University of Reading

Nonlinear data assimilation in geosciences: an extremely efficient particle filter

Article

Published Version

van Leeuwen, P. J. (2010) Nonlinear data assimilation in geosciences: an extremely efficient particle filter. Quarterly Journal of the Royal Meteorological Society, 136 Part B (653). pp. 1991-1999. ISSN 1477-870X doi: 10.1002/qj.699 Available at https://centaur.reading.ac.uk/24126/

It is advisable to refer to the publisher's version if you intend to cite from the work. See Guidance on citing.

To link to this article DOI: http://dx.doi.org/10.1002/qj.699

Publisher: Royal Meteorological Society

All outputs in CentAUR are protected by Intellectual Property Rights law, including copyright law. Copyright and IPR is retained by the creators or other copyright holders. Terms and conditions for use of this material are defined in the End User Agreement.

www.reading.ac.uk/centaur

CentAUR

University of Reading

Central Archive at the University of Reading

Reading's research outputs online

RMetS
Royal Meteorological Society

Nonlinear data assimilation in geosciences: an extremely efficient particle filter

P. J. van Leeuwen*
Department of Meteorology, University of Reading, Reading, UK
*Correspondence to: P. J. van Leeuwen, Department of Meteorology, University of Reading, Earley Gate, PO Box 243, Reading RG6 6BB, UK. E-mail: p.j.vanleeuwen@reading.ac.uk

Almost all research fields in geosciences use numerical models and observations and combine these using data-assimilation techniques. With ever-increasing resolution and complexity, the numerical models tend to be highly nonlinear and also observations become more complicated and their relation to the models more nonlinear. Standard data-assimilation techniques like (ensemble) Kalman filters and variational methods like 4D-Var rely on linearizations and are likely to fail in one way or another. Nonlinear data-assimilation techniques are available, but are only efficient for small-dimensional problems, hampered by the so-called 'curse of dimensionality'. Here we present a fully nonlinear particle filter that can be applied to higher dimensional problems by exploiting the freedom of the proposal density inherent in particle filtering. The method is illustrated for the three-dimensional Lorenz model using three particles and the much more complex 40-dimensional Lorenz model using 20 particles. By also applying the method to the 1000-dimensional Lorenz model, again using only 20 particles, we demonstrate the strong scale-invariance of the method, leading to the optimistic conjecture that the method is applicable to realistic geophysical problems. Copyright © 2010 Royal Meteorological Society

Key Words: data assimilation; particle filtering

Received 26 February 2010; Revised 15 July 2010; Accepted 16 August 2010; Published online in Wiley Online Library

Citation: van Leeuwen PJ. 2010. Nonlinear data assimilation in geosciences: an extremely efficient particle filter. *Q. J. R. Meteorol. Soc.* **136**: 1991–1999. DOI:10.1002/qj.699

1. Introduction

The solution to the full nonlinear data-assimilation problem is well known (Jazwinki, 1970; Van Leeuwen and Evensen, 1996) and based on Bayes' theorem. That theorem tells us how to update the probability density (pdf) of the model, the so-called prior pdf, with new observations to obtain the so-called posterior pdf, which is the full solution to the data-assimilation problem. Because these probability densities can be far from standard densities like the Gaussian, their representation on a computer is problematic, especially for large-dimensional systems. Hence approximations have to be made, and present-day data-assimilation methods in high-dimensional systems are all based on linearizations.

Examples are the ensemble Kalman filter (EnKF) and its variants (Evensen, 1994, 2006; Burgers *et al.*, 1998), in which the evolution of the system between observations is fully nonlinear. However, when confronted with observations the prior pdf and the observation pdf are assumed to be Gaussian, so the analysis is linear.

Other extremely popular methods search for the maximum of the posterior pdf by assuming Gaussian-distributed observations, model initial conditions and model errors, and minimizing a so-called cost function, which is the negative of the logarithm of the posterior pdf. Examples are 4D-Var (Talagrant and Courtier, 1987), the representer method (Bennett, 1992) and PSAS (Courtier, 1997). Iterative methods are used to find this minimum, but no guarantee exists to ensure convergence to the global minimum.

Copyright © 2010 Royal Meteorological Society

Furthermore, these methods search for the maximum of the pdf and not the pdf itself, e.g. they provide no error estimate. Several methods have been developed to bring more nonlinearity into the methods mentioned above (e.g. combining 4D-Var and the EnKF (Zhang *et al.*, 2009) and Gaussian mixture models (Anderson and Anderson, 1999; Bengtson *et al.*, 2003)), but the extensions are to some extent ad hoc and do not solve the full nonlinear problem.

Particle filters are fully nonlinear in both model evolution and analysis steps (Metropolis and Ulam, 1944; Gordon *et al.*, 1993; Doucet *et al.*, 2001). They have a fundamental problem, the so-called 'curse of dimensionality', which is related to the fact that it is very unlikely that a swarm of model runs, called particles, shooting at random through state space will end up close to a large set of observations in a large-dimensional system (Snyder *et al.*, 2008). The result is that the majority of the particles will end up far away from the observations and have no statistical significance for the estimate of the posterior pdf.

More complicated particle filters have been proposed that do have potential, but little experience in geoscience applications exists (Van Leeuwen, 2009). Here the so-called proposal density is explored, which allows the particles to know where the observations are, and simple choices lead to extremely encouraging results. In this article we demonstrate this for the highly nonlinear three-dimensional Lorenz (1963) model and the 40-dimensional Lorenz (1995) models. Using traditional particle filters, hundreds to tens of thousands of model runs are needed for these models, while only of the order of 20 particles are used here. We also managed to show that the method works satisfactorily in a 1000-dimensional Lorenz (1995) model using only 20 particles, showing extremely promising scaling behaviour. This shows that the 'curse of dimensionality' may have a cure.

In Chorin and Tu (2009) a procedure is depicted that is similar to our almost equal-weight scheme. That article concentrates on an application, and the scheme is not easy to extract. An article that has full details is in preparation (Xuemin Tu, private communication).

