

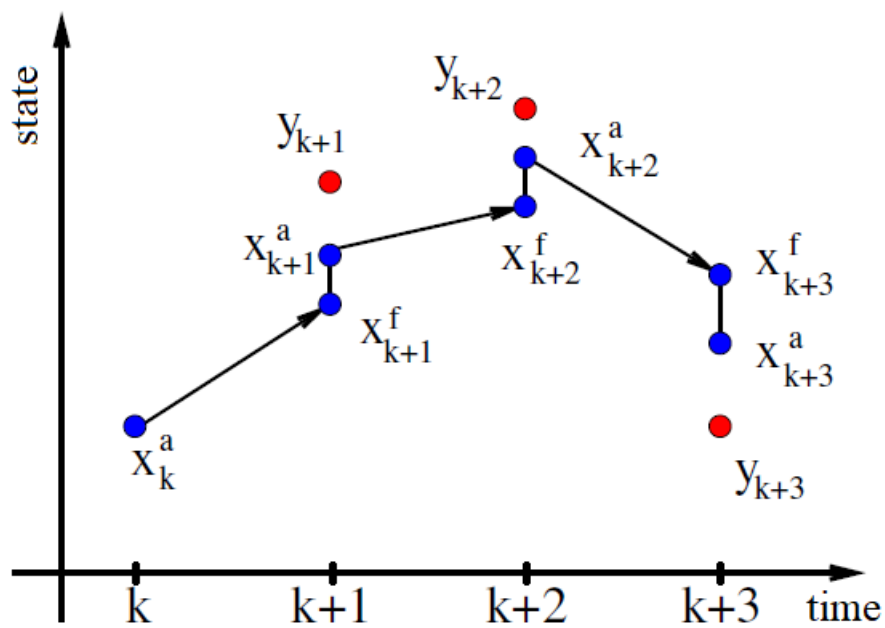
Finite-size Ensemble Transform Kalman Filter

Gido Limperg

July 3, 2018

Bachelor thesis

Supervisor: dr. Svetlana Dubinkina



Korteweg-de Vries Instituut voor Wiskunde
Faculteit der Natuurwetenschappen, Wiskunde en Informatica
Universiteit van Amsterdam



Abstract

Data assimilation is an important tool used to improve forecasts of chaotic dynamical systems such as weather systems. This is a process in which theoretical knowledge of the system state is combined with observational data, in order to improve the state estimate. In this thesis, different methods of data assimilation are investigated.

The first method we discuss is the Ensemble Transform Kalman Filter (ETKF), which uses an ensemble of estimates to approximate the true state of the system. The ETKF uses a prior Gaussian probability density function (pdf) defined by the sample mean and the sample error covariance matrix of the ensemble. A disadvantage of the ETKF is that large sampling errors arise for low ensemble sizes. As a consequence, the ETKF requires methods called inflation and localisation to prevent filter divergence.

The second method is called the Finite-size Ensemble Transform Kalman Filter (ETKF-N). This method notices a loss of information when basing the prior pdf on the sample mean and sample error covariance, as with ETKF. Instead, the prior pdf is set to be conditional on the entire ensemble. This choice should diminish the sampling errors of the ETKF and hence eliminate the need for inflation.

Both methods are tested on the Lorenz '63 and Lorenz '95 models, by comparing the methods to an artificial truth. In addition to comparing the errors of both methods, various inflation factors are also tested, to determine whether or not ETKF-N requires inflation for convergence. The results show that ETKF-N outperforms ETKF for most ensemble sizes for both models. Furthermore, inflation is not necessary for convergence of ETKF-N when the ensemble size is high enough. However, ETKF-N still requires localisation for convergence for low ensemble sizes, just like the regular ETKF.

Title: Finite-size Ensemble Transform Kalman Filter
Authors: Gido Limperg, glimperg@live.nl, 11060727
Supervisor: dr. Svetlana Dubinkina
Second grader: prof. dr. Rob Stevenson
Date: July 3, 2018

Korteweg-de Vries Instituut voor Wiskunde
Universiteit van Amsterdam
Science Park 904, 1098 XH Amsterdam
<http://www.science.uva.nl/math>

Contents

1	Introduction	4
2	Data Assimilation	6
3	Ensemble Transform Kalman Filter	7
3.1	General setup	7
3.2	Coordinate change	8
3.3	Undersampling	10
3.3.1	Methods to deal with undersampling	10
4	Finite-size Ensemble Transform Kalman Filter	12
4.1	Background distribution	12
4.2	Coordinate change	16
4.3	Minimisation	17
5	Numerical experiments	19
5.1	Lorenz '63 model	19
5.1.1	Results	21
5.2	Lorenz '96 model	22
5.2.1	Results	22
5.2.2	Inflation diagrams	23
6	Conclusion and discussion	25
	Bibliography	27
	Popular summary	29

1 Introduction

Analysis of complicated systems such as the weather is an important part of applied mathematics. In particular, we are interested in predicting the future state of such systems. To this end, we can use a numerical model for the system state together with an a priori estimate. However, such predictions will often quickly diverge from the true state of the system. This is because the systems we are interested in are chaotic, meaning that small errors in our initial estimate will eventually lead to large errors in the system state. Instead of using a model, another possibility for predicting the future state would be to use previous observations of the state. However, this method has several drawbacks as well. For example, it's possible that we can only observe part of the system state.

A possible solution to these problems comes through the use of *data assimilation*. Data assimilation is a process in which theoretical knowledge of the system state is combined with observational data, in order to improve the state estimate. This is a sequential process which uses previous data assimilation results as well as observations to obtain a state estimate. After performing data assimilation, we can apply our model to the improved estimate, giving us a forecast for the future state. In turn, this forecast can be used as an initial estimate in future data assimilation. We will discuss data assimilation in more detail in chapter 2.

Data assimilation is a widely used method in fields such as numerical weather prediction (NWP) and the geosciences. Over the past few decades, data assimilation has become an essential component of numerical weather prediction [1]. In this thesis, we study different methods of data assimilation. The data assimilation methods that we will discuss in this thesis are based on the *Ensemble Kalman Filters* (EnKF), introduced in 1994 by the Norwegian mathematician Geir Evensen [2]. One example of a meteorological institute where the EnKF is used in practice is Environment Canada [3]. Another popular method is four-dimensional variational assimilation (4D-Var). This method is used in many NWP centres [1] such as the UK Met Office [4].

One type of EnKF is the *Ensemble Transform Kalman Filter* (ETKF), developed by Brian R. Hunt in 2007 [5]. In chapter 3 we will discuss this method. We will first derive a general cost function that is used in some EnKF variants, by writing out a certain probability density function. We then apply a coordinate change and find the minimum of this cost function. Next, we give an algorithm for the ETKF. We conclude chapter 3 by explaining a major drawback to the ETKF method. Due to certain *sampling errors*, an additional process is required, called *inflation*.

In chapter 4, we discuss an extension of the ETKF method, called the *Finite-size Ensemble Transform Kalman Filter* (ETKF-N). This method was introduced by Marc Bocquet in 2011 [6]. The ETKF-N method notes the sampling errors that come with ETKF, and tries to eliminate them by choosing a different prior probability density function. This should theoretically solve the need for inflation of the ETKF method. The rest of the method is similar to the regular ETKF, but there are a lot more subtleties that arrive when deriving the cost function, as well as when finding the minimum of the cost function.

We perform various numerical experiments comparing the two methods in chapter 5. We apply both methods to the Lorenz '63 and Lorenz '96 models, comparing the average *root mean square errors*. We also test different inflation factors on both methods, to determine the influence of inflation on performance.

The aim of this thesis is to compare the ETKF and ETKF-N methods, and determine which of the two methods has the best performance when applied to the models. Furthermore, we investigate whether or not the ETKF-N method has eliminated the intrinsic need for inflation that is present in the ETKF method and other EnKF methods.

