

First steps towards the application of the Ensemble Transform Kalman Filter technique at the Hungarian Meteorological Service

Petra Csomós and Gergely Bölöni *

June 16, 2008

1 Introduction

In the ALADIN 3D-VAR data assimilation system, for the time being, background errors are considered to be constant in time, however we all know that background forecasts might be of variable quality depending on the weather situation. Kalman Filter is an approach to take into account the time dependency of background errors in the analysis by ensuring the time evolution of the background error covariance matrix from one analysis time step to another. The implementation of Kalman Filter, however, is extremely expensive in large dimension systems such as numerical weather prediction (NWP) models, which excludes the possibility of a real time operational application with the present computers. The so called Ensemble Kalman Filters aim to reduce the cost of computing the time evolution of the background error covariance matrix. They all estimate it from a small size ensemble of background forecasts but different approaches exist for the generation of initial perturbations for them. At the Hungarian Meteorological Service (HMS) the so called Ensemble Transform Kalman Filter (ETKF) has been chosen for implementation. Details about its theoretical background and practical implementation are presented in this paper.

In the following we summarize the basic concepts of data assimilation in general and Kalman Filter methods in particular. In Section 3 the theoretical background of ETKF is detailed, while in Section 4 its practical realization is shown.

2 Data assimilation and Kalman Filter

Data assimilation systems provide initial conditions or analysis (x_a) for NWP models using the actual observations (y) and a background forecast (x_f) valid at the analysis time as primary information for the analysis procedure. Optimal least-square and variational data assimilation methods compose this analysis according to the so-called BLUE (Best Linear Unbiased Estimation) estimation:

$$x_a = x_f + K(y - \mathcal{H}(x_f)) \quad (\text{BLUE})$$

where K is the Kalman gain:

$$K = P_f H^\top (H P_f H^\top + P_o)^{-1}.$$

In (BLUE), \mathcal{H} denotes the observation operator and H its linearized around the background state x_f ($H = \frac{\partial \mathcal{H}}{\partial x}(x_f)$). These operators enable the representation of the model state at the space of the observations. The P_f and P_o , background and observation error covariance matrices are defined as follows:

$$P_f := E(\varepsilon_f \varepsilon_f^\top) \quad \text{and} \quad P_o := E(\varepsilon_o \varepsilon_o^\top), \quad (1)$$

*Hungarian Meteorological Service, 1024 Budapest, Kitaibel Pál u. 1., Hungary, csomos.p@met.hu, boloni.g@met.hu

where $E(\cdot)$ denotes the expected value, $\varepsilon_f = x_t - x_f$ and $\varepsilon_o = \mathcal{H}(x_t) - y$ stand for the background and observation errors respectively where x_t is the unknown true state of the atmosphere. Further assumptions on the errors are that:

1. Each error is of expected value of zero, i.e. $E(\varepsilon_f) = E(\varepsilon_o) = 0$ (the corresponding estimate is unbiased, it does not contain systematic error).
2. Background and observation errors are uncorrelated, i.e. $E(\varepsilon_f \varepsilon_o) = 0$.

Similarly to (1) the error covariance matrix of the analysis can be defined as

$$P_a := E(\varepsilon_a \varepsilon_a^\top), \quad (2)$$

where $\varepsilon_a = x_t - x_a$. In an assimilation system based on (BLUE), the following relation holds between the analysis and the background error covariance matrices:

$$P_a = (I - KH)P_f, \quad (3)$$

where I denotes the identity matrix (Bouttier and Courtier [2]). Let us denote the degree of freedom of an NWP model by n ($n \approx 10^7$) and the number of observations by p ($p \approx 10^4 - 10^6$). Then x_f and x_a are vectors of size n , y is a vector of size p , the covariance matrices are of size $n \times n$, the Kalman gain K is a matrix of size $n \times p$, and the linearized observation operator H is a matrix of size $p \times n$.