2. Bayes theorem

Data assimilation describes the flow of information from observations of the real system at hand (i.e. the atmosphere) to the numerical model of that system. This model is based on previous experience, and is available in terms of equations that describe how the system evolves with time. The most general form in which information about a system can be represented is through a probability density function (pdf). A probability density function describes what the probability of a certain event is, compared with all other possible events in a system.

We can formulate the data-assimilation problem as trying to find the new so-called posterior probability density of the model of a system when new observations are incorporated, i.e. the pdf of the model *given* the new observations. Obviously, this is related to the pdf of the model before the new observations are taken into account, and the pdf of the observations. This can be formalized in Bayes' theorem. It is based on conditional probability densities, and given by

$$p_m(\psi|d) = \frac{p_d(d|\psi)p_m(\psi)}{\int p_d(d|\psi)p_m(\psi)\mathrm{d}\psi}. \tag{1}$$

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. The denominator is just a normalization to ensure that the probability density integrates to 1. Actually, the likelihood does not have to be a pdf, but in the following we assume it is for ease of presentation. It is important to realize that data assimilation in its purest form is a multiplication problem and not an inverse problem. When applying Bayes theorem, the probability density of the observations p_d is assumed to be known from standard calibration procedures. The difficulty in applying Bayes theorem is the probability density of the model. In large-scale geophysical applications it is a density over a million (or more) dimensional space, which is impossible to store, let alone calculate the evolution of. Even though the actual dimension of the subspace in which the model tends to reside can be much smaller, very little is known regarding this actual dimension, but its size is still considered to be substantial.

3. Particle filtering

The idea used in particle filtering is to try to represent the model pdf by a number of random draws, called ensemble members, or particles. The model is represented by a sum of delta functions positioned at the model states chosen as the particles:

$$p(\psi) = \frac{1}{N}\sum_{i=1}^{N}\delta(\psi - \psi_i). \tag{2}$$

The expected value of any function of the model state $f(\psi)$ can be approximated as

$$\overline{f(\psi)} = \int f(\psi)p(\psi)\,\mathrm{d}\psi \approx I_N(f) = \frac{1}{N}\sum_i f(\psi_i). \tag{3}$$

Common examples for $f(\psi)$ are ψ itself, giving the mean of the pdf, and the squared deviation from the mean, giving the covariance.

The technique is depicted in Figure 1. After (or during) a previous data-assimilation step a new ensemble of particles is generated. This is stage 1 in Figure 1, in which the length of the bar representing each particle gives its relative importance or weight in the ensemble of particles. Initially, all particles have equal weight. Each particle (model state) is propagated forward in time with the full nonlinear model. This part of the data assimilation represents the forward evolution of the model pdf. Formally the evolution equation of this pdf is given by the Kolmogorov equation (Jazwinsky, 1970). This equation is solved approximately by solving an ensemble of stochastic partial differential equations. The stochastic terms in these equations represent unknown external and internal terms (or factors) in the model equations. Unknown terms in external forcing and in the model equations are incorporated by adding random numbers, drawn from a known error density, to the deterministic model equations:

$$\psi^n = f(\psi^{n-1}) + \beta^n, \tag{4}$$

in which $f(..)$ denotes the deterministic part of the model, β^n is the stochastic part and n is the time index. (It is

Copyright © 2010 Royal Meteorological Society

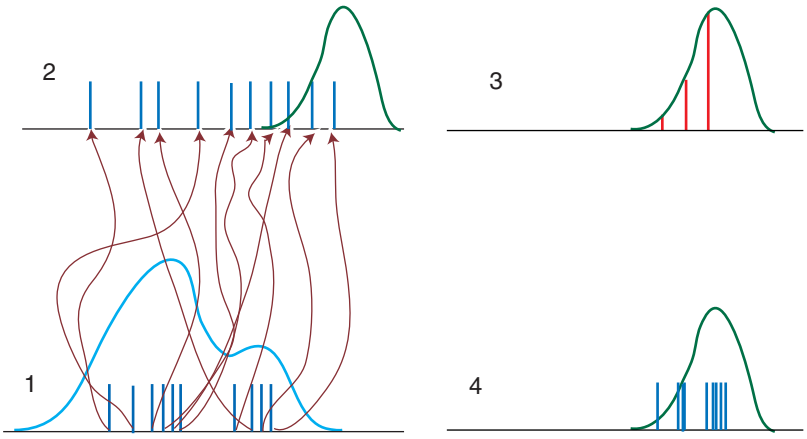

Figure 1. The standard particle filter. The prior (blue in the online article) pdf is sampled by a number of particles (10 in this case), indicated by the vertical bars (dark blue in the online article). These particles are all propagated forward in time using the full nonlinear equations, indicated by the lines (brown in the online article). When observations are present we see the prior particles as vertical bars (blue in the online article) again. The pdf of the observations is given by the curve (green in the online article). In this example a large percentage of particles ends up far from the observations and has negligible weight. The new weights are indicated by the bars (red in the online article). After the resampling step we ensure that we can continue the model integrations with 10 particles again. This figure is available in colour online at wileyonlinelibrary.com/journal/qj

also possible to multiply parts of the model equations by unknown factors, sometimes called multiplicative errors. The latter approach is usually related to unknown model parameters. We concentrate on additive random forcing here.) All particle methods have this forward propagation in common, and differ mainly in the analysis step, i.e. in the way in which model and observations are combined.

At stage 2 in Figure 1, the particles arrive at the new observations. At this point they still have equal weight. Using the particle representation in Bayes' theorem we obtain

$$p(\psi|d) = \sum_{i=1}^{N} w_i \delta(\psi - \psi_i), \tag{5}$$

in which the weights w_i are given by

$$w_i = \frac{p(d|\psi_i)}{\sum_{j=1}^{N} p(d|\psi_j)}. \tag{6}$$

The density $p(d|\psi_i)$ is the probability density of the observations given the model state ψ_i, which is often taken as a Gaussian:

$$p(d|\psi_i) = A \exp\left[-\frac{\{d - H(\psi_i)\}^2}{2\sigma^2}\right], \tag{7}$$

in which $H(\psi_i)$ is the measurement operator, which projects the model state on the observation d, and σ is the standard deviation of the observation error. When more measurements that might have correlated errors are available, the above should be the joint pdf of all these measurements.