This thesis reproduces the results of Bocquet (2011) [6].

2 Data Assimilation

In the previous chapter, we introduced the topic of data assimilation and gave a global overview of the data assimilation process. More formally, we can describe our data assimilation scheme in the following way. First of all, our initial estimate at time t_k is denoted by x_k^b , the index b standing for *background*. Usually, this estimate is the outcome of a previous forecast. Along with our initial estimate, we have a vector of observations y_k at time t_k . By combining x_k^b and y_k in a certain way, we achieve a new state estimate x_k^a at time t_k , the index a standing for *analysis*. After the data assimilation step, we use our improved estimate x_k^a to forecast the system state at time t_{k+1} , obtaining x_{k+1}^b . We can then use this forecast as an initial estimate in future data assimilation.

Consider the following equation describing the system state at time t_{k+1} :

$$x_{k+1}^t = M(x_k^t). \quad (2.1)$$

Here $x_k^t \in \mathbb{R}^n$ is the true state of the system at time t_k , the index t standing for *truth*. Furthermore, $M: \mathbb{R}^n \rightarrow \mathbb{R}^n$ is called the *model* of the system. Note that the systems we are interested in are chaotic. Therefore, M is a nonlinear operator.

Likewise, we have an equation describing the observations of our system:

$$y_k = h(x_k^t) + v_k. \quad (2.2)$$

Here $y_k \in \mathbb{R}^p$ is the vector of observations at time t_k , where $p \leq n$. This is because it may not be possible to observe all n variables [7]. Also, the *observation operator* $h: \mathbb{R}^n \rightarrow \mathbb{R}^p$ is generally nonlinear. Finally, $v_k \in \mathbb{R}^p$ is called the *observation error*. It is assumed that $v_k \sim N(0, R_k)$, meaning that v_k is drawn from a Gaussian distribution with mean 0 and covariance R_k . Here $R_k \in \mathbb{R}^{p \times p}$ is a matrix, called the observation error covariance matrix.

Remark 2.1. Note that in the context of data assimilation, it will be impossible to compute the true system state. Therefore, x_k^t will be replaced by the analysis x_k^a in equation (2.1) for the computation of the forecast:

$$x_{k+1}^b = M(x_k^a).$$

3 Ensemble Transform Kalman Filter

The most important part data assimilation is the combination of the system state and the observations. There are many different ways to implement this idea. One class of data assimilation schemes is the class of *Ensemble Kalman Filters* (EnKF). In this chapter, we will consider a variant called the *Ensemble Transform Kalman Filter* (ETKF). To simplify the notation, we will drop the time index k from chapter 2 and instead consider only the background x^b and analysis x^a .

3.1 General setup

The core concept of the EnKF comes with the use of an *ensemble*, which is a statistical sample of estimates of a certain state. Our initial estimate is the *background ensemble* of m members $x_1^b, \dots, x_m^b \in \mathbb{R}^n$. It is assumed that these ensemble members are drawn from a multivariate Gaussian distribution with mean x^b and covariance matrix B , where x^b and B are unknown. In actuality, this assumption leads to an approximation, since the ensemble members are often retrieved from the previous forecast, which in general has a non-Gaussian distribution.

The background ensemble can also be represented by a probability density function $p(x)$ of a Gaussian distribution $N(\bar{x}^b, P^b)$. Here, \bar{x}^b is the sample background mean, defined as

$$\bar{x}^b = \frac{1}{m} \sum_{i=1}^m x_i^b. \quad (3.1)$$

Furthermore, P^b is called the sample background error covariance matrix, defined as

$$P^b = \frac{1}{m-1} X^b (X^b)^T \in \mathbb{R}^{n \times n}, \quad (3.2)$$

where X^b is called the *anomaly matrix*, defined by

$$X^b = [x_1^b - \bar{x}^b \mid x_2^b - \bar{x}^b \mid \dots \mid x_m^b - \bar{x}^b] \in \mathbb{R}^{n \times m}. \quad (3.3)$$

Notice that each column in X^b represents the deviation of each ensemble member from the mean. The factor $\frac{1}{m-1}$ instead of $\frac{1}{m}$ in equation (3.2) is called *Bessel's correction*. The correction can be explained by observing the columns of X^b . The ensemble consists of m members, but since $\sum_{i=1}^m X_i^b = 0$, the space spanned by the columns of X^b only consists of $m-1$ linearly independent vectors. Therefore, the factor $\frac{1}{m-1}$ removes the bias in the estimation, whereas P^b would be biased with the factor $\frac{1}{m}$.

Remark 3.1. It is important to note that we used the sample mean \bar{x}^b and sample covariance matrix P^b for the distribution of $p(x)$. This choice is instead of the actual mean x^b and covariance matrix B of the ensemble. Naturally, this is because x^b and B are unknown. Although this seems like the natural choice, some information will be lost, since our empirical estimates \bar{x}^b and P^b generally differ from their exact counterparts x^b and B . This observation motivates the ETKF-N algorithm that we will discuss in chapter 4. See also section 3.3.

The goal of the ETKF is to find the analysis probability distribution $p(x|y)$ given our observations y and the background ensemble. Using this probability density function $p(x|y)$, we choose the analysis state x^a to be the mode of the function. Since we made assumptions of Gaussianity for $p(x)$, the mode of $p(x|y)$ is equal to its mean. Thus, maximising the distribution $p(x|y)$ will give us an analysis state x^a which should be an improved state estimate.

In order to compute this distribution, we will apply Bayes' rule:

$$p(x|y) = \frac{p(x)p(y|x)}{p(y)} \propto p(x)p(y|x), \quad (3.4)$$

where the symbol \propto means 'equal up to a constant of normalisation'. Now, we assume that $p(y|x)$ is the probability density of a Gaussian distribution $N(h(x), R)$, where h is the observation operator from (2.2) and R is the observation error covariance matrix. By explicitly writing out the probability density functions in (3.4), we obtain

$$p(x|y) \propto \exp(-J^b(x)) \exp(-J^o(x)) = \exp(-J(x)), \quad (3.5)$$

where

$$\begin{aligned} J(x) &= J^b(x) + J^o(x) \\ &= \frac{1}{2}(x - \bar{x}^b)^T (P^b)^{-1} (x - \bar{x}^b) + \frac{1}{2}(y - h(x))^T R^{-1} (y - h(x)). \end{aligned} \quad (3.6)$$

The function $J(x)$ defined in (3.6) is known as the *cost function*. Here, $J^b(x)$ is called the background term and $J^o(x)$ is the observation term. Now, by taking the logarithm, maximising $p(x|y)$ in (3.5) is equivalent to minimising $J(x)$. Thus, the analysis x^a is equal to the minimum of $J(x)$, meaning that x^a is the root of $\nabla J(x)$.

3.2 Coordinate change

In the previous paragraph, we glossed over an important fact when computing the cost function (3.6). Namely, P^b is not a full rank matrix when $m < n$, so P^b is in general not invertible. Therefore, our cost function $J(x)$ is in general not well-defined. However, $(P^b)^{-1}$ is well-defined on the *ensemble space* \mathcal{V} , a subspace of the state space.

Definition 3.2. The *ensemble space* \mathcal{V} is defined as the space spanned by the columns of the anomaly matrix X^b .

Since $J(x)$ is only well-defined on \mathcal{V} , a coordinate change is required. This is done by expressing the state vector x as the sum of the background mean \bar{x}^b and a linear combination of the background ensemble perturbations:

$$x = \bar{x}^b + X^b w, \quad (3.7)$$

where $w \in \mathbb{R}^m$ is to be determined. We also compute an approximation of the observations corresponding to the state vector x :

$$h(x) = h(\bar{x}^b + Xw) \approx h(\bar{x}^b) + Y^b w, \quad (3.8)$$

where Y^b is defined by

$$Y^b = [h(x_1^b) - \bar{y}^b \mid h(x_2^b) - \bar{y}^b \mid \dots \mid h(x_m^b) - \bar{y}^b], \quad (3.9)$$

with $\bar{y}^b = \frac{1}{m} \sum_{i=1}^m h(x_i^b)$. By definition, column i of Y^b represents the deviation of $h(x_i^b)$ from its ensemble mean.