The P_f background error covariance matrix is often assumed to be constant in time in the data assimilation systems, however, it is well known that background errors depend a lot on the actual weather situation, therefore it is desirable to release this consideration. The main idea of Kalman Filter methods is to update matrix P_f at each analysis step so that equations (BLUE) and (3) always hold. The ‘‘original’’ Kalman Filter method was introduced by Rudolf Emil Kalman in 1960 (see Kalman [9]). He has shown that taking into account the same assumptions as above, the following relation can be obtained between the covariance matrices P_f and P_a (see Kalman [9] and Kalnay [10, Chapter/Section 5.6]):

$$P_f^{k+1} = M_k^{k+1} P_a^k M_k^{k+1\top} + P_M^{k+1}, \quad (\text{KF})$$

where index k denotes the k^{th} analysis step, M is the linearized propagator of the nonlinear model \mathcal{M} (i.e. a matrix of size $n \times n$) and P_M is the covariance matrix of the linear model error (contains the linearization error as well). The P_M matrix is usually neglected in practice, however, there are also efforts on its estimate (see e.g. Tremolet [11]).

However the following problems arise for the realization of the algorithm (KF):

- The original Kalman Filter is only applicable for linear $\mathcal{M} = M$ model. In order to extend the method to nonlinear models, the Extended Kalman Filter has been introduced, where the nonlinear operator \mathcal{M} is assumed to be linearized around the background state.
- Since the degree of freedom of atmospheric models is $n \approx 10^7$, the computation of (BLUE) cannot be performed directly. Thus, the variational algorithms have been introduced where the Kalman gain matrix K is constructed implicitly (Courtier et al. [3]). Therefore, K is not available explicitly for computing the new error covariance matrix P_f from formula (KF).
- Determination of matrix P_f according to formula (KF) would need $n \approx 10^7$ integration with the tangent linear model and its adjoint (which is the transpose of the tangent linear model operator).

Several approximate methods have been proposed in order to by-pass these implementation problems. They might come from different approaches but all of them aim to reduce the dimensions used in the the Kalman Filter equation (KF). The so-called Reduced Rank Kalman Filter developed at ECMWF uses Hessian singular vectors to define a low-dimensional subspace where the P_f matrix is evolved (Fisher [4]). Other approximate Kalman Filter methods are based on the ensemble technique where the low-dimensional subspace is defined by a small size ensemble (Ensemble Kalman Filter (EnKF) and Ensemble Transform Kalman Filter (ETKF)).

3 Ensemble techniques for Kalman Filtering

Furthermore we focus on the Ensemble Kalman Filter and the Ensemble Transform Kalman Filter methods more in details.

3.1 Ensemble Kalman Filter (EnKF)

In the case of EnKF the error covariance matrices P_a and P_f are obtained from k ($k \ll n$) *ensemble* members (i.e. from a statistical population). Matrix P_a can be estimated for k ensemble members as follows (see Houtekamer and Mitchell [8], Evensen [6, Chapter 2.4.2], Gillijns et al. [5]):

$$P_a \approx \frac{1}{k-1} \sum_{j=1}^k (x_{a,j} - \bar{x}_a)(x_{a,j} - \bar{x}_a)^\top. \quad (4)$$

Let us consider the following

$$Z_a = \frac{1}{\sqrt{k-1}} (z_{a,1}, z_{a,2}, \dots, z_{a,k})$$

matrix of size $n \times k$, where $z_{a,j} := x_{a,j} - \bar{x}_a$ ($j = 1, \dots, k$) are the analysis dispersions, i.e. the differences between the j^{th} member $x_{a,j}$ and the ensemble average \bar{x}_a . Observe that with this notation the covariance matrix P_a of the analysis error can be written from (4) as:

$$P_a = Z_a Z_a^\top, \quad (5)$$

which is the product of matrix Z_a and its transpose. Then one can assume that the background dispersions are obtained by integrating the analysis dispersions with the linear model, that is,

$$Z_f = M Z_a \quad (\text{EnKF})$$

and then

$$P_f = Z_f Z_f^\top = M Z_a Z_a^\top M^\top = M P_a M^\top,$$

hence, we get back the formula (KF) of the Kalman Filter (neglecting the model error's covariance matrix P_M). The advantage of this method is that only k integrations are needed in formula (EnKF). It is remarked that the Ensemble Kalman Filter is one of the so-called *square-root filters*, because Z_a is the mathematical square-root of P_a according to formula (5) (see Tippett et al. [12]). Since

