Nonlinear Ensemble Data Assimilation for the Ocean

Peter Jan van Leeuwen

Institute for Marine and Atmospheric research Utrecht (IMAU)
Utrecht University, P.O.Box 80005, 3508 TA Utrecht, The Netherlands
leeuwen@phys.uu.nl

ABSTRACT

Data assimilation in high-resolution atmosphere or ocean models is complicated because of the nonlinearity of these models. Several methods to solve the problem have been presented, all having their own advantages and disadvantages. In this paper so-called particle methods are discussed, based on Sequential Importance Resampling (SIR). Reference is made to related methods, in particular to the Ensemble Kalman filter (EnKF). A detailed comparison between the EnKF and the SIR is made using the nonlinear KdV equation. It is shown that for problems this nonlinear the Gaussian assumption made in Kalman filter methods is too restrictive. In passing it is noted that the error covariance is not a good measure for uncertainty in strongly nonlinear systems: the common wisdom that it reduces during the update is proven to be incorrect. The entropy of the system is shown to be a more adequate measure. It is shown that SIR produces good results in a highly nonlinear multi-layer quasi-geostrophic ocean model. Since the method needs at least 500-1000 ensemble members or particles with the present-day observing system, new variants have to be studied to reduce that number in order to make the method feasible for real applications, like (seasonal) weather forecasting. We explore local updating, as is commonly used in ensemble methods and discuss its problems. Finally a new idea is discussed in which the the number of members can be reduced by at least a factor 10 by guiding the ensemble to future observations. In this way the method starts to resemble a smoother. It is shown that the new method gives promising results, and the potentials and drawbacks are discussed.

1 Introduction

In areas of the world ocean where even state-of-the-art numerical models still have serious short comings data assimilation can be of use. Especially smoothers, in which observations of the system are not only used for future evolution but also back in time, can be used to point to missing model physics, problematic forcing, or wrong boundary conditions. Another area where data assimilation is of importance is forecasting. By providing an initial condition as close as possible to observations, while still allowing information from model dynamics, more accurate predictions can be made. In this case there is no need to use a smoother because it contains no extra information compared to a filter, even for nonlinear models (see Evensen and Van Leeuwen, 2000).

The problems we are facing today regarding data assimilation are for a large part attributable to poorly known model behavior. We all agree that all observations of a system should be accompanied by an estimate on how accurate the measurement is. For numerical models this is not common practice. Still numerous studies appear in literature where one single (climate) model run is analyzed in some detail followed by exclamations on the behavior of the model in general, or even on the real world! Obviously, one needs to know the model sensitivity to various parameters and parameterizations before this kind of statements can be made.

Assuming for the moment that we know the accuracy of the numerical model, it can be combined with observations in a data assimilation scheme. The Kalman filter provides the linear least-squares solution given the observations and the model with their errors. The model errors are presented in a large matrix, the so-called error covariance matrix. This error covariance matrix is a remarkable matrix indeed. Apart from providing the errors of the model variables on the diagonal, it also contains information on how all model variables co-vary, that is, it contains all linear relationships between all model variables. For instance, for the ocean is tells one how the volume transport in a deep western boundary current is linearly related to the position of the Gulf

Stream front. For the atmosphere one can think of the wind speed in Reading as linear function of the position of the Jet Stream.

When we return to the Kalman filter, we see that we somehow have to propagate the error covariance matrix forward in time between observations, because the relations between the model variables will change. For linear models this can be done without approximation. The only problem is that the size of the computation compared to a single model run increases dramatically: if N operations are needed to propagate the model one time step, N^2 operations have to be performed in the Kalman filter. Thinking about a state-of-the-art atmospheric model with more than 1 million model variables, this becomes way too expensive. Even more problems arise when the model becomes nonlinear.

While for linear problems a Kalman filter will provide the variance minimizing solution, for nonlinear problems this is not the case (see e.g. Jazwinski, 1970). The so-called extended Kalman filter designed for nonlinear problems can only handle weakly nonlinear model behavior because the assumption is made that the error evolution evolves according to the tangent linear model evolution. Furthermore, it is assumed that the central forecast, so the model evolution assuming no errors in the dynamical equations, is the optimal state. The advantage of this assumption is that the optimal state evolves independently from the rest of the probability density of the model, so that the covariances are only needed at analysis times. This has led to several methods to approximate the error covariances at analysis times, avoiding the time-consuming propagation of the covariance fields. Obviously, when the model is strongly nonlinear, problems regarding the optimality of the solution arise.

The problems are in fact threefold. First, the evolution of the error covariances is not according to the tangent linear model evolution. Several ensemble methods based on either a root-mean-square approximation of the error covariance (e.g. Verlaan and Heemink, 1997) or a Monte-Carlo approach solve this problem. The second problem is that the central forecast is not optimal between analyses for a nonlinear model evolution. Methods like the Ensemble Kalman filter (Evensen, 1994, see also Burgers et al., 1998) address this problem by propagating an ensemble of model states forward in time, and taking (correctly) the variance minimizing state as the mean of the ensemble. Finally, the Kalman update itself is not variance minimizing (see e.g. Jazwinski, 1970). In this paper a possible solution to this last problem is presented, using ensemble integrations to solve the first two.

The method presented here remains rather close to Bayes equation, which is at the heart of the data-assimilation problem. It is shown that a frequency (or particle) representation of the probability density of the model evolution leads to a weighting of the ensemble members (or particles) related to their distance to the observations. Van Leeuwen and Evensen (1996) tried to apply this method for a smoother over a 100 day interval in a quasi-geostrophic ocean model, but it failed to work. The failure is due to the fact that only very few ensemble members have a large-enough weight over the 100 days, so that the effective ensemble size becomes too small. However, when the weighted probability density is resampled every now and then, the idea can be made to work for quite large state spaces, as is shown in this paper. Several similar methods have been proposed for these kind of problems (e.g. Anderson and Anderson, 1999; Miller et al., 1999; Pham, 2001), but these methods have only been tested for low-dimensional problems like the Lorenz63 model (although the application to large-scale models is discussed in these papers).

Before we continue along this line it is worth spending some time on its necessity. Are the models that nonlinear? By inspecting a probability density function (pdf) of a multi-layer quasi-geostrophic model of the ocean we immediately see that the nonlinearity is substantial: when starting from a Gaussian the pdf becomes multimodal in a matter of days (see figure 1). The pdf was constructed by running an ensemble of models each with a slightly perturbed initial condition drawn from a Gaussian and adding Gaussian model noise at each time step (see section 4).

The message from such a figure is that data-assimilation methods that contain linearizations of the true dynamics are expected to work not too well. Indeed, methods like the Extended Kalman filter (EKF) will not work in problems like these, because they assume that the errors in the optimal estimate evolve with linear dynamics. It is expected that when the resolution of the numerical models increases this problem becomes even more

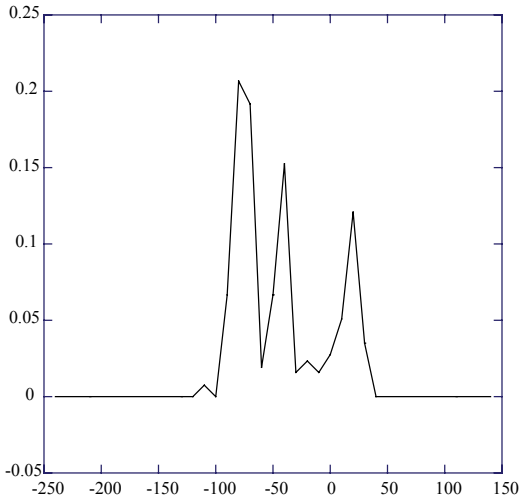

Figure 1: Example of a probability density function in a certain model point, obtained from a data-assimilation run with a multi-layer quasi-geostrophic model.

stressing because the dynamics tend to become more nonlinear.

As mentioned above, a possible way to circumvent this problem is by letting the errors evolve with the nonlinear model equations by performing an ensemble of model runs. This approach leads to the well-known Ensemble Kalman filter (EnKF), as presented by Evensen (1994, see also Burgers et al., 1998). A practical advantage of this method above the original EKF is the fact that only a limited number of ensemble members (100-500) is needed, compared to the equivalent of N ensemble runs (with linear dynamics) in the EKF, in which N is the size of the state space, which can easily be 1 million. A drawback of the original EnKF is that random perturbations drawn from the observation error density have to be added to the actual observations to produce the correct statistics. This leads to sampling errors that can be avoided by using e.g. an adjustment ensemble Kalman filter (Anderson, 2001). However, the extra sampling leads to a very efficient Kalman update as shown by Evensen (2003, see also this issue).

The number of ensemble members needed can be reduced further by considering combination of a square-root filter and the EnKF, with is superior to either variant separately (Heemink et al, 2001). Another line of research is to try to find more efficient representations of the error covariance, like the first few singular vectors, as is done in the SEEK filter (Brasseur et al., 1999). (The singular vectors are not to be confused with those of the tangent linear model operator.) Problems with these efficient methods is that they rely more and more on the linearity of the model. For strongly nonlinear models they might be less efficient because the evolving singular vectors are not the singular vectors of the evolving error covariance matrix. A way to solve this is to recompute the singular vectors after a few time steps, which makes the method much less efficient.

One might ask how an ensemble of about 100 realizations can accurately represent a covariance matrix of over 1 million model variables. The answer is that it cannot. A trick is needed to let these ensemble methods work, and that is to strongly reduce the number of degrees of freedom by doing a local analysis. This means that only those observations are taken into account that are close to a model grid point. This can be achieved by multiplying the error covariance by a correlation function, or by taking the full error covariance, but than restricted to a certain area around the grid point. The former deforms the error covariance matrix close to the grid point, while the latter introduces an unphysical cut off at larger distances. At this moment it is unclear which approach gives best results.