By using (3.7) and (3.8), we can modify the cost function (3.6) to get a function on \mathcal{V} :

$$\tilde{J}(w) = \frac{1}{2}(m-1)w^T w + \frac{1}{2}(y - h(\bar{x}^b) - Y^b w)^T R^{-1}(y - h(\bar{x}^b) - Y^b w). \quad (3.10)$$

As explained in section 3.1, we want to minimise the cost function (3.6) to get the optimal analysis. It turns out that this is equivalent to minimising the function (3.10). Therefore, we want to solve

$$\nabla \tilde{J}(w) = (m-1)w - (Y^b)^T R^{-1}(y - h(\bar{x}^b) - Y^b w) = 0.$$

The solution of this equation is

$$w^a = \tilde{P}^a (Y^b)^T R^{-1}(y - h(\bar{x}^b)), \quad (3.11)$$

where

$$\tilde{P}^a = \left((m-1)I + (Y^b)^T R^{-1} Y^b \right)^{-1}. \quad (3.12)$$

Note that the analysis error covariance matrix \tilde{P}^a in ensemble space is equal to the inverse matrix of the Hessian of the cost function (3.10). The corresponding error covariance matrix in the original state space is $P^a = X^b \tilde{P}^a (X^b)^T$. Now, we find the mean of our analysis ensemble by inserting w^a into (3.7):

$$x^a = \bar{x}^b + X^b w^a.$$

Finally, we would like to obtain the analysis ensemble of system states x_1^a, \dots, x_m^a with mean x^a and error covariance matrix P^a . First we define W^a as the symmetric square root of $(m-1)\tilde{P}^a$ and W_i^a as the i th column of W^a . These columns W_i^a determine the analysis ensemble perturbations in ensemble space of each x_i^a . This gives us the following analysis ensemble for $i = 1, \dots, m$:

$$x_i^a = x^a + X^b W_i^a = \bar{x}^b + X^b (w^a + W_i^a). \quad (3.13)$$

Summarising, we obtain the following algorithm for the ETKF:

1. Determine the background ensemble x_1^b, \dots, x_m^b from the model propagation of the previous analysis.
2. Compute the background mean \bar{x}^b and anomaly matrix X^b given by (3.3).
3. Compute Y^b as given by (3.9).
4. Evaluate \tilde{P}^a using (3.12) and w^a using (3.11).
5. Compute W^a , the symmetric square root of \tilde{P}^a .
6. Generate the analysis ensemble members using (3.13).

3.3 Undersampling

There is a major drawback to the ETKF method, specifically regarding the use of an ensemble. When the ensemble is too small to be statistically representative of the system, a problem arises called *undersampling* [9]. For high-dimensional systems such as weather systems, our ensemble size is often much smaller than the dimension of the system, i.e. $m \ll n$. Therefore, the error covariance matrices P^b and P^a will only be sampled by a small number of state estimates, and can differ from the actual error covariances (as noted in remark 3.1). In fact, the error covariances are often underestimated [9]. When this happens, too much confidence is placed in the background state (in this case, the sample mean \bar{x}^b). Moreover, the observations will be increasingly ignored as time progresses, essentially removing the aspect of data assimilation from the process. This phenomenon is called *filter divergence*, meaning that the ensemble converges to an incorrect system state.

3.3.1 Methods to deal with undersampling

Fortunately, there are some ways to prevent filter divergence. One of them is *localisation*. Localisation was first introduced by Hamill and Whitaker in 2001 [8]. The idea behind localisation comes from another negative effect of undersampling. Namely, we obtain false correlations between observations and system states that are physically remote from each other [9]. The correlations are deemed false since in practice, we would expect correlations between two distant variables to be low. Now, localisation aims to remove these long-range correlations from the error covariance matrices. This can be done by taking the *Schur product* \circ between a certain *correlation operator* and the analysis error covariance matrix. The Schur product of two matrices is computed by taking the elementwise product of the matrices, i.e. $(A \circ B)_{ij} := A_{ij} \cdot B_{ij}$.

Localisation has the effect of greatly reducing the ensemble size necessary for filter convergence [11]. It is therefore an essential tool when applying ETKF to physical systems, since the dimension of these systems is often very high, of the order of 10^7 variables [7].

This means that the necessary ensemble size for filter convergence is also high, which may involve computational costs that are too high for present computers. Localisation reduces these computational costs. We will not apply localisation in this thesis due to time constraints.

Another way to combat undersampling is called *inflation*, first introduced by Anderson and Anderson in 1999 [10]. Inflation reduces the aforementioned underestimation in the error covariance matrices. One way of applying inflation is by multiplying the analysis error covariance matrix by a factor λ slightly greater than one:

$$P^a \mapsto \lambda P^a.$$

Equivalently, we could apply the inflation to the analysis ensemble members:

$$x_i^a \mapsto x^a + \lambda(x_i^a - x^a).$$

The question remains of how to choose the inflation factor λ . In this thesis, we select λ after performing various numerical data assimilation tests, similar to the tests we will do in chapter 5, where we vary λ . We perform a data assimilation run with inflation factor λ and compare it to an artificial truth. We choose λ such that the deviation of the analysis with the truth is lowest. Then, we assume that such λ will also lead to the smallest errors when applying the method in practice. We have to choose λ in this way because in reality, the true state of the system will be unknown. As such, it is not possible to know exactly how well the method performs in practice. After performing the tests, an inflation factor of $\lambda = 1.1$ seems to be optimal for most ensemble sizes.

4 Finite-size Ensemble Transform Kalman Filter

As mentioned in remark 3.1, the probability density function $p(x)$ of the ETKF method is defined by the sample mean \bar{x}^b and sample error covariance P^b of the background ensemble. However, we noted that some information is lost, since \bar{x}^b and P^b originate from sampling, resulting in an approximation of the desired distribution.

The consequence of these sampling errors is a consistent need for inflation, as discussed in section 3.3. Instead of applying inflation, another approach is to condition the prior distribution on the entire background ensemble x_1^b, \dots, x_m^b instead of the usual choices \bar{x}^b and P^b . This choice should diminish the sampling errors and hence *eliminate the need for inflation*.

The method that applies this observation is called the *Finite-size ETKF* (ETKF-N), introduced by Marc Bocquet in 2011 [6]. In this chapter, we will describe an algorithm for the Finite-size ETKF.

4.1 Background distribution

We start by computing the prior distribution conditional on the background ensemble, $p(x | x_1^b, \dots, x_m^b)$. This will lead to a cost function, just like in the previous chapter. As in chapter 3, we assume that each ensemble member is independently drawn from a multivariate Gaussian distribution of unknown mean state x^b and error covariance matrix B . Since x^b and B are unknown, we can integrate over all potential x^b and B in the following way:

$$\begin{aligned} p(x | x_1^b, \dots, x_m^b) &= \int p(x, x^b, B | x_1^b, \dots, x_m^b) dx^b dB \\ &= \int p(x | x_1^b, \dots, x_m^b, x^b, B) p(x^b, B | x_1^b, \dots, x_m^b) dx^b dB \end{aligned} \quad (4.1)$$

$$= \int p(x | x^b, B) p(x^b, B | x_1^b, \dots, x_m^b) dx^b dB. \quad (4.2)$$

Here dB means the Lebesgue measure on all independent entries $\prod_{i \leq j} d(B)_{ij}$. In equation (4.1) we used the definition of conditional probability, and in (4.2) we used the fact that $p(x)$ is already fully characterised by x^b and B , so the conditioning of $p(x | x_1^b, \dots, x_m^b, x^b, B)$ on the ensemble is redundant.