$$\frac{1}{k} \sum_{j=1}^k z_{f,j} = \frac{1}{k} \sum_{j=1}^k (x_{f,j} - \bar{x}_f) = \frac{1}{k} \sum_{j=1}^k x_{f,j} - \frac{1}{k} \sum_{j=1}^k \bar{x}_f = \bar{x}_f - \frac{1}{k} k \bar{x}_f = 0,$$

the average of the dispersions $z_{f,j}$ is zero, hence, they are linearly dependent. Therefore, the rank of matrix P_f is only $k-1$. This means that we estimate matrix P_f of size $n \times n \approx 10^7 \times 10^7$ by a matrix with "information" of $k-1$ (this phenomenon is usually called rank deficiency). It can be shown that EnKF method is optimal in the sense of formula (3) only if the observations are also perturbed. It is also mentioned here that there are also other consequences as well, for instance, the collapse of ensemble members, sampling noise due to the small number of ensemble members, long distance correlation between them, etc.

3.2 Ensemble Transform Kalman Filter (ETKF)

The advantage of ETKF is that, unlike the Ensemble Kalman Filter, the observations are not to be perturbed in order to get an optimal estimate. The basic idea behind ETKF is that there exists a relationship between the dispersions of the analysis and the dispersions of the background, that is

$$Z_a = Z_f T, \quad (6)$$

where T denotes the (for the time being unknown) transformation matrix of size $k \times k$ describing this relationship. Matrix T can be determined from the following assumptions:

- the method is optimal in the sense of formula (3),
- the matrix of the dispersions is the square-root of the background error covariance matrix.

Hence, we are seeking matrix T under the following assumptions:

$$\begin{cases} Z_a = Z_f T, \\ P_a = (I - KH)P_f, \quad \text{where } K = P_f H^\top (H P_f H^\top + P_o)^{-1}, \\ P_f = Z_f Z_f^\top. \end{cases} \quad (\text{ETKF})$$

From formulae (ETKF) matrix T can be determined as (for the derivation see in Bishop et al. [1]):

$$T = C(\Gamma + I)^{-1/2}, \quad (7)$$

where I is the identity matrix and

$$Z_f^\top H^\top P_o^{-1} H Z_f = C \Gamma C^\top.$$

Thus, matrix C contains the normalized eigenvectors of the matrix $Z_f^\top H^\top P_o^{-1} H Z_f$, and the diagonal Γ matrix contains the corresponding eigenvalues. Matrix Z_a defined by formula (ETKF) contains the dispersions to be added to the analysis x_a computed from the control member (e.g. $x_{f,1}$). The new background members $x_{f,j}^{\text{new}}$ are determined by integrating the new analysis “ensemble” members $x_{a,j}$ with the model ($j = 1, \dots, k$). From their dispersions the new values of matrices P_f and P_a can be computed, and so on. The algorithm is shown in *Figure 1*.

Let us make the remark that the ETKF method has also a great significance in the field of observation targeting due to its potential to compute a future analysis error covariance matrix through the transform (6) and the estimation (5). It namely makes possible to minimize the analysis and the forecast errors of a later time instant (called respectively the targeting and the verification time) with respect to the assimilation of possible adaptive elements of the observing system (drop sondes, aircrafts, etc.).