One problem remains, however, and that is the analysis step, when the model is confronted with observations. In all Kalman-filter-like methods it is assumed at analysis time that either the model is linear or that the pdf of the model is Gaussian. For observations the assumption is that the pdf is Gaussian, and that the measure-

ment operator (that produces the model equivalent of an observation) is linear. In methods like the EnKF, the Kalman-filter update equations are used even when the above mentioned assumptions are not fulfilled. What is done is that the variance of the ensemble is used in the Kalman-filter equation. So, it is assumed that this equation does hold, but the rest of the method is 'as nonlinear as possible'. With the latter I mean that due to the ensemble nature of the method it is possible to use nonlinear measurement operators, and each ensemble member is updated individually. One can show that, under the assumptions of the original Kalman-filter, the updated ensemble has the correct covariance. (Note that the covariance is never explicitly calculated in ensemble methods.) However, since these assumptions are relaxed to some extend it is in fact not entirely clear what is done statistically. For instance, the prior ensemble will in general have a third-order moment (skewness) that is unequal to zero. So, extra information on the true pdf is present, which is a good thing. However, the update is done independent of this skewness. So, the updated ensemble will have a skewness that is deformed by the update process, and it is unclear what its meaning is. It might be completely wrong, deviating the estimated pdf from the true one.

So-called adjoint methods take the nonlinearity of the dynamics into account by performing a sequence of linearizations. Although good results are reported (see e.g. Weaver in this issue) there is no a-priori reason why these methods should work for highly nonlinear models. They search for the maximum of the pdf (or the minimum of a cost function) which will contain several local minima. There is no guaranty whatsoever that the gradient descent search applied will end up in the true minimum. In fact, multiple searches starting from different first guesses should be done to detect and avoid these local minima. This is however not common practice because it makes the methods very expensive. An example of failure is a highly nonlinear Richardson-number-dependent mixed layer scheme in the tropical Pacific, in which the method failed to converge (Vossepoel, private communication). Not withstanding these potential problems, very interesting results have been obtained in numerous settings, showing that the adjoint methods do work in the numerical models used today.

It is good to realize also what approximations are made in adjoint methods. First it is assumed that the adjoint of the forward model exists. However, several parameterizations in the forward model contain if statements, which cannot be linearized. Practical fixes used are to let the adjoint model just follow the backward trajectory of the forward model trough the if statement, but this is not correct mathematically. When the if statement is flow dependent this might prevent gradient descent methods to converge. Another approach is to replace the complex parameterization by a very simple one containing no if statement in the adjoint model. Also here convergence is an issue. In fact, the above mentioned failure with the tropical ocean model could be traced back to this problem. Other approximations are made in the formulation of the cost function. There, one has to assume that errors in initial conditions and observations (and model dynamics if included) are Gaussian distributed. This is not the same as saying that the model itself is Gaussian distributed, as done in the Kalman filter, however. The relation between the cost function and the pdf is explained in e.g. Van Leeuwen and Evensen (1996).

A serious problem is, however, that the method does not provide an error estimate in a highly nonlinear context. Note that the inverse of the Hessian, which is sometimes claimed to provide this error estimate, does only so in a linear context. This is because the Hessian is the local curvature of the pdf at its maximum, which has no direct relation to the width of the pdf if that is different from a Gaussian, see Fig. 2. Another problem is the assumption that the model dynamics and so-called physics (that is, the diabatic processes) are free of errors. Unfortunately, we are a long way from that assumption. Finally, the error-free model dynamics make the final estimate extremely dependent on initial conditions in the (nearly) chaotic system that the true atmosphere or ocean is. This latter problem can be avoided by avoiding long assimilation time intervals, as is done in incremental 4DVAR.

An advantage of the latter methods is the fact that the model evolution is always balanced. The Kalman-filter-like methods can produce unbalanced states at analysis times because each updated ensemble member is just a (local) linear combination of the prior ensemble. For instance, negative concentrations or unbalanced dynamics can occur, leading to spurious gravity-wave energy, or even model blow up. This is due to the Gaussian

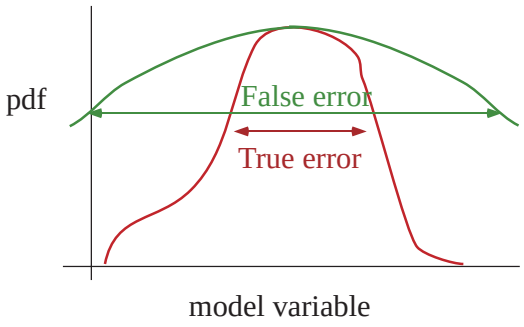

Figure 2: The Hessian gives the curvature of the pdf at its maximum, not the width of the pdf which is the correct error measure.

assumption on the pdf's, while e.g. a concentration does not have a Gaussian pdf.

An interesting method that shares the advantage of balanced states with 4DVAR, but does allow for model errors is the representer method (Bennett, 1992). It is also based on linearizations, but also here good results have been presented for nonlinear models (e.g. Bennett et al., 1996).

Another interesting fact of all methods presented above is that they have to perform matrix inversions. The size of the matrix depends on the number of observations, and can become rather large. So, it seems wise to try to keep the number of observations low for optimal numerical performance, while, from a statistical point of view, we want to use as much observations as possible.

Is it possible to create a data-assimilation method that can handle non-Gaussian pdf's, nonlinear measurement functionals, provide error estimates and has no problems with millions of observations? To answer this question one has to return to the most general description of the data-assimilation problem. Bayes has shown that the probability density of a model ψ given a new set of observations is given by:

$$f_m(\psi|) = \frac{f_d(|\psi)\, f_m(\psi)}{\int f_d(|\psi)\, f_m(\psi) d\psi}. \tag{1}$$

In words, the pdf of the model given the observations is given by the product of the prior pdf of the model and that of the observations given the model. The denominator is a normalization constant. Note that, because we need the pdf of the observations given the model, we are not troubled with the true value of the observed variable, only with its measurement and the model estimate.

A close look at the equation shows that the posterior pdf can be obtained by just multiplying the densities of model and observations. So, data assimilation is that simple. In its purest form it has nothing to do with matrix inversions, it is not an inverse problem in that sense. At the same time we see how ensemble methods can be used. The ensemble members, or particles, are a certain representation of the prior model pdf, and the posterior pdf is represented by a reweighting of these ensemble members. This weighting is dependent on the value of the observations given an ensemble member. Sequential Importance sampling is just doing that: it creates an ensemble of models, runs that ensemble forward until observations become available (so far the method is identical to the other ensemble methods like the EnKF), weight each ensemble member with these observations, and continue the integration. This procedure is depicted in figure 3, and the actual calculation is presented in the next section.

A practical problem is evident in figure 2, namely that some ensemble members have nothing to do with the observations: they are just way off. These members get a very low weight compared to the others. The ratio of these weights can easily be a million or more. Clearly, these members have no influence on the first few moments of the pdf, they contain no real information anymore. This has let people to so-called resampling (Gordon et al, 1993). In the original formulation one just draws the complete ensemble randomly from the weighted posterior ensemble. Clearly, posterior ensemble members with very low weight have a very low

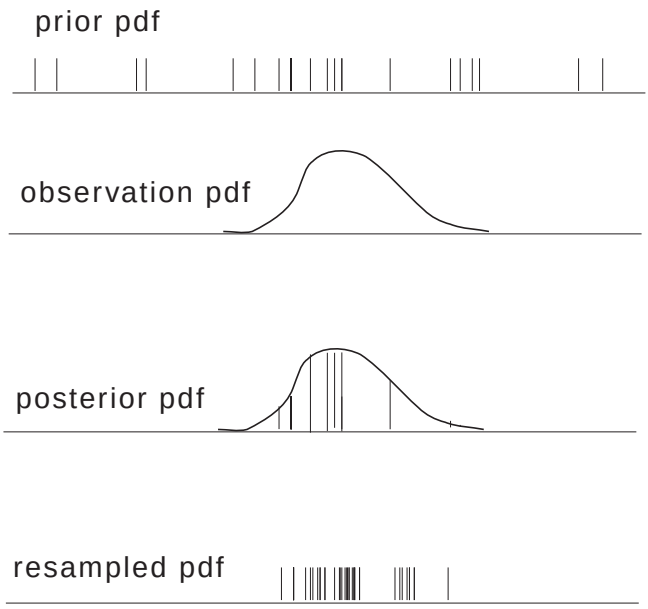

Figure 3: Sequential Importance Resampling: the prior pdf, as represented by an ensemble is multiplied with the observation pdf (not necessarily a Gaussian) to obtain the posterior pdf as represented by the ensemble. This posterior pdf is resampled to give each member equal weight again. The horizontal axis denoted one model variable.

probability to be drawn, while members with large weights can be drawn more than once. This is nothing more than abandoning those members that contain no information, and stress those that have. In the present implementation members with large weights are copied, and the number of copies is related to their weight. This copying and abandoning is done such that the total number of members remains the same as before the update. This presents, in words, Sequential Importance Resampling; we elaborate on this in the next section.

In Van Leeuwen (2003) was shown that this method works in a highly nonlinear corner of the World Ocean, where the size of the state-space was about $2\ 10^5$. Unfortunately, the size of the ensemble needed was rather high; good results were obtained with a size of 495. This is considered too large for operational purposes. The idea to reduce this number further arose from communications with Kevin Judd. He suggests to use a so-called shadowing trajectory (or rather pseudo orbit) to first detect where the observations are before an ensemble is created to follow that trajectory. He proposes to use 4DVAR to generate the shadowing trajectory, but the drawback is that that is rather expensive (of the order of 100 ensemble integrations), so that the gain is lost in generating this first trajectory. Approximate ways to construct the trajectory are under construction, as are extensions to pseudo-orbits for models including dynamical errors.

