the pdf of the prior covariances in the inverse problem described above. The actual prior information in this high-level

Bayesian problem is our prior knowledge on the pdf of λ, sometimes called the hyperprior. The posterior pdf becomes the posterior pdf for λ (or, more generally, the posterior pdf for the prior covariances in the inverse problem described above), and, again, not one value of λ.

Sometimes one uses an L1 regularization term instead of the L2 norm described above. From a Bayesian point of view that corresponds to an exponential or Laplacian prior. Also other forms of regularization are in use, like penalties on derivatives of the state vector. If one should ask why that form of the penalty term is used the answer will be that prior knowledge exists. From a Bayesian point of view that should be put in via a prior pdf.

3. Present-day data-assimilation methods and Bayes Theorem

Two main methods can be distinguished in the present-day methods used in e.g. numerical weather prediction. These are variational methods, of which the dominant one is the so-called 4DVAR method, and the (Ensemble) Kalman Filter. The 4DVAR method is strongly related to our discussion in the previous section [13,3]. It tries to find the maximum of the posterior pdf by minimizing the costfunction (3) using gradient-descent methods. For 4DVAR the data-assimilation problem boils down to finding the solution of a large set of coupled nonlinear partial differential equations. Using the calculus of variations the problem is rewritten as a two-point boundary value problem: the Euler–Lagrange equations. These are typically linearized first and solved iteratively (the so-called incremental 4DVAR). Due to its efficiency for present-day weather forecasting it is still the most popular method.

The method has a few problems. Unfortunately, there is no guarantee that the minimum found is indeed the global minimum. Furthermore, in concentrating on the mode the rest of the posterior pdf is ignored and it misses e.g. an error estimate. The error estimate is sometimes calculated as the inverse of the Hessian, but that is only correct for almost linear problems. For strongly nonlinear problems the inverse of the Hessian only gives the local curvature of the posterior pdf, which is not necessarily a good estimate of the spread of this pdf. For numerical weather prediction typically a few nonlinear iterations are done, the so-called outer loops, each with a few tens of linear inner loops. Due to the high costs for the high-dimensional models (some 10^8 model variables), one does not attempt to converge to an actual minimum.

It is actually not entirely clear why 4DVAR works so well for weather forecasting. But if one looks at the development of the implementation it becomes clear that the prior error covariances (the so-called background error covariance matrix, or B matrix) do not represent our knowledge of the actual errors but is used as a regularization term (M. Cullen, personal communication).

The Kalman Filter is developed for linear models. Applications for nonlinear models need extra closure models (e.g. the Extended Kalman Filter) which are ad hoc. (The problem is indeed similar to that in turbulence theories.) The Kalman Filter equations follow directly from Bayes Theorem when assuming Gaussian prior and observation pdf's, and linear measurement operators $H(\ldots)$. In that case the mean and the mode of the posterior pdf coincide, and we can find the solution directly from setting the gradient of the costfunction to zero in (3). Another approach is to write the posterior as one Gaussian pdf in ψ by 'completing the square'. Again another method is to minimize the trace of the posterior error covariance.

The result is:

$$\begin{aligned} \hat{\psi} &= \psi_0 + K(d - H(\psi_0)) \\ \widehat{B} &= (\mathbf{1} - KH^T)B \end{aligned} \tag{5}$$

in which K is the Kalman gain given by $K = BH^T(HBH^T + R)^{-1}$, and the $\hat{}$ denotes the updated value. One often sees the Kalman Filter equations presented in this way but now with H nonlinear, for instance in the LETKF formulation in numerical weather prediction (see also Ref. [16]). The author of this paper has never seen a proper derivation of those equations. It looks like an ad hoc extension of the original Kalman Filter equations.

One of the reasons why the Kalman Filter is so popular is that its error covariances are updated in the process so that we always have an estimate on the accuracy of the mean. For a linear model it is easy to derive equations for the propagation of the error covariance in time. However, when the models are nonlinear and high-dimensional the Kalman Filter is not optimal in any sense. First, no closed form for the propagation of the error covariance in time can be found, again due to a closure problem, which is now exactly equal to that of the quasi-normal approximation in turbulence theory. Furthermore, for a 10^8 dimensional system the error covariance matrix has 10^{16} entries. We might be able to store this matrix in some efficient way, but we cannot propagate it in time.