4 Realization of Ensemble Transform Kalman Filter at the Hungarian Meteorological Service

In this section we give an overview on the most important features about the practical realization of Ensemble Transform Kalman Filter at our Service in the framework of the ALADIN 3D-VAR data assimilation system. We remark and warmly acknowledge that the idea behind the whole realization of ETKF at the HMS was originally suggested by Sándor Kertész. Following the algorithm of ETKF shown in *Figure 1*, one should perform the following steps:

1. Build matrix $Z_f^\top H^\top P_o^{-1} H Z_f$
2. Solve the eigenvalue problem $Z_f^\top H^\top P_o^{-1} H Z_f = C \Gamma C^\top$ (normalize the eigenvectors)
3. Create transformation matrix $T = C(\Gamma + I)^{-1/2}$
4. Construct analysis dispersions with $Z_a = Z_f T$
5. Compute new analysis members with $x_{a,j} = x_a + z_{a,j}$, $j = 1, \dots, k$
6. Generate new ensemble members with $x_{f,j}^{\text{new}} = \mathcal{M}(x_{a,j})$, $j = 1, \dots, k$

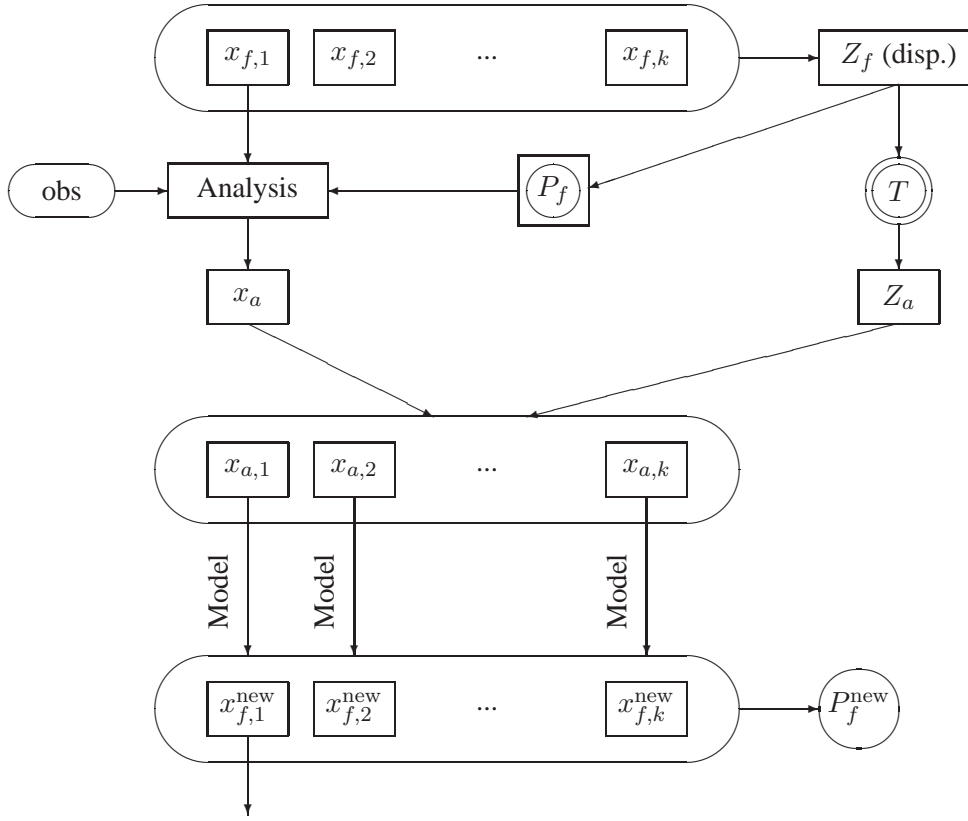


Figure 1: Algorithm of Ensemble Transform Kalman Filter (using analysis x_a computed from the control member $x_{f,1}$).

7. Compute the new error covariance matrix P_f^{new} .

Hereafter the practical realisation of these steps will be briefly explained with special emphasis on the technical details of the ALADIN data assimilation system.

