GOETHE UNIVERSITÄT
FRANKFURT AM MAIN

Derivation and Characterization of a New Filter for Nonlinear High-Dimensional Data Assimilation

Dissertation
zur Erlangung des Doktorgrades
der Naturwissenschaften

vorgelegt beim Fachbereich Geowissenschaften/Geographie
der Johann Wolfgang Goethe-Universität
in Frankfurt am Main

von
Julian Tödter
aus Frankfurt am Main

Frankfurt am Main (2015)
(D 30)

Vom Fachbereich 11 (Geowissenschaften/Geographie)
der Johann Wolfgang Goethe-Universität als Dissertation angenommen.

Dekan: Prof. Dr. Ulrich Achatz

Gutachter: Prof. Dr. Bodo Ahrens, Prof. Dr. Leopold Haimberger

Datum der Disputation: 19.06.2015

The real glory of science is that we can find a way of thinking such that the law is evident.

Richard P. Feynman, The Feynman Lectures on Physics (1965)

Abstract

Derivation and Characterization of a New Filter for Nonlinear High-Dimensional Data Assimilation

Data assimilation (DA) combines model forecasts with real-world observations to achieve an optimal estimate of the state of a dynamical system. The quality of predictions in nonlinear and chaotic systems such as atmospheric or oceanic circulation is strongly sensitive to the initial conditions. Therefore, beyond the consistent reconstruction of past states, a primary relevance of advanced DA methods concerns the proper model initialization. The ensemble Kalman filter (EnKF) and its deterministic variants, mostly square root filters such as the ensemble transform Kalman filter (ETKF), represent a popular alternative to variational DA schemes. They are applied in a wide range of research and operations. Their forecast step employs an ensemble integration that fully respects the nonlinear nature of the analyzed system. In the analysis step, they implicitly assume the prior state and observation errors to be Gaussian. Consequently, in nonlinear systems, the mean and covariance of the analysis ensemble are biased and these filters remain suboptimal. In contrast, the fully nonlinear, non-Gaussian particle filter (PF) relies on Bayes' theorem without further assumptions, which guarantees an exact asymptotic behavior. However, it is exposed to weight collapse, particularly in higher-dimensional settings, known as the curse of dimensionality.

This work presents a new method to obtain an analysis ensemble with mean and covariance that exactly match the corresponding Bayesian estimates. This is achieved by a deterministic matrix square root transformation of the forecast ensemble, and subsequently a suitable random rotation that significantly contributes to filter stability while preserving the required second-order statistics. The forecast step remains as in the ETKF. The algorithm, which is fairly easy to implement and computationally efficient, is referred to as the nonlinear ensemble transform filter (NETF). The limitation with respect to fully-nonlinear filtering is that the NETF only considers the mean and covariance of the Bayesian analysis density, neglecting higher-order moments.

The properties and performance of the proposed algorithm are investigated via a set of experiments. The results indicate that such a filter formulation can increase the analysis quality, even for relatively small ensemble sizes, compared to other ensemble filters in nonlinear, non-Gaussian scenarios. They also confirm that localization enhances the applicability of this PF-inspired scheme in larger-dimensional systems. Finally, the novel filter is coupled to a large-scale ocean general circulation model with a realistic observation scenario. The NETF remains stable with a small ensemble size and shows a consistent behavior. Additionally, its analyses exhibit low estimation errors, as revealed by a comparison with a free ensemble integration and the ETKF. The results confirm that, in principle, the filter can be applied successfully and as simple as the ETKF in high-dimensional problems. No further modifications are needed, even though the algorithm is only based on the particle weights. Thus, it is able to overcome the curse of dimensionality, even in deterministic systems. This proves that the NETF constitutes a promising and user-friendly method for nonlinear high-dimensional DA.

Kurzfassung

Herleitung und Charakterisierung eines neuen Filters
für die nichtlineare, hochdimensionale Datenassimilation

Die Datenassimilation (DA) kombiniert Modellvorhersagen mit empirischen Beobachtungen, um den Zustand eines dynamischen Systems optimal zu schätzen. Die Qualität von Prognosen in nichtlinearen und chaotischen Systemen, wie etwa Atmosphäre oder Ozean, hängt stark von den Anfangsbedingungen ab. Neben ihrer Anwendung zur konsistenten Rekonstruktion vergangener Zustände sind fortschrittliche DA-Methoden daher von großer Relevanz für die korrekte Initialisierung von Modellen. Der Ensemble Kalman Filter (EnKF) und dessen deterministische Varianten, vor allem sogenannte Quadratwurzelfilter wie der Ensemble Transform Kalman Filter (ETKF), stellen eine gängige Alternative zu variationellen DA-Methoden dar. Sie sind sowohl in der Forschung als auch in operativen Anwendungen weit verbreitet. Ihr Vorhersageschritt nutzt eine Ensembleintegration, wodurch die Nichtlinearität des betrachteten Systems uneingeschränkt berücksichtigt wird. Im Analyseschritt wird jedoch implizit angenommen, dass Zustands- und Beobachtungsfehler einer Normalverteilung folgen. Daher unterliegen Mittel und Kovarianz des Analyseensembles in nichtlinearen Systemen einem systematischen Fehler. Im Gegensatz dazu greift der vollständig nichtlineare Partikelfilter (PF) lediglich auf Bayes' Theorem zurück, was ein exaktes asymptotisches Verhalten garantiert. Allerdings sind PF insbesondere in hochdimensionalen Systemen nicht anwendbar, da die Partikelgewichte kollabieren.

Die vorliegende Arbeit präsentiert eine neue Methode, um ein Analyseensemble zu generieren, dessen Mittelwert und Kovarianz exakt den entsprechenden Bayes'schen Schätzern entsprechen. Hierzu wird das Vorhersageensemble mittels einer Wurzelmatrix transformiert. Anschließend wird eine geeignete Zufallsrotation im Ensemble-Unterraum durchgeführt, welche die Stabilität des Filters deutlich verbessert, ohne die ersten beiden Momente zu verändern. Der Vorhersageschritt bleibt gegenüber dem ETKF unverändert. Der Algorithmus ist einfach zu implementieren und rechentechnisch effizient. Er wird als Nonlinear Ensemble Transform Filter (NETF) bezeichnet. Seine grundsätzliche Einschränkung besteht darin, dass nur Mittelwert und Kovarianz der Bayes'schen Analyseverteilung berücksichtigt werden, während höhere Momente vernachlässigt werden.

Eigenschaften und Leistungsverhalten des NETF werden in verschiedenen Simulationen untersucht. Die Resultate zeigen, dass die neue Methode die Qualität der Analyse in nichtlinearen, nicht-Gauß'schen Szenarien im Vergleich zu anderen Ensemblefiltern deutlich verbessern kann, sogar für relativ kleine Ensemblegrößen. Hierbei wird die Anwendbarkeit in Systemen höherer Dimension durch eine Lokalisierung der Analyse ermöglicht. Zum Schluss wird der neue Filter mit einem großskaligen Modell der Ozeanzirkulation mit realistischem Beobachtungsszenario gekoppelt. Der NETF bleibt stabil und weist bereits bei einer kleinen Ensemblegröße Analysen mit niedrigen Schätzfehlern auf. Damit wird bestätigt, dass der Filter erfolgreich und ähnlich einfach wie der ETKF für hochdimensionale Probleme angewendet werden kann. Weitere Modifikationen sind nicht erforderlich, obwohl der Algorithmus ausschließlich auf den Partikelgewichten basiert. Somit erweist sich der NETF als eine vielversprechende und anwenderfreundliche Methode für die nichtlineare, hochdimensionale DA.

Contents

Abbreviations

3/4DVAR	**3/4**-dimensional **VAR**iational assimilation
CRPS, CRIGN	**C**ontinuous **R**anked **P**robability **S**core / **IGN**orance score
DA	**D**ata **A**ssimilation
DL	**D**omain **L**ocalization
DWD	**D**eutscher **W**etter**D**ienst (*German Weather Service*)
ECMWF	**E**uropean **C**enter for **M**edium-range **W**eather **F**orecasts
EnKF	**En**semble **K**alman **F**ilter
(L)ETKF	(**L**ocal) **E**nsemble **T**ransform **K**alman **F**ilter
ETKS	**E**nsemble **T**ransform **K**alman **S**moother
EWPF	**E**quivalent **W**eights **P**article **F**ilter
NETF	**N**onlinear **E**nsemble **T**ransform **F**ilter
NEMO	**N**ucleus of **E**uropean **M**odeling of the **O**cean
NLEAF	**N**on**L**inear **E**nsemble **A**djustment **F**ilter
NWP	**N**umerical **W**eather **P**rediction
KF	**K**alman **F**ilter
OL	**O**bservation **L**ocalization
(O)GCM	(**O**cean) **G**eneral **C**irculation **M**odel
OSSE	**O**bserving **S**ystem **S**imulation **E**xperiment
PDAF	**P**arallel **D**ata **A**ssimilation **F**ramework
pdf	**p**robability **d**ensity **f**unction
PF	**P**article **F**ilter
RMSE	**R**oot **M**ean **S**quare **E**rror
SVD	**S**ingular **V**alue **D**ecomposition
SSH	sea surface height (in NEMO)
T	(ocean) temperature (in NEMO)
U	zonal velocity (in NEMO)
V	meridional velocity (in NEMO)

Symbols

N	ensemble size (index: n)	scalar
K	state space dimension (index: k)	scalar
L	observation space dimension (index: l)	scalar
I	number of model time steps (index: i)	scalar
J	number of observations (index: j)	scalar
$\mathbb{X}$	state space	vector space of dim. K
$\mathbb{Y}$	observation space	vector space of dim. L
$\mathbb{R}^p$	real space	vector space of dim. p
$\mathbf{A}$	squared transform matrix	matrix, $N \times N$
$\mathbf{B}$	background error covariance	matrix, $K \times K$
$\mathbf{H}$	tangent linear of observation operator	matrix, $L \times K$
$\mathbf{I}_N$	identity matrix	matrix, $N \times N$
$\mathbf{K}$	Kalman gain	matrix, $K \times L$
$\mathbf{M}$	tangent linear of model operator	matrix, $K \times K$
$\mathbf{\Omega}$	random rotation matrix	matrix, $N \times (N-1)$
$\mathbf{Q}$	model error covariance	matrix, $K \times K$
$\mathbf{P}$	state error covariance	matrix, $K \times K$
$\mathbf{R}$	observation error covariance	matrix, $L \times K$
$\mathbf{T}$	transform matrix	matrix, $N \times N$
$\mathbf{\Lambda}$	random rotation matrix	matrix, $N \times N$
$\mathbf{x}, \mathbf{y}$	state/observation vector ($\in \mathbb{X}/\mathbb{Y}$)	vector, K/L
$\mathbf{X}, \mathbf{Y}$	ensemble matrix ($\in \mathbb{X}/\mathbb{Y}$)	matrix, $K/L \times N$
$\mathbf{X}', \mathbf{Y}'$	ensemble perturbation matrix ($\in \mathbb{X}/\mathbb{Y}$)	matrix, $K/L \times N$
$\mathcal{H}(\cdot)$	observation operator	operator, $\mathbb{X} \to \mathbb{Y}$
$\mathcal{M}(\cdot)$	model operator	operator, $\mathbb{X} \to \mathbb{X}$
$J(\cdot)$	cost function	scalar function, $\mathbb{X} \to \mathbb{R}$

Chapter 1

Introduction

If you can look into the seeds of time, and say which grain will grow
and which will not, speak then unto me.[1]

The prediction of future events has always been an issue of great relevance. Modern science attempts to understand the intrinsic natural laws and processes, and to formulate them in an explicit, mathematical way. The solution of the resulting equations, usually numerically, allows to generate *model* predictions, based on current and past information (Feynman 1982). Thus, a look into the future is actually possible. Nowadays, forecasts are an important tool in a broad range of applications to enable decision making. The field that probably affects most people is weather forecasting. Each day, countless decisions are based on weather forecasts, and they also influence economical and political actions. For example, this concerns extreme weather events, which may even threat lives or cause substantial damages, but also climate change, as statistical statements about the future climate can be extracted.

It is evident that the improvement of forecast quality represents an important task for research. Weather predictions are, despite a good general understanding of the underlying physics, subject to many sources of uncertainty, mostly model shortcomings and imperfect input data. The atmosphere, as a nonlinear and chaotic system, is strongly sensitive to the *initial condition*, i.e., the state of the atmosphere that is used to initialize the model forecast. Even a perfect model would not be able to deliver reasonable predictions without good initial conditions. Their specification represents a major challenge and requires huge effort, often more than the forecast itself. In a more general interpretation, this issue can be reduced to the problem of *state estimation* in a dynamical system. The state-of-the-art method to obtain an optimal state estimate, also called the *analysis*, is *data assimilation* (DA). Its principal concept consists in the combination of the model with real-world observations. Numerous DA algorithms are available, but only a few are applicable to large-scale systems. They rely, at least partly, on a Gaussian assumption for the appearing probability distributions. Non-Gaussian and nonlinear methods are more appealing by offering a full Bayesian solution, but they usually fail in high dimensions. This work proposes a simple novel nonlinear technique that is also applicable in such scenarios, but is still as generic and easy to apply as standard methods.

[1] The bard in William Shakespeare's *MacBeth* (around 1606), asking for a useful prediction.

1.1 Background

Most of the concepts and algorithms discussed in this thesis are of generic nature and apply to any forecasting problem. However, the ideas are usually illustrated by the example of numerical weather prediction (NWP) as the primary field of application for DA. In principle, the system of equations that determine the atmospheric dynamics is known well. It is governed by the Navier-Stokes equations and the conservation of mass, energy and water content. The equations can only be solved numerically, which corresponds to an integration in time and is made possible by supercomputers. The finite spatial and temporal resolution requires to parameterize many important processes that occur on the sub-grid scale, such as radiation, cloud formation or turbulence. Additionally, many model parameters and necessary input data, such as the land surface characteristics, are only known poorly. These uncertainties contribute to limited predictive skill.

As the partial differential equations contain time derivatives, an initial condition is mandatory, i.e., the full atmospheric state at initial time of the forecast. However, the atmosphere is a chaotic system. As a consequence, any *error*, i.e., a deviation from the unknown true initial state, usually grows exponentially. This property, popularly known as the ”butterfly effect”, is the main reason that predictability of weather events is rather limited. Lorenz (1963) estimated the average deterministic predictability to two weeks. Of course, the actual error growth rate depends on the current flow. Probabilistic predictions, using an ensemble of single forecasts, can better explore the predictability limit, but do not solve the general problem. Therefore, weather would be hard to predict even if a perfect model was available.

As already mentioned, DA refers to the optimization of the initial state by combining the model forecasts with observations collected in the real world. Accounting for the inherent uncertainties, a valid description is given by probability density functions (pdfs). Then, in principle, this combination is determined by a fundamental and elegant equation, *Bayes' theorem*. The prior knowledge, which is typically a model forecast, is updated with a new piece of information, the observation, resulting in the analysis distribution. However, the practical application is not as straightforward, as the pdfs are formulated in a very high dimensional space (up to 10^9 in NWP), where the actual form of the pdfs is not known, and they also evolve in time. The underlying Fokker-Planck equation cannot be solved in praxis either. Additionally, models, but also observations, often exhibit limitations such as biases towards the truth or undetected cross-correlations. Therefore, assumptions have to be imposed for computational and practical reasons. At this point, different DA algorithms can be developed, based on different assumptions and implementations. Even with high-performance computers, the limited computational power puts a major restriction on the applicable methods. This development is still in progress and nowadays occupies many scientists day by day. In every operational weather center, DA represents a major work area in both research and operations. DA systems have significantly advanced in the past decades. The good predictive skill of the European Center for Medium-Range Weather Forecasts (ECMWF) is often attributed to their sophisticated analysis scheme and the efficient use of different observational data (Richardson et al. 2013).
Figure 1.1 visualizes the basic concept. Observations within the past hours are assimilated into a repeated model run. The final state constitutes the analysis, which is used to initialize the prediction into the future.

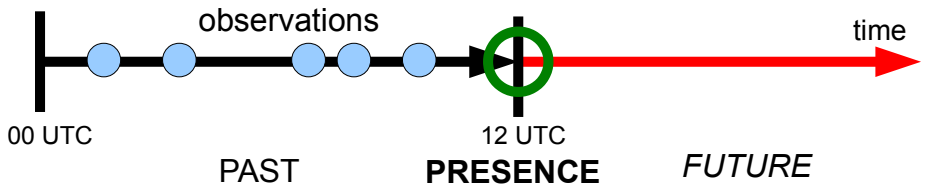


FIGURE 1.1: A schematic DA cycle. At 12 UTC, a forecast into the future is to be initialized. The states within an assimilation window in the past (here 00-12 UTC) are recomputed, assimilating the available observations into the model. The final analysis state at 12 UTC represents the optimal initial condition for the adjacent prediction.

It should also be mentioned that DA is not only used for forecast initialization. Its potential to estimate the state over a whole time range represents a useful tool in almost any application that deals with dynamical systems. For example, it has led to successful reconstructions of the atmospheric or oceanic states over extended periods. The resulting estimates, also called reanalyses, are an extremely valuable data source for climate studies and diagnostics (e.g., Bengtsson et al. 2007; Dee et al. 2011). They allow to verify or drive climate simulations, which in turn helps to identify model deficiencies. Additionally, the DA output can be used to diagnose systematic model or observation errors (e.g., Chiodo and Haimberger 2010; Haimberger 2007).

1.2 Short history of DA in NWP

At the beginning of the 20th century, Cleveland Abbe identified the equations that govern the atmospheric motion, and Vilhelm Bjerknes manifested the aim of solving them to obtain weather forecasts. The first attempt was made by Lewis Fry Richardson in 1917, based on a discretized version of the equations, but his forecasts yielded unrealistically large pressure tendencies. This failure can be related to imbalances in his initial conditions, for which he used pressure observations. With an appropriate initialization, his forecast would have performed much better (Lynch 1992), which already emphasizes the need for proper DA. Beginning in 1949, computers could be used to actually solve the underlying equations. Together with John von Neumann, Jules Charney solved a quasi-geostrophic vorticity equation to obtain large-scale circulation patterns in the earth's atmosphere. From then on, more and more advanced models were developed, in concurrence with the increasing computational power.
In the 1950's, the sparse observation network was a major challenge. Cressman (1959) suggested a successive interpolation scheme to correct existing model fields by weighting observations with their distance. Even though the weights are chosen subjectively, this approach already resembles the basic idea in modern DA. In the 1970's, *nudging* was a state-of-the art method (Hoke and Anthes 1976). It consists in adding a term to the prognostic equations to pull the model states closer to the observations at each time step. However, the determination of the nudging weights is subjective and a matter of tuning. Additionally, only prognostic model variables can be assimilated, but related quantities that are not explicitly part of the model state, such as satellite radiances, cannot be processed.
In the 1980's, the first modern, objective DA methods were developed and successfully applied, mainly optimal interpolation, variational methods and the Kalman filter. They are introduced in Chapter 2.

Since the 1990's, these methods have been further enhanced by ensemble techniques, which is the topic of Chapter 3. Nowadays, model complexity is still increasing, concerning both resolution and physics. Furthermore, the observation networks offer a variety of dense observations, particularly from remote sensing. Thus, the progress in DA will remain a key issue of NWP also in the future, particularly concerning the challenges set by high-dimensional, nonlinear models.

1.3 Motivation

In the previous section, it was outlined that DA represents an important component of a NWP system. Here, actual techniques to assimilate observations into a forecast model are briefly characterized. This motivates why the development of a new nonlinear algorithm represents a valid contribution to current research. As mentioned, the practical restrictions imply that many assumptions and approximations have to be made. The most common one consists in assuming the appearing pdfs to be Gaussian. Even though this is not appealing from a purely theoretical point of view, as nonlinear models necessarily generate non-Gaussian pdfs, the resulting algorithms often lead to satisfactory results, and nonlinearities can be considered at least partly.

Many developments concern the advancement of variational methods such as 4DVAR (Talagrand and Courtier 1987), which require the minimization of a high-dimensional cost function, and hence, a reliable tangent linear and adjoint model of the forward model have to be derived and maintained. 4DVAR can be classified as a smoother, as it aims at producing an analysis trajectory for a complete time window such that it represents an optimal fit to both the background state and all the observations. In contrast, filters only consider past observations and are usually applied in a sequential way by iterating a forecast step, which executes the forward model until observation time, and an analysis step, which turns the prior forecast state into a posterior estimate that contains observational information. The output of the analysis step is then reused as the initial condition for the next forecast step. Initialized by the introduction of the ensemble Kalman filter (EnKF, Evensen 1994), sequential, ensemble-based techniques now constitute a popular alternative to variational methods. Their particular advantage, at least for most state estimation applications, is the independence of the actual analysis algorithm of the specific forward model. Furthermore, the combination of both approaches into so-called hybrid systems is an active field of research, particularly for operational applications (e.g., Buehner et al. 2013; Lorenc et al. 2014).

The EnKF is based on the Kalman filter (KF, Kalman 1960), originally derived as an optimal solution (in a least squares sense) for linear models and linear observation operators. As this leads to Gaussian densities, the KF analysis step relies on the prior mean and covariance to compute the posteriori mean and covariance. The KF analysis can be interpreted as a linear regression of the model state on the observation vector (Anderson 2010). In order to apply it to nonlinear systems, several approximations have been suggested. They all attempt to obtain realistic estimates of the prior mean and covariance within the forecast step. The extended KF employs a linearized propagation of the error covariance matrix, which can lead to unrealistic results in nonlinear problems (Miller et al. 1994). The EnKF offers a distinct perspective by estimating the prior moments from a preceding ensemble forecast. This can be interpreted as a Monte Carlo approximation for solving the underlying Fokker-Planck equation

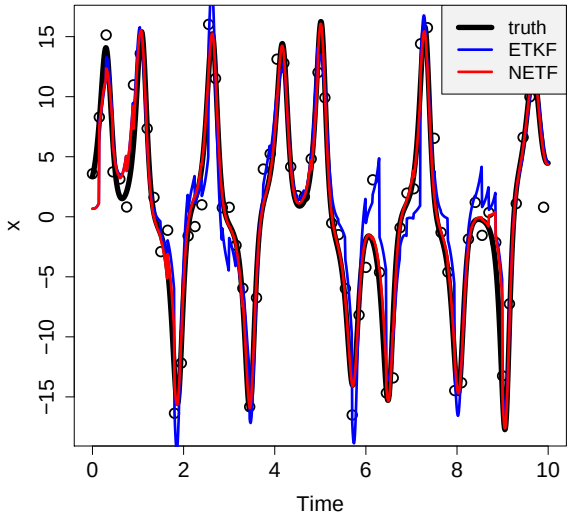


FIGURE 1.2: DA in the strongly nonlinear Lorenz-63 model in an identical twin experiment (see also Chapter 5). Shown is the temporal evolution of the x component for the truth (black), the ETKF (blue) and the new nonlinear filter (NETF, red), together with the observations (dots).

(Evensen 2009). However, the update step remains Gaussian, since it utilizes the usual KF equations to determine the mean and covariance of the analysis ensemble.

The classical EnKF requires a perturbation of the observations in the analysis step in order to maintain the correct covariance (Burgers et al. 1998), which results in suboptimal behavior because of sampling errors, particularly for small ensemble sizes. As alternative, so-called deterministic EnKFs were suggested. They generate an analysis ensemble with the desired properties without perturbing the observations (Anderson 2001; Bishop et al. 2001; Evensen 2004; Ott et al. 2004; Whitaker and Hamill 2002). Most of them belong to the class of square root filters (Nerger et al. 2012b; Tippett et al. 2003) as they require the computation of a matrix square root in order to determine an appropriate transformation matrix to be applied to the prior ensemble.

In a nonlinear, non-Gaussian scenario the linear update equations of the EnKFs necessarily produce an analysis where the mean and covariance are biased, resulting from the assumption of a Gaussian prior pdf, which, in general, is not satisfied. An illustrative example is given by Figure 1.2, showing the results of an identical twin experiment in the Lorenz-63 system. It emphasizes that the ETKF performs sub-optimally in this strongly nonlinear environment, while the new nonlinear filter that is derived in this work apparently is able to track the truth well. The corresponding experiment will be discussed in detail in Chapter 5.

A distinct approach to the EnKFs is represented by the particle filter (PF), first published by Gordon et al. (1993). Here, the properties of the analysis, mostly moments, are estimated directly via Bayes' theorem. The PF only requires some mild assumptions (Doucet et al. 2001, ch. 2) about the form of the prior pdf and does not restrict it to a particular type. Therefore, it can be regarded as a fully nonlinear, non-Gaussian filter. Even though this seems appealing, its application to high-dimensional systems is not straightforward due to the so-called curse of dimensionality (Snyder et al. 2008). In a Bayesian

update step in a high-dimensional space with many independent observations, many particles are likely to fall outside the region of significant probability mass and receive negligible weights, and therefore, their information is effectively lost, which results in a filter collapse. The review by van Leeuwen (2009) summarized different PF variants and approximations, whereas a more recent development, which is the equivalent weights PF (EWPF, van Leeuwen 2010), seems to be a promising approach. It relies on a stochastic nature of the forward model in order to pull the particles towards the observations, and to ensure almost-equal weights by employing suited proposal densities. Other recent attempts to improve the PF also apply a form of residual nudging (Luo and Hoteit 2014a) or variational methods (implicit PF, e.g., Atkins et al. 2013).

During the past years, there has been research activity concerning the development of algorithms that attempt to address the drawbacks of both the EnKF and the PF. A popular approach, initiated by Anderson and Anderson (1999) and Pham (2001), is the representation of the forecast ensemble as a Gaussian mixture, allowing a better consideration of nonlinear features contained in the ensemble (e.g., Frei and Künsch 2013a,b; Hoteit et al. 2012, 2008; Stordal et al. 2011). Other authors use the EnKF as a proposal for a PF (Papadakis et al. 2010) or apply linear programming methods (Reich 2013). However, to the author's knowledge[2], the EWPF is the PF-based algorithm that has proven to be applicable in large-scale scenarios, as it worked in a stochastically forced primitive equation model (Ades and van Leeuwen 2014).

Lei and Bickel (2011) suggested the nonlinear ensemble adjustment filter (NLEAF) which uses a stochastic update mechanism designed such that the analysis mean and covariance are unbiased. However, sampling error as an additional source of uncertainty is introduced. This becomes more apparent for relatively small ensemble sizes, as inevitable even in moderate dimensions, and the algorithm exhibits high computational costs in these cases.

In summary, much progress towards nonlinear high-dimensional DA has been made. The EWPF has much potential, but its forecast step requires a valid model error formulation, as the proposal densities are not utilizable in deterministic systems, and sufficient tuning is needed (Ades and van Leeuwen 2013). In contrast, the NLEAF represents a more generic alternative by neglecting higher-order moments of the analysis pdf, but its shortcomings hamper its large-scale applicability.

1.4 Aims

The previous section has given an overview to the scientific progress in nonlinear DA and revealed current challenges. Therefore, the principal aim of this work is to offer a novel contribution to this field by deriving an objective, generic filter algorithm that relaxes some of these issues and becomes applicable in high dimensions as well.

For a better organization of the research conducted within this thesis, three consecutive working stages with different focusses are defined. In the following, the aims for each stage are specified.
First, a thorough review of DA concepts and algorithms reveals state-of-the-art techniques and their properties, with the following aims:

[2] Also stated on `http://www.nceo.ac.uk/PFtools/`, hosted by a collaboration of various UK institutions.

- Definition of a self-consistent DA framework that allows to describe all relevant aspects and objective algorithms in probabilistic terms.
- Embedding of classical solutions (KF and variational methods) within the general framework to enable a better integration of more recent developments.
- Presentation of ensemble-based techniques (EnKF, PF) that partly account for the probabilistic nature of DA in order to highlight their advantages and practical issues, as they serve as basis and reference for the new ensemble filter.
- Discussion of recent developments in nonlinear DA in order to identify the current stage of research and open issues, which in turn gives directions for potential progress.

The second stage concerns the theoretical formulation of the new algorithm, aiming at:

- Identification of the paradigms for the new filter and its discrimination against other methods.
- Objective derivation of the equations to update the ensemble mean and perturbations in resemblance to the derivation of the ensemble transform Kalman filter (ETKF) as a current standard method.
- Investigation of mathematical properties, in particular, confirmation that the presented algorithm is able to generate a valid analysis ensemble.
- Deduction of the expected behavior and implementation issues, based on the experiences from established ensemble filters.
- Assessment of the computational complexity of the algorithm, particularly with regard to applications in high-dimensional systems.