Square-root versions of the Kalman Filter have been derived, which attacks the dimensionality problem, but not the nonlinearity problem. A big step forward has been the development of the Ensemble Kalman Filter (EnKF) by Evensen [5,6], see also Burgers et al. [2]. It attacks the nonlinear evolution problem for the error covariance by sampling from the posterior pdf and propagating the samples, so the model states, forward in time with the fully nonlinear model equations. At any time the samples can be used to calculate an approximate mean and error covariance. The success of this method in high-dimensional applications using only a very small number of samples (50–100) is surprising given the limited space spanned by the ensemble. Crucial to this success is so-called localization, in which spurious correlations are eliminated by applying a cut-off radius of influence for each observation. Space limitations do not allow me to discuss this further here. Despite its success numerous problems arise in highly nonlinear systems and systems with inequality constraints, such as concentrations that have to remain non-negative, while the Gaussian does allow negative values.

This motivated some to look into fully nonlinear data-assimilation methods. Of these, Particle Filters might have great potential, with some modifications to the standard formulation. The model pdf $p_m(\psi)$ of model state ψ is represented by a set of model states called particles ψ_i, as:

$$p_m(\psi) = \frac{1}{N}\sum_{i=1}^{N}\delta(\psi - \psi_i) \tag{6}$$

This representation of the model pdf is propagated forward in time using the model equations on each particle, as:

$$\psi^n = f(\psi^{n-1}) + \beta^n \tag{7}$$

in which n is the time index, $f(\psi^{n-1})$ is the deterministic part of the model, and β^n denotes the stochastic part of the model related to inaccuracies in the model equations. (This process approximately solves the Kolmogorov equation for the evolution of the model pdf.) This part is similar to what is done in the EnKF.

When new observations become available we can just plug the particle representation in Bayes theorem to obtain the Particle Filter update:

$$p(\psi|d) = \sum_{i=1}^{N} w_i\delta(\psi - \psi_i) \tag{8}$$

in which the weights w_i are given by:

$$w_i = \frac{p(d|\psi_i)}{\sum_{j=1}^{N} p(d|\psi_j)} \tag{9}$$

It turns out that this approach is not very efficient and that all but one particle get negligible weight after a few updates with observations. A partial solution is to use resampling, in which low-weight particles are abandoned and high-weight particles are duplicated in a systematic way ([10,8,4], see [14], for an overview of particle filtering in geophysical applications, including approximations to full Particle Filters). A schematic of the method is given in Fig. 1. Unfortunately, even for low-dimensional systems large numbers of particles, so large numbers of model integrations are needed.

4. Efficient particle filtering

This section closely follows Van Leeuwen [15]. A very interesting property of particle filters that has received little attention in the geophysical community is related to the so-called proposal density. It will allow us to slightly change the model equations to ensure that all particles (model runs) end up close to the observations, ensuring that only a very small fraction of the model runs has been a waste of computer time.

The posterior expectation value of a function of the state vector $f(\psi)$ can be written using Bayes Theorem as:

$$\overline{f(\psi^n)} = \int f(\psi^n) p(\psi^n|d^n) d\psi^n = \frac{1}{A} \int f(\psi^n) p(d^n|\psi^n) p(\psi^n) d\psi^n \tag{10}$$

in which A is a normalization factor. The prior density $p(\psi^n)$ can be obtained from integration from the previous state $p(\psi^{n-1})$ as

$$p(\psi^n) = \int p(\psi^n, \psi^{n-1}) d\psi^{n-1} = \int p(\psi^n|\psi^{n-1}) p(\psi^{n-1}) d\psi^{n-1} \tag{11}$$

in which $p(\psi^n|\psi^{n-1})$ is the so-called transition density that tells us what the probability is to go from ψ^{n-1} to ψ^n in one time step. For a purely deterministic model that pdf is a delta function: when ψ^{n-1} is given it has to end up in ψ^n. However, our models contain errors that we represent by stochastic terms. The transition pdf than becomes equal to the pdf of the stochastic term β^n centered around the deterministic part of the model:

$$p(\psi^n|\psi^{n-1}) = p(\beta^n) \tag{12}$$

So we know how to calculate this transition density when the pdf of the random forcing is given. Let us now use (11) into our expression for the expected value to find:

$$\overline{f(\psi^n)} = \frac{1}{A} \int f(\psi^n) p(d^n|\psi^n) p(\psi^n|\psi^{n-1}) p(\psi^{n-1}) d\psi^n d\psi^{n-1} \tag{13}$$