4.1 Computation of matrix $Z_f^\top H^\top P_o^{-1} H Z_f$

Let us observe that matrix $Z_f^\top H^\top P_o^{-1} H Z_f$ can be written as

$$Z_f^\top H^\top P_o^{-1} H Z_f = V^\top V \quad \text{with} \quad V := P_o^{-1/2} H Z_f, \quad (8)$$

where matrix V is of size $p \times k$. Matrix P_o is diagonal containing the variations (i.e. the squares of the standard deviations) of the observation error. Matrix $H Z_f$ can be written as follows:

$$H Z_f = H \frac{1}{\sqrt{k-1}} (z_{f,1}, z_{f,2}, \dots, z_{f,k}) = \frac{1}{\sqrt{k-1}} (H z_{f,1}, H z_{f,2}, \dots, H z_{f,k}),$$

where the terms $H z_{f,j}$ ($j = 1, \dots, k$) can be estimated as

$$H z_{f,j} = H(x_{f,j} - \bar{x}_f) \approx \mathcal{H}(x_{f,j}) - \mathcal{H}(\bar{x}_f).$$

$\mathcal{H}(x_{f,j})$ and $\mathcal{H}(\bar{x}_f)$ can be written as the difference between the observation and the so-called first guess departure (i.e. the difference between the observation and the background) (Kertész, personal

communication). These values are stored in the ALADIN observation database (ODB) by names `obsvalue` and `fg_depar`. Therefore, one can further write that

$$\mathcal{H}(x_{f,j}) = y - (y - \mathcal{H}(x_{f,j})) = \text{obsvalue} - \text{fg_depar}.$$

Then the entries of matrix V are

$$\frac{1}{\sqrt{\sigma_j^2}} (\mathcal{H}(x_{f,j}) - \mathcal{H}(\bar{x}_f)),$$

where σ_j denotes the j^{th} entry of matrix P_o , that is, the standard deviation of the j^{th} observation error stored in the ODB as `obs_error`. Thus, the above formula can be written using the variables of the ODB as

$$\begin{aligned} & \underbrace{[(\text{obsvalue} - \text{fg_depar}) - (\text{obsvalue} - \text{fg_depar})]}_{j^{\text{th}} \text{ ens. member}} / \underbrace{(\text{obsvalue} - \text{fg_depar})}_{\text{average}} / \underbrace{\text{obs_error}}_{j^{\text{th}} \text{ ens. member}} \\ & = \underbrace{[(\text{fg_depar}) - (\text{fg_depar})]}_{\text{average}} / \underbrace{(\text{fg_depar})}_{j^{\text{th}} \text{ ens. member}} / \underbrace{\text{obs_error}}_{j^{\text{th}} \text{ ens. member}}. \end{aligned}$$

According to the above formula, k new files, representing the k rows of matrix $V = P_o^{-1/2} H Z_f$, are constructed in the following way:

$$\begin{aligned} V(i, j) &= (P_o^{-1/2} H Z_f)(i, j) = \\ &= \frac{1}{\sqrt{k-1}} \frac{1}{\sigma_i} [H(x_j - \bar{x})](i) \\ &= \frac{1}{\sqrt{k-1}} \frac{1}{\text{obs_error}(i)} [\text{fg_depar}_{\bar{x}}(i) - \text{fg_depar}_{x_j}(i)], \end{aligned} \quad (9)$$

where the values `fg_depar` and `obs_error` are to be obtained from the ODB.

The content of ODB can be read with the so-called `odbviewer` which is based on `sql` requests defining the parameters we are interested in (in our case `fg_depar` and `obs_error`). The output of `odbviewer` is a simple `ascii` file containing the requested values. The `fg_depar` and `obs_error` variables in ODB are filled during the SCREENING, i.e. the quality control of the observations. This means that in order to get all the necessary information, the SCREENING procedure should be run $k + 1$ times taking the ensemble members $x_{f,j}$ and their average \bar{x}_f as background fields. Having a look at matrix V defined in (8), one expects it to be computed only for those measurements which were accepted by the quality control (i.e. active measurements) of each SCREENING runs. Consequently, we are interested only in the `fg_depar` values corresponding to the intersection of the active measurements. As a summary, for the computation of matrix $Z_f^T H^T P_o^{-1} H Z_f$ the following subsequent steps should be done:

1. Run the SCREENING taking $x_{f,j}$ ($j = 1, \dots, k$) and \bar{x}_f as background fields ($k + 1$ SCREENINGS altogether)
2. Read each ODB resulted from each SCREENING (use `odbviewer` and `sql` request files)
3. Compare the ODB REPORTS and collect the observations which appear in all of them
4. Save the values of `fg_depar` and `obs_error` only for these observations for each background field $x_{f,j}$ and \bar{x}_f (this results $k + 1$ files)
5. With the help of the above files compute $V(i, j)$ from formula (9) and save the k columns of matrix V in k different files.