Another approach is to try to perform a local analysis, as is also done with ensemble Kalman filters. This leads to the local-analysis SIR, the so-called LSIR. It turns out that that idea is not combined easily with the SIR idea of only weighting ensemble members, not combining them. On the other hand, if we could guide the ensemble members to where the observations are, a serious reduction in the ensemble size is expected. This results in a SIR-algorithm in which the SIR is applied with the same set of observations before the actual observation time. This leads to the Guided Sequential Importance Resampling. It can be shown that this is a valid procedure as long as the weighting is done properly. In this paper it is shown that this method can reduce the number of ensemble members needed to about 30 for the problem under consideration.

In the present paper the methods are presented in the next section. In section 3 a comparison of the performance of the EnKF and the SIR is presented to highlight the differences in a strongly nonlinear environment (clearly not to judge on which method is best...). Section 4 discusses the SIR, the LSIR and especially the GSIR for a real-sized problem with real observations. The paper closes with a discussion of the relative merits.

2 Sequential Importance Resampling

In this section a filter is derived for a truly nonlinear model using Bayes theorem. At the heart of nonlinear data assimilation lies the notion of combining probability densities of model and observations. By expressing the problem in terms of probability density functions a Bayesian estimation problem can be formulated. In Bayesian statistics the unknown model evolution ψ is viewed as the value of a random variable $\underline{\psi}$. The density $f_m(\psi)$ of ψ is obtained from the model somehow, and is called the prior probability density. Using the definition of a conditional probability density we can derive the new, or posterior, probability density of ψ given the observations :

$$f_m(\psi|) = \frac{f_d(|\psi)\, f_m(\psi)}{\int f_d(|\psi)\, f_m(\psi) d\psi}. \tag{2}$$

The first factor in the numerator, the density $f_d(|\psi)$, is the probability density of the observations given that the model random variable $\underline{\psi} = \psi$. It is rather convenient that the observational density f_d comes in this form, because we do not have to reference to an unknown truth, but instead to the known model counter part. The second factor is the a-priori model density $f_m(\psi)$. The denominator is the probability density of the observations, written as a marginal density of the joint density of model and observations. Obviously, this is just a normalization term (see e.g. Van leeuwen and Evensen, 1996, for details).

The first thing that comes to mind when considering this equation is that data assimilation is not an inverse problem from the start. It can be put in that form, but defining it in terms of combining probability densities is more natural. Indeed, this has direct bearings with the general idea of data assimilation: we try to combine information from observations with dynamical information in the form of partial differential equations of the underlying physics. It can be put into an inverse problem for practical purposes, but there is no need to do so in principle. As we will see, the method presented here will not be turned into an inverse problem, so no matrices have to be inverted.

As is well known the variance minimizing model evolution is equal to the mean of the posterior probability density:

$$\overline{\psi} = \int \psi\, f_m(\psi|)\, d\psi \tag{3}$$

A frequency (or particle) interpretation of this equation leads to

$$\overline{\psi} = \sum_{i=1}^{N} \psi_i \tag{4}$$

in which the particles are drawn from the posterior density $f_m(\psi|)$. Since this density is not known, we use (2) to rewrite this as:

$$\overline{\psi} = \frac{\sum_{i=1}^{N} \psi_i f_m(|\psi_i)}{\sum_{i=1}^{N} f_m(|\psi_i)} \tag{5}$$

The meaning of this equation is that each ensemble member (or particle) is weighted by its 'distance' to the observations, with the weights given by:

$$w_i = \frac{f_d(|\psi_i)}{\sum_{i=1}^{N} f_d(|\psi_i)}. \tag{6}$$

The 'distance' is found from the probability density of the observations. If the observational errors are Gaussian distributed, and , for simplicity, uncorrelated with standard deviations σ, the weights are found as:

$$w_i = \frac{1}{A} \exp\left[-\frac{(-H\psi_i)^2}{2\sigma^2} \right] \tag{7}$$

in which the normalization constant A is given by:

$$A = \sum \exp\left[-\frac{(-H\psi_i)^2}{2\sigma^2} \right] \tag{8}$$

and in which H is the measurement operator, which can be strongly nonlinear.

From the above it becomes clear that all operations that have to be performed to obtain the variance minimizing solution are direct, i.e. no inversions are present. One of the advantages is that the measurement operator can be extremely difficult to linearize without damaging the calculation: we just take it as it comes. Another is that no matrices have to be inverted, either direct or iteratively. It is also noted that higher-order moments are extremely easy to obtain as soon as we have the ensemble. For instance, any moment $g(\psi)$ is obtained as:

$$\overline{g(\psi)} = \sum_{i=1}^{N} w_i g(\psi_i) \tag{9}$$

with the weights w_i given above. Again, no inversions needed.

Now, as mentioned in the introduction, Van Leeuwen and Evensen (1996) have tried to apply these ideas in a two-layer quasi-geostrophic model of the ocean around South Africa. They fed the model with gridded altimeter observations every 10 days, but soon found that the relative weights varied to wildly. Only a few members had relatively large weights, while the rest had such low weights that it made no contribution in the posterior density to the first two moments. It was estimated that at least 1 million (!) ensemble members were needed to obtain statistically significant results in their application. A possible solution is in fact known for quite a long time (Gordon et al., 1993): resample the posterior density after some time to create a new ensemble in which all ensemble members have equal weight again. Clearly, the success of such a method depends on the density of the observations, but it is a promising candidate for the real solution. This resampling can be done in a variety of ways, and we discuss the results from importance resampling here. For an overview the reader is referred to Doucet et al.(2001).

2.1 Importance resampling

The basic idea of importance resampling is extremely simple (see Gordon et al., 1993) First calculate the weight of each ensemble member. This collection of weights forms a density in ensemble space. In the purest form of importance resampling a sequence of numbers is sampled randomly from that density, in total the amount of ensemble members N. Clearly, more numbers will be drawn at high weights than at low weights. The amount of numbers drawn at a certain weight is equal to the number of identical copies that are made of the ensemble member with that weight. So, if let's say weight w_i is chosen 4 times, 4 identical copies of ensemble member ψ_i are made. On the other hand, if a weight is so low that it is not chosen at all, no copies of that ensemble member remain. In this way a new ensemble is created, in which all ensemble members have equal weight again. It is resampled from the density defined by the weights, so by the relative closeness of the observations to the members. In Fig. 1 the procedure is depicted.

In certain variants of importance resampling the copies of an ensemble member are not identical, but some 'jitter' is applied to obtain a little more spread in the ensemble. For instance, Anderson and Anderson (1999) use the kernel method (Silverman, 1986), in which a Gaussian is created around each chosen ensemble member, and instead of identical copies new ensemble members are drawn from this Gaussian. The covariance structure of this Gaussian is of course problematic. In the standard kernel method it is taken as a factor times the covariance of the whole ensemble, but the accuracy of this procedure is questionable. There is no a-priori reason why the local structure of the probability density should resemble the global structure. Nevertheless, interesting results are obtained with several variants of the Lorenz63 model. In the application described here no jitter is applied because the errors in the model dynamics are so large that identical copies will spread relatively fast.

The procedure that I followed in this paper closely resembles importance resampling, with the following modification. Instead of choosing randomly from the distribution determined by the weights, members with large weights are chosen directly from the distribution in the following way. First, this density is multiplied by the total amount of ensemble members. For each so-obtained weight that is larger than 1 the integer part of that

weight determines the number of identical copies of the corresponding ensemble member. So, if the original weight of member i was $w_i = 0.115$, with an ensemble size of 100, the new weight $\tilde{w}_i = 100 * 0.115 = 11.5$. This results in 11 copies of the ensemble member, while $\tilde{w}_i = 0.5$ remains. This procedure is followed for all weights. Finally, all remaining parts $\tilde{w}_i$ form a new density from which the rest of the ensemble is drawn, according to the rules of stochastic importance resampling described above.

The reason for the deviation from the basic importance resampling was as follows. Statistically the basic rule is preferable if the ensemble is large enough. However, computational arguments lead to as small ensemble sizes as possible. In that case we are more strongly interested in the members with high weights and the above method minimizes the possibility that too much members are drawn with low weights. In this way I try to maximize the amount of information present in the weighted prior ensemble on physical balances in the new ensemble. So, interestingly, while we have sampled the posterior density rather bad because of the low ensemble size, we keep to this sample as close as possible to avoid ensemble members that are useless from a statistical pint of view. Later experimentation has shown that the method is not sensitive to the resampling method as long as it is statistically correct.

A smoother version of the method can be obtained by using the weights at all times in the smoother time interval. The potential problem that only a few ensemble members are retained after a few filter steps, so that the smoother is based on a much smaller ensemble, needs further investigation and is postponed to a following paper.

A final comment on the particle nature of the filter and the resampling is that because of these two approximations the filter is not exactly variance minimizing: only the mean of the continuous probability density has that privilege. However, because the continuous density cannot be calculated or stored in practice, the particle filter is as close as we can get. Indeed, the finite number of ensemble members is the only approximation made.

2.2 The LSIR algorithm

In the SIR algorithm ensemble members are only copied or removed at analysis time. This means that no 'new blood' enters the ensemble: the ensemble is not drawn towards observations as in most other methods. For instance, in the EnKF ensemble members are mixed: each new member is approximately just a linear combination of the old ones. (There is some nonlinearity due to the fact that the matrix that determines the relative weight of the old members is determined using those same members. This is related to the 'inbreeding problem' pointed out by Houtekamer and Mitchell (1998) and discussed further by Van Leeuwen (1999b) and Houtekamer and Mitchell (1999). The advantage of the SIR algorithm is that the model states are always well balanced, but the disadvantage is that the ensemble members themselves are never adjusted towards the observations.