Third, the applicability and potential usefulness of the suggested method has to be demonstrated empirically, with the following aims:

- Investigation of the filter performance in typical DA test beds of different dimensions and dynamics, focussing on stability properties and the performance in comparison with other ensemble-based filters in order to explore its benefits and limitations.
- Assessment of the behavior in larger-dimensional toy models with limited ensemble sizes in order to establish the potential large-scale applicability.
- Confirmation of the filter's large-scale applicability with a challenging assimilation problem in a high-dimensional general circulation model.
- Extraction of further conclusions from the filter properties and empirically derived findings to form a complete picture of its potential usefulness.

In combination, the aims declared for the three stages allow to achieve the general objective named in the beginning of this section.

1.5 Outline

Based on the scientific objectives presented above, the remainder of this thesis is organized as follows:

- Chapter 2 formulates the fundamentals of DA from a stochastic point of view and establishes a self-consistent probabilistic framework. Then, classical approaches based on Gaussian assumptions are reviewed. The discussion focusses on variational methods and the KF.
- Chapter 3 continues this review by turning towards more recent advances made in ensemble-based filtering. Important supplements to the EnKF are presented, which are required for successful applications in real-world problems, such as localization and inflation. Then, the PF as a non-parametric alternative is introduced. This allows to discuss rather new research findings concerning nonlinear ensemble DA, with a focus on the NLEAF and EWPF. These considerations allow a plain passage to the new algorithm.
- Chapter 4 as the heart of this thesis presents the nonlinear ensemble transform filter (NETF). The Bayesian expectations are reformulated and utilized to derive the new algorithm. Afterwards, the method is highlighted from a theoretical point of view by discussing several properties and aspects that are potentially relevant for applications.
- Chapter 5 contains several applications of the NETF to scenarios of different dimensionality and nonlinearity. These structured experiments allow a multi-faced characterization of its performance properties, particularly in comparison with other ensemble filters.
- Chapter 6 extends the empirical characterization of the NETF by a more complex application to a large-scale ocean circulation system with a realistic observation scenario. After describing the system and the setup, the applicability of the NETF in such a high-dimensional model is demonstrated by evaluating its resulting analyses.
- Chapter 7 finishes this thesis by drawing overall conclusions about the novel nonlinear filter algorithm and its implications concerning applications. This also involves a brief summary discussion on its possible advantages and limitations as well as an outlook for directions of potential continuative research paths.
- Appendix A presents some background about the mathematical methods applied throughout the thesis. Appendix B shows derivations and identities concerning the KF which are not covered in the main part. Appendix C presents the explicit algorithms to generate random rotation matrices and initial ensembles. The supplement contains a summary of the thesis in German.

The publication by Tödter and Ahrens (2015) is based on contents of this thesis, in particular, Chapters 4 and 5. A publication of the experiments and results shown in Chapter 6 is under preparation and will be submitted presumably in spring 2015.

Overview

The following flowchart, figure 1.3, presents a concise overview to the structure of this thesis (without the appendices).

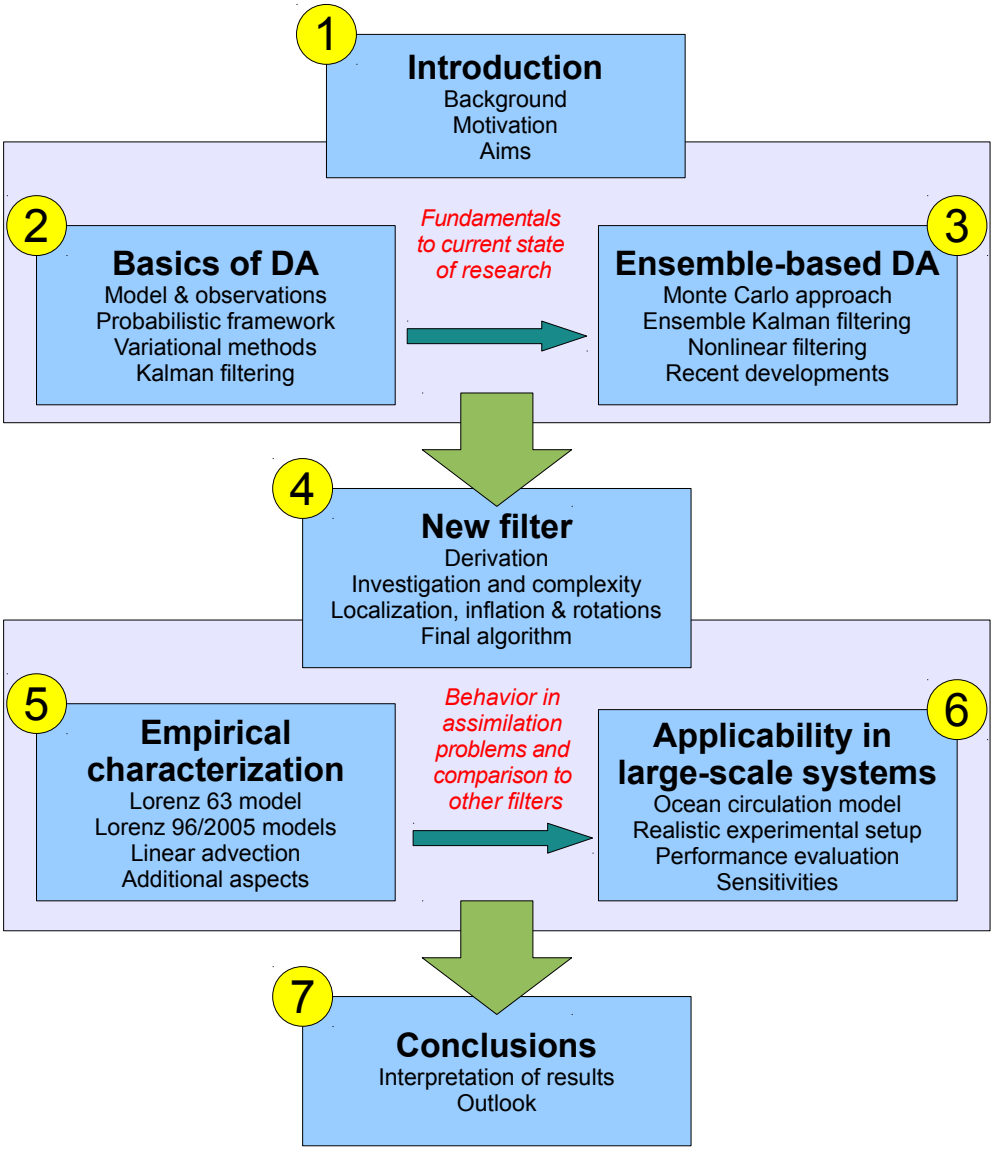


FIGURE 1.3: Flowchart for a quick overview of the contents of the thesis. The numbers in the yellow circles indicate the corresponding chapters.

Chapter 2

Data assimilation: Fundamentals and classical methods

This chapter sets the fundamentals for the presented work by introducing the basic DA concepts and solutions in a concise way. After introducing the basic terms and tasks, a probabilistic framework is established which allows to formulate DA in very general way. Its further exploration then leads to the variational approach and Kalman filtering, which are referred to as the classical methods. They play a major role particularly for prevailing operational applications and are also subject to further development. The objective of the structured presentation given in this chapter is two-fold. First, it provides a sound overview of DA and second, it serves as an initial background and motivation for the new filter algorithm that is at the heart of this thesis. For that reason, this chapter also focusses on a multi-faced interpretation and discussion of the presented techniques.

2.1 Illustration of the principal challenge

DA can be interpreted as a part of estimation theory with the principal task to combine model forecasts with real-world observations in order to achieve an optimal estimation of the state of a dynamical system. Additionally, it may be augmented to estimate static parameters that influence the model dynamics. DA represents a relatively advanced area of research, offering various different approaches. Even though here the ideas are illustrated in the field of numerical weather prediction (NWP) or ocean modeling, one should keep in mind that the methods are of general nature and apply to many other fields as well. Naturally, the relevance of particular issues depends on the area of application. For example, in NWP, the greatest challenge is imposed by dealing with the chaotic dynamics in a very high-dimensional state space. Hence, the restricted computational capacities directly prohibit many other approaches that might work well for small-scale problems.

In NWP, predictability is inherently strongly limited by the chaotic nature of its underlying complex dynamics which leads to an exponential error growth (Lorenz 1963). As a consequence, even a perfect model would not be able to generate useful forecasts as long as it is not initialized realistically, which clearly represents a nontrivial task considering the high dimensionality of the state space. It may for

example cover all atmospheric variables in the global atmosphere. The high sensitivity to the initial conditions is the reason that a well-tailored DA system is a key component in NWP. Its primary task is to supply the initial conditions, based on the model dynamics and past observations.

The development of the global observing system together with advanced DA methods have a large share in the improvement of weather forecasts throughout the previous decades (e.g., Richardson et al. 2013). However, the problem will remain challenging in the future, as models complexity still increases. First, finer resolution increases the state dimension even further, and second, state-of-the art models consider the whole earth system with the interactions between different components such as atmosphere, ocean, sea ice and land. Third, new types of observations have be assimilated both correctly and efficiently. Additionally, applications for medium-range climate predictions, such as seasonal or even decadal forecasts, as well as ensemble forecasts in general, also ask for a proper initialization, opening new fields of application. In conclusion, DA represents a very dynamic and interesting field of research. A lot of progress has already been made in the past. Yet, numerous open question still wait to be answered.

2.2 Basic quantities

At first, the quantities of interest in the field of DA are introduced. In this context, the basic notation, to be used throughout this thesis, is defined.

2.2.1 Notation

The mathematics of DA can quickly become rather complex as it involves many different variables and types (scalars, vectors, matrices, probability distributions,...) and consists of a lot of different approaches and methods. Therefore, a consistent and well-defined notation is essential when introducing and comparing DA techniques. A widely accepted attempt to unify the DA notation was established by Ide et al. (1997), but it has to be extended to include ensembles (see Chapter 3) as a relatively new degree of freedom in the DA world. The following notation is used throughout the complete thesis, unless otherwise noted. Further extensions will be inserted only when necessary. This concerns both symbols and the usage of indices. The general rules are as following:

- Vectors are marked by bold, lowercase letters, e.g., $\mathbf{x}$ or $\mathbf{y}$.
- Matrices are marked by bold, uppercase letters, e.g., $\mathbf{X}$ or $\mathbf{T}$.
- Operators are marked by uppercase Greek letters, e.g., $\mathcal{H}$ or $\mathcal{M}$.
- Vector spaces are represented by double, uppercase letters, e.g., $\mathbb{X}$ or $\mathbb{Y}$.
- Ensemble members are indicated by an upper index, usually n or m, e.g., $\mathbf{x}^n$.
- As time is usually discretized, the time level is indicated by the lower indices i and j. For further convenience, different time levels are summarized with a colon notation, e.g., $\mathbf{x}_{i:j}$ (with $j \geq i$) is to be understood as the set $\{\mathbf{x}_i, \mathbf{x}_{i+1}, \ldots, \mathbf{x}_j\}$.

- In a few cases, the vector formulation will not be sufficient and the vector components are needed explicitly. Then, a second lower index k is used, or the nature of the lower index becomes clear from the context.

For example, $x^n_{i,k}$ represents the k-th component of the state vector of ensemble member n at time level i, while $\mathbf{x}^n_i$ is the full state vector at time t_i. The corresponding continuous field would be $\mathbf{x}^n(\mathbf{r}, t)$, where $\mathbf{r}$ represents the spatial coordinates.

2.2.2 State space

2.2.2.1 State and model

The *state* of a system is given by a collection of variables which describe the relevant properties of the system at a single time. Usually, the state is dynamical, i.e., it varies with time. For example, in NWP, a state refers to a certain atmospheric condition and can be expressed by temperature, pressure, wind and humidity at each global position. The *state vector* collects all these variables into one single vector $\mathbf{x}$. The *state space* $\mathbb{X}$ is the space of all possible model states, and can be of very large dimensionality. This thesis restricts to the case where the state space is of finite dimensionality K, as follows from a spatial discretization. Then, this dimension equals the number of components in the state vector.

A state $\mathbf{x}$ only represents a snapshot of the system at a given time, while for most applications the temporal development of the state is of interest. This task is completed by the *model*, the basic component of any forecasting scheme. In general, it allows to evolve a state in time. For example, in NWP, regional or global atmospheric models are applied to integrate a given atmospheric condition. In physics and meteorology, or in geosciences in general, the underlying dynamics is typically governed by partial differential equations (PDEs) that express the underlying physical laws. The model represents a certain approximative representation of these equations, as the PDEs are usually discretized in space and time, and need to be extended by parametrizations for unresolved processes.

2.2.2.2 Model integration

In geosciences, the prognostic physical equations are typically of first order in time. Thus, the temporal evolution of a model state requires an *initial condition*, as the dynamical tendencies only depend on the previous state. More abstractly, the model can be regarded as a nonlinear operator that acts on a state vector $\mathbf{x}_i$, thereby propagating it one time step ahead:

$$\mathcal{M}_{i\to i+1} : \mathbb{X} \to \mathbb{X} \quad , \quad \mathbf{x}_{i+1} = \mathcal{M}_{i\to i+1}\left(\mathbf{x}_i\right) \tag{2.1}$$

Here, the model operator is allowed to be time-dependent. Whereas usually the physical equations used to advance a model state are actually time-invariant, a time dependency may be introduced by boundary conditions and external parameters, such as the forcing by solar radiation or at the lateral boundaries. However, the time index of the model operator can usually be dropped for reasons of convenience, since this temporal dependency is not relevant for most DA aspects.

This model integration over a finite number of I time steps can be factorized as follows, representing the repeated application of the single-time-step model operator:

$$\mathbf{x}_I = \mathcal{M}_{0\to I}(\mathbf{x}_0) = \mathcal{M}_{I-1\to I}(\dots \mathcal{M}_{1\to 2}(\mathcal{M}_{0\to 1}(\mathbf{x}_0))) \tag{2.2}$$

In assimilation algorithms, sometimes a linearized version of the model operator is needed. The first-order Taylor approximation around a certain state $\tilde{\mathbf{x}}$ is:

$$\mathcal{M}(\mathbf{x}) \approx \mathcal{M}(\tilde{\mathbf{x}}) + \mathbf{M}(\mathbf{x} - \tilde{\mathbf{x}}) \tag{2.3}$$

where matrix $\mathbf{M}$ is the Jacobian, also called the *tangent linear model*, evaluated at $\tilde{\mathbf{x}}$ (see Appendix A).

2.2.2.3 Model error

Equation (2.1) actually refers to a *deterministic* system, as the model state at any time $t_i > t_0$ is entirely determined by the initial state $\mathbf{x}_0$, which allows to produce a unique trajectory as in equation (2.2). In fact, most large-scale circulation models are based on the laws of classical physics, such as the Navier-Stokes equations and conservation of mass and energy. These laws are inherently deterministic. However, any model is only an approximation to the real-world behavior, due to numerous sources of uncertainty. They are introduced by errors arising for example from finite resolution in space and time, approximate parameterization schemes, imperfect boundary data and parameters or neglected physical processes. An important example for model error is the feedback of unresolved scales to the grid-scale variables. While these processes, such as turbulence, are approximated by parameterization schemes, they will never be quantified exactly. Another example are radiative processes or cloud microphysics that take place on a molecular scale. Consequently, even if an initial state was known exactly, there would still remain an inherent uncertainty in the model output of the next time step. This uncertainty is a result of the so-called *model error*, and it includes all contributions that render the model imperfect. Therefore, the model integration should actually be written as

$$\mathbf{x}_{i+1} = \mathcal{M}_{i\to i+1}\left(\mathbf{x}_i\right) + \boldsymbol{\eta}_i \tag{2.4}$$

where $\boldsymbol{\eta}_i$ is a random vector and represents a sample of the model error distribution. Typically, the model error is assumed to be normally distributed with zero or nonzero mean (allowing a model bias) and a covariance matrix $\mathbf{Q}_i$. The formulation (2.4), which is the discrete form of an Itô stochastic differential equation (Miller et al. 1999), actually assumes that the model error is flow-independent and not correlated in time, which is a probably not true for atmospheric models (Daley 1992). However, hardly anything is known about the model error statistics and its properties, and even the implementation of such a crude formulation is a difficult issue (Ades 2013; Houtekamer and Mitchell 2009; Tremolet 2007). In the context of this thesis, the basic formulation with $\boldsymbol{\eta}_i \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_i)$ is sufficient. A detailed treatment of model error is not required here, as this work mainly restricts to deterministic systems as mostly used in operations and research.

2.2.3 Observation space

2.2.3.1 Observations

The state space fully describes the system of interest. However, as discussed, any model forecast necessarily exhibits deviations from the true state due to multiple sources of uncertainty. Observations are real-world measurements of the system being modeled. They represent an independent source of information. In DA, this information is employed to constrain model forecasts in order to obtain an optimized state estimate. Observations are also valuable data for model evaluation, i.e., the assessment of its performance properties, which in turn supports further developments of the model, or for post-processing, i.e., the correction of model output. Let the L-dimensional *observation space* $\mathbb{Y}$ represent the set of possible observations and $\mathbf{y}$ the observation vector that contains all single observations at a given time. Often, the observation space is time dependent, in contrast to the state space which is mostly fixed. For example, a satellite measures different parts of the globe at different times or may not be able to observe a region due to clouds. However, in the context of presenting DA techniques this time dependency can usually be treated implicitly as it does not affect the algorithms used.

2.2.3.2 Observation operator

In most real-world applications, the observation space does not match the state space for two reasons. First, not all state variables are observable, and second, the observations may be connected to the state variables in an indirect way. For example, satellite radiances clearly contain information about the atmospheric state, as the measured radiance depends on atmospheric conditions. However, they are not part of the state space nor used anywhere in the model. Hence, it is essential to link the observations to the actual state variables, which is the task of the so-called *observation operator*. It maps from state into observation space:

$$\mathcal{H}_j : \mathbb{Y}_j \rightarrow \mathbb{X} \quad , \quad \mathbf{y}_j = \mathcal{H}_j(\mathbf{x}) \tag{2.5}$$

The observation operator outputs the observation vector that matches the input state, i.e., it answers the question *"Given the system's state is* $\mathbf{x}$*, what would the corresponding observation* $\mathbf{y}$ *be?"* This operator plays an essential role in DA, as most algorithms work by comparing the model-equivalent of the observation with the actual, real-world observation in order to decide how to adjust the model state. In general, $\mathcal{H}(\cdot)$ is nonlinear. For example, it may contain interpolations of the state or even complex calculations, like a full radiative transfer scheme. Similarly to the model operator in equation (2.3), the observation operator can be linearized using its tangent linear version $\mathbf{H}$, i.e., $\mathcal{H}(\mathbf{x}) \approx \mathcal{H}(\tilde{\mathbf{x}}) + \mathbf{H}(\mathbf{x} - \tilde{\mathbf{x}})$.

2.2.3.3 Observation errors

Even though observations represent an extremely valuable supplement to models, it has to be emphasized that they are not a perfect representation of the real world either. Both observations and the observation operator are subject to errors of different sources:

- Measurement error: A measurement usually contains a random and a systematic error which depend on the technique used. While the latter can be removed by post processing methods such as bias correction, the former is of statistical nature and therefore irreducible.
- Representation error: The mapping from state to observation space represents another source of error, for example, due to the interpolation of model variables or the fact that observation operators such as for radiative transfer are often imperfect models as well.

As a consequence, the observation equation (2.5) should be written more consistently as

$$\mathbf{y}_j = \mathcal{H}_j(\mathbf{x}) + \mathbf{e}_j \tag{2.6}$$

where $\mathbf{e}_j$ is a sample of the observation error distribution. The specification of its characteristics is a nontrivial, yet highly relevant task under active research (Bocquet et al. 2010), because the observation error statistics determines the weight the observation achieves in a DA procedure. An under- or overestimation of the error magnitude necessarily leads to an over- or underconfident adjustment of the model forecasts, hence preventing the optimal usage of the observations (Fowler and van Leeuwen 2013).

Usually, one assumes that the observation errors follow a Gaussian distribution with the observation error covariance matrix $\mathbf{R}_j$, $\mathbf{e}_j \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_j)$, and at least the methods presented in this chapter require this assumption. The diagonal elements of $\mathbf{R}$ represent the variances of each component, while the off-diagonal elements specify their correlations. In case different observation systems are merged into the observation vector, it is obvious that the matrix $\mathbf{R}$ is block-diagonal. However, the specification of error cross-correlations is not straightforward. For example, it can be expected that observation errors of a satellite measurement exhibit an articulate spatial dependency. Modeling of these error statistics is an open topic of research within the DA world (e.g., Miyoshi et al. 2013). An advantage of nonlinear DA methods (see Chapter 3) is that they offer much more flexibility in this context by allowing almost any error distribution. This can become an important issue when dealing with inherently non-Gaussian observation errors that can appear in real-world applications. For example, for bounded quantities such as precipitation or cloud fraction, Gaussianity is evidently not appropriate (Anderson 2010).

2.3 The data assimilation problem

2.3.1 Definition

In the previous sections, the DA problem and its relevant components were discussed in qualitative way. Aiming a more quantitative formulation of DA techniques, it is necessary to formally define the problem to be solved. In the remainder of this thesis, this definition will be referred to repeatedly. The specifications are as following, and figure 2.1 visualizes this situation on a conceptual time line.

- The assimilation window is fixed in time, $[t_0; t_I]$. At time t_0, an initial estimate about the model state, called the background, usually is available.

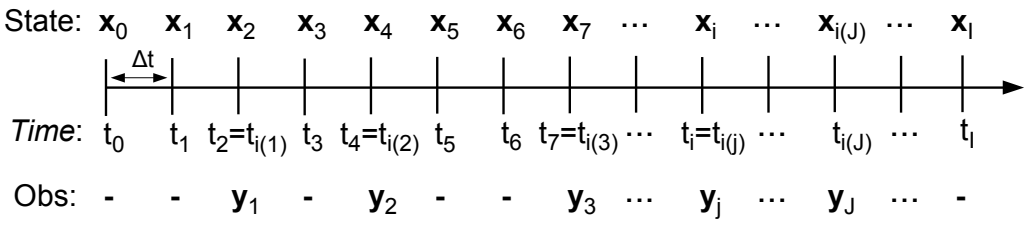


FIGURE 2.1: The assimilation time window (inspired by Evensen 2009, Fig. 7.1). At each time level t_i $(i = 0 \dots I)$ a model state $\mathbf{x}_i$ is defined. Observations can be distributed irregularly in time, and $i(j)$ specifies the time level at which observation $\mathbf{y}_j$ $(j = 1 \dots J)$ is valid. In general, the assimilation window exceeds the observation time range, i.e., $i(1) > 0$ and $J < I$.

- The model state is discretized in time with a time interval Δt, which typically corresponds to the discretization time step of the model. Thus, there are $I + 1$ time levels $t_i, i = 0 \dots I$, which are connected through the model operator as defined in equation (2.4). The model state at time t_i is denoted as $\mathbf{x}(t_i) \equiv \mathbf{x}_i$.

- A number of J observations $\mathbf{y}_j$ are available at respective time levels $i(j), j = 1 \dots J$, where $0 < i(1) < i(2) < \dots i(J) < I$. The mapping $i(j)$ specifies at which time level the j-th observation is valid. At each observation time, an observation operator $\mathcal{H}_j$, see equation (2.6), is available. The need for further specifications, e.g., an error covariance $\mathbf{R}_j$, depends on the actual algorithm. For convenience, $i(0) \equiv 0$ is defined even though no observation is available at t_0.

The principal aim of DA consists in finding an optimal estimate for the description of the states at all time levels that contains the observational information. The type of description depends on the method. In the simplest case, it is given by one point estimate at each time, $\hat{\mathbf{x}}_i$, but it may also refer to a full pdf. One distinguishes two types of solution, visualized in figure 2.2:

- A *smoother* finds the continuous model trajectory that fits best to a given background state and all the observations. The estimate at any time is influenced by past and future observations.

- In contrast, a *filter* produces estimates that, at a given time, only depend on past observations. Therefore, the filter trajectory is discontinuous at observation times, at which the state is updated by the new information.

In praxis, the problem is further simplified by introducing more restrictions, such as Gaussian assumptions for the error statistics, or the linearity of involved operators. However, a better insight into the specific algorithms is gained by interpreting them in an entirely general context at first.

2.3.2 Principal approaches

The aim is to formulate a principle solution to the just defined problem of state estimation. In this thesis, the view is held that the most general approach consists in a *probabilistic interpretation*, and the framework is based on the pioneer work of van Leeuwen and Evensen (1996). The key issue of DA is to find ways of optimally dealing with uncertain quantities, and it was already motivated that all components of

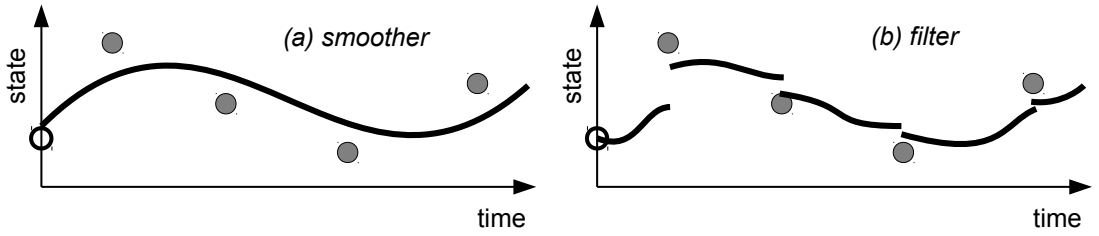


FIGURE 2.2: Conceptual form of the different DA solutions, shown is the state trajectory versus time t. The gray points represent observations, and the black circle the best estimate at initial time t_0. (a) The smoother yields a continuous trajectory throughout the assimilation window. (b) The filter begins at the initial estimate and adjusts the state each time observations are available, i.e., at the analysis steps.

a DA system are inherently subject to various sources of uncertainty. The theory of probability offers a very general, suitable framework to deal with uncertainty by the means of pdfs. The probabilistic point of view is not only useful in the context of DA, but also in many other fields related to forecasting. For example, forecast evaluation can be approached similarly, leading to a coherent verification framework that is capable of incorporating many relevant aspects of forecast quality and value (e.g., Gneiting et al. 2007; Jolliffe and Stephenson 2011; Tödter 2011). Another example concerns the initialization of ensemble forecasts, which can be interpreted as sampling from an underlying pdf. Hence, a probabilistic DA framework will be established that comprises all objective algorithms appearing in this thesis. It will always be pointed out which assumptions are added to the most general solution.

Not surprisingly, there are multiple ways to interpret the assimilation problem and to obtain a corresponding framework. While they offer distinct interpretations and have different implications, they often lead to similar explicit algorithms, given matching basic assumptions, e.g., for the underlying distributions (e.g., Bollmeyer and Hense 2014). Before proceeding to formulate the probabilistic DA framework, another well-known framework, *inverse problem theory* (Aster et al. 2005; Bennett 2002; Tarantola 2005; Yaman et al. 2013), is mentioned which has applications throughout a large variety of fields, e.g., in geophysics, neural field theory, remote sensing, etc. It considers the observation operator itself as a "forward model" which outputs the measurement, given a certain state. The actual observation controls the state, and the corresponding inverse problem is considered as finding a state that exhibits the "best" match with the given observation. In principle, a direct solution would be the *inversion* of the observation equation (2.5), $\mathbf{x} = \mathcal{H}^{-1}(\mathbf{y})$. However, this is impossible, as $\mathcal{H}^{-1}$ is typically unstable and unbounded for the following reasons (Aster et al. 2005):

- The dimension of $\mathbf{y}$ is usually much smaller than of $\mathbf{x}$ and hence, $\mathcal{H}$ is not injective.
- The observations are subject to errors as well and thus, $\mathcal{H}$ is not surjective. For any true state, usually an infinite number of possible observations exists.
- Errors coming from model, observations, observation operator and previous errors can accumulate in a dynamical system.

As a consequence, the problem is severely ill-posed, i.e., it does not have a unique solution and there may be a high sensitivity of the solutions towards small changes in the input parameters. To overcome this issue, some kind of *regularization* is required. The latter is often represented by providing a

background estimate of the model state. Furthermore, the optimal solution is defined with respect to a specific measure. For example, the well-known Tikhonov regularization (Aster et al. 2005; Tikhonov and Arsenin 1977) replaces $\mathcal{H}^{-1}$ by a stable version, which directly leads to the least-squares approach also appearing in variational DA (see section 2.5).