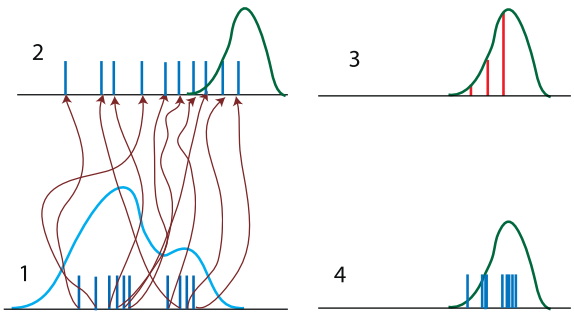

Fig. 1. The standard particle filter. The prior blue pdf is sampled by a number of particles (10 in this case), indicated by the dark blue vertical bars. These particles are all propagated forward in time using the full nonlinear equations, indicated by the brown lines. When observations are present we see the prior particles as blue vertical bars again. The pdf of the observations is given by the green curve. In this example a large percentage of particles ends up far from the observations and gets negligible weight. The new weights are indicated by the red bars. After the resampling step we ensure that we can continue the model integrations with 10 particles again. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

At the heart of this paper is the freedom in the transition density. We can rewrite (13) as

$$\overline{f(\psi^n)} = \frac{1}{A} \int f(\psi^n) p(d^n|\psi^n) \times \frac{p(\psi^n|\psi^{n-1})}{q(\psi^n|\psi^{n-1}, d^n)} q(\psi^n|\psi^{n-1}, d^n) p(\psi^{n-1}) d\psi^n d\psi^{n-1} \tag{14}$$

in which we just multiplied and divided by the so-called proposal transition density q. To make this a valid expression we have to make sure that $q(\psi^n|\psi^{n-1}d^n)$ is not zero where $p(\psi^n|\psi^{n-1}) \neq 0$, which does not pose any practical problems. The important observation is that we can make this proposal density dependent on the future observations d^n. A simple way to do this is to choose:

$$\psi^n = f(\psi^{n-1}) + \hat{\beta}^n + K^n(d^n - H(\psi^{n-1})) \tag{15}$$

in which K^n is a matrix that can be time dependent, but many other possibilities are open. Note that we have chosen the *proposal density* $q(\psi^n|\psi^{n-1}d^n)$ as the pdf of $\hat{\beta}^n$ centered on the full deterministic part of the equation above, so on $f(\psi^{n-1}) + K^n(d^n - H(\psi^{n-1}))$. Note also that we could use the same or another stochastic part of the equation, denoted by $\hat{\beta}^n$ in the equation above. Most important, however, is the new 'nudging' or relaxation term $K^n(d^n - H(\psi^{n-1}))$. This last term will 'pull' the particle towards the observations. By choosing K^n wisely one can assure that all particles end up relatively close to the observations. As one of the reviewers pointed out, there is no guarantee when, e.g. H is highly nonlinear. Note however that we have an enormous freedom here, we can choose 'any' term that forces the model towards the future observations.

If we now use a particle representation of the pdf at time $n-1$ we find that the integral in (13) is again a weighted sum over the particles, but now with weights:

$$w_i = \frac{1}{A} p(d^n|\psi_i^n) \frac{p(\psi_i^n|\psi_i^{n-1})}{q(\psi_i^n|\psi_i^{n-1}, d^n)} \tag{16}$$

To evaluate these weights we have to make choices for the pdf of the new stochastic forcing $\hat{\beta}^n$ and the matrix K^n. Suppose that the actual model error is Gaussian with mean zero and covariance Q, and suppose that we take the stochastic part of the proposal transition density from a Gaussian with zero mean and error covariance $\widehat{Q}$. Also, assume that the observations are Gaussian distributed with mean zero and covariance R. The weights can now be written as:

$$w_i \propto \exp\left[-\frac{1}{2}(\psi^n - f(\psi^{n-1}))Q^{-1}(\psi^n - f(\psi^{n-1})) + \frac{1}{2}\hat{\beta}^n \widehat{Q}^{-1} \hat{\beta}^n - \frac{1}{2}(d - H(\psi^n))R^{-1}(d - H(\psi^n))\right] \tag{17}$$

where we can recognize the contributions from the different terms in the expression for the new weights. In geophysics we usually have observations only every L time steps, where L can easily be 100 or more. In fact, only when several time steps are performed between observations the nudging-like term can do its work. In that case the weights become simply:

$$w_i \propto \exp\left\{\sum_{j=1}^{L}\left[-\frac{1}{2}\left(\psi^j - f(\psi^{j-1})\right)Q^{-1}\left(\psi^j - f(\psi^{j-1})\right) + \frac{1}{2}\hat{\beta}^j \widehat{Q}^{-1} \hat{\beta}^j\right] - \frac{1}{2}(d - H(\psi^n))R^{-1}(d - H(\psi^n))\right\} \tag{18}$$