Weighting the particles just means that their relative importance in the probability density changes, as shown in stage 3 in Figure 1. For instance, if we want to know the mean of the function $f(\psi)$ we now have

$$\overline{f(\psi)} = \int f(\psi) p(\psi|d)\, \mathrm{d}\psi \approx \sum_{i=1}^{N} w_i f(\psi_i). \tag{8}$$

A potential problem is that the weights tend to vary too much: a large number of particles have very low weight compared with the others. If the process of propagation of the ensemble and assimilation of new observations is repeated a few times (or with a large number of observations only once), only one member with large weight will remain and all others have negligible weight. This means that the statistical information in the ensemble is lost; effectively only one particle has all information available to us. A way to avoid this is so-called resampling. This results in ignoring the particles with low weight and duplicating ones with high weight, such that we end up with an ensemble of particles with equal weight again. Several ways to perform the resampling exist, see e.g. Doucet *et al.* (2003) and Van Leeuwen (2009) for a review.

The final stage 4 in Figure 1 is the resampling step that gives all particles equal weight again by taking multiple copies of particles with high weight and ignoring particles with low weights. Universal resampling is used here, in which all weights are put after each other in the interval $[0, 1]$ and a random number from the uniform density over $[0, 1/N]$ is chosen. That number is laid on to the unit interval, and the weight it points to is the first resampled particle. Then $1/N$ is added to the random number and the weight that points to denotes the second resampled particle. This process is repeated to generate N resampled particles, all with equal weight. From there we start the model integrations forward in time again.

The good thing about importance sampling is that the particles are not modified, so that dynamical balances are not destroyed by the analysis. The bad thing about importance sampling is also that the particles are not modified, so that when all particles move away from the observations they are not pulled back to the observations because only their relative weights are changed. This results in weights that vary wildly, and only a few particles will have relatively high weight and hence any statistical significance. This as called *filter degeneracy* and is a very serious problem in particle filtering (Snyder *et al.*, 2008). Several methods have been proposed to solve this problem (Doucet *et al.*, 2001), but none of these is directly applicable to the large-dimensional geophysical problems (Van Leeuwen, 2009).

Copyright © 2010 Royal Meteorological Society

4. A possible solution: the proposal density

We now discuss a very interesting property of particle filters that has received little attention in the geophysical community. It is related to the following. Suppose we want to determine the expectation value of a function of the state vector $f(\psi)$,

$$\overline{f(\psi^n)} = \int f(\psi^n)p(\psi^n|d^n)\,\mathrm{d}\psi^n = \frac{1}{A}\int f(\psi^n)p(d^n|\psi^n)p(\psi^n)\,\mathrm{d}\psi^n, \tag{9}$$

in which we used Bayes' theorem and A is a normalization factor. The prior density $p(\psi^n)$ can be obtained from integration from the previous state $p(\psi^{n-1})$, giving

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

We therefore start from the prior density at a previous time step $p(\psi^{n-1})$ and generate $p(\psi^n)$ using the *transition density* $p(\psi^n|\psi^{n-1})$. This density is given by the pdf of the model errors. If we write the model equation as

$$\psi^n = f(\psi^{n-1}) + \beta^n, \tag{11}$$

β^n denotes the stochastic model error, with pdf given by $p(\psi^n|\psi^{n-1})$.

At the heart of this article is the freedom in the transition density. We can rewrite (10) as

$$\overline{f(\psi^n)} = \frac{1}{A}\int f(\psi^n)p(d^n|\psi^n)\frac{p(\psi^n|\psi^{n-1})}{q(\psi^n|\psi^{n-1},d^n)} \times q(\psi^n|\psi^{n-1},d^n)p(\psi^{n-1})\,\mathrm{d}\psi^n\,\mathrm{d}\psi^{n-1}, \tag{12}$$

in which we just multiplied and divided by the so-called proposal transition density q. The important thing is that we can make this proposal density dependent on the future observations d^n. In this article we choose

$$\psi^n = f(\psi^{n-1}) + \hat{\beta}^n + K[d^n - H(\psi^{n-1})], \tag{13}$$

but many other more sophisticated possibilities are open. In this case $q(\psi^n|\psi^{n-1},d^n)$ is equal to the pdf of $\hat{\beta}^n$ but with mean $K[d^n - H(\psi^{n-1})]$. $\hat{\beta}^n$ is a stochastic term that could have equal pdf to β^n, but can also be chosen differently. The most important term is the new 'nudging' or relaxation term $K[d^n - H(\psi^{n-1})]$, which will 'pull' the particle towards the future observations. By choosing matrix K wisely, one can assure that all particles end up relatively close to the observations. One of the main points in this article is that we have an enormous freedom here: we can choose 'any' term that forces the model towards the future observations. (Of course, practical implementations put restrictions on K related to e.g. dynamical balances. We come back to this later.)

If we now use a particle representation of the pdf at time $n-1$ and choose random realizations for the proposal transition density, we find that the integral in (10) is again a weighted sum over the particles, but now with weights

$$w_i = \frac{1}{A}p(d^n|\psi_i^n)\frac{p(\psi_i^n|\psi_i^{n-1})}{q(\psi_i^n|\psi_i^{n-1},d^n)}. \tag{14}$$

To evaluate these weights we have to make choices for the pdf of the new stochastic forcing $\hat{\beta}^n$ and the matrix $\mathbf{K}$.