2.4 Probabilistic framework

As motivated, the general approach to DA consists in interpreting all uncertain quantities, in particular the system's state and the observation, as random variables. They are described by *probability density function* (pdfs), which are denoted by $p(\cdot)$ and contain all knowledge about the variables. In Appendix A, an overview of basic properties of pdfs is given.

2.4.1 Bayes' theorem

In DA, two sources of uncertain information exist. The model provides an estimate of the full system state, at any time level. This knowledge is contained in the *prior* pdf $p(\mathbf{x})$, i.e., an unconstrained estimate of the system's state. The observations are linked to the system and represent an additional constraint. This information can be interpreted as a conditional *likelihood* pdf $p(\mathbf{y}|\mathbf{x})$. It describes the distribution of the observations, given the system is in an arbitrary state $\mathbf{x}$. Finally, in probabilistic terms, the aim of DA is to obtain an updated estimate of the state that depends on the observations, as expressed by the conditional distribution $p(\mathbf{x}|\mathbf{y})$. It is related to the input pdfs by *Bayes' theorem*:

$$p(\mathbf{x}_{0:I}|\mathbf{y}_{1:J}) = \frac{p(\mathbf{x}_{0:I})p(\mathbf{y}_{1:J}|\mathbf{x}_{0:I})}{p(\mathbf{y}_{1:J})} \propto p(\mathbf{x}_{0:I})p(\mathbf{y}_{1:J}|\mathbf{x}_{0:I}) \tag{2.7}$$

The denominator can be neglected safely, as it only normalizes the right-hand side and is independent on $\mathbf{x}$, thereby justifying the proportional sign. Equation (2.7) quantifies how the prior pdf is updated by the observational information. Hence, from a purely probabilistic point of view, the whole DA problem can be expressed as the simple multiplication of prior and likelihood pdf. In this form, DA is not an inverse problem, but "just" a multiplication problem (van Leeuwen 2010). However, this does not imply all efforts can be stopped at this point. The pdfs are formulated in a high-dimensional space and are usually not given in a closed form, in fact, very little knowledge about their form is available. For instance, the prior pdf, $p(\mathbf{x}_{0:I})$, contains the dynamical evolution of uncertainty as imparted by the nonlinear, often chaotic, model. Also, all temporal and spatial connections of state and observations are contained in equation (2.7). Fortunately, some typical additional properties of the two pdfs involved can be utilized to simplify the most general formulation before turning to explicit algorithms that are based on such a formulation.

2.4.2 General smoother solution

Equation (2.7) simultaneously represents the state and observation vectors of all time levels, i.e., it deals with joint pdfs. As motivated in section 2.2.2, the underlying physical equations are usually of

first order in time. As a consequence, the model state at a specific time only depends on the model state at the previous time step, or, more generally speaking, the model can be interpreted as a *first order Markov process* where the random variable, here the state $\mathbf{x}$, is stochastically independent on all previous realizations expect the last one. This holds as long as the model is deterministic or the model errors are not correlated in time. First, the joint prior pdf is reformulated in terms of conditional densities (see Appendix A) and then, this important property is employed:

$$p(\mathbf{x}_{0:I}) = p(\mathbf{x}_0)p(\mathbf{x}_1|\mathbf{x}_0)p(\mathbf{x}_2|\mathbf{x}_{0:1})\cdots p(\mathbf{x}_I|\mathbf{x}_{0:(I-1)}) \tag{2.8}$$

$$= p(\mathbf{x}_0)\prod_{i=1}^{I} p(\mathbf{x}_i|\mathbf{x}_{0:(i-1)}) = p(\mathbf{x}_0)\prod_{i=1}^{I} p(\mathbf{x}_i|\mathbf{x}_{i-1}) \tag{2.9}$$

The conditional distribution $p(\mathbf{x}_i|\mathbf{x}_{i-1})$ is referred to as the *transition pdf*. It is concerned with the probabilistic evolution of the dynamic state pdf.

Next, the joint likelihood pdf, the second term in equation (2.7), is reformulated. As defined in section 2.3, the joint observation vector $\mathbf{y}_{1:J}$ can be partitioned into the individual measurements $\mathbf{y}_j$ at distinct times $t_{i(j)}$. It is reasonable to assume that the observation at a certain time $t_{i(j)}$ only depends on the state at that time, $\mathbf{x}_{i(j)}$. Additionally, a typical assumption is that the observation errors are not correlated in time. These considerations allow a factorization of the joint likelihood pdf as following:

$$p(\mathbf{y}_{1:J}|\mathbf{x}_{0:I}) = \prod_{j=1}^{J} p(\mathbf{y}_j|\mathbf{x}_{i(j)}) \tag{2.10}$$

Finally, insertion of the factorizations (2.9) and (2.10) into the general formula (2.7) yields a revised formulation of Bayes' theorem that is more convenient:

$$p(\mathbf{x}_{0:I}|\mathbf{y}_{1:J}) \propto p(\mathbf{x}_0)\prod_{i=1}^{I} p(\mathbf{x}_i|\mathbf{x}_{i-1})\prod_{j=1}^{J} p(\mathbf{y}_j|\mathbf{x}_{i(j)}) \tag{2.11}$$

This result represents the fundamental equation of DA from a probabilistic point of view. However, it still remains the question how to use this equation in praxis. In principle, one could apply it directly and attempt to find a solution for the whole time window at once, yielding a smoother solution. This simultaneous processing of all observations leads to the variational approach discussed in section 2.5. However, before showing explicit algorithms to solve equation (2.11), a further manipulation of Bayes' theorem is explored by splitting the generic solution into a subset of smaller tasks that can be solved in an iterative way, which leads to the sequential approach. The latter can be applied to both filtering and smoothing problems, and will be used repeatedly in the subsequent chapters of this thesis.

2.4.3 General sequential solutions

The sequential approach relies on the property of the model being a first order Markov process. The basic idea is to proceed through the time window and process each observation consecutively, which results in updating the previous solution with the new observation. This approach is also more suitable

for forecasting applications since new measurements can be incorporated directly after arrival, without the need to repeat the full assimilation window.

Additional to Bayes' theorem, the sequential reformulation of the assimilation problem requires the marginalization rule (see Appendix A). It states that a *marginal* pdf is achieved by integrating the joint pdf over all variables which are not of interest. For illustration, the following integral extracts the marginal prior pdf at time t_i from the joint prior pdf:

$$p(\mathbf{x}_i) = \int p(\mathbf{x}_{0:I})\, d\mathbf{x}_{0:(i-1)} d\mathbf{x}_{(i+1):I} \tag{2.12}$$

Here, and in the following, the integral is understood as a multiple integral that covers the full range of all components of the integration variables. The marginalization rule will be used below to derive the general filter solution. First, the general smoother solution needs to be reformulated in an iterative way.

2.4.3.1 Sequential smoother

In order to realize the sequential paradigm, first, the product in equation (2.11) is rewritten such that the time steps in between two successive observations are summarized.

$$p(\mathbf{x}_{i(j)}|\mathbf{x}_{i(j-1)}) = \prod_{i=i(j-1)+1}^{i(j)} p(\mathbf{x}_i|\mathbf{x}_{i-1}) \tag{2.13}$$

$$\Rightarrow \quad p(\mathbf{x}_{0:I}|\mathbf{y}_{1:J}) \propto p(\mathbf{x}_0) \left[\prod_{j=1}^{J} p(\mathbf{x}_{i(j)}|\mathbf{x}_{i(j-1)}) p(\mathbf{y}_j|\mathbf{x}_{i(j)}) \right] p(\mathbf{x}_I|\mathbf{x}_{i(J)}) \tag{2.14}$$

The last term describes the transition from time $t_{i(J)}$ to the final time t_I. In this interval no observations are available anymore. Thus, Bayes' theorem now appears as a product of pdfs, and it can be evaluated iteratively with the following algorithm:

1. Compute the posterior pdf at the first observation time level,

$$p(\mathbf{x}_{0:i(1)}|\mathbf{y}_1) \propto p(\mathbf{x}_0) p(\mathbf{x}_{i(1)}|\mathbf{x}_0) p(\mathbf{y}_1|\mathbf{x}_{i(1)}) \tag{2.15}$$

2. Iterate the following equation for $j = 2 \ldots J$ to find the posterior pdf at the j-th observation time. The previous posterior pdf can be interpreted as a new prior pdf, which is advanced and updated by the next measurement,

$$p(\mathbf{x}_{0:i(j)}|\mathbf{y}_{1:j}) \propto \underbrace{p(\mathbf{x}_{0:i(j-1)}|\mathbf{y}_{1:(j-1)})}_{\text{prev. posterior}} \underbrace{p(\mathbf{x}_{i(j)}|\mathbf{x}_{i(j-1)}) p(\mathbf{y}_j|\mathbf{x}_{i(j)})}_{\text{assimilate new obs.}} \tag{2.16}$$

3. Proceed the final posterior pdf until the end of time window,

$$p(\mathbf{x}_{0:I}|\mathbf{y}_{1:J}) \propto p(\mathbf{x}_{0:i(J)}|\mathbf{y}_{1:J}) p(\mathbf{x}_I|\mathbf{x}_{i(J)}) \tag{2.17}$$

It is obvious that the final joint pdf in equation (2.17) equals equation (2.11). The iterative solution is valid as long as the system is a first order Markov process and model and observations errors are

uncorrelated in time. Thus, this algorithm represents a general smoother as the information from any observation $\mathbf{y}_j$ influences the state estimates at all time levels. As this thesis focusses on filtering algorithms, this result will be processed to derive the sequential filter. However, it is also possible to develop explicit smoothers based on this iteration (Evensen and van Leeuwen 2000). The general smoother can be simplified to a computational more efficient version, the lagged smoother, in which observations only influence the state estimates in a specified time range rather than possibly affecting all of them. This is motivated by the fact that the impact of a measurement is negligible outside a time range that corresponds to the model's predictability limit (Evensen 2009, p. 136).

2.4.3.2 Sequential filter

Bayes' theorem and the marginalization rule, as stated in equations (2.11) and (2.12), are both of completely general nature. They can also be used to derive equations for a general filter. The characteristic property of a filter is that the observational information is not transferred backwards in time. Thus, in contrast to the smoother, past states are not updated again by a newly assimilated observation.
In the following, the general time $t_{i(j)}$, at which the j-th observation, $\mathbf{y}_j$, is assimilated, is considered. In probabilistic terms, the target is the marginal pdf at that time, as previous time levels are not updated again. This can be achieved by integrating the smoother posterior pdf, as given in equation (2.16), over all past states. Again, any normalization factors are omitted.

$$\begin{aligned} p(\mathbf{x}_{i(j)}|\mathbf{y}_{1:j}) &= \int\limits_{\mathbf{x}_{0:(i(j)-1)}} p(\mathbf{x}_{0:i(j)}|\mathbf{y}_{1:j})\, d\mathbf{x}_{0:(i(j)-1)} \\ &\propto \int p(\mathbf{x}_{0:(i(j-1))}|\mathbf{y}_{1:(j-1)}) p(\mathbf{x}_{i(j)}|\mathbf{x}_{i(j-1)}) p(\mathbf{y}_j|\mathbf{x}_{i(j)})\, d\mathbf{x}_{0:(i(j)-1)} \\ &= p(\mathbf{y}_j|\mathbf{x}_{i(j)}) \int p(\mathbf{x}_{0:i(j)}|\mathbf{y}_{1:(j-1)})\, d\mathbf{x}_{0:(i(j)-1)} \qquad (2.18) \\ \Rightarrow \quad p(\mathbf{x}_{i(j)}|\mathbf{y}_{1:j}) &\propto p(\mathbf{x}_{i(j)}|\mathbf{y}_{1:(j-1)}) p(\mathbf{y}_j|\mathbf{x}_{i(j)}) \qquad (2.19) \end{aligned}$$

Thus, at any observation time, the marginal posterior pdf is simply obtained by a marginal form of Bayes' theorem. Equation (2.19) states that the prior marginal pdf at time $t_{i(j)}$, $p(\mathbf{x}_{i(j)}|\mathbf{y}_{1:(j-1)})$,which incorporates all past observations, has to be multiplied by the likelihood pdf at that time, $p(\mathbf{y}_j|\mathbf{x}_{i(j)})$.

However, it remains to specify the prior marginal pdf, as given by the integral in equation (2.18), which expresses the marginalization rule. The first manipulation consists in factorizing the integrand into the joint pdf at the previous observation time and the transition density. Then, after re-ordering the integrals accordingly, the first integral is recognized as the marginal pdf at the previous observation time.

$$\begin{aligned} p(\mathbf{x}_{i(j)}|\mathbf{y}_{1:(j-1)}) &= \int\limits_{\mathbf{x}_{0:(i(j)-1)}} p(\mathbf{x}_{0:i(j)}|\mathbf{y}_{1:(j-1)})\, d\mathbf{x}_{0:(i(j)-1)} \\ &= \int p(\mathbf{x}_{0:i(j-1)}|\mathbf{y}_{1:(j-1)}) d\mathbf{x}_{0:(i(j-1)-1)} \int p(\mathbf{x}_{i(j)}|\mathbf{x}_{i(j)-1})\, d\mathbf{x}_{(i(j-1)):(i(j)-1)} \\ &= \int p(\mathbf{x}_{i(j-1)}|\mathbf{y}_{1:(j-1)}) p(\mathbf{x}_{i(j))}|\mathbf{x}_{i(j-1)})\, d\mathbf{x}_{(i(j-1)):(i(j)-1)} \qquad (2.20) \end{aligned}$$

This derivation used the fact that the transition density does not depend on the observation. Thus, the result is simple to interpret again. The prior density at time $t_{i(j)}$ is obtained by integrating the posterior density at time $t_{i(j-1)}$ together with the transition density. Since the joint transition density can be factorized, see equation (2.13), the integration can be performed iteratively in correspondence to the model time steps, which will be shown below.

In conclusion, the posterior filter pdf is given by updating the prior filter pdf, which depends on all past observations and arises from an integration in time, with the likelihood pdf of the new observation. The general sequential filter solution for the assimilation problem can be summarized as follows:

1. Find the posterior pdf at the first observation time:

   (a) Integrate the initial pdf until $t_{i(1)}$ via $p(\mathbf{x}_{i(1)}) = \int p(\mathbf{x}_0)p(\mathbf{x}_{i(1)}|\mathbf{x}_0)\, d\mathbf{x}_{0:(i(1)-1)}$. Using the factorization (2.13), this is achieved by an iterative procedure for $i = 1 \dots i(1)$

   $$p(\mathbf{x}_i) = \int p(\mathbf{x}_{i-1})p(\mathbf{x}_i|\mathbf{x}_{i-1})d\mathbf{x}_{i-1}$$

   (b) Assimilate the first observation, $p(\mathbf{x}_{i(1)}|\mathbf{y}_1) \propto p(\mathbf{x}_{i(1)})p(\mathbf{y}_1|\mathbf{x}_{i(1)})$.

2. Iterate the following substeps for $j = 2 \dots J$:

   (a) Forecast step: Integrate the previous posterior pdf, $p(\mathbf{x}_{i(j-1)}|\mathbf{y}_{1:(j-1)})$ according to equation (2.20) until observation time with the following iteration for $i = (i(j-1)+1) \dots i(j)$:

   $$p(\mathbf{x}_i|\mathbf{y}_{1:(j-1)}) = \int p(\mathbf{x}_{i-1}|\mathbf{y}_{1:(j-1)})p(\mathbf{x}_i|\mathbf{x}_{i-1})d\mathbf{x}_{i-1} \tag{2.21}$$

   Here, the filter directs the posterior pdf at the previous observation time, $p(\mathbf{x}_{i(j-1)}|\mathbf{y}_{1:(j-1)})$, through all model time steps until the next observation, where the resulting pdf is interpreted as new prior density $p(\mathbf{x}_{i(j)}|\mathbf{y}_{1:(j-1)})$. This step is strongly connected with the forward model. In the next subsection, this issue will be explored in more detail.

   (b) Analysis step: Update the prior pdf with the likelihood pdf according to Bayes' theorem, as derived in equation (2.19):

   $$p(\mathbf{x}_{i(j)}|\mathbf{y}_{1:j}) \propto \quad p(\mathbf{x}_{i(j)}|\mathbf{y}_{1:(j-1)})p(\mathbf{y}_j|\mathbf{x}_{i(j)}) \tag{2.22}$$

   This step concerns an objective adjustment of the prior pdf which, in principle, is independent on the model.

3. Proceed the posterior pdf at the final observation time, $p(\mathbf{x}_{i(J)}|\mathbf{y}_{1:J})$, to the end of the time window. Again, perform the iterative integration for $i = (i(J)+1) \dots I$:

   $$p(\mathbf{x}_i|\mathbf{y}_{1:J}) = \int p(\mathbf{x}_{i-1}|\mathbf{y}_{1:J})p(\mathbf{x}_i|\mathbf{x}_{i-1})\, d\mathbf{x}_{i-1}$$

   Thus, the sequential filter terminates with the final posterior pdf $p(\mathbf{x}_I|\mathbf{y}_{1:J})$, which equals the marginal of the general smoother solution (2.17) at final time t_I.

The filter is conceptually simpler than the smoother since it only concerns the marginal pdfs. At the end of the assimilation window, the solutions of the general filter and smoother are identical as the final filter pdf $p(\mathbf{x}_I|\mathbf{y}_{1:J})$ depends on all observations. In contrast, at intermediate times, the filter solution is only based on past observations, while the smoother considers the future ones as well, which is useful for re-analysis applications. However, for the initialization of a prediction after the DA cycle only the solution at the end of the time window is of relevance. In this context, a particular advantage of the sequential filter is that new observations after time t_I can be incorporated almost in real-time by simply advancing and updating the previous filter solution $p(\mathbf{x}_I|\mathbf{y}_{1:J})$.
The algorithm presented in this section is completely generic and does not impose any further restrictions apart from the general assumptions that (1) the model represents a first order Markov process and (2) model and observation errors do not exhibit temporal correlations.[1] It represents a different formulation of the general solution in terms of a probabilistic description. This algorithm is the basis for all filters appearing in this thesis, including the new method shown in Chapter 4.

2.4.4 Transition density and model integration

2.4.4.1 Integral equation

In DA, it is essential to understand the connection between the transition density and the model equation (2.4). The transition density $p(\mathbf{x}_i|\mathbf{x}_{i-1})$ serves as an example. This conditional pdf is the distribution of the random variable $\mathbf{x}_i$ given that the state at time t_{i-1} actually is $\mathbf{x}_{i-1}$. Here, two typical cases are considered. First, if the model is entirely deterministic ($\boldsymbol{\eta}_{i-1} \equiv 0$), the state at time t_i is fully determined by $\mathbf{x}_{i-1}$ through the model equations. In this case, the transition density can be interpreted as a delta distribution (see Appendix A),

$$p(\mathbf{x}_i|\mathbf{x}_{i-1}) = \delta\left(\mathbf{x}_i - \mathcal{M}_{(i-1)\to i}(\mathbf{x}_{i-1})\right) \tag{2.23}$$

Neglecting the model errors is referred to as the *perfect model assumption*. In the more general case, an additive Gaussian model error as in equation (2.4) is assumed. Then, the transition density is Gaussian, and its mean is given by an error-free model integration,

$$p(\mathbf{x}_i|\mathbf{x}_{i-1}) = \mathcal{N}(\mathbf{x}_i; \mathcal{M}_{(i-1)\to i}(\mathbf{x}_{i-1}), \mathbf{Q}_{i-1}) \tag{2.24}$$

For both situations, it becomes apparent that the transition density is strongly connected to the model operator. The latter defines the deterministic part of an integration in time, and the corresponding pdf describes the uncertainty gained during the integration due to unresolved model errors. Hence, applying the model integration as in equation (2.4) can be interpreted as sampling from the transition density (van Leeuwen 2009). The sequential approach requires to integrate the transition density along with the previous marginal pdf in order to obtain the marginal pdf at the next time level,

$$p(\mathbf{x}_i) = \int p(\mathbf{x}_{i-1})p(\mathbf{x}_i|\mathbf{x}_{i-1})\, d\mathbf{x}_{i-1} \tag{2.25}$$

[1] Actually, a similar solution can even be formulated in the presence of autocorrelated model errors. In this case, the model can still be formulated as a first order Markov process by augmenting the model errors to the state vector (Evensen 2003; Reichle et al. 2002).

Analytical solutions for this integral can only be obtained for a few special cases, e.g., if the model is linear. Such a situation applies to the Kalman filter forecast step, see section 2.6.2.

2.4.4.2 Fokker-Planck equation

Equation (2.25) constitutes the principal solution for the time evolution of the state pdf, and the system dynamics is hidden in the transition density. For a nonlinear model, the solution cannot be given anymore, even if the transition density is a delta function as in (2.23). Instead, at least for the case of additive Gaussian model errors, it is possible to transform the integral into an underlying differential equation, the Kolmogorov equation, often also referred to as Fokker-Planck equation (e.g., Jazwinski 1970; Miller et al. 1999). It describes the temporal evolution of a pdf, given the system's dynamical equations:

$$\frac{\partial p(\mathbf{x})}{\partial t} + \sum_{k} \frac{\partial \mathcal{M}_k(\mathbf{x}) p(\mathbf{x})}{\partial x_k} = \sum_{k_1 k_2} \frac{Q_{k_1 k_2}}{2} \frac{\partial^2 p(\mathbf{x})}{\partial x_{k_1} \partial x_{k_2}} \tag{2.26}$$

This equation can be interpreted as a continuity equation for pdfs since it constitutes the conservation of probability mass (Evensen 1994). The drift term, expressed by the divergence of the "probability flux", is determined by the model dynamics and speficies the local advection of probability mass. The diffusion term on the right hand side considers the influence of random forcing and tends to broaden the pdf, as uncertainty increases in time.

In principle, one could solve Kolmogorov's equation for any given initial pdf, $p(\mathbf{x}_0)$, using Sommerfeld boundary conditions. However, this is only feasible in a few cases due to several reasons. First, the derivatives required in this equation are not directly accessible unless the model has a very simple form. They become difficult if the model is a large operator, represented by a collection of computer code. Second, the computational cost of a numerical solution is very high and would exceed the cost of a single model integration by orders of magnitude, which is prohibitive even in moderate dimensions (Feynman 1982). Hence, applications are only possible in very simple cases for comparing such an "exact" pdf integration with approximative methods (Miller et al. 1999). Additionally, the equation is not valid anymore if the model error has a different form or if the model itself exhibits random components or explicit time dependencies, e.g., from lateral boundary conditions. In summary, it is fairly safe to conclude that even though Kolmogorov's equation implicitly is at the heart of any DA system, it does not represent a practical solution to the assimilation problem (Evensen 2009).

Consequently, other approaches are required to solve the problem of advancing the state pdf. In the remainder of this chapter, it will be shown that a Gaussian assumption allows practical solutions. In contrast, ensemble-based algorithms apply Monte Carlo methods to approximate the pdf integration in a stochastic fashion. This will be discussed in more detail in Chapter 3.

2.5 Variational data assimilation

This work mainly deals with sequential schemes. However, in order to discriminate them against other methods and to understand their properties and potential advantages, it is useful to gain an additional

insight by first looking at the principles of variational DA. It plays an important role in weather forecasting and climate reanalysis, and it represents the competitor to ensemble-based, sequential schemes. For example, the ECWMF relies on 4DVAR to produce analyses for different types of forecast, while the operational section of the German Weather Service (DWD) recently switched to an ensemble Kalman filter (Reich et al. 2011).

2.5.1 Characterization

The variational approach is mainly characterized by the assumption that all involved pdfs are Gaussian, with specified means and covariance matrices. These moments represent the major inputs to any variational DA system. Of particular relevance is the background error covariance matrix, which specifies the error correlation structure of the first guess. The variational scheme is a smoother. Its solution is represented by an estimate of the state at each time level within the assimilation window, and hence, they form a full trajectory $\mathbf{x}_{0:I}$. The algorithm can be reduced to a basic quantity, the scalar cost function $J(\cdot)$, which is quadratic in its variables. Its minimum defines the analysis states. However, no information about its reliability and uncertainty can be diagnosed directly.
The variational methods are presented in coherence with their historical developments. First, a static situation is considered where all observations are assimilated at one time, leading to 3DVAR. Second, the methodology is extended to consider the timing of different observations by including the model dynamics, which results in 4DVAR.

2.5.2 3DVAR and optimal interpolation

3DVAR (Lorenc 1981) is a classical, deterministic DA method. At a given point in time, it intends to create an analysis field which is statistically consistent with (1) a prior field (called first guess or background) with a specified covariance structure and (2) all observations valid at that time. 3DVAR can be interpreted as an enhanced interpolation scheme that considers both pieces of information, and is closely linked to optimal interpolation (OI).

3DVAR can be derived within the probabilistic framework described in section 2.4. An arbitrary analysis time $t_{i(j)}$ ($j \in \{1 \dots J\}$) is considered and therefore, the time index can be dropped for now. It is assumed that both prior and observation errors are Gaussian random variables. Hence, the prior pdf $p(\mathbf{x})$ is characterized by the *background* state, $\mathbf{x}_b$, and the background error covariance matrix, $\mathbf{B}$. The uncertainty of the observation $\mathbf{y}$ is described by an error covariance matrix $\mathbf{R}$. These four quantities represent the main input to a 3DVAR assimilation system. To be specific,

$$p(\mathbf{x}) = \mathcal{N}(\mathbf{x}; \mathbf{x}_b, \mathbf{B}) \propto \exp\left\{-\frac{1}{2}(\mathbf{x}-\mathbf{x}_b)^T\mathbf{B}^{-1}(\mathbf{x}-\mathbf{x}_b)\right\} \tag{2.27}$$

$$p(\mathbf{y}|\mathbf{x}) = \mathcal{N}(\mathbf{y}; \mathcal{H}(\mathbf{x}), \mathbf{R}) \propto \exp\left\{-\frac{1}{2}(\mathbf{y}-\mathcal{H}(\mathbf{x}))^T\mathbf{R}^{-1}(\mathbf{y}-\mathcal{H}(\mathbf{x}))\right\} \tag{2.28}$$

Following Bayes' theorem and according to equation (2.22), the posteriori density of the state given the observation, i.e., the analysis pdf, is given by

$$p(\mathbf{x}|\mathbf{y}) \propto p(\mathbf{x})p(\mathbf{y}|\mathbf{x}) = \exp\{-J(\mathbf{x})\} \tag{2.29}$$

$$\text{where} \quad J(\mathbf{x}) = \frac{1}{2}(\mathbf{x}-\mathbf{x}_b)^T\mathbf{B}^{-1}(\mathbf{x}-\mathbf{x}_b) + \frac{1}{2}(\mathbf{y}-\mathcal{H}(\mathbf{x}))^T\mathbf{R}^{-1}(\mathbf{y}-\mathcal{H}(\mathbf{x})) \tag{2.30}$$

3DVAR seeks for the state that maximizes the posteriori density, or, equivalently, minimizes is negative logarithm, the cost function $J(\mathbf{x})$. Mathematically, the cost function is a scalar, quadratic form in a K-dimensional space. Therefore, a unique global minimum exists. The cost function can also be interpreted as a penalty function that contains the weighted, squared deviations of any state $\mathbf{x}$ to both observations and background state. The weights are given by the respective inverse error covariances matrices. Therefore, the 3DVAR solution also corresponds to a least-squares estimate. The minimum is determined by the necessary condition that the gradient of $J(\mathbf{x})$ vanishes,

$$\nabla J = \mathbf{B}^{-1}(\mathbf{x}-\mathbf{x}_b) + \mathbf{H}^T\mathbf{R}^{-1}\left(\mathcal{H}(\mathbf{x})-\mathbf{y}\right) = \mathbf{0} \tag{2.31}$$

where $\mathbf{H}$ is the tangent linear observation operator. In a KF framework, see section 2.6, an analytical solution of this equation is specified. In contrast, in 3DVAR, the analysis is found by a numerical minimization of the cost function. A number of standard algorithms exist to perform this task, however, given the potentially high dimension of the state vector, only a few of them are applicable to typical DA problems with moderate computational cost (e.g., Fisher 1998). An example are quasi-Newton methods which approximate the Hessian matrix. As the iso-surface of the cost function may be considerably ellipsoidal, for large-scale applications a change of variables, the so-called preconditioning, is necessary prior to minimization.