The way we use this expression is as follows. We integrate the new model Eq. (15). This allows us to find ψ_i^n from ψ_i^{n-1} for each particle i. These state vectors are then used in the expression for the weights above to find the new weights of the particles when we arrive at the observations. This is followed by a resampling step, and the same

process is repeated. Fig. 2 shows how this particle filter with as proposal density a 'nudging' term works. The particles are 'drawn towards the observations', and all particles have a comparable weight (red bars). The improved efficiency compared to the standard particle filter depicted in Fig. 1 is clearly visible. The main difference with Fig. 1 is that the particles end up much closer to the observations in stage 2, so that the statistical representation of the posterior pdf is much better than before due to the fact that none of the particles is ignored.

The idea presented above is a major advantage in particle filtering for geoscience applications. The reason why it has not been explored in the particle filter community in statistics before is that the models used in the geosciences usually need a substantial number of model steps to propagate the model forward to the next observation set. Only in such a situation can the 'nudging term' be effective. Instead of running the model randomly forward in time, we force it towards the observations. The error that we make is completely compensated for by adjusting the relative weights of the particles. We note that there is an enormous freedom in choosing the proposal density, i.e. the 'nudging' part, which can be explored fully in the future to find more efficient schemes.

When a large number of observations is present the weights still tend to differ considerably, and filter divergence is still possible. Hence the problem is that the weights of the particles vary too much. We propose here to attack that problem directly. We can make all weights almost equal in the last step towards the observations by changing the proposal density in this last step. A way to do this is as follows. Assuming Gaussian errors in the model equations for the target transition densities $p(\psi_i^n|\psi_i^{n-1})$ and ignoring the proposal contribution for the moment, the weights can be written as:

$$w_i \propto w_i^{rest} \exp\left[-\frac{1}{2}(\psi^n - f(\psi^{n-1}))Q^{-1}(\psi^n - f(\psi^{n-1})) - \frac{1}{2}(d - H(\psi^n))R^{-1}(d - H(\psi^n))\right] \tag{19}$$

in which w_i^{rest} denotes the weights due to all time steps up to the last. We can now force the last time step of the model such that the weights are equal. The weights are the same for each particle i when $-\log w_i$ are constant, equal to C let's say, so

$$-\log w_i^{rest} + \frac{1}{2}(\psi^n - f(\psi^{n-1}))Q^{-1}(\psi^n - f(\psi^{n-1})) + \frac{1}{2}(d - H(\psi^n))R^{-1}(d - H(\psi^n)) = C_i = C \tag{20}$$

If the observation operator H is linear this is a quadratic equation for the new model states ψ_i^n with, in a space with dimension larger than one, an infinite number of solutions. To proceed we first

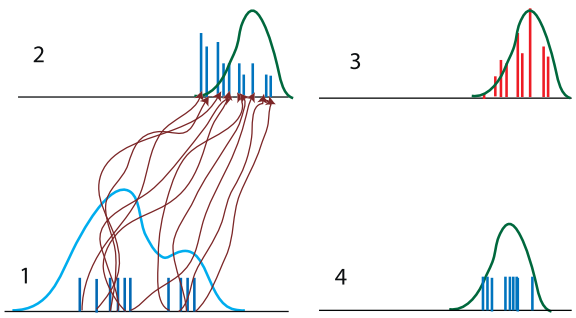

Fig. 2. The new particle filter. Same as Fig. 1, but now the particles are drawn towards the observation using the proposal density. Note that much more particles end up close to the observations in stage 2, resulting in a much better resolved posterior density in stages 3 and 4. Also note the different weights of the particles in stages 2 and 3 due to the proposal density.

calculate the minimum theoretical value of C_i for each member i, as:

$$C_i = -\log w_i^{rest} + \frac{1}{2}(d - H(f(\psi^{n-1})))(HQH^T + R)^{-1}(d - H(f(\psi^{n-1}))) \tag{21}$$