We now apply Bayes' rule to $p(x^b, B | x_1^b, \dots, x_m^b)$ in equation (4.2):

$$p(x | x_1^b, \dots, x_m^b) \propto \int p(x | x^b, B) p(x_1^b, \dots, x_m^b | x^b, B) p(x^b, B) dx^b dB. \quad (4.3)$$

The first two probability density functions in the integral of equation (4.3) are Gaussian by our previous assumptions, and can therefore be written explicitly. By definition of the multivariate Gaussian distribution, we have

$$p(x | x^b, B) = \frac{\exp\left(-\frac{1}{2}(x - x^b)^T B^{-1}(x - x^b)\right)}{(2\pi)^{n/2} \sqrt{|B|}}, \quad (4.4)$$

where $|B|$ denotes the determinant of B . Naturally, the definition for each $p(x_i | x^b, B)$ is the same, with x replaced by x_i . By combining (4.3) and (4.4), we obtain

$$p(x | x_1^b, \dots, x_m^b) \propto \int \exp(-L(x, x^b, B)) p(x^b, B) dx^b dB, \quad (4.5)$$

where

$$\begin{aligned} L(x, x^b, B) &= \frac{1}{2}(x - x^b)^T B^{-1}(x - x^b) + \frac{1}{2}(m + 1) \log |B| \\ &\quad + \frac{1}{2} \sum_{i=1}^m (x_i - x^b)^T B^{-1}(x_i - x^b). \end{aligned} \quad (4.6)$$

Here we used the fact that $\sqrt{|B|}^{-1} = \exp\left(-\frac{1}{2} \log |B|\right)$.

Now, the third function in the integral of (4.3) is $p(x^b, B)$, a prior on the background statistics of the ensemble. Since x^b and B are unknown, we will have to make some assumptions on the statistics of x^b and B . The prior that we will choose is called *Jeffreys' prior*. We first assume that x^b and B are independent, since this will make for a better prior than without this assumption [12]:

$$p(x^b, B) \equiv p_J(x^b, B) = p_J(x^b) p_J(B).$$

Jeffreys' prior then corresponds to the following choice:

$$p_J(x^b) = 1, \quad p_J(B) = |B|^{-\frac{n+1}{2}}, \quad (4.7)$$

where n is the dimension of the state space.

Lemma 4.1. *The prior $p_J(B)$ is invariant by any reparametrisation of state vectors.*

Proof. Suppose we apply a change of state variables $x = Fx'$, where F is a non-singular matrix. This implies that $B = FB'F^T$, where B' is the transformed error covariance matrix. Then, it follows (see [13]) that the Jacobian dB is equal to

$$dB = |F|^{n+1} dB', \quad (4.8)$$

such that

$$p_J(B) dB = |FB'F^T|^{-\frac{n+1}{2}} |F|^{n+1} dB' = p_J(B') dB'.$$

□

Now that we have selected a prior on the background statistics, we can integrate our previous result (4.5) over x^b . Using Jeffrey's prior (4.7), we get

$$p(x|x_1^b, \dots, x_m^b) \propto \int \exp(-\hat{L}(x, B)) dB, \quad (4.9)$$

where

$$\begin{aligned} \hat{L}(x, B) &= \frac{1}{2} \frac{m}{m+1} (x - \bar{x}^b)^T B^{-1} (x - \bar{x}^b) + \frac{1}{2} (n+m+1) \log |B| \\ &+ \frac{1}{2} \sum_{i=1}^m (x_i - \bar{x}^b)^T B^{-1} (x_i - \bar{x}^b). \end{aligned} \quad (4.10)$$

Notice that after the integration, x^b in (4.6) has been replaced by the sample mean \bar{x} as in (3.1). For future use, we rewrite the equation (4.10) as

$$\hat{L}(x, B) = \frac{1}{2} \text{Tr}(AB^{-1}) + \frac{n+m+1}{2} \log |B|, \quad (4.11)$$

where

$$A = \frac{m}{m+1} (x - \bar{x}^b)(x - \bar{x}^b)^T + (m-1)P^b. \quad (4.12)$$

The matrix P^b is the sample background error covariance matrix, defined in (3.2).

We now run into a similar problem to the one we had defining the ETKF in chapter 3. In our high-dimensional context of data assimilation A is generally rank-deficient: $\text{rank}(A) \leq m-1 \ll n$. This means that (4.11), and thus our integral (4.9), is not well-defined on our state space. We could solve this by restricting our integration over matrices B to a subspace on which $\hat{L}(x, B)$ is well-defined, similar to the coordinate change in the ETKF method. We will apply such a coordinate change in the next section. A more rigorous way to solve the rank-deficiency of A is to extend the matrix A to a full rank positive definite matrix $A_\epsilon = A + \epsilon I_n$, where I_n is the $n \times n$ identity matrix and $\epsilon > 0$. After the integration, we let ϵ go to zero.

To perform the integration on B in (4.9), we apply the change of variables $B = A_\epsilon^{1/2} B' A_\epsilon^{1/2}$. By (4.8), we have $dB = |A_\epsilon|^{\frac{n+1}{2}} dB'$. Hence, continuing from (4.9), we obtain

$$\begin{aligned} p(x|x_1^b, \dots, x_m^b) &\propto \int |B|^{-\frac{n+m+1}{2}} \exp\left(-\frac{1}{2} \text{Tr}(AB^{-1})\right) dB \\ &= \int |A_\epsilon|^{\frac{n+1}{2}} |A_\epsilon B'|^{-\frac{n+m+1}{2}} \exp\left(-\frac{1}{2} \text{Tr}(AA_\epsilon^{-\frac{1}{2}} (B')^{-1} A_\epsilon^{-\frac{1}{2}})\right) dB' \\ &= |A_\epsilon|^{-\frac{m}{2}} \int |B'|^{-\frac{n+m+1}{2}} \exp\left(-\frac{1}{2} \text{Tr}((B')^{-1})\right) dB' \end{aligned} \quad (4.13)$$

$$\propto |A_\epsilon|^{-\frac{m}{2}} \propto |A|^{-\frac{m}{2}} = \exp\left(-\frac{m}{2} \log |A|\right). \quad (4.14)$$

In equation (4.13), we used the fact that the trace is invariant under cyclic permutations of matrix products. Note that A as in (4.14) actually represents the restriction of A as a linear operator to the ensemble space \mathcal{V} (as defined in definition 3.2), such that (4.14) is well-defined.

Finally, we use (4.14) to obtain the background term $J^b(x)$ of our cost function:

$$J^b(x) = \frac{m}{2} \log |A| = \frac{m}{2} \log \left| \frac{m}{m+1} (x - \bar{x}^b)(x - \bar{x}^b)^T + (m-1)P^b \right|.$$

The observation term $J^o(x)$ is unchanged from our ETKF cost function (3.6):

$$J^o(x) = \frac{1}{2} (y - h(x))^T R^{-1} (y - h(x)).$$

Here y is the vector of observations, h is the observation operator, and R is the observation error covariance matrix. This gives us the following cost function:

$$\begin{aligned} J(x) &= J^b(x) + J^o(x) = \frac{m}{2} \log |A| + J^o(x) & (4.15) \\ &= \frac{m}{2} \log \left| \frac{m}{m+1} (x - \bar{x}^b)(x - \bar{x}^b)^T + (m-1)P^b \right| \\ &\quad + \frac{1}{2} (y - h(x))^T R^{-1} (y - h(x)). \end{aligned}$$

Remark 4.2. In his article on the ETKF-N method, Marc Bocquet argues that the choice of $p_J(x^b)$ in equation (4.7) could be considered ‘too weakly informative’ [6], meaning that it is too independent of x^b and therefore does not pass along enough information on the statistics of x^b . One could argue that this gives the cost function an unfair bias. An alternative is to make a specific choice for x^b . A logical choice would be to assume that $x^b \approx \bar{x}^b$, such that $p(x | x^b, B) \approx p(x | \bar{x}^b, B)$. This changes equation (4.2) in the following way:

$$\begin{aligned} p(x | x_1^b, \dots, x_m^b) &= \int p(x | x^b, B) p(x^b, B | x_1^b, \dots, x_m^b) dx^b dB \\ &\approx \int p(x | \bar{x}^b, B) \int p(x^b, B | x_1^b, \dots, x_m^b) dx^b dB. \end{aligned}$$

Then, the background term of the cost function becomes equal to

$$J_{alt}^b = \frac{m}{2} \log \left| (x - \bar{x}^b)(x - \bar{x}^b)^T + (m-1)P^b \right|.$$

Observe that the factor $m/(m+1)$ from (4.15) has disappeared. We will not use this alternate cost function, since the difference in result with the original cost function is not very large.