Then matrix $V^T V$ should be computed. Using the definition of the transpose

$$V^T(i, j) := V(j, i),$$

the (i, j) th entry of matrix $V^T V$ can be determined as

$$(V^T V)(i, j) = \sum_{l=1}^N V^T(i, l) V(l, j) = \sum_{l=1}^N V(l, i) V(l, j),$$

where N denotes the size of the files containing the columns of V , i.e. the number of the common measurements. We should add the product of the l th line of the i th and j th files. The result can be written to a new file, which now contains the matrix $V^T V = Z_f^T H^T P_o^{-1} H Z_f$ needed for the eigenvalue problem.

4.2 Solving the eigenvalue problem

The matrix $Z_f^T H^T P_o^{-1} H Z_f$ is of size $k \times k$ and it is real-valued and symmetric, therefore, its eigenvalues are real as well. In ALADIN we can use subroutine RG.F for solving the eigenvalue problem. A FORTRAN code should be written which reads the data from the file containing the elements of matrix $Z_f^T H^T P_o^{-1} H Z_f$ to an array and then calls RG.F in order to determine the eigenvectors and eigenvalues. We call the attention that the definition of the transformation matrix refers to C and Γ as matrices obtained from formula

$$Z_f^T H^T P_o^{-1} H Z_f = C \Gamma C^T,$$

however, routine RG.F solves the eigenvalue problem:

$$Z_f^T H^T P_o^{-1} H Z_f = C \Gamma C^{-1}.$$

This means that in order to be consistent with the derivation of the transformation matrix T , the equality $C^T = C^{-1}$ should hold, that is, C should be an orthogonal matrix. Hence, the columns of C (i.e. the eigenvectors of matrix $Z_f^T H^T P_o^{-1} H Z_f$) should build an orthonormal system. Therefore, columns of C should be normalized after calling RG.F.

4.3 Derivation of the transformation matrix

Although the transformation matrix T was initially proved to be computed as

$$T = C(\Gamma + I)^{-1/2},$$

we will use another form suggested in Wei et al. [13]. They showed that the analysis dispersions $z_{a,j}$ are not centered about their mean, not even when the background dispersions $z_{f,j}$ are centered about their mean. In order to center $z_{a,j}$ about their mean, a so-called simplex can be applied. They also showed that a possible simplex, having the necessary properties, is the matrix C^T . Hence, matrix T can be obtained as

$$T = C(\Gamma + I)^{-1/2} C^T,$$

where C denotes the matrix of the normalized eigenvectors, while Γ the matrix of eigenvalues, and I is the identity matrix. The latter two are diagonal, therefore, the power of their sum can be obtained easily. The (i, j) th entry of matrix T can be written as

$$T(i, j) = \sum_{l=1}^N \frac{1}{\sqrt{\lambda_l + 1}} v_l(i) v_l(j), \quad (10)$$

where λ_l is the l th eigenvalue and $v_l(i)$ is the i th entry of the l th normalized eigenvector ($i, j = 1, \dots, k$ and N denotes the length of the files). From the point of view of the realization this means that the FORTRAN program (i) solves the eigenvalue problem in order to determine the eigenvectors and eigenvalues of matrix $Z_f^T H^T P_o^{-1} H Z_f$, and (ii) computes the transformation matrix T defined by formula (10) from the above obtained eigenvectors and eigenvalues.