One of the problems that one encounters is that an ensemble member that performs good in some area of the total domain might do very bad in a remote area. Should we keep that member? In the SIR the weighting is done globally, i.e. all observations from the whole domain are taken into account. But what we want is that the member remains in the ensemble in one area, and removed in the other. So, we want to perform a local updating. It is expected that this might be a way to strongly reduce the number of ensemble members.

The algorithm is explained rather easily. We apply the SIR as explained in the previous section. However, this time we only weight the members at each grid point using a local batch of observations. This means that the relative weights of the ensemble members is different for each grid point. Now the following problem arises.

Suppose I have to make 5 copies of a certain member at one grid point in the resampling process, and only 4 copies of that member at a neighboring grid point. The new ensemble for the two grid points will than contain the 4 copies they have in common, but one copy remains. That copy has to be combined with another ensemble member from the neighboring point. The question is now: how do we glue these two ensemble members together?

In the experiments described below I did the most simple thing: I just glued them together with no extra measures taken. In principle this could lead to spurious gradients, but it seems that the quasi-geostrophic model in which this was tested behaved quite well. We come back to this point when discussing the results. A better procedure might be to define a transition region in which the new ensemble member slowly changes from one old ensemble member to another ensemble member, for instance using a weighted mean of the two. A problem now is that features in each member will be smeared out, and the new member might be worse than either of the original ones in the transition region.

One could think of other, more sophisticated methods to glue members together, but the bottom line is that no general approach is available, and a lot of tuning is needed for each new problem. My philosophy is to stay a closely to the general equations as possible for objectively and clarity.

2.3 The GSIR algorithm

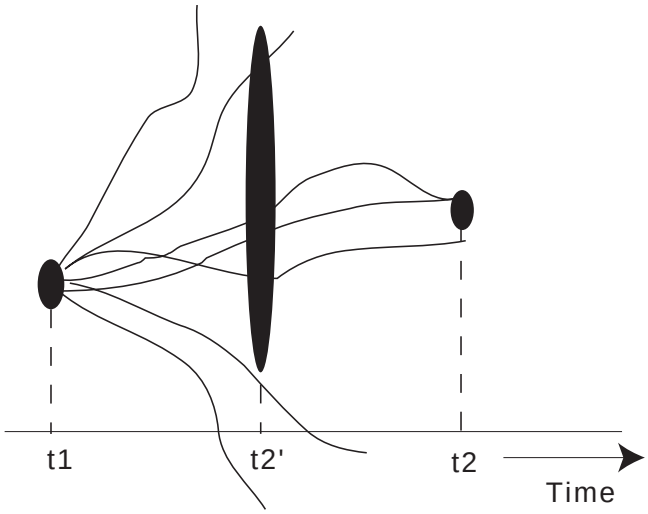

Figure 4: Guided Sequential Importance Resampling: a SIR is performed at t2' before the actual measurement time t2 to guide the ensemble towards the observations at t2. The ellipses denote the observations with their standard deviation.

Given the SIR algorithm given above, the GSIR is easily explained. Figure 4 shows how the method works. When the SIR is applied at the previous observation time, we wait until the next observations come in. Then we integrate the ensemble forward in time for a few time steps. We now assume that the new observations are performed at this time and perform a SIR-step. Obviously, we make an error here, because the observations are not to be used yet. However, we want to know already at this stage which members are going in the right direction, and which members are shooting off to remote areas in state space not supported by the observations at all. To avoid being too strict we increase the measurement error by a large factor, 100 say. The system is so nonlinear (or the model is so bad), that a number of ensemble members is unable to get a reasonable weight at this stage. By using the SIR we just abandon these already.

After performing the SIR at this intermediate time step before the actual observations, we continue integrating the ensemble, again a few time steps. Then the procedure described above is repeated, but the error in the observations is increased less than in the previous step, let's say a factor 10. Again the SIR is performed at this intermediate step. This is repeated a few times if necessary.

Then we integrate the ensemble up to the true measurement time. Again we perform a SIR step, but because the ensemble members are already relatively close to the observations (guided to them as it were), the weights will not differ too great. This allows one to greatly reduce the number of members. In the application described in section 4 the reduction is more than a factor 30! This brings the number of members needed to about 30,

which is sensible for real applications, given the ensemble forecasts and number of iterations performed in the 4-DVAR at this moment. It is even cheaper...

It can be shown that no approximation is made compared to the original SIR when the relative weights of the ensemble members (or rather the number of copies made of them) in the too-early applications of the SIR algorithm are corrected for (Van Leeuwen and Doucet, in preparation).

3 Nonlinear filtering: A simple example with the KdV equation

We study the influence of the Gaussian assumption made in Kalman-filter-like methods on the solution of a highly nonlinear data assimilation problem in this section.

The Korteweg-DeVries (KdV) equation describes the nonlinear evolution of a field u subject to advection and dispersion. It reads:

$$u_t + 6uu_x + u_{xxx} = 0 \tag{10}$$

(In fact several forms of the KdV equation exist, see e.g. Drazin and Johnson, 1989) Shape conserving solutions called solitons are allowed by a balance between the steepening of the wave form due to nonlinear advection and the dispersion from the third spatial derivative. We start by a field of the form:

$$u(x,0) = \frac{0.5a}{(\cosh(\sqrt{a}(x-x_0)))^2} \tag{11}$$

in which x_0 is the position of the maximum of the wave form, and $0.5a$ its amplitude (see Fig. 5). The KdV equation will move the wave form towards positive x values with a speed a, while conserving its shape. Important for the following is that the soliton is stable to small perturbations (see e.g. Drazin and Johnson, 1989).

Several experiments have been performed with the SIR and the ensemble Kalman filter. We present one experiment in detail here that highlights the differences between the two methods and highlight some issues in nonlinear filtering.. We first form a true solution by integrating this form with $a = 1$ over a domain of length 50, with periodic boundary conditions, $x_0 = 20$ and $\Delta x = 0.5$. The time stepping scheme is leap frog, with an Euler step every 67 time steps to suppress the numerical mode. The time step is $\Delta t = 0.1$.

This solutions is measured six times, on $t = 10$, at $x = 37$, $x = 40$, and at $x = 43$, and on $t = 20$, at $x = 57$, $x = 60$, and at $x = 63$. To these pseudo observations random Gaussian noise with zero mean and standard deviation 0.05 was added. Note that the observations are taken around the peak values of the wave form.

An ensemble was created with the amplitudes of the ensemble members drawn from a Gaussian with zero mean and standard deviation 0.5. To mimic errors in model dynamics, random numbers are drawn from a Gaussian distribution with zero mean and standard deviation 0.001. These numbers are added to the solution at each time step. The ensemble size was 250, but the results converged at $N = 150$. This ensemble was integrated forward in time, and data were added at $t = 10$ and $t = 20$ as explained above. Figure 5 gives the mean of the forecasting ensemble at $t = 10$. The decrease of the amplitude can be attributed to the spread in amplitudes a, leading to an ensemble of soliton-like waves with different propagation speed.

In the same figure the mean of the ensemble after analysis at $t = 10$ is given for the Ensemble Kalman filter and for the sequential importance resampling filter. The measurements are indicated by the crosses. The first thing that strikes the eye is that the EnKF solution comes much closer to the observations than the SIR. However, the EnKF solution is not variance minimizing, and is in fact too close to the observations. The EnKF assumes that the prior probability density of the model is Gaussian, but that is not the case. In Fig. 6 the prior, posterior and observational density are given at the peak of the true soliton, on $t = 10$, at $x = 40$. The densities are created using the frequency interpretation on pre-specified intervals. Varying the intervals within reasonable bands showed that the features visible are robust. Clearly, the prior is non Gaussian because the solitons are always

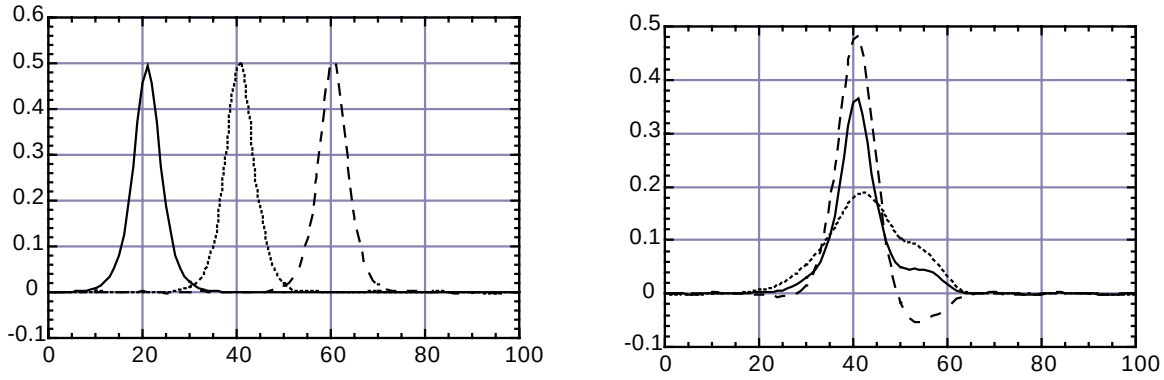

Figure 5: Left: True solution of the KdV equation at time 0, 10 and 20. Right: Prior (dotted), EnKF (dashed) and SIR (solid) ensemble mean solutions on t=10.

nonnegative. Several ensemble members moved too slow or too fast to have a significant value for the solution at $x = 40$. The variance of the prior is 0.3, which is more than a factor ten larger than that of the observations. So, indeed, the EnKF posterior solution, being a weighted mean between the prior ensemble mean and the observations, has to be very close to the observations. If the prior is not Gaussian distributed, as is the case here, the posterior is not only determined by the mean and the variance of the prior and the observations, but by the whole density. The variance minimizing solution is the mean of the posterior density, which gives much more credit to the prior model estimate. The SIR does exactly give this estimate. (Note that it has converged for 250 members.) So, the fact that the model is drawn close to the observations in the EnKF does not automatically mean that the EnKF analysis is good!