That this form comes up is easy to see when one realizes that the second term is just the value of the costfunction at the minimum (see e.g. [1]). This is the lowest value for C_i for each member. To make all C_i's equal they have to be equal to the largest C_i, so $C = max_i(C_i)$. However, we do not want all weights equal to that of the worst performing particle. (Note that even if the C_i's are similar, the weights are proportional to the exponent of them, and can still vary significantly.) In the application described below we have chosen C such that 80% of the particles can achieve that weight. The last 20% of the particles is too far from the observations to take into account. These numbers are a compromise between being close to all observations and keeping enough particles in the ensemble. With this choice, we typically keep 80% of the particles in the ensemble, while 20% will have very low weight, and will re-enter only through resampling later on. It is good to realize that other choices might lead to better overall performance of the filter. We leave that for future research. Still, we are left with a quadratic equation (if H is linear) in the state at time n for each particle, again with an infinite number of solutions. One can imagine several ways to choose one of these solutions. In this paper we simply assume

$$\psi_i^n = f(\psi_i^{n-1}) + \alpha_i K(d - H(f(\psi_i^{n-1}))) \tag{22}$$

in which $K = Q\,H^T(HQH^T + R)^{-1}$ and α_i is a scalar. So we reduce the problem to a quadratic equation in a scalar, which is easily solved as

$$\alpha = 1 - \sqrt{1 - b_i/a_i} \tag{23}$$

in which $a_i = 0.5x_i^T R^{-1} HKx$ and $b_i = 0.5x_i^T R^{-1} x_i - C - \log w_i^{rest}$. Here $x = d - H(f(\psi_i^{n-1}))$ (see Appendix A).

As mentioned before, from Eq. (16) we observe that taking the proposal deterministically would lead to division by zero since the proposal would just be a delta function centered around the deterministic value. To avoid that we introduce an extra random step from a pdf with small amplitude to make only small changes to the particles. In our example with the Lorentz-95 model we used a Gaussian distribution with a width of $\gamma\sigma$, in which σ is the standard deviation of the model error and γ is a small dimensionless number. We calculate the new weights using the new ψ_i^n as before, and divide by the new Gaussian proposal density.

$$\exp\left[-\frac{1}{2}\left(\psi_i^n - \hat{\psi}_i^n\right)\widehat{Q}^{-1}\left(\psi_i^n - \hat{\psi}_i^n\right)\right] \tag{24}$$

in which $\widehat{Q} = \gamma^2 Q$, with γ small, e.g. 10^{-5}, and $\hat{\psi}_i^n$ is the particle after the equal weight scheme. A final step now is a resampling to ensure that all particles have equal weight again.

Finally, it is stressed that by construction the particles are independent, and as such the particles form a random sample from the posterior pdf.

5. Application to the Lorenz-95 model

A challenging example for a particle filter is the 40-variable Lorenz 1995 model [9], which for the settings given below typically needs tens of thousands of particles [11]. The model equations are given by:

$$\frac{dx_j}{dt} = (x_{j+1} - x_{j-2})x_{j-1} - x_j + F \tag{25}$$

using $dt = 0.01$, and $F = 8$, with 40 grid points. The size of F ensures the model operates in the chaotic regime. The model was initialized

by choosing $F = 8.01$ at grid point 20, and running the model for 2000 time steps. The end point of that run was used as the initial condition for the data-assimilation experiment. In the application of the new particle filter we chose $K = 1$ in the nudging term (except for the last time step before the new observations, where the 'almost equal weight' scheme was used, as explained above), multiplied by a linear function that is zero half way the two updates and growing to one at the new observation time. The random forcing was multiplied by one minus that function. This allows the ensemble to spread out due to the random forcing initially, and pulling harder and harder towards the new observation the closer to the new update time. It is stressed again that an enormous freedom exists in choosing the form of this nudging term, or, more generally, the proposal density. Whatever we do is always compensated for by using the correct corresponding relative weights from (16).

The truth was generated by solving the stochastic model with the above parameters, with observations every other grid point, every 10 time steps. The observation error was $\sigma_{obs} = 1$, the initial condition standard deviation was $\sigma_{initial} = 2$, and the model error standard deviation was chosen as $\sigma_{model} = 0.5$.

Fig. 3 shows what the new particle filter generates: a swarm of particles that follows the observations smoothly in time. The red crosses denote the observations, with the red bars indicating their standard errors. Fig. 4 shows a similar plot but now for an unobserved variable. Also here the swarm of particles closely follows the truth.