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 $\hat{Q}$. Also, assume that the observational errors are Guassian-distributed with mean zero and covariance R. The weights can now be written as

$$w_i \propto \exp\left[-\frac{1}{2}\left(\psi^n - f(\psi^{n-1})\right)Q^{-1}\left(\psi^n - f(\psi^{n-1})\right) + \frac{1}{2}\hat{\beta}^n\hat{Q}^{-1}\hat{\beta}^n - \frac{1}{2}(d^n - H(\psi^n))R^{-1}(d^n - H(\psi^n))\right], \tag{15}$$

where we can recognize the contributions from the original transition density $p(\psi_i^n|\psi_i^{n-1})$, the proposed transition density $q(\psi_i^n|\psi_i^{n-1},d^n)$ and the likelihood $p(d^n|\psi_i^n)$, respectively. The actual calculation of this term is as follows: one first chooses a realization for the proposed transition pdf for each particle in equation (13), i.e. a random value for $\hat{\beta}^n$ for each particle. We thus know ψ^{n-1} and ψ^n for each particle and use these to evaluate $p(\psi_i^n|\psi_i^{n-1})$ in the equation for the weights. Finally, we evaluate the likelihood.

In geophysics we usually have observations only every L time steps, where L can easily be 100 or more. This is an advantage since it allows us to keep the nudging term relatively small while still bringing the model towards the observations. In that case the weights become simply

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

which for our example boils down to

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

The way we use this expression is as follows. We integrate the new model equations (13). 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. Figure 2 shows how this particle filter with a 'nudging' term as proposal density works.

Copyright © 2010 Royal Meteorological Society

The particles are 'drawn towards the observations', and all particles have a comparable weight (shaded bars, red in the online article). The improved efficiency compared with the standard particle filter depicted in Figure 1 is clearly visible. The main difference from Figure 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.

5. Application to the Lorenz-63 model

As an illustration of the efficiency of the proposal transition density, we apply it to the Lorenz (1963) model (hereafter Lorenz-63). The parameters for the Lorenz-63 model are given by $\mathrm{d}t = 0.01$, $\sigma = 10$, $\rho = 28$, $\beta = 8/3$, $\sigma_{\mathrm{obs}} = \sqrt{2}$, $\sigma_{\mathrm{model}} = \sqrt{2\Delta t}, \sigma_{\mathrm{initial}} = \sqrt{2}$, in which σ_{model} is the standard deviation of the model error pdf.

The starting point was $(x_0, y_0, z_0) = (1.508870, -1.531271, 25.46091)$. The truth was generated by solving the stochastic model with the above parameters. Observations from this truth were sampled from the x-variable of the truth run every 40 time steps. Random noise was added chosen from a Gaussian distribution with error variance as indicated above, and a correlation matrix with 1 on the diagonal, 0.5 on the first sub- and superdiagonals and 0.25 on the second sub- and superdiagonals.

With these parameters, the solutions of the Lorenz equation show chaotic behaviour and the data-assimilation problem is a difficult one. To make the problem even harder (and more realistic) we provide only measurements on the x variable of the system every 40 time steps.

Given that the Lorenz-63 model has only three dimensions, the standard particle filter with resampling performs very poorly: even 20 particles cannot trace the true solution. One tends to need a few hundred particles to solve this problem with the standard particle filter (Nakano *et al.*, 2007).

In the application of the new particle filter we chose the **K** matrix in the nudging term as 25 times the model correlation matrix described above. It does depend on time by multiplied it by a linear function that is zero to half way the two updates and growing exponentially to 1 at the new observation time. The random forcing was multiplied by 1 minus that function. This allows the ensemble to spread out due to the random forcing initially, and to pull harder and harder towards the new observation the closer to the new update time it is. The results are not very sensitive to these choices. 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).

Figure 3 shows the same set up of the Lorenz-63 model, but not using the new particle filter using only 3(!) particles. This small number of particles is chosen to emphasis the strength of the proposal transition density idea, and the result does rely heavily on the nudging term. Obviously, to represent the full pdf more particles would be needed. The truth is followed very closely for the measured x variable, but also the non-measured y variable is traced very well, see Figure 4. The figures show that the new procedure is much more efficient than the standard one.

6. Almost equal weights

There is more, however. When a large number of observations is present, the weights still tend to differ considerably and filter divergence is still possible. To avoid this we can make all weights almost equal in the last step towards the observations by changing the proposal density in this last step. Assuming Gaussian errors in the model equations for the target transition densities and ignoring the proposal contribution for the moment, the weights can be written as

$$\begin{aligned} w_i \;\propto\; & w_i^{\mathrm{rest}} \exp\Big[-\frac{1}{2}\left(\psi^n - f(\psi^{n-1})\right) Q^{-1} \\ & \times \left(\psi^n - f(\psi^{n-1})\right) \\ - \;& \frac{1}{2}(d^n - H(\psi^n))R^{-1}(d^n - H(\psi^n))\Big], \end{aligned} \tag{18}$$

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, say equal to C, so

$$\begin{aligned} C_i \;=\; & -\log w_i^{\mathrm{rest}} \\ & +\frac{1}{2}\left(\psi^n - f(\psi^{n-1})\right) Q^{-1} \left(\psi^n - f(\psi^{n-1})\right) \\ & +\frac{1}{2}(d^n - H(\psi^n))R^{-1}(d^n - H(\psi^n)) = C. \end{aligned} \tag{19}$$

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 1, an infinite number of solutions. To proceed we first calculate the minimum theoretical value of C_i for each member i, as

$$\begin{aligned} C_i^{\min} \;=\; & -\log w_i^{\mathrm{rest}} \\ + \;& \frac{1}{2}\{d^n - H(f(\psi^{n-1}))\}\tilde{Q}^{-1}\{d^n - H(f(\psi^{n-1}))\}, \end{aligned} \tag{20}$$

in which $\tilde{Q} = HQH^{\mathrm{T}} + R$. This is the lowest value for C_i for each member. The problem we just solved is similar to that solved in the Kalman filter and in 3D-Var, with Q now having a different meaning (model error covariance instead of model state covariance). For nonlinear observation operators in particular, the 3D-Var methodology might be useful to find the minimum. However, it is stressed that this minimum is only an intermediate result as explained