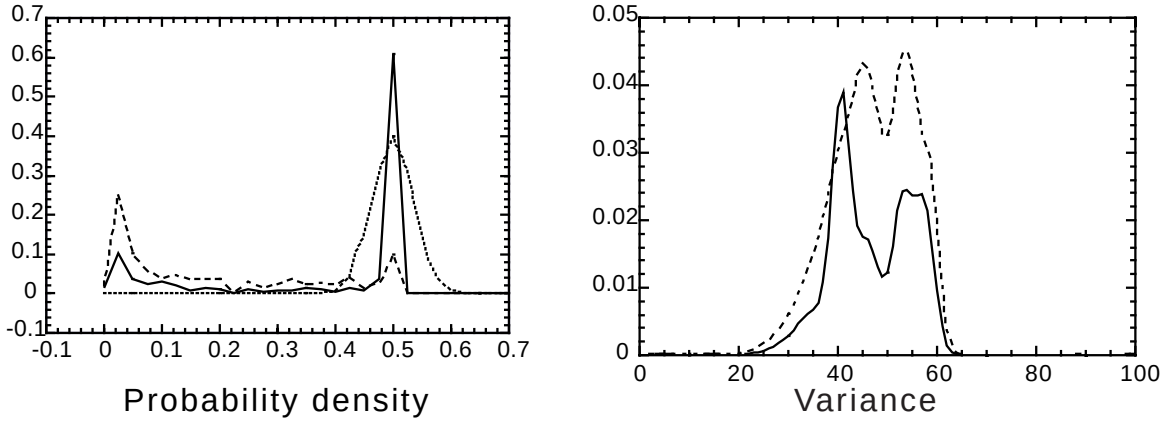

Figure 6: Left: Prior (dashed), posterior (solid) and observational (dotted) pdf on t=10 at x=40. Right: prior (dashed) and posterior (solid) variance on t=10.

Variance estimates for a truly variance minimizing solution also show unfamiliar behavior. The right-hand-side of figure 6 shows the prior and posterior variance estimates for the SIR at $t = 10$. Interestingly, the posterior variance is higher than the prior variance at the measurement point $x = 40$. So, in a conventional way of thinking the uncertainty in the estimate at that point is increased due to the measurement. A more accurate inspection of the full prior and posterior densities shows that the uncertainty has decreased, but the second moment of the density does not show it. A way to quantify this uncertainty is by introducing the so-called entropy of the system as (see Shannon, 1948):

$$H = -\sum_{k=1}^{M} p_k \log p_k \tag{12}$$

in which M is the number of bins with a probability density unequal to zero. One can easily show that the

entropy of a Gaussian distributed variable is proportional to its variance. For probability densities other than the Gaussian this is not necessarily true. For instance, for bimodal densities the variance looses its meaning. In that case the entropy is a more sensible estimate, as Shannon showed. In our case we find that the entropy of the prior density is $H = 1.1587$, and that of the posterior density is $H = 0.7286$. So, the entropy has decreased. We will come back to this in the next section.

We now turn to the analysis at $t = 20$. Figure 7 shows the mean states of the EnKF and SIR ensembles after analysis. The SIR is very close to the observations this time. This has to do with the correct update of the ensemble at $t = 10$, leading to a relatively large part of the ensemble rather close to the truth. The model error increases the spread in this ensemble a bit, so that the largest variance (about 0.045) is around $x = 60$ at $t = 20$. This variance is so large, and the ensemble is so close to the observations that an almost perfect match is possible. The EnKF analysis looks rather strange. In fact, the program crashes some time after the update. Several reasons for this behavior can be found. First, Fig. 5 shows that a negative tail exists in the analysis of the EnKF. However, as is well known, negative values do give rise to fast wave motion towards the left in the KdV equation. The reason for this behavior is that the nonlinear advection and the dispersion enhance each other instead of balancing each other. This is indeed what happens with the updated ensemble members. They have negative parts, and also the perturbation due to the update is too large, so that the soliton falls apart rapidly. So, the mean is not evolving as a soliton rightwards anymore, but it will break up. If one recalls that the figure shows the mean, one can imagine what the individual members must look like at analysis time, and how their subsequent evolution will be. The negative values arise due to the fact that each new EnKF ensemble member is a linear combination of the old ones, without taking into consideration that no negative values exist, so without realizing that the prior is not a Gaussian. The resulting over fitting at the measurements leads to negative values away from the measurement positions, due to the large gain. Another reason for the wild character of the update is the fact that the ensemble is way off. The prior variance is extremely high, leading again to a large gain all over the domain, resulting in problems as mentioned earlier.

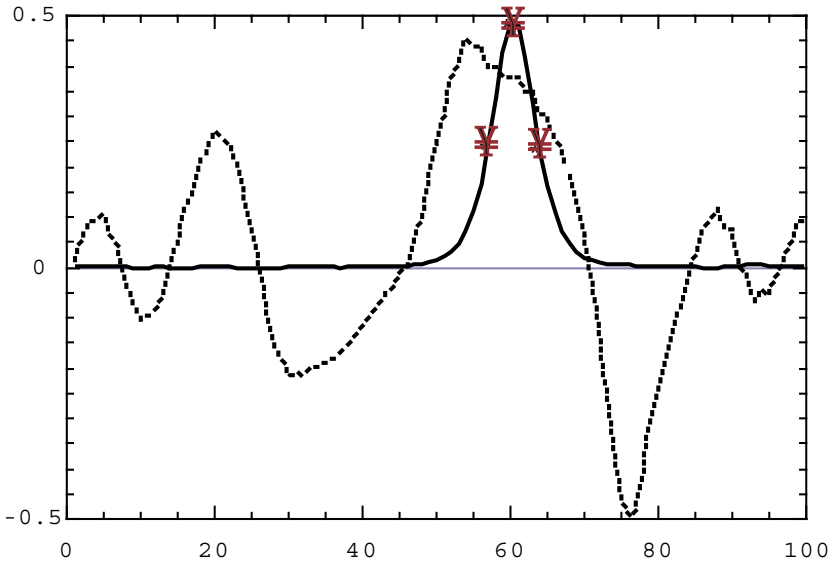

Figure 7: EnKF (dotted) and SIR (solid) ensemble mean solutions on t=20. The three observations are also indicated.

As mentioned above, several experiments have been performed with the KdV equation, and, depending on the system parameters, the above differences between EnKF and SIR are more or less pronounced. Since little new can be learned from them they are not discussed.

A serious problem is the issue of ensemble collapse. When the weights are such that the relative weight of one member is much larger than that of all the others, only that member is present in the determination of the moments of the posterior density. So, a mean will be produced, equal to that member, but the variance is close to zero. The filter thinks it is doing perfect, but that is not necessarily the case. The resampling will produce

N (the size of the whole ensemble) identical copies of that member, consistent with the weighted ensemble, but again (of course) having the same problem. (Generally, the filter will diverge from the true evolution, so ensemble collapse is also termed filter divergence. Since divergence is one of the possible consequences of the collapse we use the term ensemble collapse throughout this paper.) The experiments showed that the SIR is more sensitive to ensemble collapse than the EnKF (with perturbed observations). On the other hand, When the EnKF collapses it will be extremely difficult to get it spreading again because the variance in the ensemble remains very low. The observations are unable to pull the ensemble to the correct state because their variance will generally be much larger. For the SIR the situation is different. The variance in the ensemble is not directly related to the weighting with the observations via the Kalman gain, but instead each member is weighted individually. The best member in the SIR will by duplicated and the new ensemble will spread again during the integration. The probability that more members are close to the next observation set increases strongly this way. As a result, the SIR can be expected to be pulled more easily on the correct track than the EnKF. Crucial in the collapse problem is the choice of the probability density for the observations. The next section, that deals with a real application, elaborates further on this.

It should be mentioned that modifications to the standard EnKF can be used to overcome the problems presented here. For instance, one could put all negative values after analysis to zero, with or without compensation for the 'mass loss'. Also, one could perform a local analysis to avoid the problems. This is almost always used in large-scale problems to prevent spurious correlations to destroy the solution. So, several relatively simple fixes are possible to make the EnKF work in practice. The fundamental point is, however, that the EnKF is not variance minimizing. The new method proposed here does not have this kind of problems by construction.

4 The SIR, the LSIR, and the GSIR in a large-scale application

In this section the SIR, the LSIR, and the GSIR are applied to a large-scale problem to study its behavior in such a setting. We study the ocean area around South Africa. The Agulhas Current runs along the east coast of South Africa southward and retroflects just after leaving the continent at its most southern tip, back into the Indian Ocean. At the retroflection point large Agulhas rings are shed, moving into the South Atlantic. The area is an ideal test bed for data assimilation methods due to the highly nonlinear dynamics and the availability of high-quality satellite altimeter measurements of the height of the sea surface. Since this height is directly related to the pressure it is an important dynamical constraint on the flow.

The area is modeled by a 5-layer quasi-geostrophic ocean model with a horizontal resolution of 10 km, with 251*150 gridpoints. The first baroclinic Rossby deformation radius is about 35 km in this area. The layer depths are 300, 300, 400, 100 and 3000 m, respectively, with densities of 1026, 1026.9, 1027.25, 1027.65 and 1027.75 kg/m^3. The time stepping was done with leap-frog, with an Euler step every 67th step to suppress the computational mode. A time step of 1 hour was chosen, close to the CFL-limit, for optimal accuracy. Small-scale noise was reduced by a Shapiro filter of order 8. Boundary conditions are such that features are leaving the domain with twice the speed of the fastest 5 wave modes. Only the inflow of the Agulhas Current at the eastern side of South Africa was prescribed. Because this inflow is super critical for all baroclinic (Kelvin) waves this last condition is well posed for those waves. Problems with barotropic waves did not arise. (Note that the values at the boundaries are part of the data assimilation problem.) The model is able to produce realistic ring-shedding events and general meso-scale motion in the area, as compared to satellite observations and in situ ship measurements.