4.2 Coordinate change

As with the ETKF method, we apply a coordinate change to the state vector x :

$$x = \bar{x}^b + X^b w.$$

This coordinate change means that $x - \bar{x}^b = X^b w$, so, using (3.2), we can rewrite A from equation (4.12) as

$$A = \frac{m}{m+1} X^b w w^T (X^b)^T + X^b (X^b)^T.$$

For future use, we simplify the background term of the cost function $J(x)$:

$$\begin{aligned} |A| &= \left| \frac{m}{m+1} X^b w w^T (X^b)^T + X^b (X^b)^T \right| \\ &= \left| X^b (X^b)^T \right| \left| I_{\mathcal{V}} + \frac{m}{m+1} \left[X^b (X^b)^T \right]^{-1} X^b w w^T (X^b)^T \right| \\ &\propto 1 + \frac{m}{m+1} w^T (X^b)^T \left[X^b (X^b)^T \right]^{-1} X^b w. \end{aligned} \quad (4.16)$$

Again, A here signifies the restriction of A as a linear operator to \mathcal{V} .

We would now like to obtain a cost function $\tilde{J}(w)$ such that this function has the same minimum as $J(x)$. First, we note that the columns of the anomaly matrix X^b are not linearly independent, since $\sum_{i=1}^m X_i^b = 0$. As a result, the parametrisation $J^b(x) = J^b(\bar{x}^b + X^b w)$ entails a so-called *gauge invariance*. This means that the background term $J^b(\bar{x}^b + X^b w)$ is invariant under translations of w . However, our obtained expression (4.16) is not invariant under translations of w . One way to make it invariant is by restricting the domain of the cost function. Instead of the entire ensemble space \mathcal{V} , we could only choose w with a null orthogonal projection on the kernel of X^b :

$$(I_m - (X^b)^T \left[X^b (X^b)^T \right]^{-1} X^b) w = 0.$$

Then, by inserting the previous equation into (4.16), we would obtain that

$$|A| \propto 1 + \frac{m}{m+1} w^T w. \quad (4.17)$$

We will not apply this domain restriction, since it turns out to be too inefficient [6].

Instead, we will insert a *gauge-fixing term*

$$\mathcal{G}(w) = \frac{m}{m+1} w^T \left(I_m - (X^b)^T \left[X^b (X^b)^T \right]^{-1} \right) w$$

into the cost function $J(x)$ such that the same result (4.17) is achieved. This results in the following augmented cost function $\tilde{J}(w)$ that we will use for our ETKF-N algorithm:

$$\begin{aligned}\tilde{J}(w) &= J^o(\bar{x}^b + X^b w) + \frac{m}{2} \log(|A| + \mathcal{G}(w)) \\ &= \frac{1}{2} (y - h(\bar{x}^b + X^b w))^T R^{-1} (y - h(\bar{x}^b + X^b w)) \\ &\quad + \frac{m}{2} \log \left(1 + \frac{1}{m} + w^T w \right).\end{aligned}\tag{4.18}$$

Lemma 4.3. *The cost functions $J(x)$ from (4.15) and $\tilde{J}(w)$ from (4.18) have the same minimum.*

Proof. Since \log is an increasing function, we have $\tilde{J}(w) \geq J(\bar{x}^b + X^b w)$ for all $w \in \mathbb{R}^m$, with equality if and only if $\mathcal{G}(w) = 0$. Furthermore, for any x there exists a w in the null space of X^b such that $J(x) = \tilde{J}(w)$. In particular, if x' is the minimum of $J(x)$, we can find a w' with $J(x') = \tilde{J}(w')$, such that w' is the minimum of $\tilde{J}(w)$. We conclude that the two cost functions have the same minimum. \square

4.3 Minimisation

The outline of the ETKF-N method is mostly similar to the regular ETKF method. We start by minimising the cost function $\tilde{J}(w)$ as defined in (4.18). Unlike in the ETKF method, however, we cannot easily find an exact solution for this minimisation. Therefore, we minimise $\tilde{J}(w)$ iteratively, starting with an initial guess of $w = 0$. In this thesis, we apply Newton's method to $\nabla \tilde{J}(w)$ for the iterative minimisation, such that we find a root of the gradient of $\tilde{J}(w)$ (which is of course a minimum of $\tilde{J}(w)$). The choice for $w = 0$ is arbitrary, but has been tested to work well and is also the simpler choice as an initial guess. Our result is a minimiser w^a such that $\nabla \tilde{J}(w^a) \approx 0$.

After the minimisation, the mean of the analysis ensemble is given by

$$x^a = \bar{x}^b + X^b w^a.\tag{4.19}$$

We would also like to know an estimate of the error covariances at the minimum. It can be checked that the Hessian of $\tilde{J}(w)$ is equal to

$$\mathcal{H}(w) = \nabla^2 \tilde{J}(w) = m \frac{\left(1 + \frac{1}{m} + w^T w\right) I_m - 2ww^T}{\left(1 + \frac{1}{m} + w^T w\right)^2} + (HX)^T R^{-1} HX,\tag{4.20}$$

where H is the Jacobian matrix of the observation operator h . Then, the analysis error covariance matrix \tilde{P}^a in ensemble space is approximately equal to the inverse matrix of the Hessian at the minimum:

$$\tilde{P}^a \approx (\mathcal{H}(w^a))^{-1}.\tag{4.21}$$

As in the ETKF method, we define W^a as the symmetric square root of $(m-1)\tilde{P}^a$ and W_i^a as the i th column of W^a . We then obtain the following analysis ensemble with mean x^a and error covariance matrix $P^a = X^b \tilde{P}^a (X^b)^T$, for $i = 1, \dots, m$:

$$x_i^a = x^a + X^b W_i^a. \quad (4.22)$$

In conclusion, we have described the following algorithm for ETKF-N:

1. Determine the background ensemble x_1^b, \dots, x_m^b from the model propagation of the previous analysis.
2. Compute the background mean \bar{x}^b and anomaly matrix X^b given by (3.3).
3. Use Newton's method to iteratively minimise the cost function $\tilde{J}(w)$ from (4.18) starting with $w = 0$, to obtain w^a .
4. Compute x^a using (4.19) and the analysis error covariance matrix \tilde{P}^a using (4.20) and (4.21).
5. Compute W^a , the symmetric square root of \tilde{P}^a .
6. Generate the analysis ensemble members using (4.22).

5 Numerical experiments

In this chapter, the ETKF-N method of chapter 4 is numerically tested and compared against the original ETKF method of chapter 3. We will test the data assimilation methods on two chaotic dynamical ‘toy models’, namely the Lorenz ’63 [15] and Lorenz ’96 [16] models. In practice, the methods are usually applied to high-dimensional systems, but this is not possible for this thesis due to the large computational costs. Therefore, we instead use simplified low-dimensional models for our experiments. The toy models are still deemed to be representative, since they exhibit the same chaotic behaviour as that of the models used in practice.