3DVAR was introduced as the operational DA method at ECMWF in 1996, and many details concerning the cost function minimization were developed (Courtier et al. 1998). Before 1996, ECMWF had relied on OI (Lorenc 1986) for 17 years. In principle, OI solves the same problem as 3DVAR but splits the global analysis into a number of local analyses for which equation (2.31) can be solved analytically. Interestingly, the analyses produced by the dynamical successive correction method (nudging, mentioned in section 1.2) converge to the OI results if the nudging weights are chosen properly (Bratseth 1986). However, these techniques only allow for simple observation operators, which is particularly problematic for the assimilation of satellite radiances as they are not directly linked to the prognostic model variables. In contrast, 3DVAR naturally deals with an arbitrarily complex observation operator.

The major drawback of 3DVAR is the challenging specification of the prior information (Bouttier 1994; Derber and Bouttier 1999). It may be easy to obtain a realistic prior state estimate $\mathbf{x}_b$ from a previous model integration, using 3DVAR in a cycled way. However, the quality of the analysis is sensitive to the error covariance matrix $\mathbf{B}$ as it weights the increments and spreads the observational information in space. Various statistical methods have been developed to estimate $\mathbf{B}$ (Fisher 2003), but a realistic, flow-dependent correlation structure is difficult to achieve in a large-scale, chaotic system (Rabier et al. 1998). Additionally, while 3DVAR produces a consistent analysis in space, it only combines the model and observations at a fixed time. This disregards the fact that observations are collected nearly continuously in time.

2.5.3 Strong- and weak-constraint 4DVAR

As shown, 3DVAR is principally designed for a stationary situation, even though it can be applied to time-dependent problems as well. In contrast, 4DVAR is an actual smoother that fully considers the dynamical situation, thereby tackling the main drawback of 3DVAR. It aims at assimilating all observations during a given time window by finding an optimal model trajectory which offers the best fit to the background state at the beginning of the window and to the observations (LeDimet and Talagrand 1986; Talagrand and Courtier 1987).

2.5.3.1 Strong-constraint 4DVAR

In order to derive 4DVAR, the general smoother solution given by equation (2.11) is sufficient. Strong-constraint 4DVAR implies that the model is assumed to be perfect. As discussed in section 2.4.4, this leads to a delta function representation of the transitions densities,

$$p(\mathbf{x}_i|\mathbf{x}_{i-1}) = \delta\big(\mathbf{x}_i - \mathcal{M}_{(i-1)\rightarrow i}(\mathbf{x}_{i-1})\big) \tag{2.32}$$

In other words, given any initial state $\mathbf{x}_0$, all successive states $\mathbf{x}_i$ ($i \geq 1$) are uniquely determined through a model integration. Consequently, it is sufficient to optimize the state at the beginning of the time window. Then, the full smoother solution is given by the corresponding model trajectory $\mathbf{x}_{0:I}$. In order to formalize these considerations, the full smoother pdf is marginalized with respect to all time levels by integrating equation (2.11). The integrals can be solved successively with the help of the delta functions. Furthermore, property (2.2) is used to shortly denote a full model integration.

$$\begin{aligned} p(\mathbf{x}_0|\mathbf{y}_{1:J}) \quad &\propto \quad \int p(\mathbf{x}_{0:I}|\mathbf{y}_{1:J})\, d\mathbf{x}_{1:I} = \int p(\mathbf{x}_0) \prod_{i=1}^{I} \delta\big(\mathbf{x}_i - \mathcal{M}_{(i-1)\rightarrow i}(\mathbf{x}_{i-1})\big) \prod_{j=1}^{J} p(\mathbf{y}_j|\mathbf{x}_{i(j)})\, d\mathbf{x}_{1:I} \\ &= \quad p(\mathbf{x}_0) \prod_{j=1}^{J} p(\mathbf{y}_j|\mathcal{M}_{0\rightarrow i(j)}(\mathbf{x}_0)) \end{aligned} \tag{2.33}$$

In 4DVAR, it is assumed that the initial prior pdf as well as the likelihood pdfs are given by Gaussian distributions with the corresponding means and covariances. Then, using equation (2.33), it can easily be seen that the posterior density, as for 3DVAR, takes the form $\exp\{-J(\mathbf{x}_0)\}$. Again, its negative logarithm defines the cost function:

$$p(\mathbf{x}_0|\mathbf{y}_{1:J}) \quad \propto \quad \mathcal{N}(\mathbf{x}_0; \mathbf{x}_b, \mathbf{B}) \cdot \prod_{j=1}^{J} \mathcal{N}(\mathbf{y}_j; \mathcal{H}_j\left(\mathcal{M}_{0\rightarrow i(j)}(\mathbf{x}_0)\right), \mathbf{R}_j) \propto \exp\{-J(\mathbf{x})\} \tag{2.34}$$

$$\begin{aligned} J(\mathbf{x}_0) \quad &= \quad \frac{1}{2}(\mathbf{x}_0 - \mathbf{x}_b)^T \mathbf{B}^{-1}(\mathbf{x}_0 - \mathbf{x}_b) \\ &\quad + \frac{1}{2}\sum_{j=1}^{J} \Big(\mathbf{y}_j - \mathcal{H}_j(\mathcal{M}_{0\rightarrow i(j)}(\mathbf{x}_0))\Big)^T \mathbf{R}_j^{-1} \Big(\mathbf{y}_j - \mathcal{H}_j(\mathcal{M}_{0\rightarrow i(j)}(\mathbf{x}_0))\Big) \end{aligned} \tag{2.35}$$

The minimum of this strong-constraint 4DVAR cost function yields the initial state $\mathbf{x}_0$ that produces the *optimal* trajectory $\mathbf{x}_{0:I}$. Here, as in 3DVAR, the term optimality refers two to distinct definitions:

- The solution is the most probable estimator in the sense that it maximizes the posteriori pdf, given the Gaussian assumption for prior and likelihood pdf and a perfect model.
- It is also the least-squares estimator that minimizes that weighted, squared deviations to background and observations. These weights are defined by the entries of the inverse covariance matrices.

2.5.3.2 Minimization of the 4DVAR cost function

When evaluating the cost function, repeated model integrations can be avoided by an iterative procedure. Usually, the state is integrated until the next observation time, where the corresponding observational cost term is added. Then, the next time level is processed. As for 3DVAR, the minimization algorithms require to evaluate the gradient, which is

$$\nabla J = \mathbf{B}^{-1}(\mathbf{x}_0 - \mathbf{x}_b) + \sum_{j=1}^{J} \mathbf{M}_{0\rightarrow i(j)}^{T} \mathbf{H}_j^T \mathbf{R}_j^{-1} \left(\mathbf{y}_j - \mathcal{H}_j(\mathcal{M}_{0\rightarrow i(j)}(\mathbf{x}_0))\right) \tag{2.36}$$

Here, the adjoints of model and observation operator appear. The evaluation of the second term of the gradient is characterized by a backward integration of the adjoint model, as $\mathbf{M}_{0\rightarrow i(j)}^{T}$ can be factorized to $\mathbf{M}_{0\rightarrow 1}^{T}\mathbf{M}_{1\rightarrow 2}^{T}\cdots\mathbf{M}_{(i(j)-1)\rightarrow i(j)}^{T}$ (see Appendix A).
Hence, the cost function and its gradient can be evaluated by one integration of forward and adjoint model, respectively, and the latter typically requires considerably more computational time. Therefore, even with efficient minimization algorithms which require only 10-100 iterations, for a rather long time window of 12 hours as in NWP up to 200 days of model time may be needed, which is prohibitive. Instead, the minimization is solved by an incremental method, in which the tangent linear model is applied to minimize an approximately quadratic version of $J(\mathbf{x}_0)$. This technique is for example used at ECWMF (Tremolet 2004), where 4DVAR runs operationally since 1997 (Rabier et al. 2000). As of today, it still represents the backbone of their DA system, even though it has been enhanced in 2011 with an improved estimation of the background error covariance by using an ensemble of 4DVAR analyses (Bonavita et al. 2012). Finally, it should be mentioned that even though the cost function has a unique global minimum, in case of nonlinear observation or model operators additional local minima appear and may pose a great challenge to the minimization algorithm (Courtier et al. 1994).

2.5.3.3 Weak-constraint 4DVAR

Due to the perfect model assumption in strong-constrained 4DVAR, the length of the assimilation window is rather limited (Tremolet 2006). In order to account for model errors that are already relevant within this time range and to possibly allow longer time windows, the perfect model assumption is relaxed (Ahrens 1999) using a model error formulation as introduced in equation (2.4). The model error for each time step is expressed as the difference between the new state and its deterministic counterpart. It is assumed to be drawn from a Gaussian distribution with zero mean and covariance $\mathbf{Q}$ (Zupanski 1997). Hence, equation (2.11), which is based on the model evolution as a first order Markov chain, is still valid. The transition densities in presence of model error have already been shown in section

2.4.4. It remains to evaluate equation (2.11) with the typical assumption that all appearing densities are Gaussian:

$$p(\mathbf{x}_{0:I}|\mathbf{y}_{1:J}) \propto \mathcal{N}(\mathbf{x}_0; \mathbf{x}_b, \mathbf{B}) \cdot \prod_{i=1}^{I} \mathcal{N}\left(\mathbf{x}_i; \mathcal{M}_{i-1\rightarrow i}(\mathbf{x}_{i-1}), \mathbf{Q}_{i-1}\right) \cdot \prod_{j=1}^{J} \mathcal{N}(\mathbf{y}_j; \mathcal{H}_j\left(\mathbf{x}_{i(j)}\right), \mathbf{R}_j)$$

This leads to a cost function which looks similar as in the strong-constraint case in equation (2.35), but with an additional term due to the model errors:

$$\begin{aligned} J(\mathbf{x}_{0:I}) \quad = \quad & -\ln\left(p(\mathbf{x}_0|\mathbf{y}_{1:J})\right) = \frac{1}{2}(\mathbf{x}_0 - \mathbf{x}_b)^T \mathbf{B}^{-1}(\mathbf{x}_0 - \mathbf{x}_b) \\ & + \frac{1}{2}\sum_{j=1}^{J}\left(\mathbf{y}_j - \mathcal{H}_j(\mathbf{x}_{i(j)})\right)^T \mathbf{R}_j^{-1}\left(\mathbf{y}_j - \mathcal{H}_j(\mathbf{x}_{i(j)})\right) \\ & + \frac{1}{2}\sum_{i=1}^{I}\left(\mathbf{x}_i - \mathcal{M}_{i-1\rightarrow i}(\mathbf{x}_{i-1})\right)^T \mathbf{Q}_{i-1}^{-1}\left(\mathbf{x}_i - \mathcal{M}_{i-1\rightarrow i}(\mathbf{x}_{i-1})\right) \end{aligned} \tag{2.37}$$

A substantial difference is that now the cost function depends on all model states $\mathbf{x}_{0:I}$. Thus, the degrees of freedom in this high-dimensional minimization problem strongly increase, which represents the first, rather technical, challenge of weak-constraint 4DVAR (Tremolet 2007). Furthermore, an issue currently under intensive research concerns the suitable representation of model error. In reality, it is likely that model errors are correlated in time (Daley 1992). However, even the apparently simple approach using a Gaussian distribution requires knowledge about the statistical properties of model error and their representation in form of a covariance matrix $\mathbf{Q}$. At ECMWF, an approach is to estimate it with an ensemble integration (Bonavita et al. 2012).

2.6 Kalman filter and its extension

The Kalman Filter (KF, Kalman 1960; Stratonovich 1960) represents the sequential counterpart to 4DVAR. Originally, it intended to solve the estimation problem with the constraint that both model and observation operator are linear. A straightforward extension to nonlinear filtering problems was proposed shortly after (Kalman and Bucy 1961). Often, it is referred to as the extended KF (EKF). In this thesis, the term KF is used for the generalized formulation as well. Already in the 1960's, the KF gained a famous reputation for its successful application in aerospace systems, particularly in the first manned spacecraft missions (Schmidt 1981; Smith et al. 1962).

2.6.1 Characterization

The KF is a sequential scheme that iterates forecast and analysis steps over the assimilation window and is, as 4DVAR, based on Gaussian distributions (Sorenson 1970). Here, the KF is derived within the probabilistic framework, using the general filter formulation developed in section 2.4.3. For a general description of such a sequential scheme, it is sufficient to concentrate on one single assimilation cycle, which consists of forecast and analysis step. A "hat" notation indicates the KF estimates. For example, $\hat{\mathbf{x}}_i$ represents the KF solution at time t_i.

In Appendix B, an alternative to the probabilistic approach is presented by deriving the KF from a purely statistical point of view. The results are completely identical to the equations presented here, but the statistical derivation offers further insight into the properties and assumptions behind the KF that may not be revealed directly by the probabilistic derivation.

2.6.2 Forecast step

The derivation of the forecast step refers to step 2.a of the general algorithm in section 2.4.3. The purpose of the forecast step is to advance the state pdf in time such that it can be used as prior pdf when being confronted with the next observation. For the time integration step, the marginal pdf at the next time step has to be computed from the marginal pdf at the previous time step. As the transition density is independent of the observation, equation (2.21) can be simplified by omitting the dependency on all past observations:

$$p(\mathbf{x}_i) = \int p(\mathbf{x}_{i-1}) p(\mathbf{x}_i|\mathbf{x}_{i-1})\, d\mathbf{x}_{i-1} \tag{2.38}$$

Now, it is assumed that at time t_{i-1}, the pdf is Gaussian with mean and covariance given by the corresponding KF estimates, i.e., $p(\mathbf{x}_{i-1}) = \mathcal{N}(\mathbf{x}_{i-1}; \hat{\mathbf{x}}_{i-1}, \hat{\mathbf{P}}_{i-1})$. If t_{i-1} is an observation time, this pdf is the posteriori pdf, which is conditioned on all previous observations. The transition pdf is given by $p(\mathbf{x}_i|\mathbf{x}_{i-1}) = \mathcal{N}(\mathbf{x}_i; \mathcal{M}_{i-1\to i}(\mathbf{x}_{i-1}), \mathbf{Q}_{i-1})$. As discussed in section 2.4.4, the integral in equation (2.38) is, in general, not solvable. However, in the Gaussian case, a solution can be found by linearizing the model integration around the previous estimate as follows, using the tangent linear model:

$$\mathbf{x}_i = \mathcal{M}_{i-1\to i}(\mathbf{x}_{i-1}) + \boldsymbol{\eta}_{i-1} \approx \underbrace{\mathcal{M}_{i-1\to i}(\hat{\mathbf{x}}_{i-1}) + \mathbf{M}_{i-1\to i}(\mathbf{x}_{i-1} - \hat{\mathbf{x}}_{i-1})}_{\text{LT}} + \boldsymbol{\eta}_{i-1} \tag{2.39}$$

In this case, the model integration (2.38) can be interpreted as a linear transformation applied to the Gaussian random variable $\mathbf{x}_{i-1}$ (marked "LT" in the equation), followed by the addition of another independent Gaussian random variable, $\boldsymbol{\eta}_{i-1} \sim \mathcal{N}(\mathbf{0}; \mathbf{Q}_{i-1})$. Then, basic properties of Gaussian densities (see Appendix A) imply that

- The mean of the transformed variable is the same as applying the transformation to the previous mean, $\hat{\mathbf{x}}_{i-1}$, which simply leads to $\hat{\mathbf{x}}_i = \mathcal{M}_{i-1\to i}(\hat{\mathbf{x}}_{i-1})$. Addition of the centered random variable $\boldsymbol{\eta}_{i-1}$ does not change the mean.
- The covariance of the transformed variable is given by $\mathbf{M}_{i-1\to i}\hat{\mathbf{P}}_{i-1}\mathbf{M}_{i-1\to i}^T$. Addition of model error further adds $\mathbf{Q}_{i-1}$ to the covariance term.

In short, the integrated pdf is described as

$$\begin{aligned} p(\mathbf{x}_i) &= \mathcal{N}(\mathbf{x}_i; \hat{\mathbf{x}}_i, \hat{\mathbf{P}}_i) \\ \text{with} \quad \hat{\mathbf{x}}_i &= \mathcal{M}_{i-1\to i}(\hat{\mathbf{x}}_{i-1}) \quad \text{and} \quad \hat{\mathbf{P}}_i = \mathbf{M}_{i-1\to i}\hat{\mathbf{P}}_{i-1}\mathbf{M}_{i-1\to i}^T + \mathbf{Q}_{i-1} \end{aligned} \tag{2.40}$$

Consequently, the KF forecast step consists in advancing the estimates of both state and covariance since the state pdf remains Gaussian throughout the forecast step, given the approximate linearity assumption. While the state estimate is simply obtained by performing a (deterministic) model integration, the covariances are updated with the linearized model. It can be seen that both tangent linear and adjoint models are required. Additionally, the model error covariance matrix has to be added in order to account for the increasing uncertainty due to model shortcomings. However, it should be mentioned that the propagation of the error covariance, as essential as it is, suffers from two drawbacks. First, it relies on the linearized model, which is questionable in nonlinear systems. Second, its integration corresponds to K^2 model integrations, which gets prohibitive in larger-dimensional cases for reasons of computational expenses and storage capabilities. Higher-order closures of equation (2.39) in principle exist, but are only feasible for small-scale systems (Miller et al. 1994).

2.6.3 Analysis step

An arbitrary observation time is considered, $t_{i(j)}$, $j \in \{1, \ldots, J\}$. For the description of the analysis step, the time index is usually dropped, as all quantities are valid at the same time. The prior pdf at that time is assumed to be available. Usually, it arises from a preceding forecast step as described in the previous subsection, which allows to state $p(\mathbf{x}_{i(j)}|\mathbf{y}_{1:(j-1)}) = \mathcal{N}(\mathbf{x}_{i(j)}; \tilde{\hat{\mathbf{x}}}_{i(j)}, \tilde{\hat{\mathbf{P}}}_{i(j)})$. The tilde notation indicates that these moments are only the prior, but not the final estimates at time $t_{i(j)}$. In order to make the notation as clearly arranged as possible, the abbreviations $\mathbf{x}_f \equiv \tilde{\hat{\mathbf{x}}}_{i(j)}$ and $\mathbf{P}_f \equiv \tilde{\hat{\mathbf{P}}}_{i(j)}$ for the forecast moments as well as $\mathbf{x}_a \equiv \hat{\mathbf{x}}_{i(j)}$ and $\mathbf{P}_a \equiv \hat{\mathbf{P}}_{i(j)}$ for the analysis moments are introduced.

The analysis step is performed according to the general filter algorithm given in section 2.4.3, specifically by equation (2.22), which represents the Bayesian update of the forecast pdf with the observational likelihood pdf. Plugging in the functional forms of both pdfs yields

$$p(\mathbf{x}_{i(j)}|\mathbf{y}_{1:j}) \propto p(\mathbf{x}_{i(j)}|\mathbf{y}_{1:(j-1)})\, p(\mathbf{y}_j|\mathbf{x}_{i(j)}) = \mathcal{N}(\mathbf{x}; \mathbf{x}_f, \mathbf{P}_f) \cdot \mathcal{N}(\mathbf{y}; \mathcal{H}(\mathbf{x}), \mathbf{R}) \tag{2.41}$$

A similar posteriori pdf appears in 3DVAR, equation (2.30). However, 3DVAR only focusses on the state that maximizes the posterior density. In contrast, the KF solves for the full pdf and hence, the multiplication in equation (2.41) has to be performed explicitly. The computation is rather technical and not presented here, but is demonstrated in Appendix B. The result is the following analysis pdf,

$$p(\mathbf{x}_{i(j)}|\mathbf{y}_{1:j}) \propto \exp\left\{-(\mathbf{x}-\mathbf{x}_a)^T \mathbf{P}_a^{-1} (\mathbf{x}-\mathbf{x}_a)\right\} \propto \mathcal{N}(\mathbf{x}; \mathbf{x}_a, \mathbf{P}_a) \tag{2.42}$$

and its moments are given by

$$\mathbf{x}_a = \mathbf{x}_f + \mathbf{K}_j \left(\mathbf{y}_j - \mathcal{H}_j(\mathbf{x}_f)\right) \tag{2.43}$$

$$\mathbf{P}_a = \left(\mathbf{I} - \mathbf{K}_j \mathbf{H}_j\right) \mathbf{P}_f = \left(\mathbf{P}_f^{-1} + \mathbf{H}^T \mathbf{R}^{-1} \mathbf{H}\right)^{-1} \tag{2.44}$$

$$\text{where} \quad \mathbf{K}_j = \mathbf{P}_f \mathbf{H}_j^T \left(\mathbf{H}_j \mathbf{P}_f \mathbf{H}_j^T + \mathbf{R}_j\right)^{-1} \tag{2.45}$$

It is important to notice that the analysis pdf remains Gaussian, with mean and covariance given by equations (2.43) and (2.44). These results are identical to the outcome of a statistical approach shown in Appendix B, where the equality of both analysis covariance representations in equation (2.44) is

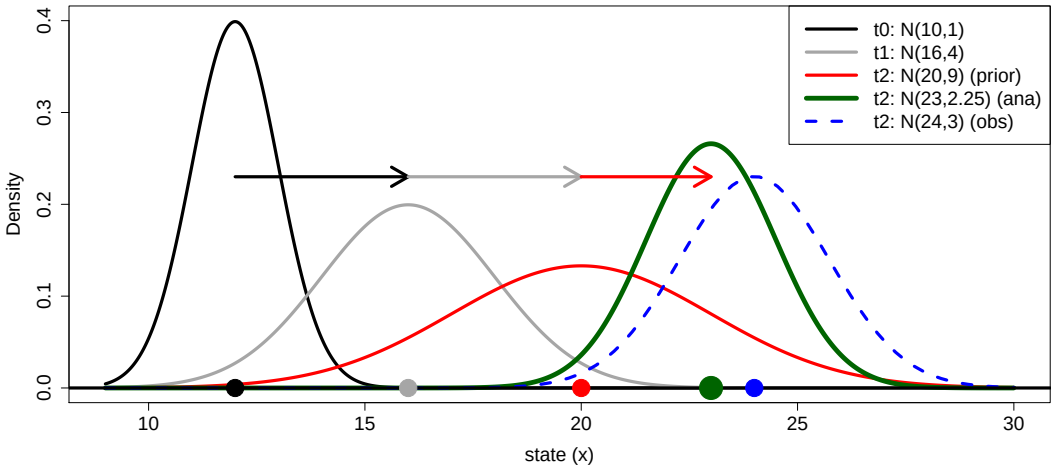


FIGURE 2.3: Conceptual visualization of the KF with a scalar, linear "model" (with error) that shifts x to the right by 4 each time step. At t_0, there is an initial Gaussian density (black). During the two forecast steps from $t_0 \rightarrow t_1$ (gray) and $t_1 \rightarrow t_2$ (red), the uncertainty increases, broadening the densities in addition to the advection. At t_2, an observation with Gaussian likelihood density (blue) is available. It updates the prior density (red) to the analysis density (green). The uncertainty is decreased again. The arrows indicate the increments added to the state each time level. The dots show the mean values of the densities.

also proved. It is emphasized that the key element in the KF is the Gaussian assumption for the initial pdf and the model error. Together with the linearized model integration and the Gaussian observation errors, it follows that the state pdf remains Gaussian for the whole assimilation window. Figure 2.3 visualizes the principle for a scalar case, showing the increasing uncertainty during the forecast step and the assimilation of the observation via Bayes' theorem. Even though the Gaussian assumptions allow a convenient analytical solution of the assimilation problem, they also imply a strong restriction, which may deteriorate the results in nonlinear scenarios.

2.6.4 Interpretation of the update equation

The linear update equations are the central part of the KF. While equation (2.43) quantifies the adjustment of the forecast state to assimilate the observational information, equation (2.44) describes the change in the error correlation structure associated with this update. These basic equations are directly adapted in the ensemble-based KFs, as shown in Chapter 3. Therefore, a detailed investigation is beneficial to gain a deeper understanding of the functionality of these algorithms.

- The analysis itself consists in adding a linear transformation of the so-called *innovation*, $\mathbf{i}_j = \mathbf{y}_j - \mathcal{H}_j(\mathbf{x}_f)$ to the prior state estimate. The innovation is a vector in observation space and describes the difference between the actual observation and its model equivalent. Its statistics can also be used to diagnose and correct systematic model or observation deficiencies (Haimberger et al. 2012). The Kalman gain can be interpreted as a proper weight matrix,

$$\mathbf{K} = \underbrace{\mathbf{P}_f\mathbf{H}^T}_{\text{distribution}} \underbrace{\left(\mathbf{H}_j\mathbf{P}_f\mathbf{H}_j^T + \mathbf{R}_j\right)^{-1}}_{\text{weighting}} \tag{2.46}$$

It computes the weights by mapping the forecast error covariance into observation space, relating it to the observation error covariance matrix. Then, the Kalman gain distributes the innovation to an *increment* in state space. The increment $\mathbf{K}\left(\mathbf{y}-\mathcal{H}(\mathbf{x}_f)\right)=\mathbf{x}_a-\mathbf{x}_f$ is a vector that quantifies the difference between posterior and prior estimate in state space.

- The covariance equation (2.44) states that the diagonal elements of the analysis error covariance matrix are always smaller than the corresponding elements in $\mathbf{P}_f$. Hence, the uncertainty, as expressed by these variances, is always reduced thanks to the observation, and the reduction is again determined by the Kalman gain. The second variant of the covariance shown in equation (2.44) shows that the sharpness of the analysis, as expressed by the inverse covariance matrix, is given by the sum of the prior sharpness and the observation's sharpness, where the latter is mapped into state space.

It is also illustrative to investigate the scalar case with direct observation of the state, $\mathcal{H}(x)=x$. Then, the KF equations reduce to

$$\begin{aligned} x_a &= x_f + K\left(y-x_f\right) \\ \sigma_a^2 &= (1-K)\sigma_f^2 = \frac{1}{\sigma_f^2}+\frac{1}{\sigma_o^2} \\ \text{where}\quad K &= \frac{\sigma_f^2}{\sigma_f^2+\sigma_o^2} = \left(1+\frac{\sigma_o^2}{\sigma_f^2}\right)^{-1} \end{aligned}$$

In this case, the Kalman gain is a real number in $[0;1]$ and can be interpreted as relative weight assigned to the observation, since the analysis is given by a weighted adjustment of the prior state towards the observation. This weight is simply determined by the magnitude of the forecast error variance relative to the total error variance. Thus, the observation receives more weight the smaller the observation error is compared to the forecast error, and vice versa. Furthermore, the extreme cases are reasonable. A useless observation ($\frac{\sigma_o}{\sigma_f}\to\infty$) is simply ignored and the analysis remains equal to the prior. A perfect observation ($\frac{\sigma_o}{\sigma_f}\to 0$) renders the prior information irrelevant, and the observation entirely dominates the analysis.

The Kalman gain is computed with the tangent linear and adjoint observation operator and therefore incorporates approximations in case of a nonlinear observation operator. The Kalman gain strongly depends on the error covariance structure specified by $\mathbf{P}_f$ as it contains the variances and correlations that in turn determine the weights and therefore the increments. However, even the specification of $\mathbf{R}$ is an issue not to be underestimated. For example, the ignorance of existing observation error correlations necessarily renders the analysis suboptimal.

2.6.5 Comparison of KF and 4DVAR

Even though the KF and 4DVAR represent considerable different assimilation techniques, both can be derived within the same probabilistic framework by introducing the Gaussian assumption into the general formulations. 4DVAR solves the smoothing problem while the corresponding sequential filter formulation leads to the KF. Therefore, it is interesting to discuss the differences which arise despite their common basis.

First, a single analysis is considered, with a Gaussian assumption for the prior state. The KF states that the analysis pdf remains Gaussian and specifies its parameters. However, if the prior moments are associated as the background, this situation becomes formally equal to the problem 3DVAR solves. In Appendix B, it is explicitly shown that the analytical minimum of the cost function corresponds to the KF analysis mean. Hence, the variational approach is equal to the KF update mechanism, the difference lies in the way the problem is solved. While the KF uses the analytical update equation, in 3DVAR, the cost function is minimized numerically. This proves that the maximum likelihood estimator applied for the variational approach is not only equal to the least squares solution. It is also identical to the mean of a Bayesian solution, which further corresponds to the minimum variance estimator.