A comment on the relative size of the model domain is in order. The spatial dimensions are 2500 by 1500 km, which seems rather small from an atmospheric perspective. However, when we compare the Rossby radius of deformation with the domain size and keep that ratio constant when turning to the atmosphere, we see that we model the equivalent of twice the Earth, using the Rossby radius at midlatitude. This shows that the application can be qualified as large scale indeed.

4.1 Statistics

The initial streamfunction error (uncertainty) was taken space independent, with values of 4000, 3000, 2000, 1000 and 1000 m^2/s for the layer models. Every day a random error of 0.05 times these values was added to describe the model error. The spatial correlation of the errors was Gaussian with a decorrelation length of twice the Rossby radius of deformation. The state space, consisting of the 5 streamfunction fields for the 5 layers, has a dimension of about $2\ 10^5$. The ensemble size was 1024, where the SIR seems to have converged (see Van Leeuwen, 2003), and 32 for the LSIR and the GSIR.

The initial streamfunction error (uncertainty) was taken space independent, with values of 4000, 3000, 2000, 1000 and 1000 m^2/s for the layer models. Every day a random error of 0.05 times these values was added to describe the model error. The spatial correlation of the errors was Gaussian with a decorrelation length of twice the Rossby radius of deformation. The value of the initial errors was rather low because the model was initialized from an interpolated altimeter sea-surface height field. The time-mean field is always problematic because it is not well known. We used a similar field as used by Van Leeuwen (2001), but now interpolated over 5 layers. In this application we added the same time-mean field to all observations, leading to consistency between initial conditions and observations, but the real world might be inconsistent with this mean field. This will be visible in the data assimilation system by a bias in model dynamics (apart from the bias due to quasi-geostrophy). However, as explained by Van Leeuwen (2001), a bias does not prevent us from using the data-assimilation equations because they are still valid.

The state space, consisting of the 5 streamfunction fields for the 5 layers, has a dimension of about $2\ 10^5$.

4.2 Observations

The observations were satellite altimeter height data from the TOPEX/Poseidon and the ERS-2 satellites. These two satellites cover the model area with tracks that are about 150 (T/P) and about 70 (ERS) km apart. T/P has a repeat orbit of 10 days, ERS has a repeat orbit of 35 days. The along-track resolution is 7 km.

A problem with satellite altimeter data is that the time mean signal contains information about both the ocean circulation and the geoid. Since the geoid is not well known at the length scales of interest, only the time-varying part of the altimeter signal can be used. The time-mean field has to come from other sources. Here we used a field derived from in-situ measurements, as also used in Van Leeuwen (1999a) and Van Leeuwen (2001). Unfortunately, the accuracy of this field is not well known. (One could try to estimate the time-mean oceanic field by using it as the unknown in a data assimilation experiment, because the time-mean and the time-varying part of the signal are dynamically coupled. This is done by Van Leeuwen (1999a), but, although the results are encouraging, the quality of the field is still questionable. An independent estimate would definitely be preferable, and we have good hope that the Global Ocean Circulation Experiment (GOCE), that will determine the shape of the geoid, will help us out on this problem.)

For the initial field an interpolated image was produced from the observations over a time period of 35 days, representative for the oceanic situation of January 1st 2000. The observations that are used in the data-assimilation experiment are collected over 1 day and the resulting batches are offered to the model. So, each batch has only a partial coverage of the domain, a few tracks, that differs from day to day. Every 5th observation was used in the data-assimilation experiment, while every 5th observation with a offset of 2 was used as independent data to check the results from the assimilation experiment.

The observational error was specified as 5000 m^2/s, which corresponds to about 4 cm in sea-surface height. This value is probably a bit too high for T/P data (2 cm), but a little too low for the ERS data (5 cm). The universal value was chosen here for simplicity, and because both data sources suffer from a not well defined time-mean sea-surface topography. Recall that the purpose here is to demonstrate the abilities of the data-assimilation system, not the best reproduction of the oceanic state.

The shape of the probability density of the altimeter observations is a difficult matter. Due to the weighting procedure the SIR is very sensitive to the tails of that density. In general, we know little of those tails. It has been suggested, however, that the tails of a Gaussian are too small: the square in the exponent cuts off large deviations from the observations very drastically. So, outliers, meaning bad measurements here, may have a tremendous effect on the behavior of the filter. Indeed, the first experiments with Gaussian densities for the observations led to ensemble collapse directly at the first analysis time. Even increasing the observational errors with a factor of 10 did not help much.

This formed the motivation to look for densities with larger tails. The Lorentz density is used in this paper, but better alternatives can probably be found quite easily. The density is given by

$$f_d(|\psi) = \frac{1}{1+\frac{(d-H(\psi))^2}{\sigma^2}} \tag{13}$$

for uncorrelated observational errors. Advantage of this density is that it has a shape very similar to a Gaussian near the peak (symmetric and quadratic), but it is much broader away from the peak. The observational error is taken equal to σ, half the full-width at half maximum, so equal to a Gaussian in this respect. The similarity with the Gaussian close to the peak is an important reason to use this density, although other choices might be just as good. A disadvantage is that the density has infinity variance, but that is only a theoretical problem, not a practical one in this case.

4.3 Implementation

The SIR filter was implemented on a Origin 3500 parallel computer using up to 64 processors. The distribution over the processors was done in MPI. The analysis done was done serially, but because no inversions have to be performed the code remained extremely parallel. The speedup was close to 100 %, while the f90 code is extremely simple. The straight forward observational data set required no special treatment. However, for a more involved assimilation parallel IO seems to be in order.

4.4 Results

The carefully chosen error covariances lead to a sensible solution. First the statistics of the results are discussed, including a comparison with independent observations, then a short physical discussion of the obtained results is given.

4.4.1 Nonlinear filtering

In this section the emphasis is on the nonlinear aspects of the filtering problem. First we recall figure 1, which shows the probability density of the streamfunction at a point in the middle of the retroflection area (20E, 40S). This figure was created during the assimilation with the SIR filter, so not from a free run. It confirms that the exercise undertaken in this paper is needed because the pdf does not resemble a Gaussian. One could ask the question what gradient descent methods would produce with such a pdf. Apart from that, any linearizations seems doomed to fail.

Figure 8 shows the evolution of the total rms error of the ensemble for each model layer when the SIR is applied. The rms is defined here as the rms deviation from the ensemble mean, since the truth is not known. The original large uncertainty is reduced strongly at the first measurement time, to stay close to this level for the rest of the assimilation period. Since the method weights each ensemble member as a whole, the error evolution in the lower layers is just a constant fraction of that in the upper layer (represented by the top curve, in Fig. 10). The dashed line that starts at day 4 shows the unconstrained evolution of the rms error of the upper layer when no

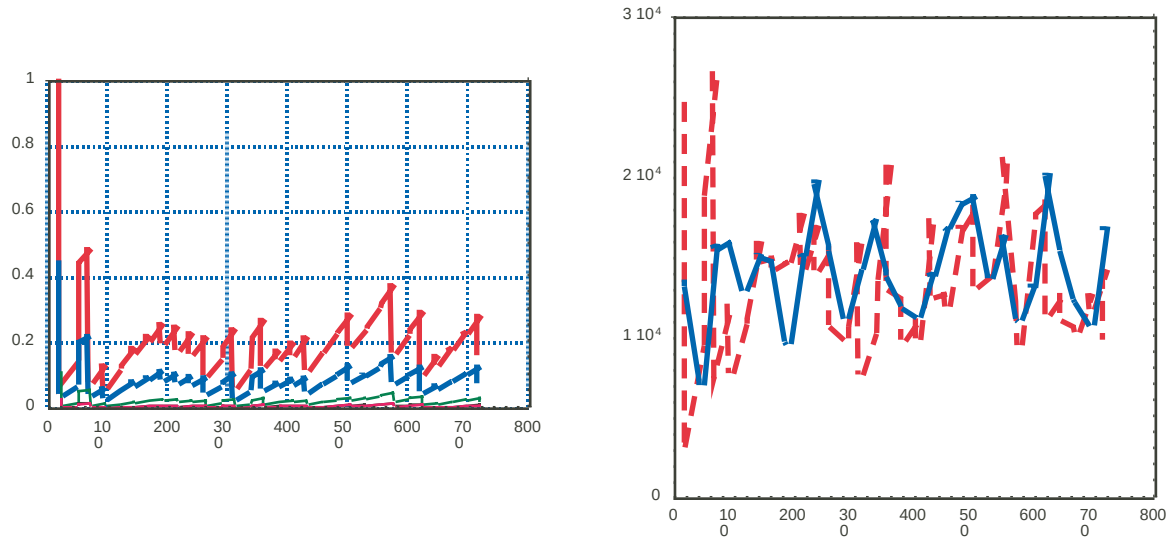

Figure 8: Left: Total rms error of each model layer versus time as determined from the ensemble. Note that the error sometimes goes up at analysis times, as is to be expected for non-Gaussian pdf's. Right: Rms error of the ensemble (dotted) compared to the difference of the ensemble mean with independent observations.

observations are assimilated after day 3. It shows that the error growth is indeed more or less exponentially, as one would expect from such a nonlinear model.