Copyright © 2010 Royal Meteorological Society

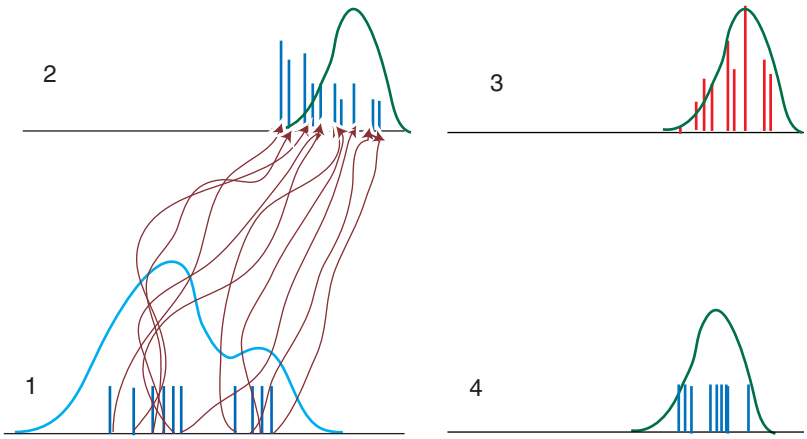

Figure 2. The new particle filter. Same as Figure 1, but now the particles are drawn towards the observation using the proposal density. Note that many more particles end up close to the observations in stage 2, resulting in a much better resolved posterior density in stage 3 and 4. Also note the different weights of the particles in stage 2 and 3 due to the proposal transition density, which changes the relative weights of the particles during the forward integration. This figure is available in colour online at wileyonlinelibrary.com/journal/qj

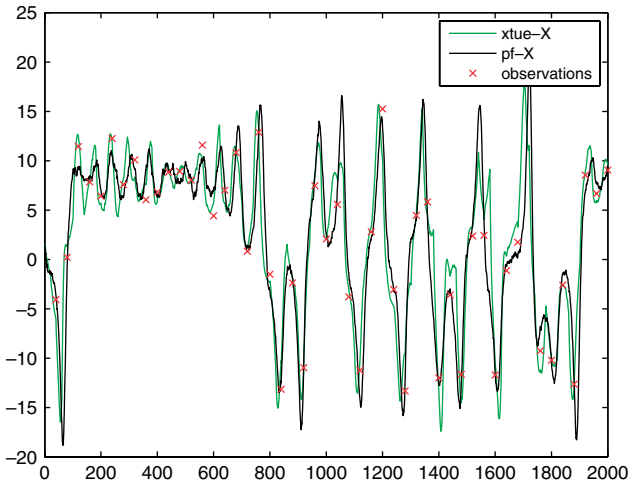

Figure 3. The new particle filter for the Lorenz-63 model: x-variable. The shaded crosses (red in the online article) denote the observations, the black line is the true solution and the shaded line (green in the online article) is the mean of the three-particle ensemble. Note that this three-member particle filter is able to follow the truth quite well, much better than traditional methods which typically need hundreds of particles. This figure is available in colour online at wileyonlinelibrary.com/journal/qj

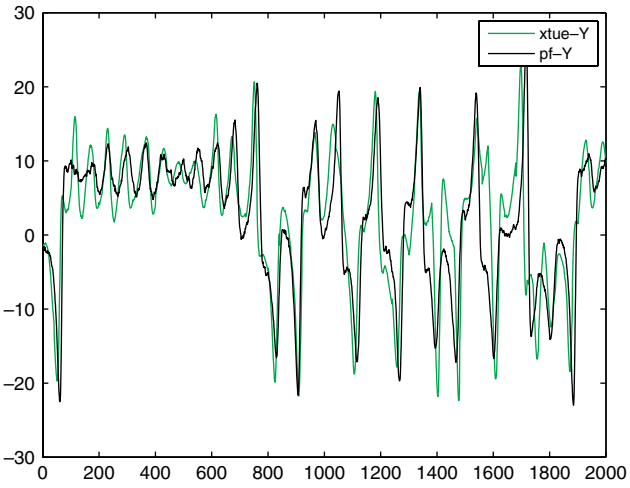

Figure 4. The new particle filter for the Lorenz-63 model: y-variable. The black line is the true solution, and the shaded line (green in the online article) is the mean of the three-particle ensemble. Note the closeness of the estimated solution to the truth for this non-observed variable, showing information flow from the observed variable to this variable through the model equations. This figure is available in colour online at wileyonlinelibrary.com/journal/qj

below, and the minimum value does not have to be known with extreme accuracy.

To make all C_i 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 particle. We have chosen C such that 80% of the particles can achieve that weight. The last 20% are 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. 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. We now choose solutions

$$\psi_i^n = f(\psi_i^{n-1}) + \alpha_i K(d^n - H(f(\psi_i^{n-1})), \qquad (21)$$

in which $K = QH^{\mathrm{T}}(HQH^{\mathrm{T}} + R)^{-1}$ and α_i is a scalar. Other choices might be equally valid. We thus reduce the problem to a quadratic equation in a scalar, which is easily solved as

$$\alpha = 1 - \sqrt{1 - b_i/a_i}, \qquad (22)$$

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

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 centred 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, and with large width to ensure that the weights will not change much. In our example with the Lorenz-95 model we used a Cauchy 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

Copyright © 2010 Royal Meteorological Society

divide by the new proposal density

$$\frac{1}{1+\left(\psi^n - f(\psi^{n-1})\right)\hat{Q}^{-1}\left(\psi^n - f(\psi^{n-1})\right)}, \quad (23)$$

in which $\hat{Q} = \gamma^2 Q$, with γ small, e.g. 10^{-10}. 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 the particles form a random sample from the posterior pdf.

To conclude, we present a short flow diagram of the computations needed.

(1) Generate the initial ensemble of model states, e.g. by perturbing the best initial guess using the pdf of the initial state. This is not trivial in large-dimensional systems, but we recall that this ensemble is only used to start the process. When model error is present it is soon 'forgotten', hence some freedom regarding accuracy exists.