Next, a dynamic situation with a *linear* model is assumed, again with Gaussian densities. In this case, the linear approximation used in the derivation of KF update step is exact, and the KF represents the exact solution for the general filter, in the same way as 4DVAR represents the exact solution of the general smoother. Therefore, KF and 4DVAR are equal (given the same random errors in the model integration) in the sense that the KF estimate at the final time, $\hat{\mathbf{x}}_I$, equals the final state of the 4DVAR analysis trajectory $\mathbf{x}_{0:I}$, since it depends on all observations (Lorenc 1986). Of course, intermediate values differ as the smoother solution also depends on future observations. It is possible to derive a Kalman smoother by introducing another backward iterative procedure that updates all past states with future observations. In the linear case, this yields exactly the same results as 4DVAR, even in the weak-constraint case (Fisher et al. 2005). However, an advantage of the KF is that it not only determines the analysis state but also gives an expression for the analysis error covariance, which contains additional uncertainty information about the analysis and can be used in further assimilation steps. In contrast, variational schemes only specify the analysis state itself. Their main advantage is the strong relative efficiency compared to the KF in larger-dimensional problems.

In the presence of *nonlinear* dynamics, it is evident that KF and 4DVAR produce different results, and their performances depend on the actual situation, as given by model, observations, errors, validity of tangent linear approximations, length of the assimilation window, etc. Therefore, no method is superior by construction (e.g., Li and Navon 2001; Lorenc 1986). Even though 4DVAR does not explicitly require a linearized model integration, the incremental approach required in praxis also introduces linear approximations. Furthermore, Courtier et al. (1994) showed that, in chaotic systems, with increasing assimilation window length the number of minima in the cost function strongly increases. This can easily trap the minimization algorithm in a local minimum. For that reason, the length of the assimilation window has to be limited, particularly for strong-constraint 4DVAR. The big advantage of 4DVAR, and the main reason why it is often preferred in the operational context, is more of technical nature. 4DVAR can be applied to high-dimensional systems, if combined with the incremental approach. In contrast, the propagation of the KF error covariance in the forecast step is not only an undesirable approximation, but also computationally very expensive and not feasible in such systems.

2.7 Summary and conclusions

This chapter reviewed the basic issues of DA and the most important classical solutions. It was motivated that the probabilistic framework yields a suitable description that allows to derive general solutions for the smoothing and filtering problem. The smoother solution, together with an assumption of Gaussian distributions, leads to 4DVAR, which serves as the core of DA systems at many operational centers. The sequential filter is characterized by an iteration of forecast and analysis steps, and the analog solution to 4DVAR is the KF which relies on a propagation of the covariance matrix in time.

The central conclusions gained in this chapter are:

- The general probabilistic framework is suitable to describe a large variety of objective DA methods.
- The assumption of Gaussian densities leads to feasible schemes. Though, it should be considered that in a nonlinear system, the results are necessarily suboptimal.
- 4DVAR has proven to be a successful scheme, but its practical implementation is extensive as it requires not only sophisticated minimization algorithms but also the development and maintenance of reliable tangent linear and adjoint models.
- The KF represents an appealing alternative from a technical point of view as it separates the model integration from the actual analysis step. However, its main drawback is the linearized integration of the error covariance matrix within the forecast step. This is not only computationally expensive, but can also lead to filter failure in nonlinear scenarios (Evensen 1992; Miller et al. 1994).

This thesis intends to achieve progress in sequential filtering in order to enhance its applicability as contrasted to 4DVAR, as the independency of the analysis step of the particular model is appealing. Based on these insights, the presentation proceeds with more recent filter developments that attempt to deal with the issues raised here.

Chapter 3

Ensemble-based data assimilation

In the previous chapter, a coherent probabilistic framework for DA was established. It comprises various explicit algorithms, and allows to derive variational schemes and the KF, which are characterized by a Gaussian assumption. Their properties were discussed, with a focus on their drawbacks in nonlinear, high-dimensional systems such as the earth's atmosphere. A large amount of research activity has taken place to advance these classical methods. The most important innovation consists in the adaptation of *ensemble* techniques, originally developed for forecasting purposes.
This chapter concentrates on ensemble-based filtering methods and reviews the properties and advances made with the ensemble KF (EnKF) and the particle filter (PF). Thus, it does not only contain important background information, but also several discussions that serve as direct preparation and motivation for the new filter algorithm in Chapter 4.

3.1 Ensembles and their application in data assimilation

The common basis of the EnKF and PF is an ensemble-based realization of the forecast step. Before investigating their mechanisms, the ensemble technique is introduced in general.

3.1.1 Ensemble forecasting

An *ensemble* of model states, shortly denoted as "ensemble", represents a set of N possible realizations of a system's state at a given time t_i, $\{\mathbf{x}_i^n\}_{n=1...N}$. Each individual state $\mathbf{x}_i^n$ is called an ensemble *member*. While a single, deterministic state hides the inherent uncertainty about the state, it is reflected by an ensemble that accounts for the fact that many other states may be true as well. Therefore, the ensemble can be interpreted as an N-sized sample of the underlying probability distribution $p(\mathbf{x}_i)$. Usually, each ensemble member is chosen to be equally likely, yielding an independent and identically distributed (i.i.d.) sample, conveniently written as $\mathbf{x}_i^n \sim p(\mathbf{x}_i)$.

Ensembles are extremely useful in forecasting applications. A chaotic system, such as in NWP, is highly sensitive to the initial conditions, meaning that errors grow exponentially in time. Due to the nonlinearity of the system, a deterministic integration of the best estimate at initial time does in general

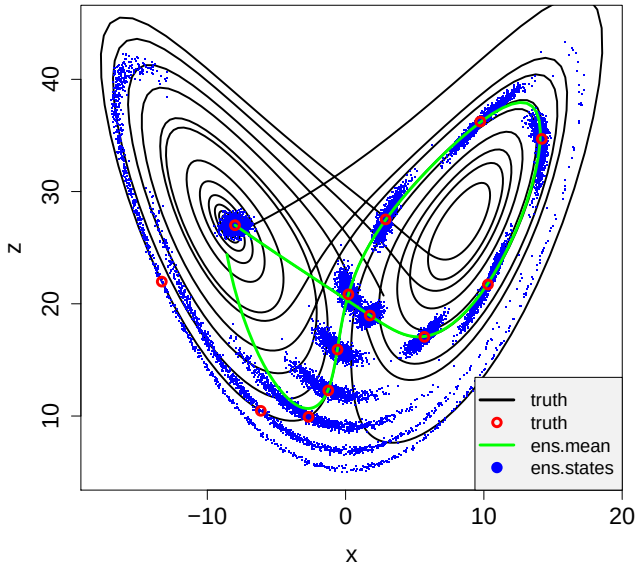


FIGURE 3.1: Ensemble prediction in the Lorenz-63 system (see Ch. 5). An integration from (-8,8,27) for 15 non-dimensional time units yields the true trajectory, visible by the Lorenz attractors in the x-z-plane. An ensemble (1000 members) is initialized by perturbing the true initial state with a variance of 0.25 for each variable (x,y,z). At times 0.0,0.1,...,1.2, the ensemble states are plotted (blue) to visualize their distribution, in comparison with the true state at these times (red) and the ensemble mean (green).

not result in the best estimate at future times. In contrast, in a probabilistic forecast the uncertainties in phase space are covered and propagated throughout time. Thus, stochastic predictability usually exceeds deterministic predictability (Kalnay 2003). The principal solution to gain a full probability forecast is the integration of the initial pdf, $p(\mathbf{x}_0)$, in time, using equation (2.25), which may be expressed by the Kolmogorov equation (2.26). As already discussed in section 2.4.4, this is not feasible except for simple small-scale systems. Instead, an ensemble $\{\mathbf{x}_0^n\}$ is drawn to represent the pdf $p(\mathbf{x}_0)$, and each member is integrated by the model, shortly denoted as *ensemble integration* (Epstein 1969). The resulting ensemble represents a sample of the pdf $p(\mathbf{x}_i)$ $(i > 0)$, which will be formally shown in the next subsection. Another option is to gather forecasts of different models in order to properly account for model error uncertainty as well.

This method corresponds to a Monte Carlo (Metropolis and Ulam 1949) solution of the Kolmogorov equation, and therefore only introduces sampling errors. An advantage is that the convergence rate of the Monte Carlo approach only depends on ensemble size and not on the systems' dimensionality. This makes is suitable for high-dimensional forecasting problems, and its implementation is fairly simple. The first operational ensemble prediction system (EPS) in NWP was introduced at ECMWF in 1992 (Molteni et al. 1996). The integrated ensemble implicitly carries all moments of the underlying pdf (see Appendix A). For example, while the ensemble mean often is the best estimate, the ensemble spread should quantify the flow-dependent uncertainty (Palmer et al. 2005). Furthermore, ensemble predictions allows to draw probability forecasts (e.g., Anderson 1996; Tödter 2011; Wilks 2011). Figure 3.1 schematically illustrates an ensemble prediction in the chaotic Lorenz-63 system, which will be used in Chapter 5. The ensemble distribution, initially Gaussian, broadens with time, reflecting the rising uncertainty. After the regime change, some members end up in the wrong attractor, and the ensemble distribution even becomes bimodal. Then, the ensemble mean ceases to represent a reliable prediction.

It is important to note that the quality of an ensemble forecast is influenced by the technique to generate the initial ensemble. Interestingly, the output of an ensemble-based DA system can be utilized for that purpose (Buizza et al. 2008; Hamill and Whitaker 2011; Wang and Bishop 2003), without resorting to more classical methods such as singular vectors or breeding (e.g., Toth and Kalnay 1997). In conclusion, ensemble integration is a valid and feasible approach to probabilistic forecasting. Next, it is demonstrated that it can be incorporated into sequential DA in a straightforward fashion.

3.1.2 Ensemble formulation of the general forecast step

In section 2.4, the general sequential filter was derived in a probabilistic formulation. It is only based on rather unproblematic assumptions, for instance, the model has to be a first order Markov process and temporal correlations are neglected. The sequential filter is characterized by an iteration of forecast and analysis steps through the time window. As discussed, the forecast step, formally described by the marginalization rule, requires to solve an integral over the joint-in-time pdf (e.g., eq. 2.21). Analytical solutions are limited, which is why classical methods resort to the Gaussian assumption.

3.1.2.1 Derivation

The introduction of an ensemble of states allows an elegant and efficient solution of this problem. As already illustrated, an ensemble forecast solves the underlying Fokker-Planck equation by a Monte Carlo approximation, which in particular does not require any linearization of the model. Now, the forecast step is carried out explicitly for an ensemble-based filter. This formulation is valid for all ensemble algorithms considered in this thesis. Hence, the following formulation can be viewed as an extension of the probabilistic framework formulated in section 2.4.

The starting point is equation (2.21), which is repeated here for reasons of clarity.

$$p(\mathbf{x}_i|\mathbf{y}_{1:i(j-1)}) = \int p(\mathbf{x}_{i-1}|\mathbf{y}_{1:i(j-1)})p(\mathbf{x}_i|\mathbf{x}_{i-1})d\mathbf{x}_{i-1} \tag{3.1}$$

It describes how to obtain the marginal prior pdf at the next time t_i ($i > i(j-1)$), given the state pdf at time t_{i-1} as well as the transition density. The dependence on past observations $\mathbf{y}_{1:i(j-1)}$ is stated for completeness, but is not relevant for the propagation of the pdf in time. Now, it is assumed that an ensemble is available whose members are identical and independently distributed (i.i.d.) according to the marginal pdf at time t_{i-1}, i.e., $\{\mathbf{x}_{i-1}^n\} \sim p(\mathbf{x}_{i-1}|\mathbf{y}_{1:i(j-1)})$. Therefore, the ensemble representation of this pdf can be written as (van Leeuwen 2009)

$$p(\mathbf{x}_{i-1}|\mathbf{y}_{1:i(j-1)}) \approx \frac{1}{N}\sum_{n=1}^{N} \delta\left(\mathbf{x}_{i-1} - \mathbf{x}_{i-1}^n\right) \tag{3.2}$$

After plugging this representation into equation (3.1), the linearity of the integral allows to exchange the order of integration and summation. Then, the resulting integrals are easily carried out using the

properties of the delta function (see also Appendix A).

$$p(\mathbf{x}_i|\mathbf{y}_{1:i(j-1)}) \approx \frac{1}{N}\sum_{n=1}^{N}\int p(\mathbf{x}_i|\mathbf{x}_{i-1})\,\delta\left(\mathbf{x}_{i-1}-\mathbf{x}_{i-1}^n\right)d\mathbf{x}_{i-1} = \frac{1}{N}\sum_{n=1}^{N} p(\mathbf{x}_i|\mathbf{x}_{i-1}^n) \tag{3.3}$$

Usually, the ensemble size is required to be conserved. Hence, this mixture pdf can simply be approximated by drawing one sample from each transition density $p(\mathbf{x}_i|\mathbf{x}_{i-1}^n)$ $(n = 1 \ldots N)$. As explained in section 2.4.4, sampling from $p(\mathbf{x}_i|\mathbf{x}_{i-1}^n)$ equals a model integration as in equation (2.4) and yields a value of $\mathbf{x}_i^n$. Hence, an ensemble representation of the new prior pdf at time t_i can be written as

$$p(\mathbf{x}_i|\mathbf{y}_{1:i(j-1)}) \approx \frac{1}{N}\sum_{n=1}^{N}\delta\left(\mathbf{x}_i-\mathbf{x}_i^n\right) \quad \text{where} \quad \mathbf{x}_i^n = \mathcal{M}_{(i-1)\to i}\left(\mathbf{x}_{i-1}^n\right)+\boldsymbol{\eta}_{i-1}^n \tag{3.4}$$

In conclusion, a new, equally-weighted ensemble $\{\mathbf{x}_i^n\}$ is simply gained by integrating each member with the full model operator, possibly including random model errors.

3.1.2.2 Discussion

A simple ensemble integration, using the full model, equals an approximate Monte Carlo solution of the underlying integral equation. For $N \to \infty$, this solution becomes exact. This approach offers three major advantages over the forecast step formulations used in the algorithms shown in Chapter 2:

1. The full, nonlinear model is used, which avoids the linearizations needed in the classical methods. This also releases the user from developing a tangent linear and adjoint version of the model code, which allows a simpler maintenance of the DA system. Forecast and analysis step are technically decoupled, and the former one can always use the latest version of the forward model.

2. The convergency rate of such a Monte Carlo solution does not depend on the dimension of the state space, but only on ensemble size ($\propto 1/\sqrt{N}$), offering an applicability also for high-dimensional assimilation problems. Indeed, the EnKF has already been applied successfully to large-scale atmospheric or oceanic systems, even operationally, with ensemble sizes of order $\mathcal{O}(10^2)$ (e.g., Bonavita et al. 2008; Houtekamer and Mitchell 2005). For comparison, the state dimensionality of such models may easily reach $\mathcal{O}(10^9)$, with $\mathcal{O}(10^7)$ observations.

3. Properties of the forecast ensemble can directly be computed from the ensemble, for example, the forecast error covariance. It is implicitly contained in the ensemble as a low-rank approximation of the true error covariance matrix, and therefore, it is implicitly integrated in time with the full, nonlinear model (Evensen 1994). This releases from the expensive, linearized approximation required by the KF.

In the following, it will be demonstrated that the new approach to the forecast step can be combined with different analysis step formulations, resulting in different flavors of ensemble-based sequential filtering algorithms. In principle, most of the developments are either based on the KF or the PF, or employ a combination of both methods. The presentation will focus on the ensemble Kalman filter (EnKF) and its deterministic variants, in particular the ETKF, and also discusses some enhancements to

the basic formulations. Then, the basic PF will be introduced. Some recent advances in this area will be highlighted, in particular the equivalent weight PF (EWPF) and nonlinear ensemble adjustment filter (NLEAF). This treatment will almost directly lead to the nonlinear ensemble transform filter (NETF) in the following chapter, which is based on the PF, but borrows its methodology from the ETKF.

3.2 Ensemble Kalman filter

As of today, the EnKF is a very popular filter. In principle, it combines the ensemble solution of the general forecast step with the analysis step of the KF. This section discusses the implications of this approach and its explicit mathematical formulation. The review begins with the original KF formulation as given by Evensen (1994) and its first revisions. The next sections then turn to more recent developments by presenting alternative analysis step mechanisms. Finally, practical adjustments of the basic filter are emphasized that are decisive for successful applications in larger-dimensional systems with relatively small ensemble sizes.

3.2.1 Motivation

Knowing the advantages of Monte Carlo forecasting, it seems almost natural to apply it to the general forecast step of sequential DA, as shown in the previous section. Using a quasi-geostrophic ocean model, Evensen (1992) emphasized that the simplified closure given by the error covariance equation of the KF inherently causes an unbounded error growth in chaotic systems due to nonlinearly evolving dynamical errors. Based on this insight, Evensen (1994) applied the Monte Carlo method to this model and found the latter to be superior to an approximate stochastic dynamic prediction (Epstein 1969) that consists in employing a Taylor expansion of the model operator in order to propagate moments of the pdf (Cohn 1993). The reason is that dynamical errors are avoided, only statistical noise is introduced. This also holds for sophisticated higher-order closures of approximate stochastic dynamic predictions, which are anyway impractical in larger-dimensional systems (Miller et al. 1994). In summary, Evensen (1994) holds the view that the major drawback of the KF is not its update mechanism, but the poor quality of the input data to the analysis step, in particular, the error covariance matrix.

3.2.2 Forecast step

These insights led Evensen (1994) to the seminal idea of replacing the KF forecast step by an ensemble forecast, as introduced in section 3.1.1. It integrates the analysis ensemble of the previous observation time until the next observation time, using the fully-nonlinear model. Hence, for a complete description of the EnKF, it is sufficient to discuss the analysis step formulation as a specific solution of the general probabilistic framework.

For the following presentations, it is convenient to extend the notation. Similar to the presentation of the KF in section 2.6, the forecast or prior ensemble at the current observation time $t_{i(j)}$, $\{\tilde{\mathbf{x}}^n_{i(j)}\}$, is denoted as $\{\mathbf{x}^n_f\}_{n=1\ldots N}$. However, working with the individual ensemble members is impractical considering a numerical implementation. Instead, the ensemble states are collected into the columns of

a $K \times N$ *ensemble matrix*, i.e., $\mathbf{X}_f = \left(\mathbf{x}_f^1, \ldots, \mathbf{x}_f^N\right)$. Furthermore, the *ensemble perturbation matrix* is defined, which collects the ensemble anomalies by subtracting the ensemble mean from each column, $\mathbf{X}'_f = \left(\mathbf{x}_f^1 - \overline{\mathbf{x}}_f, \ldots, \mathbf{x}_f^N - \overline{\mathbf{x}}_f\right)$. A shortened notation is $\mathbf{X}'_f = \mathbf{X}_f - \overline{\mathbf{X}}_f$ where each column of the matrix $\overline{\mathbf{X}}_f$ consists of the mean $\overline{\mathbf{x}}_f$. Then, the ensemble's empirical first- and second-order statistics is easily computed by (see Appendix A)

$$\overline{\mathbf{x}}_f = \frac{1}{N}\sum_n \mathbf{x}_f^n = \frac{1}{N}\mathbf{X}_f\mathbf{1} \quad \text{and} \quad \mathbf{P}_f = \frac{1}{N-1}\mathbf{X}'_f\mathbf{X}'^T_f \tag{3.5}$$

where $\mathbf{1} = (1, \ldots, 1)^T$.

3.2.3 Original analysis step formulation

As motivated, the EnKF relies on the usual KF analysis step, with the "only" advantage that the input variables, in particular the forecast mean and covariance, are expected to be a better approximation of the truth since the forecast steps fully respects the nonlinearity of the system. The direct adaption of the KF analysis step (Evensen 1994) consists in applying the KF update equation (2.43) to each individual ensemble member, yielding

$$\mathbf{x}_a^n = \mathbf{x}_f^n + \mathbf{K}_j\left(\mathbf{y}_j - \mathcal{H}(\mathbf{x}_f^n)\right) \tag{3.6}$$

Taking the ensemble average of equation (3.6), it is easily verified that the analysis mean,

$$\overline{\mathbf{x}}_a = \overline{\mathbf{x}}_f + \mathbf{K}_j(\mathbf{y} - \overline{\mathbf{y}}_f) \quad \text{where} \quad \overline{\mathbf{y}}_f = \overline{\mathcal{H}(\mathbf{X}_f)} = \frac{1}{N}\sum_n \mathcal{H}(\mathbf{x}_f^n) \tag{3.7}$$

corresponds to the KF mean (see equation 2.43), at least for linear observation operators. The Kalman gain is computed as usual. After the analysis ensemble $\{\mathbf{x}_a^n\}$ is obtained, it is considered as the new ensemble $\{\mathbf{x}_{i(j)}^n\}$ valid at time $t_{i(j)}$, replacing the prior ensemble. Then, the next forecast phase can be initialized.

3.2.4 Observation space treatment

The evaluation of the Kalman gain, $\mathbf{K}_j = \mathbf{P}_f\mathbf{H}_j^T\left(\mathbf{H}_j\mathbf{P}_f\mathbf{H}_j^T + \mathbf{R}_j\right)^{-1}$, requires the tangent linear and adjoint of the observation operator, $\mathbf{H}$ and $\mathbf{H}^T$, as well as the full error covariance matrix $\mathbf{P}_f$. However, this can be avoided by realizing that, in equation (3.6), the model equivalents to the observation are needed. Hence, all ensemble states are mapped into observation space via $\mathbf{y}_f^n = \mathcal{H}(\mathbf{x}_f^n)$. These vectors are stored in the $L \times N$ ensemble matrix $\mathbf{Y}_f = \left(\mathcal{H}(\mathbf{x}_f^1), \ldots, \mathcal{H}(\mathbf{x}_f^N)\right)$. Then, the update equation (3.6) can be written compactly as

$$\mathbf{X}_a = \mathbf{X}_f + \mathbf{K}_j\left(\mathbf{Y} - \mathbf{Y}_f\right) \tag{3.8}$$

Here, $\mathbf{Y}$ is a matrix containing the observation $\mathbf{y}_j$ in each of its columns. For $\mathbf{P}_f$ in the Kalman gain formula, the ensemble representation, equation (3.5), is used, leading to terms $\mathbf{H}\mathbf{X}'_f$ and $(\mathbf{H}\mathbf{X}'_f)^T$. Here, $\mathbf{H}\mathbf{x}_f^{n\prime}$ represents a linearized mapping of the prior perturbations into observation space. This is

actually not needed, as the mapped ensemble $\mathbf{Y}_f$ is already available. This enables to compute the corresponding anomalies directly, if $\mathbf{HX}'_f$ is associated with $\mathbf{Y}'_f = \mathbf{Y}_f - \overline{\mathbf{Y}}_f$. This technique does not only free from applying a tangent linear and adjoint version of the observation operator, but also avoids linearizations[1]. To summarize, the Kalman gain is computed efficiently in ensemble subspace by

$$\mathbf{K}_j = \frac{\mathbf{X}'_f \mathbf{Y}'^T_f}{N-1} \left(\frac{\mathbf{Y}'_f \mathbf{Y}'^T_f}{N-1} + \mathbf{R}_j \right)^{-1} = \mathbf{X}'_f \mathbf{Y}'^T_f \left(\mathbf{Y}'_f \mathbf{Y}'^T_f + (N-1)\mathbf{R}_j \right)^{-1} \tag{3.9}$$

3.2.5 Perturbation of the observations

Evensen (1994) and Evensen and van Leeuwen (1996) applied the just presented EnKF formulation to multilayer, nonlinear and quasi-geostrophic ocean models, using both artificial and real observation data. The results were quite promising, compared to the KF, which induced further research about the method. Burgers et al. (1998) revealed that the naive adaption of the KF update equation (2.43) for all ensemble members is incorrect. The analysis mean matches the KF prediction, but the analysis covariance is underestimated as compared to its theoretical value. Therefore, it produces a too tight ensemble, which in the successive analysis steps might lead to an overconfident prior ensemble that cannot properly account for the observation anymore. This might even lead to filter divergency, a situation, where the ensemble collapses and the filter simply ignores the observations. This problem did not occur in the mentioned applications because the observation frequency was relatively low, which allowed the ensembles to gain sufficient spread during the long forecast steps.

For a more formal treatment of this issue, the analysis mean, equation (3.7), is subtracted from equation (3.6), yielding the analysis perturbations:

$$\mathbf{X}'_a = \mathbf{X}'_f + \mathbf{K}_j \left(\overline{\mathbf{Y}}_f - \mathbf{Y}_f \right) = \mathbf{X}'_f - \mathbf{K}_j \mathbf{Y}'_f \tag{3.10}$$

This allows to evaluate the ensemble covariance matrix $\mathbf{P}_a$, where again $\mathbf{Y}'_f$ is interpreted as the ensemble equivalent to $\mathbf{HX}'_f$.

$$\begin{aligned} \mathbf{P}_a &= \frac{1}{N-1} \mathbf{X}'_a \mathbf{X}'^T_a = \frac{1}{N-1} \left[\mathbf{X}'_f \mathbf{X}'^T_f - \mathbf{X}'_f \mathbf{Y}'^T_f \mathbf{K}^T - \mathbf{K} \mathbf{Y}'_f \mathbf{X}'^T_f + \mathbf{K} \mathbf{Y}'_f \mathbf{Y}'^T_f \mathbf{K}^T \right] \\ &= \mathbf{P}_f - \mathbf{P}_f \mathbf{H}^T \mathbf{K}^T - \mathbf{K} \mathbf{H} \mathbf{P}^T_f + \mathbf{K} \mathbf{H} \mathbf{P}_f \mathbf{H}^T \mathbf{K}^T = (\mathbf{I} - \mathbf{KH}) \mathbf{P}_f (\mathbf{I} - \mathbf{KH})^T \end{aligned}$$

Compared with the theoretical KF result in equation (2.44), there is an additional factor $(\mathbf{I} - \mathbf{KH})^T$ which further reduces the analysis covariance and confirms the discussed problem. The reason is that the uncertainty in observation space coming from the observation is not considered, only the mapped perturbations of the prior ensemble are present in equation (3.10).

Based on this insight, Burgers et al. (1998) suggested a simple workaround. Instead of updating each member with the fixed observation, an ensemble of perturbed observations, $\{\mathbf{y}^n\}$, is drawn from the likelihood density, $\mathcal{N}(\mathbf{y}, \mathbf{R})$. It is collected in the ensemble matrix $\mathbf{Y}$, which can now be written as $\mathbf{Y} = \overline{\mathbf{Y}} - \mathbf{Y}'$, where the perturbations are samples of $\mathcal{N}(\mathbf{0}, \mathbf{R})$. This procedure does not affect the

[1] However, it should be mentioned that the form of the Kalman gain itself is derived with a linear ansatz as motivated in Chapter 2.

ensemble mean in equation (3.7), as $\overline{\mathbf{Y}'} = \mathbf{0}$ and $\overline{\mathbf{Y}}$ contains the actual observation, at least in the statistical limit. Furthermore, the idea is reasonable because in an ensemble approach it should be accounted for that the actual observation is only one realization, and any other sample from $\mathcal{N}(\mathbf{y}, \mathbf{R})$ is equally likely to be true. In other words, the observation is treated as a random variable as well.

The impact of the modification can directly be checked, as the perturbations now read $\mathbf{X}'_a = \mathbf{X}'_f + \mathbf{K}_j(\mathbf{Y}' - \mathbf{Y}'_f)$, in contrast to equation (3.10). As the observation and ensemble perturbations are uncorrelated, cross-terms such as $\mathbf{X}'_f\mathbf{Y}'^T$ or $\mathbf{Y}'\mathbf{X}'^T_f$ vanish, and furthermore $\mathbf{Y}'\mathbf{Y}'^T/(N-1) \approx \mathbf{R}$ holds in a statistical sense. The analysis covariance therefore exhibits an additional term,

$$\mathbf{P}_a \approx \mathbf{P}_f - \mathbf{P}_f\mathbf{H}^T\mathbf{K}^T - \mathbf{K}\mathbf{H}\mathbf{P}_f^T + \mathbf{K}\mathbf{H}\mathbf{P}_f\mathbf{H}^T\mathbf{K}^T + \mathbf{K}\mathbf{R}\mathbf{K}^T$$

This expression also appears in the derivation of the KF, see equation (B.28) in Appendix B, where it is shown to be equal to $(\mathbf{I} - \mathbf{K}\mathbf{H})\mathbf{P}_f$. This proves the validity of the perturbed-observations EnKF. In section 3.3, its implications will be further discussed.