4.4 Construction of the analysis dispersions

The next step is the computation of matrix Z_a of the analysis dispersions by the formula

$$Z_a = Z_f T,$$

with

$$Z_f = \frac{1}{\sqrt{k-1}} (z_{f,1}, z_{f,2}, \dots, z_{f,k}), \quad (11)$$

where $z_{f,j} := x_{f,j} - \bar{x}_f$ ($j = 1, \dots, k$). Let us observe that the constant $\frac{1}{\sqrt{k-1}}$ is on both sides of equation (11), because it appears in both matrices Z_f and Z_a . Dividing both sides of the equation by this constant, we obtain that

$$(z_{a,1}, z_{a,2}, \dots, z_{a,k}) = (z_{f,1}, z_{f,2}, \dots, z_{f,k}) T,$$

that is,

$$z_{a,j}(i) = \sum_{l=1}^k z_{f,l}(i) T(l, j) \quad \text{for all } i, j = 1, \dots, k, \quad (12)$$

where $z_{a,j}(i)$ denotes the i^{th} entry of the j^{th} analysis dispersion vector. Formula (12) can be written as follows:

$$\begin{aligned} z_{a,j}(1) &= z_{f,1}(1)T(1, j) + z_{f,2}(1)T(2, j) + \dots + z_{f,k}(1)T(k, j) \\ z_{a,j}(2) &= z_{f,1}(2)T(1, j) + z_{f,2}(2)T(2, j) + \dots + z_{f,k}(2)T(k, j) \\ &\vdots \\ \underbrace{z_{a,j}(n)}_{z_{a,j}} &= \underbrace{z_{f,1}(n)T(1, j)}_{z_{f,1}T(1,j)} + \underbrace{z_{f,2}(n)T(2, j)}_{z_{f,2}T(2,j)} + \dots + \underbrace{z_{f,k}(n)T(k, j)}_{z_{f,k}T(k,j)}. \end{aligned}$$

One can see that it is enough to multiply all elements in files $z_{f,j}$ ($j = 1, \dots, k$) by the entries of the transformation matrix T in order to get the analysis dispersion files $z_{a,j}$ ($j = 1, \dots, k$). In order to fasten this procedure, each script computing the j^{th} analysis dispersion can be run on a different processor (that is, all scripts at the same time), since they are independent from each other. It is remarked that one can obtain the average \bar{x}_f and the differences from it (i.e. the background dispersions $z_{f,j}$) from the available background fields $x_{f,j}$ for instance by using a program similar to the subroutine BLEND.F, which is a simple routine for adding two ALADIN files.

4.5 Computation of the new values

New analysis members. After determining the analysis dispersions $z_{f,j}$, the new analysis members are obtained by

$$x_{a,j} = x_a + z_{a,j}, \quad j = 1, \dots, k$$

where x_a denotes the analysis field computed by a data assimilation method (e.g. 3D-VAR) from the control ensemble member (e.g. from $x_{f,1}$).

New ensemble members. The new background fields are obtained from integrating the new analysis ensemble members $x_{a,j}$ with the model \mathcal{M} as

$$x_{f,j}^{k+1} = \mathcal{M}(x_{a,j}^k) \quad \text{for all } j = 1, \dots, k.$$

New background error covariance matrix (P_f). A new sample of background errors can be estimated as the difference of the ensemble members from their mean for instance:

$$\delta_j = \bar{x}_f - x_{f,j}, \quad j = 1, \dots, k.$$

or, also as the differences between the ensemble members themselves:

$$\tilde{\delta}_j = x_{f,i} - x_{f,l}, \quad i = 1, \dots, k \text{ and } l = 1, \dots, k.$$

The P_f background error statistics needed for the ALADIN 3D-VAR run can be then computed by the FESTAT program, which was developed at ECMWF and then adapted to ALADIN (available from Météo-France or the Hungarian Meteorological Service).

5 Summary

In the present paper we showed the theoretical background and the practical realization of Ensemble Transform Kalman Filter method as suggested by Sándor Kertész at the Hungarian Meteorological Service. The big advantage of ETKF is that its application results time dependent background errors and also an ensemble system at each analysis time without the replacement of the already existing data assimilation method (e.g. 3D-VAR). The main difference between the computational time of EnKF and this kind of realization of ETKF is the number of minimizations needed. Meanwhile in ETKF only one analysis step (e.g. 3D-VAR) is needed, in EnKF an analysis should be obtained from each background field (k altogether). On the other hand, in ETKF we should run $k + 1$ SCREENINGS, which means that we could "only" save the time of the k minimization steps (which is still meaningful). The heart of ETKF is the transformation matrix, which should be computed in addition compared to EnKF on the other hand. However its derivation, is still less computer intensive than a 3D-VAR minimization.