The problem discussed above is already a nonlinear one. To test the method in an even more nonlinear setting we performed the same experiment using 50 time steps between observations. To the knowledge of this author, this experiment has not been described before. Fig. 5 shows the results with again only 20 particles. The results show that also in this case the new particle filter works satisfactorily.

Finally, an experiment is performed to test the scalability of the method. To this end a 1000-dimensional Lorenz-95 model was constructed, and the system is again observed every other grid point, so 500 observations, every 10 time steps of the model. This is a very hard problem, but, as can be seen from Fig. 6, the particles

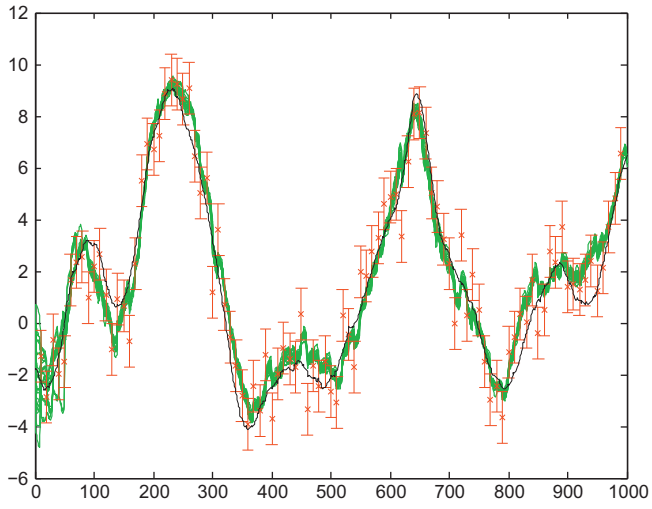

Fig. 3. The new particle filter with almost equal weights for the Lorenz-95 model. The chaotic 40 dimensional Lorenz-95 model in which every other model variable is observed every 10 time steps. The black line is the true solution, the red crosses observations of this truth, and the green lines depict the evolution of the particles in time. Note that the particles follow the truth remarkably well, using only 20 particles. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

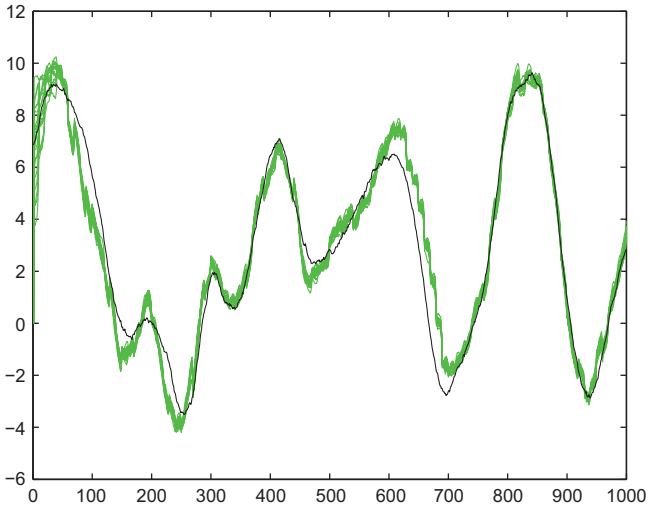

Fig. 4. Similar to Fig. 3, but now for an unobserved variable.

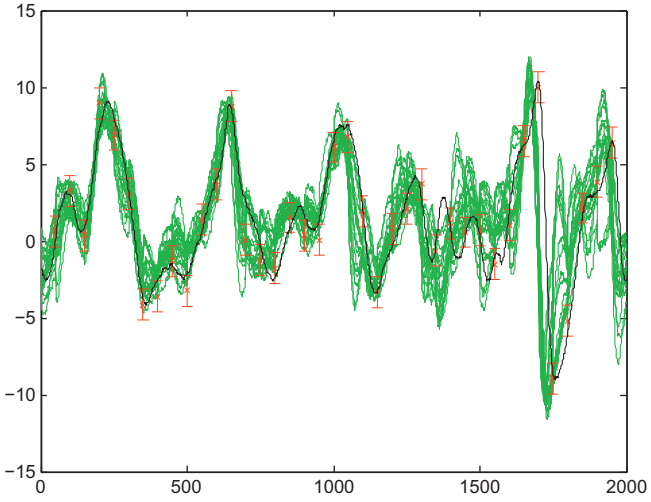

Fig. 5. Similar to Fig. 3, but now for observations every 50 time steps. Even in this case the nudging is working quite effectively.