3.2.6 Final EnKF algorithm

The EnKF represents a Monte Carlo solution of the general sequential filter within the probabilistic framework of Chapter 2. As it is a fundamental algorithm and highly relevant for this work, the complete algorithm is now summarized in compact form.

1. At time t_0, draw an initial ensemble $\{\mathbf{x}_0^n\}_{n=1\ldots N}$. For example, this can be done by drawing N samples from a multivariate Gaussian density with desired mean and covariance.

2. Proceed from one observation to the next by iterating the following steps for $j = 1 \ldots J$:

   (a) Forecast step: Integrate the ensemble until the next observation time $t_{i(j)}$ using the full model, possibly with random forcing. This is realized by repeated application of the model operator for $i = (i(j-1)+1) \ldots i(j)$ for each member:

   $$\mathbf{x}_i^n = \mathcal{M}_{i-1\to i}(\mathbf{x}_{i-1}^n) + \boldsymbol{\eta}_{i-1}^n \tag{3.11}$$

   The result is the prior ensemble at time $t_{i(j)}$. Assign $\mathbf{x}_{i(j)}^n \to \mathbf{x}_f^n$, put the states in the ensemble matrix $\mathbf{X}_f$ and evaluate $\overline{\mathbf{x}}_f$ and $\mathbf{X}'_f$.

   (b) Analysis step:

      i. Map the forecast ensemble into observation space by applying the fully nonlinear observation operator, $\mathbf{y}_f^n = \mathcal{H}_j(\mathbf{x}_f^n)$, and put it into the ensemble matrix $\mathbf{Y}_f$. Evaluate $\overline{\mathbf{y}}_f$ and $\mathbf{Y}'_f$.

      ii. Generate the perturbed observations $\{\mathbf{y}_j^n\}$ by drawing N samples from $\mathcal{N}(\mathbf{y}_j, \mathbf{R}_j)$ and put them into the ensemble matrix $\mathbf{Y}$.

      iii. Compute the Kalman gain matrix using the ensemble matrices in state and observation space, $\mathbf{K}_j = \mathbf{X}'_f\mathbf{Y}'^T_f \left(\mathbf{Y}'_f\mathbf{Y}'^T_f + (N-1)\mathbf{R}_j\right)^{-1}$.

iv. Update the forecast to the analysis ensemble:

$$\mathbf{X}_a = \mathbf{X}_f + \mathbf{K}_j \left(\mathbf{Y} - \mathbf{Y}_f \right) \tag{3.12}$$

Analysis properties, such as the mean $\overline{\mathbf{x}}_a$, can now be derived. Before proceeding, replace the forecast ensemble with the analysis ensemble by assigning $\mathbf{x}_a^n \rightarrow \mathbf{x}_{i(j)}^n$.

3. Integrate the analysis ensemble of the final observation time, $t_{i(J)}$, to the final time t_I by repeating single-step integrations as in equation (3.11), except that the index range now is $i = (i(J) + 1) \ldots I$. This results in an analysis ensemble $\{\mathbf{x}_I^n\}$ as a sample of $p(\mathbf{x}_I|\mathbf{y}_{1:J})$.

This analysis step formulation only involves basic matrix computations. Therefore, the algorithm can be implemented efficiently in most programming languages. Furthermore, the error covariance matrices $\mathbf{P}_f$ and $\mathbf{P}_a$ are never computed or stored explicitly.

3.3 Deterministic EnKFs

This section analyzes different EnKF formulations. They represent alternatives to the algorithm presented in the previous section, referred to as the classical EnKF.

3.3.1 Motivation and techniques

Perturbing the observation ensures that the theoretical value of the analysis covariance, as given by the KF, is achieved in a statistical sense (Burgers et al. 1998). However, such a stochastic component within the analysis step does not only require an appropriate technical implementation, but also introduces sampling errors. The latter can be of notable magnitude for small and medium-sized ensembles that are typical for high-dimensional applications (Whitaker and Hamill 2002). In conclusion, the classical EnKF is suboptimal by construction. For that reason, alternative algorithms have been derived to avoid the perturbation of the observation. They are classified as *deterministic* EnKFs, and borrow the idea from a square root formulation of the classical KF (Andrews 1968).

The analytical result for the analysis covariance, $\mathbf{P}_a^{\mathrm{KF}}$, is given by equation (2.44). In the stochastic EnKF, this equation is never used as the analysis covariance is implicitly contained in the updated ensemble. It can be estimated by $\mathbf{X}'_a(\mathbf{X}'_a)^T/(N-1)$, which only converges to $\mathbf{P}_a^{\mathrm{KF}}$ for $N \rightarrow \infty$. The paradigm of a deterministic EnKF consists in creating an analysis ensemble such that its empirical covariance is exactly equal to $\mathbf{P}_a^{\mathrm{KF}}$, even for finite N. The common mathematical basis of these variants is a matrix transformation of the forecast ensemble anomalies, instead of adjusting each member individually.

$$\mathbf{X}'_a = \mathbf{X}'_f \mathbf{T} \quad \text{where} \quad \frac{1}{N-1} \mathbf{X}'_f \mathbf{T} \mathbf{T}^T \mathbf{X}'_f = \mathbf{P}_a^{\mathrm{KF}} \tag{3.13}$$

Note that the transformation is carried out in ensemble subspace, and $\mathbf{T}$ is a $N \times N$ matrix. Furthermore, Sakov and Oke (2008b) emphasized that, in order to preserve the analysis mean, $\mathbf{X}'_a \mathbf{1} = \mathbf{0}$, the

transform matrix should also exhibit $\mathbf{T1} = \alpha\mathbf{1}$ ($\alpha \in \mathbb{R}$). Usually, $\mathbf{T}$ arises as a matrix square root. Afterwards, the analysis mean can be updated by the usual equation (3.7), and the full analysis ensemble is regained by $\mathbf{X}_a = \mathbf{X}'_a + \overline{\mathbf{X}}_a$.

Tippett et al. (2003) reviewed three alternative algorithms within the framework of Kalman square root filters (Maybeck 1982). The ensemble transform Kalman filter (ETKF, Bishop et al. 2001) can be formulated as in equation (3.13), while the ensemble square root filter (EnSRF, Whitaker and Hamill 2002) and the ensemble adjustment Kalman filter (EAKF, Anderson 2001) both employ a left-sided multiplication in state space,

$$\mathbf{X}'_a = \mathbf{T}\mathbf{X}'_f \tag{3.14}$$

Another deterministic EnKF has been suggested by Sakov and Oke (2008a). It is applies a transformation as in equation (3.14), but does not require to compute a matrix square root.

3.3.2 Ensemble transform Kalman filter

The analysis technique used in the ETKF (Bishop et al. 2001) offers a good insight into the principle of deterministic EnKFs. Additionally, is also highly relevant for this thesis as the new filter will be derived in a similar way in Chapter 4. The basis is the theoretical KF covariance equation (2.44). Then, the Kalman gain is plugged in and the ensemble representation of the prior covariance matrix is used:

$$\begin{aligned}
\mathbf{P}_a^{\mathrm{KF}} &= \mathbf{P}_f - \mathbf{K}\mathbf{H}\mathbf{P}_f = \frac{1}{N-1}\left[\mathbf{X}'_f\mathbf{X}'^T_f - \mathbf{X}'_f\mathbf{Y}'^T_f\left(\mathbf{Y}'_f\mathbf{Y}'^T_f + (N-1)\mathbf{R}\right)^{-1}\mathbf{H}\mathbf{X}'_f\mathbf{X}'^T_f\right] \\
\mathbf{P}_a^{\mathrm{KF}} &= \frac{1}{N-1}\mathbf{X}'_f\left[\mathbf{I}_N - \mathbf{Y}'^T_f\left(\mathbf{Y}'_f\mathbf{Y}'^T_f + (N-1)\mathbf{R}\right)^{-1}\mathbf{Y}'_f\right]\mathbf{X}'^T_f \qquad (3.15) \\
\Rightarrow \mathbf{P}_a^{\mathrm{KF}} &= \mathbf{X}'_f\underbrace{\left[(N-1)\mathbf{I}_N + \mathbf{Y}'^T_f\mathbf{R}^{-1}\mathbf{Y}'_f\right]^{-1}}_{\mathbf{A}^{\mathrm{KF}}}\mathbf{X}'^T_f \qquad (3.16)
\end{aligned}$$

In the last step, a variant of the Sherman-Morrison-Woodbury identity (Horn and Johnson 1985, p. 19, see also Appendix B) was used. Note that using equation (3.15) instead of (3.16) for defining $\mathbf{A}^{\mathrm{KF}}$ in the following leads to another square root filter (Evensen 2004), which therefore differs from the ETKF only by its technical implementation. Next, it has to be ensured that the empirical analysis covariance exactly matches $\mathbf{P}_a^{\mathrm{KF}}$, i.e.,

$$\frac{1}{N-1}\mathbf{X}'_a\mathbf{X}'^T_a = \mathbf{X}'_f\mathbf{A}^{\mathrm{KF}}\mathbf{X}'^T_f \tag{3.17}$$

A solution for this equation is found by the help of the matrix square root of $\mathbf{A}^{\mathrm{KF}}$, which leads to a matrix transform as already introduced in equation (3.13).

$$\mathbf{X}'_a = \sqrt{N-1}\,\mathbf{X}'_f\mathbf{T}^{\mathrm{KF}} \quad \text{where} \quad \mathbf{T}^{\mathrm{KF}}(\mathbf{T}^{\mathrm{KF}})^T = \mathbf{A}^{\mathrm{KF}} \tag{3.18}$$

Since $\mathbf{A}^{\mathrm{KF}}$ is positive definite, the unique positive definite square root (Horn and Johnson 1985, p. 406) is usually used (Sakov and Oke 2008b). It can be obtained via a singular value decomposition (SVD, see Appendix A).

Finally, the analysis mean could be computed as in equation (3.7), but a more efficient formulation can be obtained. Using the new expression (3.16) in a different formulation of the Kalman gain, $\mathbf{K} = \mathbf{P}_a\mathbf{H}^T\mathbf{R}^{-1}$ (see Appendix B, equation B.5), directly yields $\mathbf{K} = \mathbf{X}'_f\mathbf{A}^{\mathrm{KF}}\mathbf{Y}'^T_f\mathbf{R}^{-1}$, if again $\mathbf{X}'^T_f\mathbf{H}^T$ is associated with $\mathbf{Y}'^T_f$. Thus, the analysis mean is

$$\overline{\mathbf{x}}_a = \overline{\mathbf{x}}_f + \mathbf{X}'_f\mathbf{w}^{\mathrm{KF}} \quad \text{where} \quad \mathbf{w}^{\mathrm{KF}} = \mathbf{A}^{\mathrm{KF}}\mathbf{Y}'^T_f\mathbf{R}^{-1}(\mathbf{y} - \overline{\mathbf{y}}_f) \tag{3.19}$$

The advantage is that both $\mathbf{A}^{\mathrm{KF}}$ and the matrix $\mathbf{Y}'^T_f\mathbf{R}^{-1}$ have already been evaluated to obtain $\mathbf{T}^{\mathrm{KF}}$. The ETKF computes the analysis increment as a weighted linear combination of the prior perturbations. In conclusion, the ETKF represents an appealing deterministic EnKF formulation because the analysis is computed in ensemble subspace, allowing an efficient numerical implementation. Compared to the EnKF algorithm presented in section 3.2.6, only steps iii and iv of the analysis step are modified, and step ii is omitted.

3.3.3 Random rotations

A general deterministic EnKF is given by equation (3.13). Even though the symmetric matrix square root of $\mathbf{A}^{\mathrm{KF}}$ is unique, this does not hold for the transform matrix itself. For instance, choosing $\tilde{\mathbf{T}} = \mathbf{T}\mathbf{U}$, where $\mathbf{U}$ is an *orthogonal* matrix, does not change the empirical analysis covariance due to $\mathbf{U}\mathbf{U}^T = \mathbf{I}$. As a consequence, the freedom in the determination of the transform matrix led to the suggestion to use some random matrix $\mathbf{\Lambda}$ at this stage (Evensen 2004). As long as it is orthogonal, $\mathbf{\Lambda}^T\mathbf{\Lambda} = \mathbf{I}$ and has the vector $\mathbf{1} = (1, \ldots, 1)^T$ as eigenvector, $\mathbf{\Lambda}\mathbf{1} = \mathbf{1}$, such an augmented rotation does not affect the first- and second-order statistics of the analysis ensemble.

Furthermore, several studies (Anderson 2010; Lawson and Hansen 2004; Lei et al. 2010) pointed out that a purely deterministic EnKF update may possibly generate a skewed ensemble. Then, a few outliers guarantee the required moments, but the remaining states exhibit a too tight ensemble. In chapters 4 and 5, the potential usefulness of random rotations will be further discussed. Appendix C is devoted to the efficient construction of such random matrices.

3.4 Supplements for EnKFs

Even though the EnKF and its variants represent an appealing alternative to variational schemes, practical issues arise in applications which are mainly caused by the restriction of a rather limited ensemble size (Anderson 2007b; Lorenc 2003). If unrecognized, they can severely deteriorate the filter performance (Houtekamer et al. 2005). However, also 4DVAR implementations have evolved since more than 20 years, hence, this is not an a priori disadvantage of EnKFs. Here, some principle supplements are shown which are of relevance for an actual EnKF implementation. *Inflation* aims at a better representation of the ensemble uncertainty, and *localization* suppresses spurious correlations (e.g., Janjić et al. 2011; Ott et al. 2004). It has been shown that in combination with such well-implemented techniques, the filter performance in high-dimensional systems can be enhanced significantly and made comparative to established variational schemes (Buehner et al. 2010; Kalnay et al. 2007; Yang et al. 2012a).

3.4.1 Covariance inflation

Inflation tackles the problem that EnKFs tend to underestimate the prior variances and to produce a too tight ensemble, particularly for two reasons: (1) the finite-sized ensemble forces to use a low-rank approximation of the covariance matrix and (2) unregarded model shortcomings (e.g., missing or crude model error handling) lead to an underestimation of the actual uncertainty (Whitaker and Hamill 2012).

The most straightforward solution (Anderson and Anderson 1999) is the multiplication of the ensemble perturbations by a factor $\sqrt{\gamma}$ which is slightly larger than 1,

$$\mathbf{X}' \rightarrow \sqrt{\gamma}\,\mathbf{X}' \tag{3.20}$$

This empirical procedure, which can be applied to either the forecast or the analysis ensemble, inflates the covariance matrix by the factor γ. In most applications, it turns out that the performance of EnKFs can be significantly increased with the help of inflation, particularly for small ensemble sizes, and that the EnKF results are sensitive to inflation (Houtekamer et al. 2005). However, it should be emphasized that inflation is a pragmatic ad-hoc enhancement to overcome the effects associated with limited ensemble size, but it renders the filter suboptimal. Thus, the proper tuning of the inflation factor represents an important, yet time-consuming issue. It should counteract the underestimation of $\mathbf{P}$ but simultaneously avoid overshooting. For large N, the inflation factor should approach 1, at least in a perfect model environment.

The basic approach applies equation (3.20) with a constant inflation factor, which implies that the structure the of variance underestimation is stationary in time and space. The development and investigation of more advanced adaptive inflation schemes is an active field of research. For example, a general hierarchical Bayesian approach to account for spatial and temporal variation the inflation factor has been established (Anderson 2007a, 2009), which can be simplified using Gaussian inflation pdfs (Li et al. 2009; Miyoshi 2011). Other approaches assume additive inflation (Houtekamer and Mitchell 2009; Whitaker et al. 2008) by adding random perturbations to the ensemble or perform a relaxation of the analysis perturbations to the prior ones (Zhang et al. 2004).

3.4.2 Localization

Another severe problem of the EnKFs in large-scale applications arises from the low-rank approximation of the covariance matrix due to limited ensemble size. It leads to random errors in the estimated covariance matrix. In particular, spurious correlations may be predicted between distant points, which in reality exhibit no connection. In the analysis step, this can lead to unrealistic increments due to distant observations. The basic concept to counteract this issue is to compute the analysis increments only with observations that lie within a certain radius of influence[2]. This can be realized by directly filtering the state covariance matrix appearing in the Kalman gain or by partitioning the state vector to compute a set of local analyses.

[2]This also suppresses potential long-range correlations that can appear for example in atmospheric systems. Their consideration in localization schemes is currently under research (Wu et al. 2014).

3.4.2.1 Correlation functions and matrices

Correlation functions $\rho(\cdot)$ model the correlation structure of a multivariate random variable (see also Appendix A). Together with the variances, they allow to form a covariance matrix. In its general form, $\rho(\mathbf{r}, \tilde{\mathbf{r}})$ is a continuous function that describes the correlation of a random variable at two locations $\mathbf{r}$ and $\tilde{\mathbf{r}}$. A comprehensive framework about correlation functions in multivariate settings has been established by Gaspari and Cohn (1999), who focussed on homogenous and isotropic functions. They are characterized by a monotonous decay with distance in physical space, which is usually defined by the Euclidean norm in $\mathbb{R}^3$, $r = ||\mathbf{r} - \tilde{\mathbf{r}}||$, such that $\rho(\mathbf{r}, \tilde{\mathbf{r}}) = \rho_0(r)$. In most applications, the space is discretized. Then, the correlation function can be used to create a corresponding correlation matrix $\boldsymbol{\rho}$ with elements $\rho_{ij} = \rho(\mathbf{r}_i, \mathbf{r}_j)$. The Schur product theorem (Horn and Johnson 1985, p. 458) guarantees that the Schur (element-wise) product of two covariance matrices, denoted by $\circ$, remains a proper covariance matrix. Hence, the correlation function can be used to taper an existing covariance matrix $\mathbf{C}$ via $\mathbf{C} \circ \boldsymbol{\rho}$, which conserves the variances.

Numerous choices for $\rho_0(r)$ with the desired properties exist, for example, an exponential or Gaussian function. However, in DA, the most common choice (Houtekamer and Mitchell 2001; Kirchgessner et al. 2014) is a piecewise 5th-order polynomial (Gaspari and Cohn 1999, Eq. 4.10) because it decays in a Gaussian fashion but has compact support. It can be derived by self-convolution of a triangular function over $\mathbb{R}^3$, resulting in (with $\tilde{r} = r/c$),

$$\rho_0(r) = \begin{cases} -\frac{1}{4}\tilde{r}^5 + \frac{1}{2}\tilde{r}^4 + \frac{5}{8}\tilde{r}^3 - \frac{5}{3}\tilde{r}^2 + 1 & \text{for } r \leq c \\ \frac{1}{12}\tilde{r}^5 - \frac{1}{2}\tilde{r}^4 + \frac{5}{8}\tilde{r}^3 + \frac{5}{3}\tilde{r}^2 - 5\tilde{r} + 4 - \frac{2}{3}\tilde{r}^{-1} & \text{for } c < r \leq 2c \\ 0 & \text{for } r > 2c \end{cases} \tag{3.21}$$

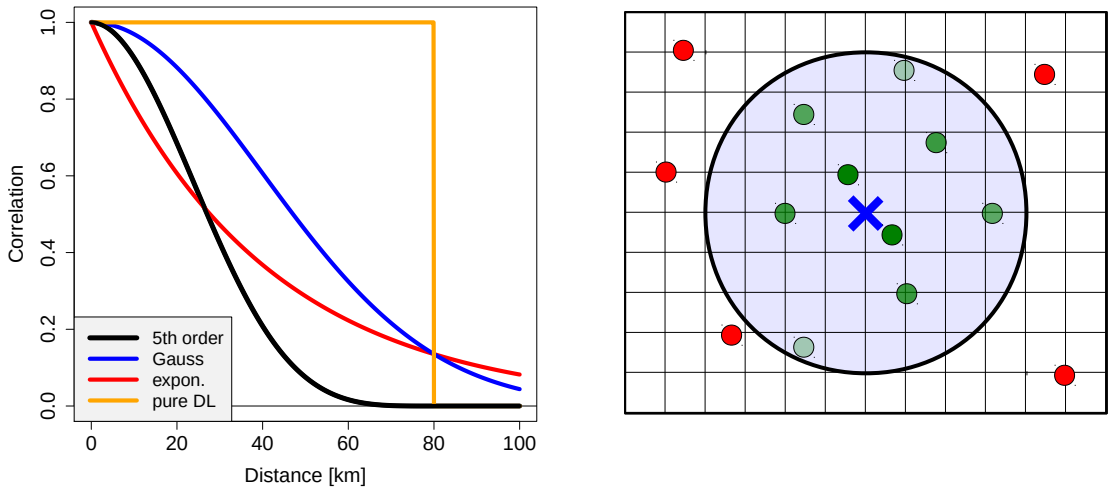
The parameter c defines the correlation length scale. The cut-off radius, also called *localization radius*, is at $2c$. Figure 3.2a visualizes equation (3.21) in comparison to an exponential and a Gaussian correlation function.

3.4.2.2 Covariance localization (CL)

The EnKF relies on the prior error covariance matrix $\mathbf{P}_f$. As it is estimated from the ensemble, effectively a low-rank approximation of the true error covariance matrix is used. This may result in spurious correlations particularly for large distances. A possible solution, first proposed by Houtekamer and Mitchell (2001), is to manually suppress them by adjusting the forecast covariance matrix using a suitable correlation function, $\tilde{\mathbf{P}}_f = \mathbf{P}_f \circ \boldsymbol{\rho}$. Such a preconditioning of $\mathbf{P}_f$ effectively filters out the unwanted noisy correlations with distant locations and guarantees a smooth decay of an observation's impact on the analysis increments with distance.

3.4.2.3 Domain and observation localization (DL/OL)

A distinct approach, first suggested by Houtekamer and Mitchell (1998), is to divide the whole region, represented by the state vector $\mathbf{x}$, into *local domains*. Typically, all state variables belonging to each grid



(a) Correlation functions

(b) Domain localization

FIGURE 3.2: (a) Example of correlations functions: 5th order polynomial (black), Gaussian (blue), exponential (red), hard cut-off (orange). The scale parameter is set to $80\,km$.
(b) Principle of DL/OL on a 2D grid: The current local domain is the grid point marked by the blue cross. Only the green observations, lying within a localization radius of 4 grid points (gray circle), are considered for its update, while the red ones are ignored. The green's thickness roughly illustrates the weight of the observation as used in OL.

point are collected into a local domain, but it may also represent a larger region, for example, a vertical atmospheric column. Then, the analysis is computed independently for each local domain, and only local observations are considered. For instance, one may choose to include only observations within a specified localization radius. The global analysis consists of the composition of all local analyses. Figure 3.2b visualizes the principle of DL on an exemplary 2D grid. In effect, the global analysis is not a linear combination of the prior ensemble anymore. Thus, the analysis subspace dimension and the rank of the empirical error covariance matrix are artificially increased, which is seen as a benefit.

However, DL can lead to a global analysis that negatively affects balances and is not continuous (Hunt et al. 2007). The reason is that observations considered for one grid point may not be included in the analysis of the neighboring grid point. Hence, the weight of an observation abruptly decreases to zero, see Figure 3.2a. In order to produce a more smooth analysis, DL can be combined with CL applied in observation space, where observations are weighted according to their distance from the local domain. This is achieved by constructing a Schur product of the inverse of the observation error covariance matrix, $\mathbf{R}^{-1}$ and a matrix obtained from a suitable correlation function (Miyoshi and Yamane 2007). Here, a typical choice is again the 5th order polynomial function for the same reasons as discussed previously. Thus, first the subset of relevant observation for the current local domain is extracted by searching within the localization radius and then, each remaining observation is weighted by the correlation function. This type of localization is referred to as observation localization (OL), and has been shown to outperform pure DL (e.g., Janjić et al. 2011). The latter can be viewed as a special case of OL where the weight function is equal to 1 inside the localization radius, and vanishes outside.

3.4.2.4 Discussion and comparison

Localization has emerged as an essential supplement for EnKF-based filters. It allows the usage of moderate ensemble sizes even in high-dimensional systems, such as in NWP. Therefore, its practical benefits usually outweigh the theoretical drawback that the filter is rendered suboptimal as it does not solve the original assimilation problem anymore (Nerger et al. 2014). However, many other assumptions of the EnKF are also satisfied only partly in reality, particularly concerning the specification of observation and model error properties or the Gaussian assumption. Hence, any EnKF system inherently contains compromises and approximations in order to allow practical applicability (Anderson 2007b).

OL represents a more natural and general approach to localization as it only modifies the observation variables $\mathbf{y}$ and $\mathbf{R}$ that are the input to any analysis step. In contrast, CL is more specific to KF-based algorithms that compute the increments based on the forecast error covariance matrix. It is not directly applicable to variants such as the ETKF where $\mathbf{P}_f$ is never used explicitly. The relation between CL and OL has been examined in several studies (e.g., Greybush et al. 2011; Nerger et al. 2012a; Sakov and Bertino 2011), which found that in general the performances are similar, even though the optimal localization radius depends on the method. A more important premise for favorable results is an extensive tuning of the localization parameters. For this thesis, exclusively OL is used, as its generality allows a straightforward extension to the new nonlinear filter in Chapter 4. Additionally, OL naturally allows for parallelization of the analysis step.

Finally, it is noted that adaptive localization schemes can be applied to reduce manual tuning, which is an issue of current research. For example, Anderson (2007b) proposed an augmented hierarchical filter, Anderson (2012) employed a statistical model of forecast errors, Bishop and Hodyss (2009) applied ensemble correlations to obtain flow-dependent estimates of the localization functions, Nerger et al. (2012b) derived a regulated OL scheme and Kirchgessner et al. (2014) related ensemble size and localization radius.

3.4.3 Local ensemble transform Kalman filter

A first local formulation of the ETKF, based on a sliding-window technique similar to DL, was given by Ott et al. (2004). It was revisited by the seminal work of Hunt et al. (2007) who supplied a coherent framework for simple filter implementation, with additional focus on computational efficiency (Szunyogh et al. 2008). The local ETKF (LETKF) applies OL, and as of today, it serves as standard reference even for operational DA systems (e.g., Bonavita et al. 2010; Reich et al. 2011). Besides its importance in the EnKF world, a good understanding of its mechanism is beneficial here, as its localization methodology will be adapted for the new filter proposed in Chapter 4.

In order to summarize the LETKF analysis step algorithm, all global quantities carry the additional subscript $[g]$. The input is a global forecast ensemble, $\mathbf{X}_{f[g]}$, and a global observation vector, $\mathbf{y}_{f[g]}$.

1. Map the ensemble states into observation space using the full observation operator $\mathbf{y}^n_{f[g]} = \mathcal{H}(\mathbf{x}^n_{f[g]})$, and form $\mathbf{Y}_{f[g]}$.

2. For each local domain, perform the following procedure. Due to the independency of the local analyses, this can be done in parallel.

   (a) Obtain the local forecast ensemble $\mathbf{X}_f$ by selecting the rows of $\mathbf{X}_{f[g]}$ which correspond to the local domain. Compute the perturbations, $\mathbf{X}'_f$.

   (b) Extract the local observation $\mathbf{y}$ from the global observation vector.

   (c) Similarly, extract the same rows from $\mathbf{Y}_{f[g]}$ to form the local ensemble equivalents of the observation, $\mathbf{Y}_f$ and their perturbations, $\mathbf{Y}'_f$.

   (d) The same rows and columns are also selected from the global error covariance matrix $\mathbf{R}_{f[g]}$ to form its local counterpart, $\mathbf{R}$.

   (e) Its inverse can now be multiplied by a proper correlation function in order to decrease the weight of distant observations.[3]

   (f) Compute the analysis perturbations and mean as given by equations (3.18) and (3.19) and compose them to the local analysis ensemble, $\mathbf{X}_a = \mathbf{X}'_a + \overline{\mathbf{X}}_a$.

3. Combine all local analysis ensembles to obtain the global analysis ensemble $\mathbf{X}_{a[g]}$.

3.5 Advances in linear ensemble filtering

Before continuing with the presentation of nonlinear ensemble-based approaches, some advances for the EnKF are shortly discussed, in order to allow a better subsumption of the EnKF. This section focusses on a particular aspect, namely, the introduction of the ensemble technique into variational methods, which were presented in Chapter 2 as an alternative approach. This highlights the relevance of the Monte Carlo approach in DA. A full review of other efforts made for the EnKF is beyond the scope of this work. The most important developments from a practical point of view have already been shown in the previous two sections.