At the moment only the basic realization of ETKF is ready at HMS, we do not have any cycling yet and therefore we are unable to show any results. Nevertheless, some open problems can be mentioned: How does the sampling noise, related to the small size of ensemble, and the choice of the same lateral boundary information effect the result obtained by ETKF? In order to answer these (and other) open issues, we plan to implement a real cycling with ETKF and investigate the background error statistics with the help of diagnostics.

Acknowledgment

The authors are indebted to Sándor Kertész for having shared them his idea about the realization of ETKF within the framework of ALADIN, and for the lot of help he gave them during the work. We also thank András Horányi for his continuous encouragement, and Claude Fischer for the very useful discussions during the stay of P. Csomós at Météo-France in Toulouse.

References

- [1] C. H. Bishop, B. J. Etherton, and S. J. Majumdar, Adaptive sampling with the Ensemble Kalman Transform Filter. Part I: Theoretical aspects, *Monthly Weather Review* **129**, 420–436 (2001)
- [2] F. Bouttier and Ph. Courtier, Data assimilation concepts and methods. *ECMWF Meteorological Training Course Lecture Series* (http://www.ecmwf.int/newsevents/training/lecture_notes/pdf_files/ASSIM/Ass_cons.pdf) (1999)

- [3] Ph. Courtier, E. Andersson, W. Heckley, J. Pailleux, D. Vasiljevic, M. Hamrud, A. Hollingsworth, F. Rabier and M. Fisher, The ECMWF implementation of three-dimensional variational assimilation (3D-Var). Part 1: formulation. *Quarterly Journal of the Royal Meteorological Society* **124**, 1783–1807 (1998)
- [4] M. Fisher, Assimilation Techniques (5): Approximate Kalman Filters and Singular Vectors, *ECMWF Meteorological Training Course Lecture Series* (http://www.ecmwf.int/newsevents/training/lecture_notes/pdf_files/ASSIM/Kalman.pdf) (2001)
- [5] S. Gillijns, O. Barrero Mendoza, J. Chandrasekar, B. L. R. De Moor, D. S. Bernstein, and A. Ridley, What is the ensemble Kalman filter and how does it work?, *Proc. 2006 American Control Conference*, Minneapolis, Minnesota, USA (June 14–16, 2006), 4448–4453 (2006)
- [6] G. Evensen, *Data Assimilation, The Ensemble Kalman Filter*, Springer–Verlag, Berlin, pp 275 (2007)
- [7] P. L. Houtekamer and H. L. Mitchell, Data assimilation using ensemble Kalman filter technique, *Monthly Weather Review* **126**, 796–811 (1998)
- [8] P. L. Houtekamer and H. L. Mitchell, A sequential ensemble Kalman filter for atmospheric data assimilation, *Monthly Weather Review* **129**, 123–137 (2001)
- [9] R. E. Kalman, A new approach to linear filtering and prediction problems, *ASME Journal of Basic Engineering* **82**, 35–45 (1960)
- [10] E. Kalnay Atmospheric Modeling, Data Assimilation and Predictability, *Cambridge University Press* , pp 341 (2003)
- [11] Y. Trémolet, Model-error estimation in 4D-Var, *Quarterly Journal of the Royal Meteorological Society* **133**, 1267–1280 (2007)
- [12] M. K. Tippett, J. L. Anderson, C. H. Bishop, T. M. Hamill, and J. S. Whitaker, Ensemble square root filters, *Monthly Weather Review* **131**, 1485–1490 (2003)
- [13] M. Wei, Z. Tóth, R. Wobus, Y. Zhu, and C. H. Bishop, Ensemble Transform Kalman Filter-based ensemble perturbations in an operational global prediction system at NCEP, *Tellus* **58A**, 28–44 (2006)