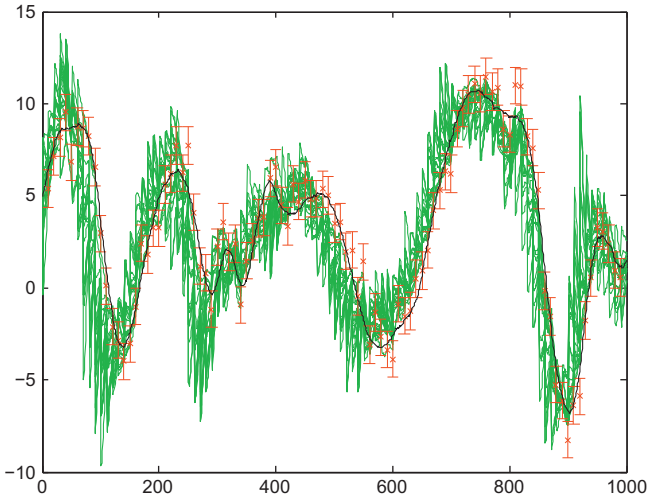

Fig. 6. Similar to Fig. 3, but now for the 1000-dimensional Lorenz-95 model. The model is capable of following the truth quite well, using only 20 particles.

are able to rack the truth remarkably well. To obtain this result we increased the nudging strength by a factor 2, but kept all other system variables the same as in the first Lorenz-95 experiment. It is possible that smoother solutions might be possible with other choices for the proposal density; that was not investigated any further here.

6. Conclusions and discussion

We have discussed the relation betwen inverse problems and data assimilation and shown how they are connected. The two main present-day data-assimilation methods for geophysical flows have been discussed and their linearity assumptions have been high-lighted. A new data-assimilation method is introduced that is fully nonlinear and potentially has enormous impact on large-dimensional applications. We presented an application to the complex 40-dimensional Lorenz 1995 model, were we show that the method needs of the order of 20 particles, showing that the method is very efficient indeed. One might argue that the 10 time steps between different observation sets were not enough for the model to develop full nonlinearity. Fig. 5 shows that the method also produces good results with 50 time steps between observation sets. So the simple nudging proposed here does not hinder application of the method to highly nonlinear applications. And indeed, when the nudging is not appropriate other more complicated schemes can be envisaged, for instance a complete 4DVar on each particle (although that would be quite expensive).

An application to the 1000 dimensional Lorenz 95 model with 20 particles stresses the perfect scaling of the new method.

The freedom in proposal density to ensure almost equal weights for the particles allows for the development of more efficient schemes than the nudging scheme presented here.

One may question what the usefulness is of representing a pdf in a say 10^8 dimensional space with only a few tens or perhaps hundreds of particles. Experience with Ensemble Kalman Filters on this kind of systems shows that useful information is present in these ensembles. The main difference is that we want to include non-Gaussian features too, and the potential to do that is still an open question.

Acknowledgement

The author was sponsored by the National Centre for Earth Observation (NCEO), funded by the National Environmental Research Council (NERC).

Appendix A. Derivation of Eq. (23)

In this appendix we derive Eq. (23). If we plug expression (22) in (20) we find:

$$-\log w_i^{rest}+\frac{1}{2}\alpha^2 x^T K^T Q^{-1} K x+\frac{1}{2}(x-\alpha H K x)^T R^{-1}(x-\alpha H K x)=C \tag{A.1}$$

in which $x=d-H(f(\psi_i^{n-1}))$. Separating equal powers of α gives:

$$\alpha^2\left[\frac{1}{2}x^T K^T Q^{-1} K x+x^T K^T H^T R^{-1} H K x\right]+\alpha\left[-\frac{1}{2}x^T R^{-1} H K x+x^t K^T H^T R^{-1} x\right]+\frac{1}{2}x^T R^{-1} x-C-\log w_i^{rest}=0 \tag{A.2}$$