(2) Propagate each model state i, or particle i from now on, forward using random samples from the proposal transition density $q(\psi_i^n|\psi_i^{n-1}d^n)$, i.e. choose $\hat{\beta}_i^n$ and the nudging term in Eq. (13). (Note that other choices for this proposal transition density can be made, e.g. a 4D-Var on each particle.) At each time step j calculate

$$\begin{aligned}\log w_i^j &= \log w_i^{j-1} \\ &\quad + \log p(\psi_i^j|\psi_i^{j-1}) - \log q(\psi_i^j|\psi_i^{j-1}, d^n).\end{aligned} \quad (24)$$

Use the log to avoid under- or overflow. Obviously, any constants in the pdfs do not matter. For Gaussian pdfs one obtains

$$\begin{aligned}\log w_i^j &= \log w_i^{j-1} \\ &\quad - \frac{1}{2}\left(\psi_i^j - f(\psi_i^{j-1})\right) Q^{-1}\left(\psi_i^j - f(\psi_i^{j-1})\right) \\ &\quad + \frac{1}{2}\hat{\beta}_i^j \hat{Q}^{-1}\hat{\beta}_i^j.\end{aligned} \quad (25)$$

Obviously, $w_{-1} = 1$. Do this until the last time step before the new observations. Note that when the random forcing vector is large, the last term in this equation is close to the size of the random forcing vector for each particle, so that term can be ignored (Andrew Lorenc, private communication).

(3) Use the almost equal-weight procedure by first calculating the minimum value for the weights for each particle $C_i^{\min}$, e.g. using (21) for our example.

(4) Determine the 80% minimum (or another percentage of particles retained).

(5) Determine the new state vectors for each of the resulting particles, e.g. using (21) for our example.

(6) Perturb these state vectors to make this step random, using a small-amplitude large-tail proposal pdf, e.g. using (23) for our example. For large-dimensional random forcing fields this step can be omitted again.

(7) Recalculate the full weights and resample to obtain a full ensemble again.

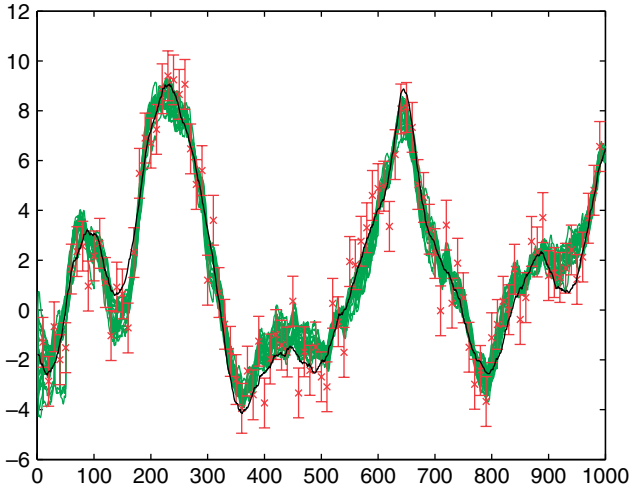

Figure 5. The new particle filter with almost equal weights for the Lorenz-95 model. The chaotic 40-dimensional Lorenz-95 model in which every other model variable is observed every 10 time steps is shown. The black line is the true solution, the shaded crosses (red in the online article) represent observations of this truth, and the shaded lines (green in the online article) depict the evolution of the particles in time. Note that the particles follow the truth remarkably well using only 20 particles, whereas traditional methods need thousands of particles. This figure is available in colour online at wileyonlinelibrary.com/journal/qj

7. Application to the Lorenz-95 model

A much more challenging example is the 40-variable Lorenz (1995) model (hereafter Lorenz-95), in which just pulling the particles towards the observations still results in wildly varying weights. Also this model is used in the chaotic regime and the data-assimilation problem is much more difficult than before. For the Lorenz-95 model we use $\mathrm{d}t = 0.01$ and $F = 8$, with 40 grid points. 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.

The truth run and the observations were generated as described for the Lorenz-63 model, with observations every other grid point, every 10 time steps. The observation error was $\sigma_{\mathrm{obs}} = 1$, the initial condition standard deviation was $\sigma_{\mathrm{initial}} = 2$ and the model error covariance was chosen as $\sigma_{\mathrm{model}} = 0.5$ times a correlation matrix with 1 at the diagonal and 05. at the first sub- and superdiagonals.

The nudging scheme explores a $\mathbf{K}$ matrix of 1 times the correlation matrix described above. The random terms $\hat{\beta}$ were chosen from a Gaussian with an error covariance twice as large as that of the original model, to compensate for reduction in ensemble spread due to the nudging term. This modification was properly accounted for via the weights, as explained in the previous sections. The last time step before the new observations uses the 'almost equal-weight' scheme explained in the previous section.

This problem has been studied before by Nakano *et al.*, (2007), and we use similar model parameters. They showed that tens of thousands of particles were needed with the standard particle filter with resampling for the result that we are able to achieve with 20 particles with the new method. Figure 5 shows what the new particle filter generates for an observed point: a swarm of particles that follows the observations and the truth (black line) smoothly in time. Figure 6 shows the ensemble for an unobserved variable.

Copyright © 2010 Royal Meteorological Society

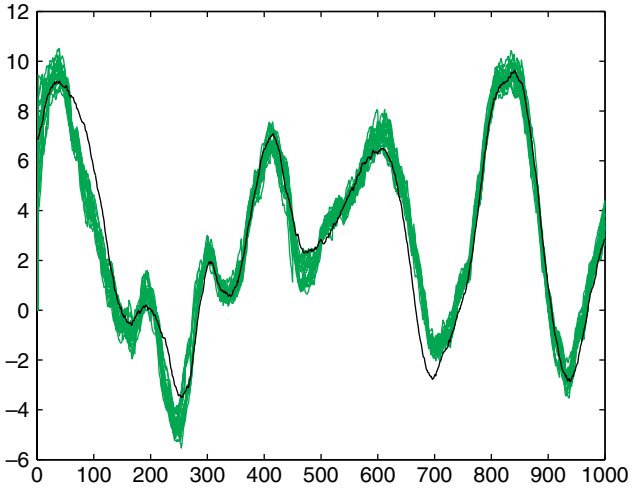

Figure 6. Same as Figure 5, but now for an unobserved variable. Although the truth lies outside the ensemble cloud for about 10% of the time (it should do so for 1/20th of the time), it is followed remarkably well. This figure is available in colour online at wileyonlinelibrary.com/journal/qj