Both methods as well as the toy models are implemented in MATLAB, which is also where we perform the numerical experiments. The models are propagated using the popular fourth-order Runge-Kutta method (RK4). The implementation of the ETKF method is largely based on that of Sanita Vetra-Carvalho [14]. The numerical experiments partly follow those of Bocquet (2011) [6].

5.1 Lorenz ’63 model

The Lorenz ’63 model is a three-dimensional system of ordinary differential equations (ODEs), developed by the American mathematician Edward Lorenz [15]. The model is defined by the following equations:

$$\begin{aligned}\frac{dx}{dt} &= \sigma(y - x) \\ \frac{dy}{dt} &= \rho x - y - xz \\ \frac{dz}{dt} &= xy - \beta z,\end{aligned}$$

where σ , ρ and β are parameters. We set the parameters to the values $(\sigma, \rho, \beta) = (10, 28, 8/3)$, since these are known to lead to chaotic dynamics. Before testing either of the data assimilation methods, we perform a ‘nature run’ with an initial condition of $(-10, 10, 20)$, which is a simulation of the model without the use of any data assimilation. This nature run serves as the true state of the model. An example of such a nature run is plotted in figure 5.1. The three variables at each time step are plotted as (x, y, z) -coordinates in \mathbb{R}^3 . The plot of solutions of the Lorenz ’63 model is called the *Lorenz attractor* and resembles the wings of a butterfly (see figure 5.1).

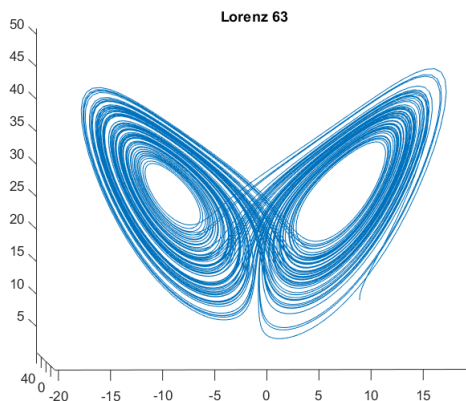


Figure 5.1: Lorenz '63 model attractor.

After performing the nature run, we generate observations each $\Delta t = 0.10$ for use in the data assimilation. The observations are generated from the nature run (the ‘truth’) and independently perturbed with a *Gaussian white noise* (a random error with pairwise independent components) of mean 0 and standard deviation 2, such that the observation error covariance matrix is equal to $R = 4 \cdot I_3$. All variables are observed, so the observation operator is the identity: $H = I_3$.

Finally, we perform our test run for both methods. The initial condition of the test runs is obtained by perturbing the initial condition for the nature run $(-10, 10, 20)$ with an error with the magnitude of the observational error covariance matrix R . In all simulations, the time step for numerical approximation is 0.01. The time interval for data assimilation is $\Delta t = 0.10$, in accordance with the generation of observations. All simulations consist of $5 \cdot 10^4$ cycles of time Δt . We also use a *burn-in period* of 10^4 cycles, in which only the model propagation takes place (without any data assimilation). This should reduce the impact of the initial condition choice on the results. We vary the ensemble size from $m = 3$ to $m = 9$. Lastly, for the ETKF method, we apply multiplicative inflation to the background ensemble as explained in section 3.3. We use an inflation factor of 1.1, since after comparing various inflation factors, it seems to give the lowest errors for most ensemble sizes.

We compare the methods by the value of the time-averaged *root mean square error* (rmse) between the analysis and the truth (nature run). Naturally, the method with the lower rmse is better for that ensemble size.

Definition 5.1. The time-averaged root mean square error is defined as

$$\sqrt{\frac{1}{N} \sum_{i=1}^N \left(x_{(i)}^a - x_{(i)}^t \right)^2}.$$

Here N is the number of data assimilation cycles, $x_{(i)}^a$ is the analysis ensemble mean at time i and $x_{(i)}^t$ is the true state at time i .

5.1.1 Results

The root mean square errors for both methods are plotted in figure 5.2, for ensemble sizes 3 through 9.

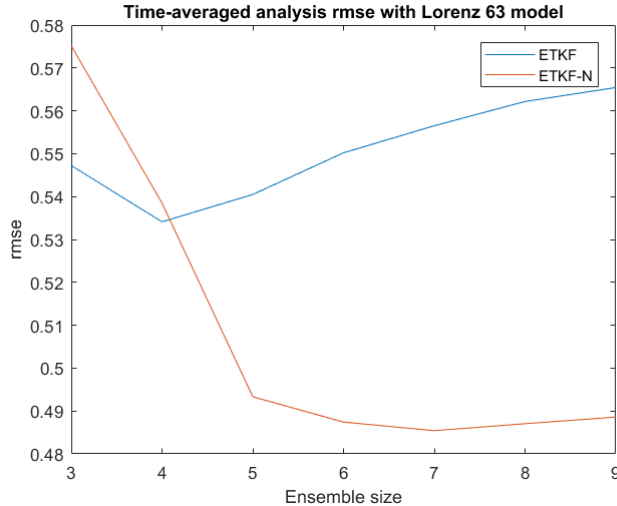


Figure 5.2: Time-averaged analysis rmse for ETKF and ETKF-N with the Lorenz '63 model, for various ensemble sizes.

For ensemble sizes $m = 3$ and $m = 4$, we see that the rmse of the ETKF-N method is slightly higher than the rmse of ETKF. However, for $m \geq 5$ we observe that the rmse of ETKF-N is significantly lower. It is surprising that ETKF-N performs worse than ETKF for the lowest ensemble sizes. For other Ensemble Kalman Filters, it is known that the effects of undersampling are larger for lower ensemble sizes [9]. Therefore, it is possible that ETKF-N performs better for $m = 3$ and $m = 4$ with some inflation.

5.2 Lorenz '96 model

Next, we test both methods on the Lorenz '96 model, by Edward Lorenz and Kerry Emmanuel [16]. The model is defined as follows, for $i = 1, \dots, N$:

$$\frac{dx_i}{dt} = (x_{i+1} - x_{i-2})x_{i-1} - x_i + F,$$

where we assume that $x_{-1} = x_{N-1}$, $x_0 = x_N$ and $x_{N+1} = x_1$, and $N = 40$. Here F is a 'forcing constant'. We choose $F = 8$ since it is known to cause chaotic behaviour.

The procedure of our experiments is mostly similar to that of the tests with the Lorenz '63 model. At first, we perform a nature run. The initial condition is set to $x_i = 8$ for each $i \neq 20$, $x_{20} = 8.01$. The choice $i = 20$ is arbitrary, but a perturbation of at least one variable is necessary for chaotic dynamics. Afterwards, we generate observations each $\Delta t = 0.05$. These observations are obtained from the nature run and perturbed with a Gaussian white noise of mean 0 and standard deviation 1, such that R is the identity. Again, all variables are observed, so the observation operator is the identity.

Like the previous experiments, we obtain the initial condition for the test runs by perturbing the initial condition of the nature run with a small error. The time step of numerical approximation is again 0.01. The data assimilation takes place each $\Delta t = 0.05$. The simulations consist of a burn-in period of $5 \cdot 10^3$ cycles and a data assimilation period of 10^4 cycles. The number of cycles is shorter than in the Lorenz '63 experiments, since the convergence is shown to take place more quickly [6]. The ensemble size is varied from $m = 5$ to $m = 30$. Again, we apply inflation to ETKF with an inflation factor of 1.1.