As we have seen in the KdV example, the error occasionally goes up at analysis time. For Kalman filter like methods, in which the assumption is made that model and observational prior densities are Gaussian, this would point to serious problems. The reason is that when combining two Gaussian distributed variables the variance of the posterior density always has to be lower than either of the prior variances, as can easily be shown. For variables that are not Gaussian distributed this doesn't have to be the case. So, the notion from modern control theory that the errors of the updated model should be at most as large as those of the observations is not valid for our data assimilation problem.

This fact, already demonstrated in the previous section, is illustrated here with a simple example. Consider two variables on the domain $[0,1]$, with one having a density that is 4/3 in $[0,0.5]$ and 2/3 in the rest of the interval, and the other variable has a density that is 2/3 in the first half and 4/3 in the second half of the interval. Obviously, the posterior density is 1 along the complete interval, and its variance is larger than that of either of the prior densities. In fact, this example is a bit too strong in the sense that also the entropy grows. So, the information in the observation is counteracting that in the model, and shows that we should be less certain than either the model or the observation predicts.

In the present case with the quasi-geostrophic model the entropy as measure of our uncertainty is calculated at the point corresponding to the probability density in figure 1. The entropy of the prior probability density is $H = 0.875$, while that of the posterior is $H = 0.731$. So, indeed, the entropy has reduced here too. As an informal statement one might say that when the model probability density is multi modal, and the observations favor only a few modes, the entropy, so the uncertainty will decrease, but the variance can increase because of the shifting mean value. No doubt more can be said on this, but that is beyond the scope of the present paper.

Another interesting feature is the fast initial drop of the error to a relatively low value, and the nearly constant value afterwards. Kalman-filter-like methods tend to show a more gradual decrease of the error. The reason must be that only a few ensemble members are close to the observations initially, so that only those members get a nonzero relative weight. The spread in those few members is relatively low. The resampling step then draws the complete ensemble to that part of state space.

In the right-hand-side of Fig. 8 the rms error between the unused altimeter observations and the mean of the

ensemble are compared to the variance of the ensemble at those observation points. The comparison shows that the ensemble spread is indeed what it should be. We thus can conclude that the SIR is doing quite a good job.

4.4.2 Reducing the ensemble size using LSIR and GSIR

In figure 9 a comparison is shown between the SIR with 1024 members, and the LSIR with 32 members. First of all, no numerical problems arise with the simple local updating implemented here. The QG-dynamics allows gluing together different members to one new member. It must be stressed however, that a primitive equation model will have more problems, especially due to spurious gravity wave generation. Assuming for now that these problems might be solved in the (near) future, we concentrate on the results first.

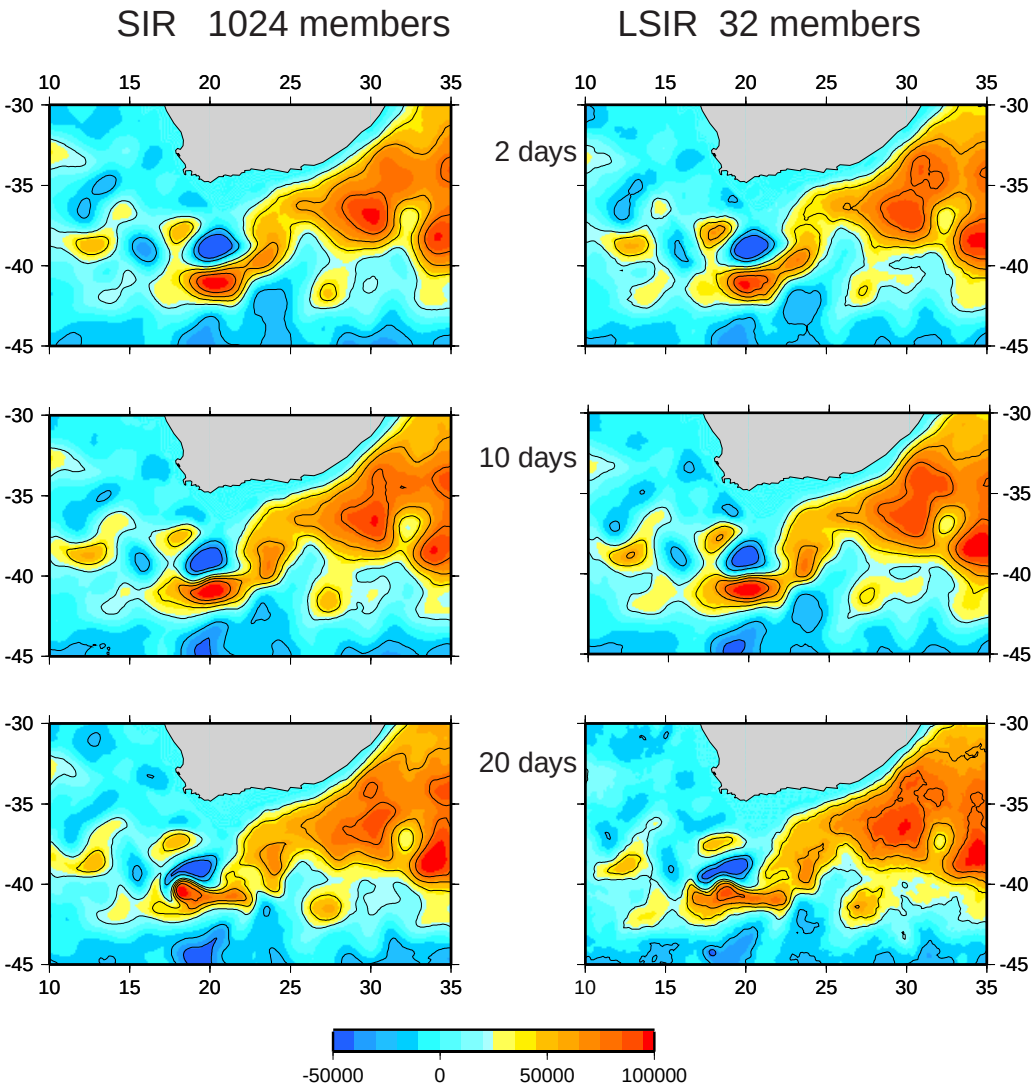

Figure 9: Upper-layer streamfunction for SIR with 1024 members (left) and LSIR with 32 members on day 2, 10 and 20.

Even after 2 days small differences can be observed in the retroflection area between the SIR and the LSIR results. Inspection of the error estimates these differences are however within the error bounds (not shown here). At day 10 the differences become more pronounced, and at day 20 serious differences are visible in the retroflection area, which are much larger than the error bound permit. This indicates that the LSIR with 32 members does not perform the way it should. A possible explanation is that we reduced the ensemble size too much. However, similar runs with 64, 128 and even 256 members showed that the large differences at day 20 remain although the stream function field does get smoother in the eastern part of the domain (not shown).

These results lead to the conclusion that local updating does not work, or at least needs serious adaptation.

Let us now look at the behavior of the GSIR. In figure 10 the same SIR run with 1024 members is compared to a GSIR run with 32 members. Again some small differences can be seen at day 2 between the two runs, but the runs are statistically the same given the error bounds on the SIR. Interestingly the GSIR results closely resemble those of the LSIR. The reason for this liking is not clear. It might be due to the smaller ensemble size, but even the 265 member LSIR does show this differences. But, no matter what, these difference don't have to worry us because statistically the results are the same. Note that this statement is true strictly speaking only true for a Gaussian, but since we started out with a Gaussian at day 0 the pdf will still be relatively close to a Gaussian after 2 days.