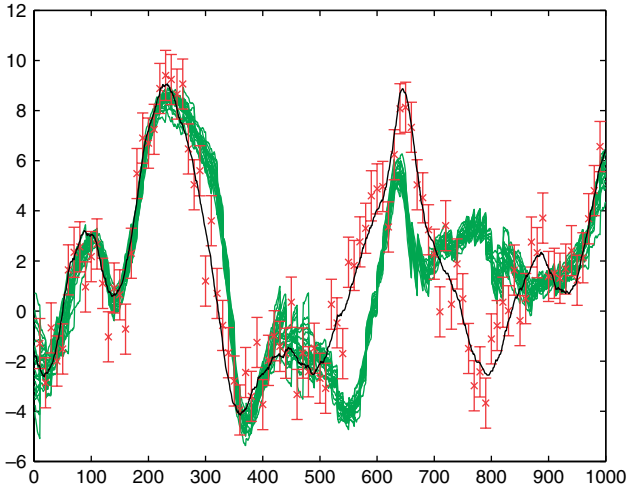

Figure 8. Same as Figure 5, but now for the EnKF with 20 particles, without localization. Although the filter does reasonably well, it does miss the truth for some period of time on several occasions. This figure is available in colour online at wileyonlinelibrary.com/journal/qj

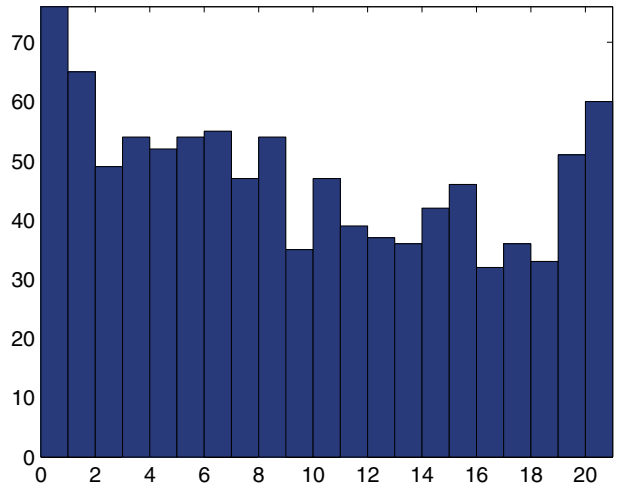

Figure 7. Rank histogram of how the truth ranks in the particle filter ensemble for a 10 000 time-step run. The flatness of the histogram shows that the truth is indistinguishable from any of the ensemble members. This figure is available in colour online at wileyonlinelibrary.com/journal/qj

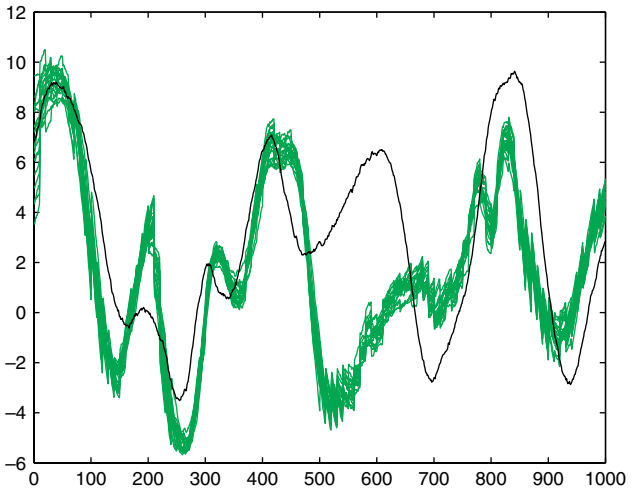

Figure 9. Same as Figure 7 but now for an unobserved variable. The filter loses track of the truth quite often. This figure is available in colour online at wileyonlinelibrary.com/journal/qj

Again, the truth (black line) is followed faithfully. To study the quality of the posterior ensemble we show the rank histogram of point $x = 20$ in Figure 7, derived from a 10 000 time-step run. This rank histogram scores where the truth ranks in the ensemble, and the flatness of the histogram in Figure 7 shows that the truth run is indistinguishable from any of the ensemble members.

Finally, the new method is compared with an EnKF with perturbed observations, as explained in Burgers *et al.* (1998). Figures 8 and 9 show the equivalents of Figures 5 and 6 for the EnKF solutions. Clearly, the EnKF has difficulty following the truth both in the observed and unobserved variables. The normalized root-mean-square difference from the truth over the whole time interval is 1.3 for the new particle filter and 3.5 for the EnKF. Obviously, this does not mean that the new particle filter is better than the EnKF in all cases. For instance, increasing the random forcing is beneficial to the EnKF compared with the new particle filter in that their performance becomes more comparable.

8. Conclusions and discussion

A new data-assimilation method is introduced that is fully nonlinear and has enormous potential for large-dimensional applications. We managed to track the true solution of the chaotic Lorenz-63 model with only partial observations of the state vector with only three particles. This has never been shown before, and this method outperforms all other existing data-assimilation methods in terms of efficiency. Obviously, this example only shows that the proposal transition density idea is a powerful one, not that a three-sample particle filter has much statistical value. We also presented an application to the much more complex 40-dimensional Lorenz-95 model, where we used 20 particles and found very encouraging results. The method outperfomed the EnKF in the present settings. The freedom in proposal density to ensure almost equal weights for the particles allows for the development of more efficient schemes than presented here. A new era of nonlinear filtering/smoothing is opening. Our method has

Copyright © 2010 Royal Meteorological Society

similarities with Chorin and Tu (2009), but is tailored to large-dimensional problems.