5.2.1 Results

In figure 5.3, the time-averaged rmse between the analysis and the truth is plotted for both methods. Note that for both methods, the rmse is high for $m \leq 15$, meaning that the filter diverges (see section 3.3) for low ensemble sizes. This means that the ensemble size is too low relative to the dimension of the system (in this case 40). We see that for small ensemble sizes $m \leq 15$, the ETKF-N method performs considerably better. This is in contrast with our results with the Lorenz '63 model, where ETKF-N underperformed ETKF for the lowest ensemble sizes. In the range $16 \leq m \leq 22$, the ETKF-N method actually has a slightly higher rmse. However, for $m > 22$ the rmse of ETKF-N becomes noticeably lower. This also seems to be the case for m beyond 30. To give an indication of the difference between both methods, the rmse of ETKF-N is 21 percent lower than the rmse of ETKF, for $m = 30$.

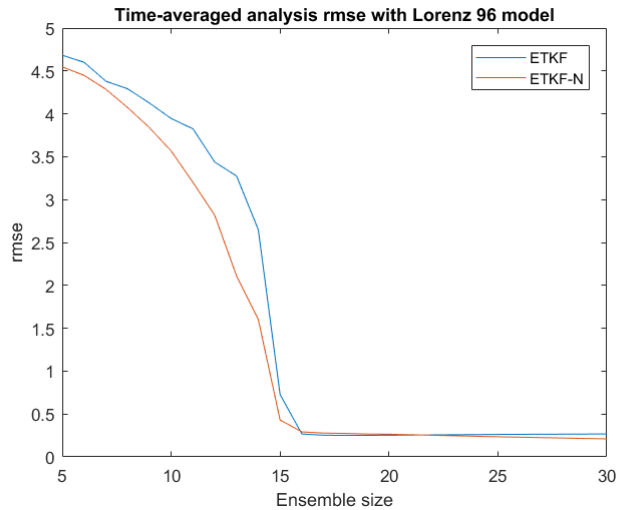


Figure 5.3: Time-averaged analysis rmse for ETKF and ETKF-N with the Lorenz '96 model, for various ensemble sizes.

5.2.2 Inflation diagrams

In addition to the previous tests investigating the rmse of the analysis with the truth at different ensemble sizes, we also perform tests considering various inflation factors for both methods. The tests follow the procedure of section 5.2, but with one major difference. For the ETKF method, we vary the inflation factor from 1 to 1.09, with increments of 0.01. For the Finite-size ETKF, we also test *deflation*, inflation factors lower than 1. We do not apply deflation for ETKF, since it is already known that (for most ensemble sizes) inflation is necessary for filter convergence for ETKF [10]. However, the effect of inflation factors on ETKF-N is unknown, so we also apply deflation. We range the inflation factors from 0.95 to 1.09. The ensemble size is varied from $m = 5$ to $m = 50$.

The results for ETKF are shown in figure 5.4. Not all results are shown, since for some combinations of ensemble sizes and inflation factors, the rmse is too high, meaning that the filter diverges. These combinations are shown as blank spaces in the figure. We see that the filter diverges for ensemble sizes $m \leq 15$, in accordance with the results of figure 5.3. Furthermore, the rmse is lowest with an inflation factor of 1.01 for high ensemble sizes $m \geq 25$. Without any inflation, ETKF only converges for ensemble sizes $m \geq 45$.

The results for ETKF-N are shown in figure 5.5. As with ETKF, the filter diverges for $m \leq 15$. Moreover, the rmse is generally the lowest for an inflation factor of 0.99. The errors for ETKF-N without any inflation or deflation are not that much higher.

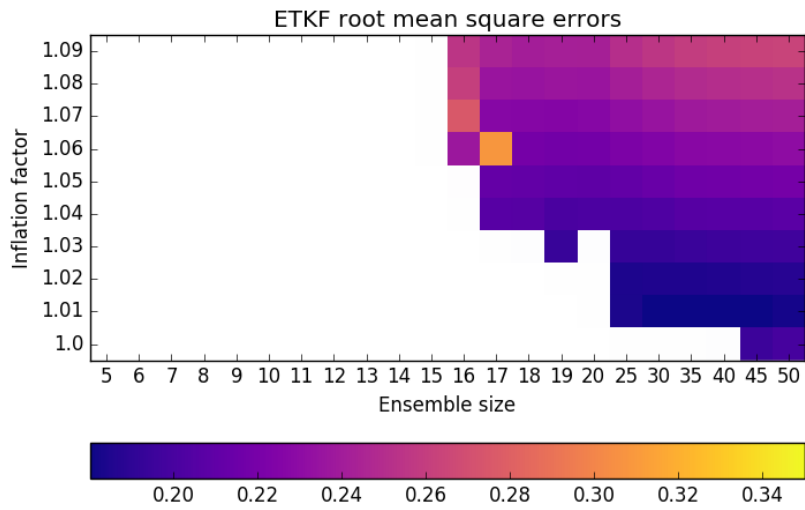


Figure 5.4: Root square mean errors of ETKF with the Lorenz '96 model, for various ensemble sizes and inflation factors.

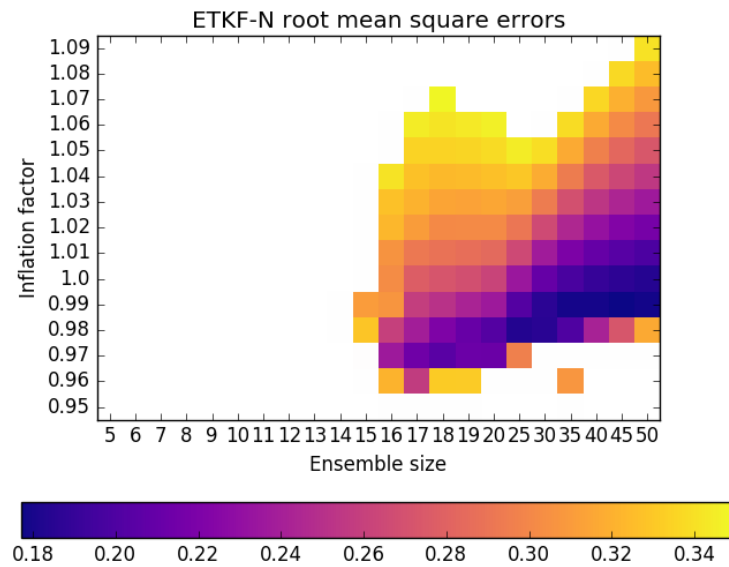


Figure 5.5: Root square mean errors of ETKF-N with the Lorenz '96 model, for various ensemble sizes and inflation factors.

6 Conclusion and discussion

First of all, we can see from our experiments with the Lorenz '63 model that the ETKF-N has a significantly better performance than the ETKF. For the lowest ensemble sizes that we tested, ETKF is slightly better, but since we are mainly interested in performance for higher ensemble sizes, this is not of much importance. Similar conclusions can be made when observing the results with the Lorenz '96 model. Even though the ETKF method is slightly better for a small range of ensemble sizes, ETKF-N has lower errors for both small and high ensemble sizes.

In addition to comparing ETKF and ETKF-N, another aim of this thesis was to find out if ETKF-N indeed eliminates the need for inflation. To this end, we investigated the performance of both methods with various inflation factors, tested on the Lorenz '96 model. We noticed that inflation is necessary for convergence of the ETKF, for ensemble sizes $m < 45$. The optimal inflation factor for high ensemble sizes $m \geq 25$ is 1.01 for ETKF, so inflation is beneficial even for ensemble sizes $m \geq 45$. We also see that the filter diverges for ensemble sizes $m \leq 15$, meaning that inflation alone is not enough to prevent filter divergence for low ensemble sizes.

In contrast to ETKF, we see that ETKF-N does not require any inflation for convergence, for ensemble sizes $m \geq 15$. Deflation seems to be beneficial for some ensemble sizes, the optimal inflation factor being somewhere around 0.99, but it is not necessary. It is surprising that deflation sometimes improves the performance, since this points to an overestimation of the error covariances rather than an underestimation.