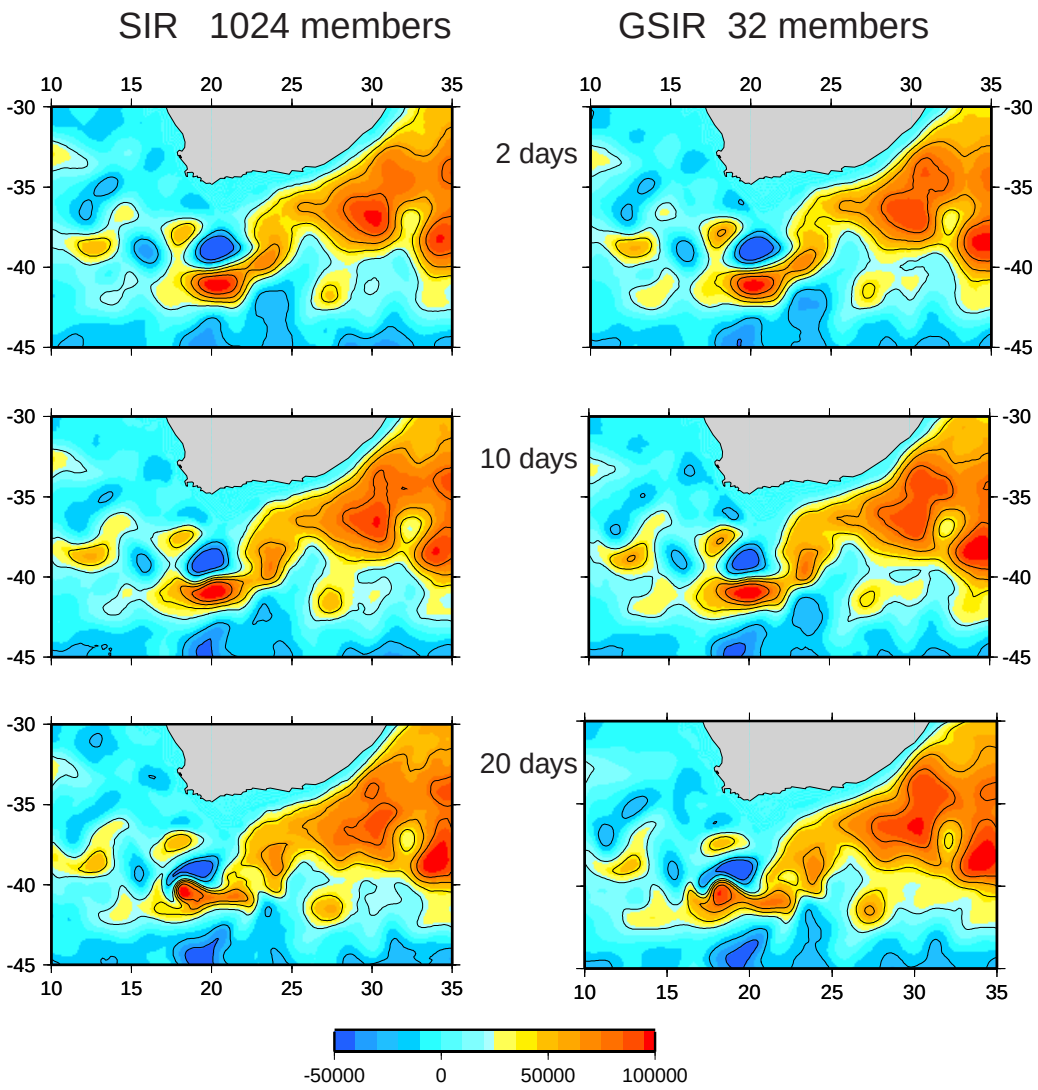

Figure 10: Upper-layer streamfunction for SIR with 1024 members (left) and GSIR with 32 members on day 2, 10 and 20.

At day 10 the differences are still minor, and even at day 20 the GSIR behaves quite well. The difference between the SIR and the GSIR is significant in the whole domain (measured using the variance of the SIR), but only slightly so. Only in a small area just west of the retroflection area, so in the most nonlinear part of the model domain, larger differences occur (around 17.5E, 41S). The conclusion from these runs is that the GSIR is doing quite a good job. One could define an error measure and obtain a more objective conclusion (although the error measure has to be subjectively chosen first), but I refrain from doing that here. The purpose of the present exercise is to show that the GSIR is promising in being able to reduce the ensemble size significantly,

and so to draw the SIR idea towards operationally possible.

5 Summary and discussion

From Bayes theorem we learned that data assimilation is not an inverse problem is its purest form. Since inverse problems are hard to solve it makes sense to try to avoid them, and one example is presented here. Because we stay that close to the basic formulation of the data assimilation problem, the only approximation needed is the finite ensemble size. In principle this is something that can be controlled by the experimenter, but, of course, the size of the problem encountered today sets serious limitations on this ensemble size. An economic problem is that the convergence rate of ensemble members is rather low, it goes like $1/\sqrt{N}$ in which N is the ensemble size. Bearing this in mind, the advantages of sequential importance resampling are impressive:

1. no linearizations of the model equations needed
2. a fully nonlinear measurement operator can be handled without any problem
3. no inversions needed
4. always balanced model states
5. higher-order moments of the pdf (like error variance) readily available
6. (non-Gaussian) model dynamics errors easily included

To compare with conventional methods: Kalman-filter-like methods fail on points 3, and 4, while the EKF fails on all. So-called adjoint methods like 4DVAR fail on 1, 2, 3, 5 and 6. The representer methods can handle Gaussian model dynamics errors, but performs the same as 4DVAR on the other points. This shows that the potential of the method is quite large. One might think that several points are not of interest, like 2 and non-Gaussian model dynamics errors in 6, but when science progresses there is little doubt that we have to deal with them. It is stresses however that the SIR should not be oversold: the large ensemble size prevents its every-day use.

By studying the performance of the SIR we encountered a few interesting new features. First, the pdf of the model can be quite non Gaussian, so that some data assimilation methods are not expected to perform well. And second, the variance of the estimated state can increase during the analysis, which is not possible when model and observations are Gaussian distributed. We found that a better measure of the uncertainty of the estimates is the entropy.

In order to try to reduce the ensemble size we turned to local updating, as done in present-day ensemble Kalman filter applications. It turns out that idea is not implemented straight forward in a SIR. We also found that the results deviated significantly from the original SIR in the oceanic example discussed here, even with only a factor 4 reduction is ensemble size. Some new ideas seem to be necessary here to get this to work, especially when working with a primitive equation model.

Another idea was to guide the ensemble towards future observations using the GSIR. Here very promising results have been achieved, while it can be shown that no extra approximations are made compared to the original SIR. Not withstanding this success, the method does still not allow new blood to enter the ensemble.

A possible way forward might be to try to combine the LSIR and the GSIR. Because of the problems of the LSIR with primitive equation models (gravity wave generation), it might be possible to change the random nature of the random forcing provided to the ensemble members at each time step. The idea is than to do a GSIR, but force the ensemble members with model dynamics errors that are partly random and partly local. The local part can be a forcing toward that ensemble member that is closest to the observations in that area. One could term this forcing state-vector nudging because the model state is forced towards another model state.

The difference with the observation nudging is that in that method model equivalents are nudged towards the observations, but it is unclear what to do with the rest of the state vector. That problem is not present in the above approach. A problem that remains is of course how strong this local nudging should be. Up to now, I have no solution. The best way is to try to relate what is been done to Bayes theorem. That theorem should than be used in its smoother form, e.g. the probability density is now a function of space and time, leading to a lagged filter to be practical. Research in this area is underway: these are exciting times!

References

[1] Anderson, J.L. An ensemble adjustment Kalman filter for data assimilation *Monthly Weather Rev.*, **129**, 2884-2903, 2001.

[2] Anderson, J.L., and S.L. Anderson, A Monte Carlo implementation of the nonlinear filtering problem to produce ensemble assimilations and forecasts *Monthly Weather Rev.*, **127**, 2741-2758, 1999.

[3] Bennett, A.F., em Inverse methods in Physical Oceanography Cambridge University Press, 346 pp., 1992.

[4] Bennett, A.F., B.S. Chua and L.M. Leslie, Generalized Inversion of a Global Numerical Weather Prediction Model, *Meteorol. Atmos. Phys.*, **60**, 165-178, 1996.

[5] Brasseur, P, J.M. Ballabrere, J. Verron, Assimilation of altimetric data in the mid-latitude oceans using the SEEK filter with an eddy-resolving primitive equation model *J. Mar. Syst.*, **4**, 269-294, 1999.

[6] Burgers, G., P. J. van Leeuwen, and G. Evensen, On the analysis scheme of the Ensemble Kalman Filter, *Monthly Weather Rev.*, **125**, 1719–1724, 1998.

[7] Doucet, A., N. de Freitas, N.J. Gordon (Eds.) *Sequential Monte Carlo methods in practice* Springer, New York, 581p., 2001.

[Drazin and Johnson, 1989] Drazin, P.G. and R.S. Johnson, *Solitons: an introduction* Cambridge University Press, 226pp, 1989.

[8] Evensen, G., Sequential data assimilation with a nonlinear quasi-geostrophic model using Monte Carlo methods to forecast error statistics, *J. Geophys. Res.*, **99**(C5), 10,143–10,162, 1994.

[9] Evensen, G., and P.J. van Leeuwen An ensemble Kalman smoother for nonlinear dynamics, *Mon. Weather Rev.*, **128**, 1852-1867, 2000.

[10] Jazwinski, A.H., *Stochastic processes and filtering theory*, Academic Press, New York, 376p., 1970.

[11] Gordon, N.J., D.J. Salmond, and A.F.M. Smith Novel approach to nonlinear/non-Gaussian Bayesian state estimation, *IEEE Proceedings-F*, **140**, 107-113, 1993.

[12] Heemink, A.W., M. Verlaan, and A.J. Segers Variance reduced ensemble Kalman filtering *Mon. Weather Rev.*, **129**, 1718-1728, 2001.

[13] Houtekamer, P.L. and H.L. Mitchell, Data assimilation using an ensemble Kalman filter technique, *Mon. Weather. Rev.*, **126**, 796-811, 1998.

[14] Houtekamer, P.L. and H.L. Mitchell, Reply *Mon. Weather. Rev.*, **127**, 1378-1379, 1999.

[15] Miller, R.N., E.F. Carter and S.T. Blue, Data assimilation into nonlinear stochastic models using the ensemble Kalman filter with a quasi-geostrophic model, *Tellus*, **51A**, 167-194, 1999.

[16] Pham T.P., Stochastic methods for sequential data assimilation in strongly nonlinear systems *Mon. Weather Rev.*, **129**, 1194-1207, 2001.

[17] Shannon, C.E. A mathematical theory of communication, *Bell Syst. Tech. J.*, **27**, 379-423, 1948.

[Silverman, 1986] Silverman, B.W., *Density estimation for statistics and data analysis* Chapman and Hall, 175 pp, 1986.

[18] Van Leeuwen, P.J. An variance-minimizing filter for large-scale applications, *Mon. Weather Rev.*, **131**, 2071-2084, 2003.

[19] Van Leeuwen, P. J., The time-mean circulation in the Agulhas region determined with the ensemble smoother, *J. Geophys. Res.*, **104**, 1393-1404, 1999a.

[20] Van Leeuwen, P.J. Comment on "Data assimilation using an ensemble Kalman filter technique", *Mon. Weather Rev.*, **127**, 1374-1377, 1999b.

[21] Van Leeuwen, P.J. An ensemble smoother with error estimates, *Mon. Weather Rev.*, **129**, 709-728, 2001.

[22] Van Leeuwen, P. J., and G. Evensen, Data assimilation and inverse problems in a probabilistic formulation, *Mon. Weather Rev.*, **124**, 2898–2913, 1996.

[23] Verlaan, M. and A.W. Heemink, Tidal flow forecasting using reduced rank square root filters *Stochastic Hydrology and Hydraulics*, **11**, 349-368, 1997.