The method is easily implemented for large-dimensional (up to 1000 dimensions) problems, and work is now being carried out to study its performance in much higher dimensional systems. As Andrew Lorenc has pointed out, the value of the proposal transition density $q(\psi_i^n|\psi_i^{n-1}d^n) = p(\hat{\beta}^n)$ will be similar for each particle (equal to n, the dimension of the state vector) when the state space is large because it is a sum of a very large number of random variables with mean equal to 1. Hence it can be ignored in the evaluation of the weights, even for the almost equal-weights step, strongly decreasing the computational load of the method.

We have obtained similar results with a 1000-dimensional Lorenz-95 model using also just 20 particles (not shown), proving that the dimensionality problem can be attacked very efficiently. Obviously, one cannot properly represent the full pdf in a million-dimensional space by only 20 model states, but due to computer limitations one can never represent the pdf faithfully. We have shown here that one can solve the data-assimilation problem in large systems with a small number of states that capture some essential features of the full pdf, similar to the present-day practice with EnKFs in these systems (but now including non-Gaussian features).

This new method will help us to concentrate on other outstanding problems in data assimilation, not hindered by linearity assumptions. Examples are the structure of model errors and observation pdfs and finally the improvement of models using data assimilation.

Finally, a word on the proposal transition density is in order. We have employed a simple nudging term in our experiments. Clearly, if the nudging is too weak the solutions are too far away from the observations and the equal-weighting scheme leads to ensembles with too wide a spread, identifiable in e.g. rank histograms. When the nudging is too strong, all particles tend to collapse on the observations. The theory presented in this article shows that with an 'infinite-size' ensemble the size of the details of the nudging do not matter for the posterior pdf. However, for small-size ensembles this will matter. The rank histogram in Figure 7 shows that our choices for the Lorenz-95 model were in the correct range.

Using a nudging term, as is done in the examples presented here, might not work for systems where delicate balances can easily be destroyed. We should keep in mind that we have entered the era in data assimilation where errors in the model equations cannot be ignored anymore, so small random changes to the traditional deterministic equations will be present. However, it is possible that the nudging terms will become too large and destroy the balance. In that case several solutions can be envisioned. One of them is to control the size of the nudging term. Another is to use other proposal transition densities than those explored here, such as a full 4D-Var on each particle or an ensemble smoother. Note that a stochastic term still has to be added to each deterministic 4D-Var solution to avoid division by zero, or the observations could be perturbed to obtain the random element. On a practical note, in the latter case the proposal density weights will be rather complicated, and this method is expected to be specifically efficient when the dimension of the system is large, so that the proposal weights for each particle tend to be the same (see comment above). This is a research area that needs further exploration.

References

Anderson JL, Anderson SL. 1999. A Monte-Carlo implementation of the nonlinear filtering problem to produce ensemble assimilations and forecasts. *Mon. Weather Rev.* **127**: 2741–2758.

Bengtsson T, Snyder C, Nychka D. 2003. Toward a nonlinear ensemble filter for high-dimensional systems. *J. Geophys. Res.* **108**: DOI: 10,1029/2002JD002900.

Bennett A. 1992. *Inverse Methods in Physical Oceanography*. Cambridge University Press: Cambridge, UK.

Burgers G, Van Leeuwen PJ, Evensen G. 1998. Analysis scheme in the ensemble Kalman filter. *Mon. Weather Rev.* **126**: 1719–1724.

Chorin AJ, Tu X. 2009. Implicit sampling for particle filters. *PNAS* **106**: 17249–17254.

Courtier P. 1997. Dual formulation of four-dimensional variational assimilation. *Q. J. R. Meteorol. Soc.* **123**: 2449–2461.

Doucet A, De Freitas N, Gordon N. 2001. *Sequential Monte-Carlo Methods in Practice*. Springer: Berlin.

Evensen G. 1994. Sequential data assimilation with a nonlinear quasi-geostrophic model using Monte-Carlo methods to forecast error statistics. *J. Geophys. Res.* **99**: 10143–10162.

Evensen G. 2006. *Data assimilation: The Ensemble Kalman Filter*. Springer: Berlin.

Gordon NJ, Salmond DJ, Smith AFM. 1993. Novel approach to nonlinear/non-Gaussian Bayesian state estimation. *IEE Proc. – F* **140**: 107–113.

Jazwinki AH. 1970. *Stochastic Processes and Filtering Theory*. Academic Press: New York.

Lorenz EN. 1963. Deterministic nonperiodic flow. *J. Atmos. Sci.* **20**: 130–141.

Lorenz EN. 1995. Predictability: A problem partly solved, In *Proc. Sem. Predictability, Vol. 1*. ECMWF: Reading, UK; pp 1–18.

Metropolis N, Ulam S. 1944. The Monte Carlo Method. *J. Am. Stat. Assoc.* **44**: 335–341.

Nakano S, Ueno G, Higuchi T. 2007. Merging particle filter for sequential data assimilation. *Nonlin. Processes Geophys.* **14**: 395–408.

Snyder C, Bengtsson T, Bickel P, Anderson J. 2008. Obstacles to high-dimensional particle filtering. *Mon. Weather Rev.* **136**: 4629–4640.

Talagrand O, Courtier P, 1987. Variational assimilation of meteorological observations with the adjoint vorticity equation. I: Theory. *Q. J. R. Meteorol. Soc.* **113**: 1311–1328.

Van Leeuwen PJ. 2009. Particle filtering in geosciences. *Mon. Weather Rev.* **137**: 4089–4114.

Van Leeuwen PJ, Evensen G. 1996. Data assimilation and inverse methods in terms of a probabilistic formulation. *Mon. Weather Rev.* **127**: 2741–2758.

Zhang F, Zhang M, Hansen JA. 2009. Coupling ensemble Kalman filter with 4-dimensional variational data assimilation. *Adv. Atmos. Sci.* **1**: 1961–9533. DOI:10.1007/s00376-009-0001-8.

Copyright © 2010 Royal Meteorological Society