It is unfortunate that both filters diverge for ensemble sizes $m \leq 15$. This was already known for ETKF [10], but the ETKF-N method was designed to remove the effects of undersampling. As it is, ETKF-N only solves the need for inflation, but this turns out to be not enough to prevent filter divergence. This means that localisation (see section 3.3) is still necessary for convergence of ETKF-N for low ensemble sizes.

Overall, the Finite-size ETKF seems to be a worthwhile substitute to the regular ETKF method. The experiments with the Lorenz '63 and Lorenz '96 models show that the errors are generally lower when using the ETKF-N method. Moreover, ETKF-N does not require inflation for convergence of the method, in contrast to ETKF.

One downside to ETKF-N is the higher computational cost. More work needs to be done to minimise the cost function, for example, since the minimisation has to be done iteratively. A test using MATLAB shows that the computational cost of ETKF-N (measured in running time of the data assimilation) is approximately 20 percent higher than that of ETKF. Therefore, one could argue that the removal of need for inflation together with the small decrease of errors is not worth the increase in computational cost. Personally, I would prefer the simplicity of the ETKF method, but I nevertheless feel that it is important to always strive for minimal errors, regardless of computational complexity.

There is still a lot of future research that can be done to improve the ETKF-N method. For one, the choice of the prior on the background ensemble could have a large influence on the results. In this thesis, we used Jeffreys' prior (4.7), but as we mentioned in remark 4.2, there are different choices that could be considered. Bocquet also argues in his summary that the ETKF-N method is 'mainly a proof of concept', and that 'more work is needed to explore the strengths and limitations of the methodology' [6].

As a final note, the goal of the ETKF-N method is to eliminate sampling errors. In practice, there are also model errors that arrive when propagating the model, but we did not simulate these errors in this thesis. Even though there is no need for inflation with ETKF-N in this theoretical setting, we would still need to apply inflation in practice to account for the model errors.

Bibliography

- [1] Rabier, F., *Overview of global data assimilation developments in numerical weather-prediction centres*, Quarterly Journal of the Royal Meteorological Society, 131, 3215-3233, 2005.
- [2] Evensen, G., *Sequential data assimilation with nonlinear quasi-geostrophic model using Monte Carlo method to forecast error statistics*, Journal of Geophysical Research, 99(C5), 143162, 1994.
- [3] Carrera, M. L., Bélair, S., and Bilodeau, B., *The Canadian Land Data Assimilation System (CaLDAS): Description and Synthetic Evaluation Study*, Journal of Hydrometeorology, 16, 1293-1312, 2015.
- [4] *Data assimilation methods - Met Office*, <https://www.metoffice.gov.uk/research/weather/data-assimilation/data-assimilation-methods>, 2017.
- [5] Hunt, B. R., Kostelich, E. J., and Szunyogh, I., *Efficient data assimilation for spatiotemporal chaos: A local ensemble transform Kalman filter*, Physica D, 230, 112-126, 2007.
- [6] Bocquet, M., *Ensemble Kalman filtering without the intrinsic need for inflation*, Nonlinear Processes in Geophysics, 18, 735-750, 2011.
- [7] Bocquet, M., *Introduction to the principles and methods of data assimilation in the geosciences*, Lecture notes Master M2 OACOS & WAPE, 19, 2016.
- [8] Hamill, T. M., and Whitaker, J. S., *Distance-Dependent Filtering of Background Error Covariance Estimates in an Ensemble Kalman Filter*, Monthly Weather Review, 129, 27762790, 2001.
- [9] Petrie, R. E., and Dance, S. L., *Ensemble-based data assimilation and the localisation problem*, Weather, 65(3), 65-69, 2010.
- [10] Anderson, J. L., and Anderson, S. L., *A Monte Carlo Implementation of the Non-linear Filtering Problem to Produce Ensemble Assimilations and Forecasts*, Monthly Weather Review, 127, 2741-2758, 1999.
- [11] Oke, P., Sakov, P., and Corney, S., *Impacts of localisation in the EnKF and EnOI: experiments with a small model*, Ocean Dynamics, 57(1), 32-45, 2007.
- [12] Jeffreys, H., *Theory of Probability*, Oxford University Press, 3rd edition, 1961.

- [13] Muirhead, R. J., *Aspect of Multivariate Statistical Theory*, Wiley-Interscience, 1982.
- [14] Vetra-Carvalho, S., *Practical 4: Ensemble Kalman Filter*,
http://www.met.reading.ac.uk/~darc/nerc_training_2014/, 2014.
- [15] Lorenz, E. N., *Deterministic nonperiodic flow*, Journal of the Atmospheric Sciences, 20(2), 130-141, 1963.
- [16] Lorenz, E. N., and Emmanuel, K. E., *Optimal sites for supplementary weather observations: simulation with a small model*, Journal of the Atmospheric Sciences, 55, 399-414, 1998.

Popular summary

An important part of applied mathematics is predicting the future state of complicated systems such as the weather. One way to predict these systems is by using the *model* of the system, which is a collection of mathematical equations. Such a model determines the future state of a system given an initial estimate. The difficulty of the prediction comes in finding the best possible initial estimate, because the true state of the system is usually unknown. Moreover, we are interested in *chaotic dynamical systems*, which have the property that any small errors in our initial estimate will eventually lead to large errors in the future state. Hence, it is very important that our initial estimate is as close as possible to the truth. Using observations of the system state as an initial estimate is not a good idea, since there are always large measurement errors present.

A popular way to find a better initial estimate is through a process called *data assimilation*. Data assimilation combines our theoretical knowledge of the system with observational data, in order to improve the initial estimate. We can apply the model to the improved estimate, giving us a forecast for the future state. To be precise, we start with an initial estimate x_k^b at time t_k , the index b standing for *background*. This x_k^b is usually the outcome of a previous forecast. By combining the background state with our observations in a certain way, we obtain an improved state estimate x_k^a , the index a standing for *analysis*. After the data assimilation, we can apply our model to x_k^a , giving us a forecast x_{k+1}^b for the system state at time t_{k+1} .

In this thesis, we discuss two methods of data assimilation. The first is called the *Ensemble Transform Kalman Filter* (ETKF). The ETKF is part of a group of methods which uses an *ensemble*, which is a statistical sample of estimates of a certain state. Instead of having a single background state, we have an ensemble x_1^b, \dots, x_m^b as an initial estimate. This ensemble can also be represented by a certain *probability density function* $p(x)$ of a Gaussian distribution. We will have to make certain assumptions about the parameters of this Gaussian distribution. It turns out that these assumptions can lead to large errors called *sampling errors*. These errors can become large enough such that the prediction of the state moves away from the true state, converging to an incorrect state far away from the truth. We call this *filter divergence*. Fortunately, there are methods to prevent filter divergence, called *inflation* and *localisation*. It is necessary to apply these methods during data assimilation if we want to have a good performance of the ETKF.

The second method we will investigate is called the *Finite-size Ensemble Transform Kalman Filter* (ETKF-N). This method is based on the ETKF, but notices the flaws in the assumptions which are made about $p(x)$. The ETKF-N makes different assumptions, which should in theory lead to less sampling errors, such that inflation is not necessary anymore.

In addition to describing these methods of data assimilation, we also perform some experiments. We would like to know which of the methods performs better, and whether or not ETKF-N still requires inflation. The methods are applied to the Lorenz '63 and Lorenz '96 models. We perform the data assimilation and compare it to an artificial truth run. The tests show that ETKF-N generally performs better than ETKF. Furthermore, we conclude that inflation is not necessary for the ETKF-N method. However, sometimes the ETKF-N still requires localisation for good performance, just like the regular ETKF.