3.5.1 Comparison to 4DVAR

Since the EnKF was made applicable to large-scale NWP problems by Houtekamer and Mitchell (1998), the assessment of its relative performance with respect to 4DVAR is an active area of research. Lorenc (2003) presented a coherent discussion from a theoretical viewpoint, comparing their advantages and downsides. He concluded that the EnKF is attractive due to its relative ease of implementation as compared to 4DVAR, the explicit quantification of uncertainties by the ensembles and the opportunity to initialize ensemble predictions. However, he preferred 4DVAR in large-scale applications as the EnKF with a small ensemble is subject to sampling errors and requires covariance localization. Kalnay et al. (2007) refined the investigation and showed that recent developments can eliminate some disadvantages of the EnKFs, for example the consideration of asynchronous observations, which is more natural in 4DVAR. This was confirmed by a set of experiments in systems of different complexity, showing that both methods deliver comparable results, given sufficient tuning and experience. In the EnKF, this

[3] This step is given for clarity. It can be included more efficiently into the computation of $\mathbf{A}^{\mathrm{KF}}$ (Hunt et al. 2007).

mainly concerns localization and inflation while 4DVAR is sensitive to the length of the assimilation window and the specification of the background error covariance matrix. Both methods share the issue of a proper consideration of model error. Kalnay et al. (2007) concluded that even in the operational context the EnKF can be made competitive to established 4DVAR systems. These findings are confirmed by many studies focussing on empirical comparisons of both methods in different situations (e.g., Caya et al. 2005; Fairbairn et al. 2013; Miyoshi et al. 2010; Whitaker et al. 2009; Zhang et al. 2011), with the overall finding that the relative performance depends on system properties and implementation issues, and no method is superior to the other in general.

3.5.2 Hybrid approaches

Since both EnKF and 4DVAR have advantages and drawbacks and different performances are mainly due to implementation issues, it seems natural that a general improvement could be gained by combining them (Wang 2010). The basic idea is that ensembles potentially allow a better specification of the flow-dependent error structures needed in 3/4DVAR (Buehner 2005; Hamill and Snyder 2000; Lorenc 2003).

A typical hybrid approach to realize this idea is to feed a usual 4DVAR system with a background error covariance matrix $\mathbf{B}$ that contains both static and dynamical information, where the latter is extracted from ensemble covariances, as for instance estimated from an EnKF. Numerous variants of this concept have been suggested and are also used operationally (Buehner et al. 2013; Clayton et al. 2013; Wang et al. 2013). Another variant combines the gain matrices (Penny 2014). A quite common conclusion is that the resulting analyses are superior to that ones obtained from a stand-alone DA system.

A distinct method, used at ECMWF since 2011, concerns the perturbation of the observations, boundary data and physical parameterizations used for a deterministic 4DVAR system. This yields an *ensemble of DA* (EDA) that accounts for analysis error, which in turn is used for estimating flow-dependent background error variances, i.e., the diagonal of the matrix $\mathbf{B}$. An extension to the error correlations is planned, but represents a stronger challenge due to sampling errors, similarly as outlined for the EnKFs in section 3.4. For more details concerning this approach, see Bonavita et al. (2012). They also showed that EDA results in a significant increase of deterministic forecast skill and that it might be suitable to initialize ensemble predictions.

3.5.3 Discussion

The superiority of hybrid DA systems seems to be well confirmed during the past years. It is evident that they require a higher degree of maintenance and tuning, increasing the complexity of the DA system even further. For that reason, its focus are operational applications, where practical considerations have absolute priority. However, hybrid systems are not objective anymore, and from a theoretical point of view, it is unclear what problem they actually solve (Ades 2013). Therefore, primary research should focus on developing objective methods at first. In turn, this can contribute to a much broader range of possible practical applications. This conclusion also holds for the research performed in this work, concentrating on the analysis step, which might help to facilitate the usage of PF-based techniques in hybrid systems as well.

3.6 Particle filtering

Now, an alternative way of computing the analysis is presented, as compared to the EnKF. The PF (Gordon et al. 1993; Kitagawa 1996), constitutes an elegant Monte Carlo solution to the analysis step within a probabilistic framework (Wikle and Berliner 2007). A *particle* resembles an ensemble member and is also a sample from the underlying pdf. The distinction is mainly historical, as the EnKF was derived in the field of geophysical applications where ensemble prediction already was an established method, while the PF originated in sequential Monte Carlo methods (SMC), used in statistics and signal theory. The only new property is that particles may have varying weights within an ensemble.
First, the limitations of the EnKF are summarized in order to motivate the PF approach. Then, the PF is formulated as a variant of the general sequential filter within the probabilistic framework of Chapter 2 by deriving its forecast and analysis step.

3.6.1 Motivation: Limitations of the EnKF

Even though the EnKF and its variants are an appealing and successful improvement of the KF, they necessarily do not behave optimally in nonlinear environments due to multiple sources of error (Anderson 2007b). General errors common to all assimilation methods, as discussed in Chapter 2, are neglected in the following discussion.

- The EnKF partially respects the nonlinear nature of the system and its consequence on the analysis by feeding the KF update equation with more realistic prior moments, as they result from a fully nonlinear ensemble prediction. However, the update is still linear and implicitly based on a Gaussian assumption, which can be violated strongly in case of nonlinear models, as nonlinear dynamics produces non-Gaussian densities (Bocquet et al. 2010). From a different point of view, the EnKF analysis step can be formulated as a *linear regression* of the state on the observations (Anderson 2010).

- Further limitations are caused by a combination of the previous point with the necessity of using a rather limited ensemble size, particularly in high-dimensional systems, leading to sampling errors in the error covariances. In turn, this yields suboptimal analysis increments (Anderson 2012). This issue is usually counteracted by covariance inflation and localization techniques, see section 3.4.

- The observation error covariance structure is assumed to be Gaussian. The necessary approximation of non-Gaussian observation characteristics (Anderson 2010) can lead to significant deviations when estimating the observation impact (Fowler and van Leeuwen 2013).

The PF attempts to encounter all these limitations by fully relaxing the assumption of a Gaussian prior, the linear update and the restriction to Gaussian likelihoods. This leads to a fully nonlinear, non-Gaussian solution of the DA problem. The following presentation highlights why this is theoretically preferable, but also exhibits some severe practical problems.

3.6.2 Basic forecast step

The forecast step of the basic PF resembles its analog in the EnKF (van Leeuwen 2009). Here, the transition from $t_{i(j-1)}$ to $t_{i(j)}$ is considered and it is assumed that at time $t_{i(j-1)}$, the ensemble $\{\mathbf{x}^n_{i(j-1)}\}$ represents a potentially weighted sample of the corresponding analysis pdf $p(\mathbf{x}_{i(j-1)}|\mathbf{y}_{1:(j-1)})$ with weights $w^n_{i(j-1)}$, where the weights sum to 1.

All particles are integrated with the fully nonlinear model until observation time $t_{i(j)}$ by iterating $\mathbf{x}^n_i = \mathcal{M}_{i-1\to i}(\mathbf{x}^n_{i-1}) + \boldsymbol{\eta}^n_{i-1}$ for $i = (i(j-1)+1)\ldots i(j)$ for each particle. The resulting prior pdf can be written in an ensemble representation as a linear combination of delta functions,

$$p(\mathbf{x}_{i(j)}|\mathbf{y}_{1:(j-1)}) = \sum_n w^n_{i(j-1)}\delta(\mathbf{x} - \mathbf{x}^n_{i(j)}) \tag{3.22}$$

In the special case of an equally-weighted ensemble, this resembles equation (3.4) of the usual forecast ensemble pdf. During such a basic forecast step the weights are not modified, and these prior weights at time $t_{i(j)}$ are denoted as w^n_f. As before, the prior ensemble is denoted $\{\mathbf{x}^n_{i(j)}\} \to \{\mathbf{x}^n_f\}$. The EnKF concentrates on the prior mean and covariance, which are to be modified, implicitly assuming a Gaussian distribution. In contrast, the PF views the ensemble as a Monte Carlo representation of the unknown underlying pdf, and makes no parametric assumptions about its form.

3.6.3 Analysis step

The PF directly applies the general analysis step as in section 2.4.3 to assimilate the observation. As usual, the dependence on past observations $\mathbf{y}_{1:i(j-1)}$ is omitted in the notation, and furthermore, $\mathbf{x}_{i(j)}$ is denoted as $\mathbf{x}$ and $\mathbf{y}_j$ as $\mathbf{y}$. Using Bayes' theorem as in equation (2.19), the posteriori expectation of any function of $\mathbf{x}_{i(j)} \equiv \mathbf{x}$ can be written as

$$\langle f(\mathbf{x})\rangle_{p(\mathbf{x}|\mathbf{y})} \quad = \quad \int f(\mathbf{x})p(\mathbf{x}|\mathbf{y})d\mathbf{x} = \frac{\int f(\mathbf{x})p(\mathbf{y}|\mathbf{x})p(\mathbf{x})d\mathbf{x}}{\int p(\mathbf{y}|\mathbf{x})p(\mathbf{x})d\mathbf{x}} \tag{3.23}$$

Plugging in the ensemble representation of the prior ensemble, as given by equation (3.22), a Monte Carlo approximation of this expected value can be calculated:

$$\langle f(\mathbf{x})\rangle_{p(\mathbf{x}|\mathbf{y})} \approx \overline{f(\mathbf{x})} = \sum_{n=1}^{N} w^n_f \frac{p(\mathbf{y}|\mathbf{x}^n_f)}{\sum_{m=1}^N w^m_f p(\mathbf{y}|\mathbf{x}^m_f)} f(\mathbf{x}^n_f) \tag{3.24}$$

This equation can be regarded as a weighted mean, where each prior ensemble state is assigned a specific weight that incorporates the observational information,

$$\overline{f(\mathbf{x})} = \sum_{n=1}^{N} w^n f(\mathbf{x}^n_f) \quad \text{with} \quad w^n = w^n_f \frac{p(\mathbf{y}|\mathbf{x}^n_f)}{\sum_{m=1}^N w^m_f p(\mathbf{y}|\mathbf{x}^m_f)} \tag{3.25}$$

The analysis weight w^n is given by the product of the prior weight and the likelihood of the particle. The latter quantifies how likely the actual observation is, given the system is in the corresponding ensemble member's state $\mathbf{x}^n_f$. In other words, particles that are reasonably close to the observation receive

a large weight, and vice versa. As the analysis pdf has to be normalized, the discrete weights are subject to a normalization constraint, $\sum_n w^n = 1$. In praxis, the weights are computed from the observational pdf without the denominator in equation (3.25), yielding the relative likelihoods of each particle. Afterwards, they are normalized by dividing each weight by the total sum of the unnormalized weights. For example, in case of Gaussian observation errors, the weights are easily obtained by evaluating the multivariate normal density, neglecting its normalization factor (which is also expensive to evaluate),

$$w^n \propto w_f^n \cdot \exp\left\{-\frac{1}{2}\left(\mathbf{y}-\mathcal{H}(\mathbf{x}_f^n)\right)^T \mathbf{R}^{-1}\left(\mathbf{y}-\mathcal{H}(\mathbf{x}_f^n)\right)\right\} \tag{3.26}$$

This also shows that the weights are determined by the innovations of the ensemble states in observation space, as in the EnKF, but now, the relationship is nonlinear. However, any other type of observation error distribution may be used within a PF, while all KF-based schemes require Gaussian likelihood pdfs. It is emphasized that the PF makes absolutely no assumption about the form of the prior pdf, as it takes the ensemble in its pure form and assumes that it carries all accessible information about the prior pdf. This marks the central difference to the EnKF, where the update implicitly relies on a Gaussian assumption when dealing with the prior ensemble.

The PF is appealing because it performs a full Bayesian analysis without any additional assumptions; thus, it entirely considers the nonlinear nature of the system and the resulting non-Gaussian densities. For example, the analysis mean as best estimate of the state is easily computed by using $f(\mathbf{x}) = \mathbf{x}$ in equation (3.25), yielding $\overline{\mathbf{x}}_a = \sum_{n=1}^N w^n \mathbf{x}_f^n$. In a similar manner, it is possible to derive any other statistic of interest using the Monte Carlo estimator (see Appendix A). The basic PF analysis step is completed by assigning the posteriori weight $w^n \rightarrow w_{i(j)}^n$. Notably, the ensemble members are not changed at all, $\mathbf{x}_{i(j)}^n = \mathbf{x}_f^n$. Only their relative weights are modified, and hence, they contain all the information extracted from the observation.

3.6.4 Curse of dimensionality

In the basic PF, the ensemble states are not changed in the analysis step, only their weights are adjusted. In sequential DA, the PF is iterated over multiple cycles. At each analysis step, the new weights are obtained by multiplying the previous weights by the new likelihoods. This has the advantage of maintaining any dynamical balances in the states, but it also leads to so-called *filter divergence*, or *filter degeneracy*, *the* primary challenge in the application of PFs (van Leeuwen 2009). The particle closest to the observation receives a larger relative weight, while all others loose relative weight. Consequently, after some analysis steps, one particle exhibits a weight close to 1, while all other weights become negligible, if the ensemble size is not large enough. Then, the ensemble does not contain any useful information anymore and the filter has collapsed. This issue, numerically diagnosed by an increasing variance of the weights, becomes particularly problematic in larger-dimensional spaces with many independent observations (Bengtsson et al. 2008; Snyder et al. 2008). However, even in low-dimensional systems a large number of particles may be required to suppress the probability of filter divergence.

An early idea to solve this issue is to augment the analysis step by a *resampling* step to get rid of particles with low weights, as they are too far away from the observation and are likely to diverge even more in the next analysis step. Thus, high-weight particles are duplicated as they are more valuable.

Specifically, one resamples N particles from the set $\{\mathbf{x}^n_{i(j)}\}$ with the associated discrete probability distribution $\{w^n_{i(j)}\}$. Thus, after resampling, the ensemble is equally-weighted again. In this form, the PF is also known as sequential importance resampling (SIR) filter. Numerous algorithms to carry out the resampling procedure have been suggested in the literature, leading to different SIR flavors (van Leeuwen 2009). It is also possible to apply resampling conditionally. For example, Hoteit et al. (2012) monitor whether the entropy, a measure of the variance of the weights, exceeds a predefined threshold.

Even though resampling appears as reasonable solution, it does not prevent filter collapse in general. The likelihood pdf only covers a small region of the observation space. For illustration, M independent Gaussian observations with standard deviations σ are considered, i.e., in equation (3.26) $\mathbf{R}$ is diagonal. Then, the volume of the hypersphere within three standard deviations of the means is approximately proportional to $(\sigma/M)^{M/2}$ (Ades 2013, p. 34). Thus, the region of significant probability mass in observation space tends to zero with increasing dimensionality, regardless of the broadness of each individual observation. Additionally, the highly-peaked likelihood pdf is multiplied by the prior pdf, which is usually flatter. This leads to an even stronger peaked posteriori pdf. Furthermore, resampling does not help in deterministic systems where the duplicated particles remain identical throughout the whole forecast phase. In this case, resampling directly decreases the effective ensemble size, while model errors at least allow duplicated particles to diffuse during the forecast phase.

As this thesis aims at deriving a new PF-based algorithm, it is important to analyze the reasons of weight collapse in a structured way. The ensemble representation of the posterior pdf is determined by the position and the weights of each particle, and the two claims of an ideal particle representation that is stable with respect to filter degeneracy are (Ades 2013):

1. The positions of the particles should all be samples from the high-probability region, as defined by the likelihood pdf, i.e., they should be close to the observation.
2. The relative weights should be similar such that all particles contribute significant information about the pdf.

Resampling only attempts to ensure the second requirement, by reducing the variance of the weights. However, filter collapse cannot be avoided particularly in higher-dimensional spaces, as it is still likely that the majority of particles does not cover the significant area in observation space at the next analysis step. Snyder et al. (2008) found that the number of particles required to avoid filter divergency strongly exceeds a critical number which grows exponentially with the square of an effective dimension of the system. This is known as the "*curse of dimensionality*" in the context of PFs. For example, they impressively estimated that in a 100- or 200-dimensional space the required ensemble size would be about 10^6 or 10^{11}, respectively. In the following, recent attempts to overcome the curse of dimensionality by improving the likelihoods of the particles are discussed.

3.6.5 Revision of the forecast step

As revealed, the principal problem is that many particles achieve very low relative weight in the analysis step as they are too distant from the observation. A promising approach consists in changing the

forecast step such that more particles end up close to the observations. Formally, this is realized by the introduction of so-called proposal densities, a standard tool in the mathematical literature (Doucet et al. 2001). This extension will be important to understand the EWPF in the next section.

First, a single transition $t_{i-1} \rightarrow t_i$ is considered. Here, the dependence of the state on past observations $\mathbf{y}_{1:i(j-1)}$ is omitted because it is irrelevant for the next forecast and analysis steps. The key idea is to introduce a proposal density $q(\cdot)$ into equation (3.1):

$$p(\mathbf{x}_i) = \int p(\mathbf{x}_{i-1}) \frac{p(\mathbf{x}_i|\mathbf{x}_{i-1})}{q(\mathbf{x}_i|\mathbf{x}_{i-1}, \mathbf{y}_j)} q(\mathbf{x}_i|\mathbf{x}_{i-1}, \mathbf{y}_j) d\mathbf{x}_{i-1} \tag{3.27}$$

Even though this appears to be a trivial reformulation, it offers many new opportunities if the integral is interpreted in an importance sampling fashion, a standard Monte Carlo technique (Robert and Casella 2004). Applying the Monte Carlo representation of the prior ensemble as given in (3.22) yields

$$p(\mathbf{x}_i) \approx \sum_n w_{i-1}^n \frac{p(\mathbf{x}_i|\mathbf{x}_{i-1}^n)}{q(\mathbf{x}_i|\mathbf{x}_{i-1}^n, \mathbf{y}_j)} q(\mathbf{x}_i|\mathbf{x}_{i-1}^n, \mathbf{y}_j) \tag{3.28}$$

Instead of approximating this pdf by drawing one sample from each transition density $p(\mathbf{x}_i|\mathbf{x}_{i-1}^n)$ as done in section 3.1.2, a sample $\mathbf{x}_i^n$ is drawn from $q(\mathbf{x}_i|\mathbf{x}_{i-1}^n, \mathbf{y}_j)$ such that the new ensemble is distributed as $\mathbf{x}_i^n \sim q(\mathbf{x}_i|\mathbf{x}_{i-1}, \mathbf{y}^j)$. This results in the following representation of the new prior pdf (van Leeuwen 2010)

$$p(\mathbf{x}_i) \approx \sum_n w_i^n \delta(\mathbf{x}_i - \mathbf{x}_i^n) \quad \text{where} \quad w_i^n = w_{i-1}^n \frac{p(\mathbf{x}_i^n|\mathbf{x}_{i-1}^n)}{q(\mathbf{x}_i^n|\mathbf{x}_{i-1}^n, \mathbf{y}_j)} \tag{3.29}$$

Note that the factor $p(\cdot)/q(\cdot)$ in the weight equation automatically corrects for the fact that the particles are not directly drawn from the transition density. Thus, the ensemble still represents a valid sample of the state pdf. The nominator can be computed through knowledge of the model error density. An appealing property is the huge freedom in choosing $q(\cdot)$, as the new ensemble could be sampled from an arbitrary distribution. However, this would result in very poor likelihood weights in the analysis step, and it should be ensured that $q(\cdot)$ has the same support as the transition density in oder to avoid division by zero (Doucet et al. 2001). Hence, it is reasonable to choose it similar to the transition pdf. Additionally, $q(\cdot)$ should be conditioned on the next observation $\mathbf{y}_j$, as already indicated in the equations shown above. This allows to ensure that more members of the new ensemble lie in the high probability region of the posterior, thus avoiding weight collapse by a better positioning of the particles. However, it should be noted that the variance of the weights still increases at each time step. Therefore, the next section deals with further advances in this direction.

3.7 Advances in nonlinear filtering

Since its first introduction by Gordon et al. (1993), the PF has received a large amount of attention for applications dealing with state and parameter estimation. This section deals with important advances made in this field, concentrating on nonlinear geophysical applications. The PF offers an appealing alternative to KF-based schemes by relaxing the assumption of Gaussian densities, as these systems usually imply non-Gaussian distributions. However, as discussed, the basic PF requires an ensemble

size that is prohibitive for typical large-scale models. The mathematical literature contains much more approaches (Aster et al. 2005; Doucet et al. 2001), but most of them are restricted to small-dimensional problems. A thorough review concerning the current status about PFs in geophysical systems at that time was provided by van Leeuwen (2009).

It is beyond the scope of this work to discuss all developments. Therefore, the review here concentrates on aspects relevant for the aims of this thesis. First, the EWPF as a promising formulation of the PF is presented, which has mainly been developed during time research for this thesis was performed. Second, the NLEAF, a reduced-order PF, is discussed. Third, other interesting efforts are mentioned briefly. The conclusions drawn here not only motivate the NETF to be derived in Chapter 4, but also allow a better classification of its mechanism, particularly concerning its properties and potential advantages.

3.7.1 Equivalent weight particle filter

The development of the EWPF has been induced by the comprehensive review by van Leeuwen (2009). The main idea consists in employing suitable proposal densities during the forecast step to guide the particles such that they remain close to the observation. Shortly after, van Leeuwen (2010) extended the concept by using a distinct proposal density before assimilating the observation. It adjusts almost all particles such that they have equivalent weights in the analysis step. The filter behaved efficiently in a 1000-dimensional Lorenz system. A thorough exploration of the filter properties was given by Ades and van Leeuwen (2013), and van Leeuwen and Ades (2013) reported a successful application in a strongly nonlinear system of larger dimensionality. Here, a concise description of the EWPF is given, focussing on the principal mechanism. The algorithm exhibits numerous technical questions and implementation details, which are covered in detail by Ades (2013). There are two reasons for discussing the EWPF in the context of this thesis. First, it allows a better classification of the new NETF in Chapter 4 concerning algorithm complexity. Second, the NETF can be combined with the forecast step of the EWPF, which may allow a comparison of both analysis mechanisms in future.

3.7.1.1 Forecast step: nudging

In section 3.6.5, the formulation of the forecast step including a proposal density was introduced. Typically, there are numerous of single-step forecasts in between two successive observations. Therefore, the repeated application of the procedure outlined by equation (3.29) leads to a prior pdf at time $t_{i(j)}$ at which each member exhibits the prior weight

$$\tilde{w}^n_{i(j)} \propto w^n_{i(j-1)} \prod_{i=i(j-1)+1}^{i(j)} \frac{p(\mathbf{x}^n_i|\mathbf{x}^n_{i-1})}{q(\mathbf{x}^n_i|\mathbf{x}^n_{i-1},\mathbf{y}_j)} \tag{3.30}$$

Then, according to equation (3.25), the analysis weights are simply computed by multiplying the prior weights with the likelihood weights, $w^n_{i(j)} \propto \tilde{w}^n_{i(j)} p(\mathbf{y}_j|\mathbf{x}^n_i)$. As motivated in section 3.6.5, the proposal pdf should be chosen similar to the transition pdf. In order to improve their *future* likelihoods, the particles can be pulled closer to the observation by generating the new ensemble as following (van

Leeuwen 2010),

$$\mathbf{x}_i^n = \mathcal{M}_{i-1\to i}(\mathbf{x}_{i-1}^n) + \tilde{\boldsymbol{\eta}}_{i-1}^n + \mathbf{N}_{i-1}(\mathbf{y}_j - \mathcal{H}(\mathbf{x}_{i-1}^n)) \tag{3.31}$$

Compared with the usual model integration, equation (2.4), a different error distribution to generate $\tilde{\boldsymbol{\eta}}_{i-1}$ can be used, for instance $\mathcal{N}(\mathbf{0}, \tilde{\mathbf{Q}})$. The last term nudges the particle towards the observation, using a weight matrix $\mathbf{N}$. The nudging term, which can be chosen freely, is supposed to be relatively small because it is applied at each time step. Furthermore, other options for choosing the proposal density might also lead to more sophisticated approaches. Using the three-dimensional Lorenz system, van Leeuwen (2010) demonstrated that such a Gaussian relaxation density enables stable PF runs with very few particles.

Nevertheless, nudging alone is not sufficient to avoid filter divergency, particularly in larger-dimensional systems. The reason is that usually $p(\mathbf{x}_i^n|\mathbf{x}_{i-1}^n)$ is smaller than $q(\mathbf{x}_i^n|\mathbf{x}_{i-1}^n, \mathbf{y}_j)$ because the particle is not generated by the actual system dynamics anymore. Thus, the accumulated weight, given by the product in equation (3.30), becomes very small, and the variance of the weights increases.

3.7.1.2 Analysis step: equivalent weights

The relaxation density ensures that the particles are positioned in the high probability region in observation space. However, as explained, the relative weights may still vary strongly. This is resolved by modifying the proposal density in the last step before the observation, i.e., $t_{i(j)-1} \to t_{i(j)}$. Separating this density from equation (3.30) and including the likelihood weights yields the final analysis weights at $t_{i(j)}$ as

$$w_{i(j)}^n \propto p(\mathbf{y}_j|\mathbf{x}_i^n) \frac{p(\mathbf{x}_{i(j)}^n|\mathbf{x}_{i(j)-1}^n)}{q(\mathbf{x}_{i(j)}^n|\mathbf{x}_{i(j)-1}^n, \mathbf{y}_j)} \underbrace{\prod_{i=i(j-1)+1}^{i(j)-1} \frac{p(\mathbf{x}_i^n|\mathbf{x}_{i-1}^n)}{q(\mathbf{x}_i^n|\mathbf{x}_{i-1}^n, \mathbf{y}_j)}}_{\propto w_{i(j)-1}^n} \tag{3.32}$$

Now, the final proposal density $q(\mathbf{x}_{i(j)}|\mathbf{x}_{i(j)-1}^n, \mathbf{y}_j)$ is chosen such that the nominator is constant. Assuming Gaussian densities for the transition and observation density, this corresponds to solving the following equation for $\mathbf{x}_i^n$ (van Leeuwen 2010, Eq. 19)

$$\begin{aligned} -\ln(w_{i(j)-1}^n) + \frac{1}{2}\left[(\mathbf{x}_{i(j)}^n - \mathcal{M}(\mathbf{x}_{i(j)-1}^n))^T \mathbf{Q}^{-1}(\mathbf{x}_{i(j)}^n - \mathcal{M}(\mathbf{x}_{i(j)-1}^n))\right] + \\ \frac{1}{2}\left[(\mathbf{y}_j - \mathcal{H}(\mathbf{x}_{i(j)}^n))^T \mathbf{R}_j^{-1}(\mathbf{y}_j - \mathcal{H}(\mathbf{x}_{i(j)}^n))\right] = -\ln w^{target} \end{aligned}$$

where $\mathcal{M}$ shortly represents $\mathcal{M}_{(i(j)-1)\to i(j)}$. The fixed target weight should be as large as possible, and it is set such this equation is solvable for a majority (80%) of the particles by examining their maximum achievable weights (van Leeuwen 2010). The other 20% are typically too far from the observations to be relevant anymore. In order to regain a full ensemble, the $0.2N$ missing particles re-enter through resampling the surviving ensemble. For the 80% particles, an infinite number of solutions for this equation exists, and the following ansatz is used (van Leeuwen 2010), similar to the nudging step

(3.31):

$$\mathbf{x}^n_{i(j)} = \mathcal{M}(\mathbf{x}^n_{i(j)-1}) + \alpha^n_j \mathbf{N}_j \big(\mathbf{y}_j - \mathcal{H}(\mathcal{M}(\mathbf{x}^n_{i(j)-1}))\big) + \tilde{\boldsymbol{\eta}}^n_{i(j)-1} \tag{3.33}$$

where $\mathbf{N}_j = \mathbf{Q}\mathbf{H}_j^T(\mathbf{H}_j\mathbf{Q}\mathbf{H}_j^T + \mathbf{R}_j)^{-1}$. This adjustment resembles an EnKF analysis step, but α^n_j controls the position of the posterior particles such that they receive equal weight, and an explicit solution can be found (Ades and van Leeuwen 2013, Eqs. 28-32). The term $\tilde{\boldsymbol{\eta}}^n_{i(j)-1}$ represents a random deviation from the deterministic solution, but it is essential as otherwise $q(\mathbf{x}_{i(j)}|\mathbf{x}^n_{i(j)-1}, \mathbf{y}_j)$ would refer to a delta function and lead to division by zero in equation (3.32). Hence, a small stochastic term must be kept. A uniform density around the deterministic move would guarantee $q(\mathbf{x}^n_{i(j)}|\mathbf{x}^n_{i(j)-1}, \mathbf{y}_j)$ to be constant in equation (3.32), keeping the weights equal, but it still has finite support. The currently best solution is to enhance it by Gaussian tails, using a strongly asymmetric mixture density (Ades and van Leeuwen 2013, Eq. 36). Then, the final analysis weights (3.32) are *almost* equal.