Using the expression for $K = QH^T(HQH^T + R)^{-1}$ in the factor for α^2 we find

$$\begin{aligned}&\frac{1}{2}x^T(HQH^T+R)^{-1}HQQ^{-1}Kx+x^T(HQH^T+R)^{-1}HTQH^TR^{-1}HKx\\&=\frac{1}{2}x^T\left[(HQH^T+R)^{-1}HK+(HQH^T+R)^{-1}HQH^TR^{-1}HK\right]x\\&=\frac{1}{2}x^T\left[(HQH^T+R)^{-1}RR^{-1}HK+(HQH^T+R)^{-1}HQH^TR^{-1}HK\right]x\\&=\frac{1}{2}x^T\left[(HQH^T+R)^{-1}(R+HQH^T)R^{-1}HK\right]x=\frac{1}{2}x^TR^{-1}HKx\end{aligned} \tag{A.3}$$

Similarly, the factor corresponding to α becomes:

$$\begin{aligned}&-\frac{1}{2}x^TR^{-1}HKx+x^tK^TH^TR^{-1}x\\&=-\frac{1}{2}x^T\left[R^{-1}HK+K^TH^TR^{-1}\right]x\\&=-\frac{1}{2}x^T\left[R^{-1}HK+(HQH^T+R)^{-1}HQH^TR^{-1}\right]x\\&=-\frac{1}{2}x^T\left[R^{-1}HK+(HQH^T+R)^{-1}(HQH^T+R-R)R^{-1}\right]x\\&=-\frac{1}{2}x^T\left[R^{-1}HK+R^{-1}-(HQH^T+R)^{-1}\right]x\\&=-\frac{1}{2}x^T\left[R^{-1}HK+R^{-1}(HQH^T+R)(HQH^T+R)^{-1}-(HQH^T+R)^{-1}\right]x\\&=-\frac{1}{2}x^T\left[R^{-1}HK+R^{-1}(HQH^T+1)(HQH^T+R)^{-1}-(HQH^T+R)^{-1}\right]x\\&=-x^TR^{-1}HKx\end{aligned}$$

So we find:

$$\frac{1}{2}x^TR^{-1}HKx\alpha^2-x^TR^{-1}HKx\alpha+\frac{1}{2}x^TR^{-1}x-C-\log w_i^{rest}=0 \tag{A.5}$$

with solution Eq. (23).

References

[1] Bennett A. Inverse methods in physical oceanography. Cambridge: Cambridge University Press; 1992.
[2] Burgers G, Van Leeuwen PJ, Evensen G. Analysis scheme in the ensemble Kalman filter. Monthly Weather Rev 1998;126:1719–24.
[3] Courtier P. Dual formulation of four-dimensional variational assimilation. Q J Roy Meteor Soc 1997;123(B):2449–61.
[4] Doucet A, De Freitas N, Gordon N. Sequential Monte-Carlo methods in practice. Berlin: Springer; 2001.
[5] Evensen G. Sequential data assimilation with a nonlinear quasi-geostrophic model using Monte-Carlo methods to forecast error statistics. J Geophys Res 1994;99:10143–62.
[6] Evensen G. Data assimilation: the ensemble Kalman filter. Berlin: Springer; 2006.
[7] Jaynes ET. Probability theory the logic of science. Cambridge: Cambridge University Press; 2003.
[8] Gordon NJ, Salmond DJ, Smith AFM. Novel approach to nonlinear/non-Gaussian Bayesian state estimation. IEE Proceedings – F 1993;140:107–13.
[9] Lorenz, E.N., 1995. Predictability: A problem partly solved, Proc. Sem. Predictability, Vol 1 ECMWF, Reading, UK, 1-18.
[10] Metropolis N, Ulam S. The Monte-Carlo method. J Am Stat Assoc 1944;44:335–41.
[11] Nakano S, Ueno G, Higuchi T. Merging particle filter for sequential data assimilation. Nonlinear Process Geophys 2007;14:395–408.
[12] Snyder C, Bengtsson T, Bickel P, Anderson J. Obstacles to high-dimensional particle filtering. Monthly Weather Rev 2008;136:4629–40.
[13] Talagrand O, Courtier P. Variational assimilation of meteorological observations with the adjoint vorticity equation I: theory. Q J Roy Meteor Soc 1987;113:1311–28.
[14] Van Leeuwen PJ. Particle filtering in geosciences. Monthly Weather Rev 2009;137:4089–114.
[15] Van Leeuwen PJ. Nonlinear data assimilation in geosciences: an extremely efficient particle filter, Q J Roy Meteor Soc; in press. Available from: http://www.met.reading.ac.uk/~xv901096/research/publications.html.
[16] Jazwinski AH. Stochastic processes and filtering theory. London: Academic Press; 1970.