3.7.1.3 Discussion

The EWPF employs an appealing PF variant and in principle allows to explore the full posteriori pdf with a relatively small amount of particles by guiding them through the proposal densities. This was confirmed in Lorenz systems and with a barotropic vorticity equation with state dimensionality of about 65000 (van Leeuwen and Ades 2013). These promising results should encourage further research about the EWPF. The EWPF is more successful than other PF variants because both reasons for filter collapse (see section 3.6.4) are considered. Nudging during the forecast phase ensures that the particles end up in the high-probability region in observation space, and the analysis step ensures that the variance of weights tends to zero. However, the explicit implementation has to be adapted carefully to the model at hand (Ades 2013), particularly concerning the relaxation density, as the results are sensitive to the choice of the weight matrix $\mathbf{N}$ (Ades and van Leeuwen 2013). As the support of $q(\cdot)$ has to be at least equal to the support of the transition density, this implies that a deterministic model cannot be used and that the choice of $q(\cdot)$ is strongly connected to the model error term. For instance, a small model error only allows a small nudging of the particles. A large model error increases the freedom in $q(\cdot)$, but also increases the noise in the forecast states. Hence, a good balance between both distributions has to be found, which is a complex task, particularly for real-world models where even the model error itself is a highly uncertain quantity. As a consequence, a direct application of the EWPF requires non-trivial tuning to be successful and it may be an option to develop alternatives to the relaxation density (3.31) (P. Kirchgessner, AWI Bremerhaven, personal communication). In conclusion, the EWPF currently is at an appealing stage but requires an extensive manipulation of the forecast step in a sequential DA system. The filter developed in this work follows a different paradigm. It only adjusts the analysis step, but still attempts to ensure good positions *and* equal weights for all particles.

3.7.2 Nonlinear ensemble adjustment filter

Another interesting development in the field of PF research is the update algorithm suggested by Lei and Bickel (2011). Their basic idea motivates the novel filter algorithm derived in this work. They analyzed the EnKF from a statistical point of view and argued that in a nonlinear, non-Gaussian setting the EnKF

analysis is necessarily biased because of the linear ensemble update. This bias particularly concerns the first moment, which leads to an additional error in location, and the second moment, leading to a mis-estimation of the ensemble's shape, which is expressed by the variances and covariances. As a result from their statistical investigation, Lei and Bickel (2011) proposed the NLEAF which aims at updating the ensemble such that biases occurring in an EnKF update are eliminated. Thus, the paradigm of their filter is the requirement that the analysis ensemble should have, on average, a mean and potentially a covariance which matches the prediction of the non-parametric Bayes' theorem. It applies a stochastic update of the ensemble, similar to the classical EnKF. The NLEAF is a sequential filter that fits in the general framework of Chapter 2 and hence, both its forecast and analysis step are discussed here.

3.7.2.1 Forecast step

The forecast step employs the usual ensemble integration as described in section 3.1.2. As will be shown next, the output of the analysis step, which is the input to the forecast step, is an equally-weighted ensemble. Thus, the forecast steps of NLEAF and EnKF are identical.

3.7.2.2 Ambiguity of "perturbed observations"

For the classic EnKF, the usage of perturbed observations was motivated in section 3.2. Usually, N samples $\{\mathbf{y}^n\}$ are drawn from the Gaussian density $\mathcal{N}(\mathbf{y};\mathbf{R})$. Equivalently, one may generate N samples $\{\mathbf{e}^n\}$ from the observation error distribution, $\mathcal{N}(\mathbf{0};\mathbf{R})$, and compute the perturbed observations as $\mathbf{y}^n = \mathbf{y} + \mathbf{e}^n$. Another valid option is to leave the actual observation $\mathbf{y}$ unchanged and to add the sampled error to the forecast state in observation space instead, $\tilde{\mathbf{y}}^n = \mathcal{H}(\mathbf{x}_f^n) + \mathbf{e}^n$. These alternatives lead to differently structured EnKF update equations,

$$\text{Variant (1):} \quad \mathbf{x}_a^n = \mathbf{x}_f^n + \mathbf{K}(\mathbf{y} + \mathbf{e}^n - \mathcal{H}(\mathbf{x}_f^n)) = \mathbf{x}_f^n + \mathbf{K}\left(\mathbf{y}^n - \mathcal{H}(\mathbf{x}_f^n)\right) \tag{3.34}$$

$$\text{Variant (2):} \quad \mathbf{x}_a^n = \mathbf{x}_f^n + \mathbf{K}(\mathbf{y} - \mathcal{H}(\mathbf{x}_f^n) + \mathbf{e}^n) = \mathbf{x}_f^n + \mathbf{K}(\mathbf{y} - \tilde{\mathbf{y}}^n) \tag{3.35}$$

Even though both equations are identical, their formal difference can be an important issue for practical implementations. When using equation (3.34), the actual observation is replaced by the perturbed observation $\mathbf{y}^n$, while in equation (3.35), the background state is replaced by the perturbed observation $\tilde{\mathbf{y}}^n$. This presentation reveals that the term *perturbed observation* is ambiguous, as of course $\mathbf{y}^n \neq \tilde{\mathbf{y}}^n$. For instance, using the first variant with $\tilde{\mathbf{y}}^n$ leads to an entirely wrong analysis[4], and vice versa. Hence, attention has to be paid to the correct choice of the "perturbed observation" according to the actual implementation.

This potential danger becomes evident in the NLEAF, as here perturbed observations are defined as $\tilde{\mathbf{y}}^n = \mathcal{H}(\mathbf{x}_f^n) + \mathbf{e}^n$ (Lei and Bickel 2011), but in the innovation vector $\mathbf{i}$ they actually have to replace $\mathbf{y}$, leading to a mix-up of the variants shown above, where $\mathbf{i} = \tilde{\mathbf{y}}^n - \mathcal{H}(\mathbf{x}_f^m)$ ($m \in \{1,\ldots,N\}$, see below). Even though the ambiguity of "perturbed observations" revealed here is a rather trivial issue, if unrecognized, it can quickly lead to unsuccessful implementations. It might also explain why Frei

[4]Note that in this case the innovation is simply given by $\mathbf{e}^n$, hence, it is not related to either observation or forecast state at all.

and Künsch (2013a) state that they were not able to achieve any stable runs with the NLEAF despite elaborate efforts, in contrast to the simulations shown in this work in Chapter 5.

3.7.2.3 Analysis step formulations

The NLEAF offers two options for adjusting the forecast ensemble. The first-order variant (NLEAF1) aims at an unbiased analysis ensemble whose mean approximates the Bayesian expectation. The second-order variant (NLEAF2) additionally considers the Bayesian covariance. Both variants can be described by a synthesized algorithm:

1. Create N perturbed observations as discussed in the previous section via $\tilde{\mathbf{y}}^n = \mathcal{H}(\mathbf{x}_f^n) + \mathbf{e}^n$, where each $\mathbf{e}^n$ is a sample from the observation error distribution, e.g., $\mathcal{N}(\mathbf{0}, \mathbf{R})$.
2. Given the forecast ensemble $\{\mathbf{x}_f^n\}$, evaluate the PF mean and covariance using equation (3.25), ...
   (a) ... with $\mathbf{y}$ being the actual observation, i.e., with weights $w^n \propto p(\mathbf{y}|\mathbf{x}_f^n)$. This yields the actual analysis mean $\boldsymbol{\mu}_a$ and covariance $\mathbf{P}_a$.
   (b) ... replacing $\mathbf{y}$ with a perturbed observation, $\tilde{\mathbf{y}}^n$, for $n = 1 \dots N$. For each fixed n, the ensemble weights are $w^{m,n} \propto p(\tilde{\mathbf{y}}^n|\mathbf{x}_f^m)$ $(m = 1 \dots N)$. This yields, for each member, the conditional moments $\boldsymbol{\mu}_a^n$ and $\mathbf{P}_a^n$.
3. Adjust the forecast ensemble members to obtain the analysis members:

$$\text{for NLEAF1:} \quad \mathbf{x}_a^n = \mathbf{x}_f^n + \boldsymbol{\mu}_a - \boldsymbol{\mu}_a^n \tag{3.36}$$

$$\text{for NLEAF2:} \quad \mathbf{x}_a^n = \boldsymbol{\mu}_a + \mathbf{P}_a^{1/2}(\mathbf{P}_a^n)^{-1/2}\left(\mathbf{x}_f^n - \boldsymbol{\mu}_a^n\right) \tag{3.37}$$

It is evident that the ensemble generated by equation (3.36) has a mean of $\boldsymbol{\mu}_a$ apart from sampling fluctuations. The addition of $\boldsymbol{\mu}_a^n$ guarantees that the members are distributed in an equally-likely region around the mean. The second-order update in equation (3.37) relies on transforming the ensemble covariance such that it converges to $\mathbf{P}_a$. In both cases, it is apparent that low-weight particles, which would be lost in a PF step, are pulled towards the analysis mean in order to retain an equally-weighted ensemble. For actual proofs concerning the asymptotic behavior of the filter, the reader is referred to the technical description given by Lei and Bickel (2009).

3.7.2.4 Discussion

The NLEAF represents an interesting alternative within the large variety of PF-based filters due to its relatively simple design. It only affects the analysis step of a sequential, ensemble-based DA system. This can be a major advantage concerning practical applications, in contrast to other suggestions which often require substantial, complex modifications of the forecast step, such as the EWPF. The adjustments applied to the prior ensemble offer the advantage of generating a new, equally-weighted ensemble. Thus, no resampling or similar techniques are required, which in principal helps to maintain filter stability as compared to the PF.

However, this comes at the price that the NLEAF does not, in contrast to full PFs, employ the full posteriori pdf but only its first- or second-order statistics. Nevertheless, the results from experiments with Lorenz models presented by Lei and Bickel (2011) are quite promising. They show that such rather simple modification of the analysis step is able to generate significantly better analyses than both EnKFs and other PF-based techniques. However, they mainly use relatively large ensemble sizes with observations available every time step, which is not an image of typical real-world situations. As already mentioned by Lei and Bickel (2011), the stochastic update mechanism also introduces sampling errors that can degrade the performance, particularly for smaller ensemble sizes. This will be explicitly demonstrated in Chapter 5. Furthermore, the NLEAF exhibits high computational cost as $\mathcal{O}(N^2)$ weights have to be computed, and the NLEAF2 even requires to evaluate and multiply $\mathcal{O}(N)$ matrix square roots in state space. It can be concluded that the NLEAF represents a reasonable compromise between KF-based and full particle filtering. However, its performance has to be further investigated and there are some drawbacks which can be tackled by employing a distinct update methodology. This will be within the scope of the new algorithm, presented in Chapter 4.

3.7.3 Other efforts

Finally, other efforts in nonlinear filtering are briefly mentioned to emphasize the broad research spectrum built around the PF. However, a complete review is beyond the scope of this work.

3.7.3.1 Markov chains

An example for a more recent development in this context concerns the combination of the PF with Markov chain Monte Carlo (MCMC) methods by the seminal work of Andrieu et al. (2010). MCMC originates in statistical physics (Metropolis et al. 1953) and is a popular method to explore the high probability region of a pdf by constructing a suitable Markov chain that converges to the pdf (Hastings 1970; Tierney 1994). However, is it unclear how to apply it to the dynamical DA problem in a feasible fashion (Ades 2013). Posselt and Bishop (2012) compared MCMC and the ensemble transform Kalman smoother (ETKS) in an idealized column model of deep convection, aiming at the estimation of ten fixed parameters. They showed that MCMC outperforms the ETKS as it aims at the true posterior density and is not limited to linear updates. However, it required $\mathcal{O}(10^5)$ MCMC iterations to achieve robust sampling from the true pdf in the ten-dimensional phase space, given a reasonable initial range for each variable.

3.7.3.2 Implicit PF

As the EWPF, the implicit PF (IPF, Chorin et al. 2010) aims at an importance sampling method such that the particles end up in the high probability region of the next posterior pdf. This is equivalent to using the posterior density as proposal density (Bocquet et al. 2010). While the EWPF relies on nudging, the IPF basically applies a minimization procedure as in 4DVAR to each particle to find the optimal state value (Atkins et al. 2013). Therefore, it is a smoother by construction. Even though it has shown strong benefits over the SIR filter in Lorenz models and low-dimensional geophysical

applications (Morzfeld and Chorin 2012), it is still subject to filter degeneracy in large-dimensional system with many observations as the relative weights may exhibit a high variance (Ades 2013).

3.7.3.3 Gaussian mixture filters

Another popular approach, initiated by Anderson and Anderson (1999) and Pham (2001), is the representation of the forecast ensemble as a Gaussian mixture, allowing a better consideration of non-Gaussian features contained in the prior ensemble (e.g., Frei and Künsch 2013b; Hoteit et al. 2012, 2008; Stordal et al. 2011). Typically, the particles are updated by a KF analysis step, together with an additional assignment of weights resulting from the analytical solution of Bayes' theorem. Thus, resampling is required as well. These filters can be interpreted as a bridge between the EnKF and PF (Frei and Künsch 2013a). They have been shown to offer potential benefits in nonlinear scenarios, where the relative performance depends on the careful balancing of the EnKF and PF contributions. The implementation of Hoteit et al. (2008) was applied successfully in an ocean general circulation model. However, as hybrid filters they do not constitute objective techniques.

3.7.3.4 Forecast step adjustments

Apart from the EWPF, other PFs also explore the freedom given by the proposal density. For example, Papadakis et al. (2010) used the EnKF as proposal, while other variants rely on repetitions of the forecast step or the introduction of pseudo-observations in the forecast step in order to improve the prior likelihoods (e.g., Pitt and Shephard 1999). However, none of them could prove its applicability in high-dimensional systems.

3.7.3.5 Approximations

Other methods abandon the aim of a full Bayesian solution and approximate the PF in different ways, as also the NLEAF. It is argued by van Leeuwen (2009) that localization might be helpful for PFs as it reduces the effective dimensionality of the system, thereby counteracting filter collapse. However, he stated that is unclear how to deal with the space-dependent weights that arise from a localized analysis. In Chapter 4, a natural answer to this question can by given.

It is interesting to look at two proposals in more detail. In the merging PF (MPF, Nakano et al. 2007), each analysis state is replaced by a tailored linear combination of some states drawn from a large set of resampled analysis members such that their mean and covariance are asymptotically preserved. Thus, it basically represents a less objective, rather manual version of the NLEAF. A similar, more objective approach is given by PF with Gaussian resampling (PFGR, Xiong et al. 2006) which samples the analysis states from a Gaussian distribution whose mean and covariance are determined by the Bayesian expectations. However, both filters have only been applied to systems with rather strong model error and quite large ensemble sizes (e.g., $N > 100$ in the Lorenz 63 model). Lei and Bickel (2011) found that they diverge in experiments with a deterministic Lorenz 96 model of intermediate dimensionality, while the NLEAF performed better. For this reason, the new filter derived in Chapter

4 will be compared with the NLEAF as it appears to be the only filter from this family that offers a potential applicability in higher-dimensional, deterministic settings.

3.8 Summary and conclusions

In this chapter, the ensemble forecasting technique, a Monte Carlo approach to the dynamical evolution of the state's pdf, was discussed in context of the forecast step of a sequential DA system. The combination with the KF directly leads to the EnKF as an revolutionary milestone. Different variants of the EnKF were presented, focussing on a deterministic implementation of the analysis step, and practical issues and findings that have been arisen from applying the EnKF since its introduction were discussed. The EnKFs rely on Gaussian distributions and perform a linear update. In contrast, the PF relaxes these restrictions by directly solving the general analysis step, as given by Bayes' theorem. Thus, it fully respects the nonlinear, non-Gaussian nature of the system. However, it cannot be applied in a straightforward fashion due to the curse of dimensionality, leading to filter divergency unless the ensemble size is extremely high. Recent developments that attempt to tackle this problem were summarized. Many PF variants or approximations have been suggested within the last years, yet the applicability in high dimensions represents a challenge difficult to overcome. At the current stage, it appears that only the EWPF is capable of actually solving the curse of dimensionality, at least in stochastic systems. The NLEAF follows an interesting paradigm that is also applicable to deterministic systems.

The main conclusions gained in this chapter are:

- The EnKF separates the forward model and analysis step without resorting to tangent linear or adjoint models. The fully nonlinear nature of the forecast step using an ensemble integration improves the prior estimates and provides a reasonable low-rank approximation of the covariance matrix. The deterministic variants are appealing as they avoid the perturbation of observations. In order be competitive to variational schemes in higher-dimensional systems, the EnKF needs to be enhanced by inflation and localization.
- The PF represents an appealing alternative by fully relaxing the Gaussian assumption, but is exposed to filter degeneracy due to the curse of dimensionality. Thus, a nonlinear DA technique has to guarantee that the particles (1) end up close to the observations at analysis time and (2) additionally exhibit relatively equal weights.
- The exploration of proposal densities within the forecast step leads to sophisticated algorithms, in particular the EWPF. However, the algorithmic complexity is increased in comparison to EnKFs. Its forecast step needs to be adapted to the model, and the latter has to be stochastic.
- The NLEAF does not rely on stochastic models, however, its stochastic update mechanism exhibits high computational costs and negatively affects the performance for small ensemble sizes. Both issues are highly relevant concerning possible practical applications.

The insights gained here will be used in the next chapter to derive the new filter algorithm. It attempts to tackle some of these issues to render it applicable in high dimensions as well.

Chapter 4

Nonlinear ensemble transform filter: Theory

The previous chapter showed how ensemble-based techniques, as offered by the EnKFs and PFs, attempt to consider non-Gaussian features in the forecast pdf. An overview of the current state of research with respect to nonlinear DA techniques was given and recent developments were discussed. These considerations constitute the motivation for further advancing these DA techniques. This chapter summarizes the motivation and then derives a new nonlinear filter. The focus is on a thorough demonstration of the mathematical details and a discussion of the filter properties from a theoretical point of view.

4.1 Prerequistes

4.1.1 Motivation

For the purpose of deriving a new filter algorithm, we take the perspective that the PF, as introduced in section 3.6, principally offers the most natural approach to solve the DA problem, as it applies Bayes' theorem in a direct way, and does not require any assumptions about the prior pdf. Furthermore, the likelihood pdf is not restricted to be Gaussian. However, a solution needs to be found to overcome the curse of dimensionality, which leads to weight collapse if the ensemble size is not extremely large. The EWPF applies proposal densities during the forecast step to position the particle in a desirable way. Here, we follow a more generic paradigm. In order to simplify future implementations, the new filter is supposed to be independent on the forecast step, which should remain a standard ensemble integration. As approximation to fully-nonlinear filtering, we pursue the idea of the NLEAF and do not focus on the full analysis pdf but only on its first two moments. Additionally, an attempt is made to overcome its shortcomings in order to enable large-scale applicability.

4.1.2 Requirements

The following list summarizes the desired properties of the new filter.

1. The filter should offer an *objective* solution to the assimilation problem.
2. The filter should fully consider the nonlinear nature of the forward model.
3. No assumption about the prior pdf should be posed.
4. The filter should be able to deal with an arbitrary likelihood pdf.
5. The filter should avoid particle divergency by ensuring an analysis ensemble whose members not only lie in the high-probability region of the observation space but additionally exhibit equal relative weights.
6. In contrast to the EWPF, the filter should only act on the analysis step and should be independent of the model and the forecast step.
7. No sampling error should be introduced that might degrade the performance.
8. The filter should be computationally efficient even in larger-dimensional settings.

4.1.3 Strategy and questions

The new filter aims at a second-order exact analysis in the sense that the first two moments of the analysis ensemble exactly match the corresponding Bayesian estimates. In section 3.3.2, we motivated that the ETKF utilizes the fact that the KF covariance can be written in the form $\mathbf{X}'_f\mathbf{A}(\mathbf{X}'_f)^T$ which allowed to identify a suitable matrix operation that transforms the prior perturbations $\mathbf{X}'_f$ into analysis perturbations $\mathbf{X}'_a$. Here, we follow the same derivation strategy, but instead of using the KF covariance, we utilize the Bayesian estimate, as in the PF. Consequently, we need to write its covariance in a similar way. In conjunction, this requires the evaluation of the Bayesian mean. Thus, the first step towards the new filter is a reformulation of the second-order statistics of the PF analysis, given any prior ensemble. Afterwards, we proceed with this result to derive the proper transform matrix, as for the ETKF. The remaining sections in this chapter investigate the properties of the analysis algorithm and show further extensions and supplements. These considerations will answer the following questions:

- What are the properties of the transform matrix, in comparison to its counterpart in the ETKF?
- How can the non-uniqueness of a second-order exact transformation be utilized to increase filter stability?
- How can we deal with the curse of dimensionality in higher-dimensional spaces? In particular, can the filter be localized?
- How can the filter stability be judged and can covariance inflation be useful, as for the EnKFs?

4.1.4 Filter input and output

The suggested filter only operates on the analysis step. Therefore, any time dependencies are omitted to describe an arbitrary analysis step at time $t_{i(j)}$. The notation remains as defined in chapters 2 and 3. Thus, the input is an *equally-weighted* prior ensemble $\mathbf{X}_f$ and an observation $\mathbf{y}$, together with its observation operator $\mathcal{H}(\cdot)$. The aim is to output an equally-weighted analysis ensemble $\mathbf{X}_a$.

4.2 Reformulation of the Bayesian moments

As shown in Chapter 3, any posterior Bayesian expectation can be approximated by a corresponding Monte Carlo estimate, using the forecast ensemble as a sample of the prior pdf (see eq. 3.25):

$$\langle f(\mathbf{x})\rangle_{p(\mathbf{x}|\mathbf{y})} \approx \overline{f(\mathbf{x})} = \sum_{n=1}^{N} w^n f(\mathbf{x}_f^n) \tag{4.1}$$

In the following, the focus is put on the functions $f(\mathbf{x})$ that generate the analysis mean and covariance. For convenience, the weight vector $\mathbf{w} = \left(w^1, \ldots, w^N\right)$ of length N is defined. Each element corresponds to the normalized particle's weight, which for an equally-weighted prior ensemble is

$$w^n = \frac{p(\mathbf{y}|\mathbf{x}_f^n)}{\sum_{m=1}^{N} p(\mathbf{y}|\mathbf{x}_f^m)} \tag{4.2}$$

as discussed in section 3.6.3 in detail.

4.2.1 Bayesian mean

To obtain the analysis mean, equation (4.1) is evaluated with $f(\mathbf{x}) = \mathbf{x}$, leading to

$$\overline{\mathbf{x}}_a^{\mathrm{PF}} = \sum_{n=1}^{N} w^n \mathbf{x}_f^n \tag{4.3}$$

The weight vector $\mathbf{w}$ allows to write this sum as a matrix-vector product. Then, the prior ensemble is decomposed into the prior mean, $\overline{\mathbf{x}}_f$, and perturbations, $\mathbf{X}'_f$.

$$\overline{\mathbf{x}}_a^{\mathrm{PF}} = \mathbf{X}_f \mathbf{w} = \overline{\mathbf{X}}_f \mathbf{w} + \mathbf{X}'_f \mathbf{w}$$

Here, $\overline{\mathbf{X}}_f$ represents a $K \times N$ matrix with all columns equal to the prior ensemble mean. Together with the constraint $\sum_n w^n = 1$, this leads to

$$\overline{\mathbf{x}}_a^{\mathrm{PF}} = \overline{\mathbf{x}}_f + \mathbf{X}'_f \mathbf{w} \tag{4.4}$$

Surprisingly, the PF analysis mean turns out to have the same form as the ETKF analysis mean in equation (3.19). A linear combination of the ensemble perturbations, determined by a weight vector, is added to the prior mean. The only difference is that the weight vector now contains the "true" weights, as derived from the fully nonlinear, non-Gaussian Bayes' theorem.

4.2.2 Bayesian covariance

The covariance predicted by Bayes' theorem uses equation (4.1) with $f(\mathbf{x}) = (\mathbf{x} - \overline{\mathbf{x}}_a^{\mathrm{PF}})(\mathbf{x} - \overline{\mathbf{x}}_a^{\mathrm{PF}})^T$, resulting in

$$\mathbf{P}_a^{\mathrm{PF}} = \sum_{n=1}^{N} w^n \left(\mathbf{x}_f^n - \overline{\mathbf{x}}_a^{\mathrm{PF}}\right) \left(\mathbf{x}_f^n - \overline{\mathbf{x}}_a^{\mathrm{PF}}\right)^T \tag{4.5}$$

The aim is to write the analysis covariance matrix in the same form as in equation (3.13) in order to identify the appropriate transform matrix for the new perturbations. In the first step, we substitute the analysis mean with the help of equation (4.4) to introduce the matrix of prior perturbations, and then, we expand the product within the sum:

$$\begin{aligned}
\mathbf{P}_a^{\mathrm{PF}} &= \sum_{n=1}^{N} w^n \left((\mathbf{x}_f^n - \overline{\mathbf{x}}_f) - \mathbf{X}'_f\mathbf{w}\right) \left((\mathbf{x}_f^n - \overline{\mathbf{x}}_f) - \mathbf{X}'_f\mathbf{w}\right)^T \\
&= \sum_n w^n (\mathbf{x}_f^n - \overline{\mathbf{x}}_f)(\mathbf{x}_f^n - \overline{\mathbf{x}}_f)^T - \sum_n w^n (\mathbf{x}_f^n - \overline{\mathbf{x}}_f)\mathbf{w}^T(\mathbf{X}'_f)^T \\
&\quad - \sum_n w^n \mathbf{X}'_f\mathbf{w}(\mathbf{x}_f^n - \overline{\mathbf{x}}_f)^T + \sum_n w^n \mathbf{X}'_f\mathbf{w}\mathbf{w}^T(\mathbf{X}'_f)^T
\end{aligned}$$

As the first term contains the columns of $\mathbf{X}'_f$, it can be written as matrix product $\mathbf{X}'_f\mathbf{W}(\mathbf{X}'_f)^T$ where the matrix $\mathbf{W} \equiv \mathrm{diag}(\mathbf{w})$ is defined as a $N \times N$ diagonal matrix created from the weights. The last term is $\mathbf{X}'_f\mathbf{w}\mathbf{w}^T(\mathbf{X}'_f)^T$ owing to the sum of weights being 1, and the second and third term can also be written as the same matrix-vector product (but with a negative sign). Therefore, the final result is

$$\mathbf{P}_a^{\mathrm{PF}} = \mathbf{X}'_f \left(\mathbf{W} - \mathbf{w}\mathbf{w}^T\right) (\mathbf{X}'_f)^T \tag{4.6}$$

A slight modification of this direct result is recommended for reasons of consistency. Usually, the covariance of an equally-weighted ensemble is computed by $\mathbf{X}'_f(\mathbf{X}'_f)^T/(N-1)$, where the factor of $N-1$ ensures an unbiased estimate. In contrast, a direct evaluation of the Monte Carlo formula (4.1) yields $\mathbf{X}'_f(\mathbf{X}'_f)^T/N$ for an i.i.d. ensemble ($w^n = 1/N \ \forall \ n$), which is a biased estimate, and inconsistent with the definition of empirical covariance used throughout this work. As a consequence, it was suggested (L. Nerger, AWI Bremerhaven, personal communication) to introduce a manual correction factor into the Monte Carlo estimate,

$$\mathbf{P}_a^{\mathrm{PF}} = \frac{N}{N-1}\mathbf{X}'_f \left(\mathbf{W} - \mathbf{w}\mathbf{w}^T\right) (\mathbf{X}'_f)^T \tag{4.7}$$

Apart from consistency, the modified formula results in a slightly larger covariance estimate, particularly if N is small. It is well-known that small ensembles tend to be under-variant, see the more detailed discussion in section 3.4. Therefore, the correction factor is conform to the expected filter behavior. It may also be interpreted as a marginal built-in inflation factor that acts particularly on small ensembles.

Interestingly, it is also possible to derive a different representation of the Bayesian covariance by using $\overline{\mathbf{x}}_a^{\mathrm{PF}} = \mathbf{X}_f\mathbf{w}$ in equation (4.5) instead of $\overline{\mathbf{x}}_a^{\mathrm{PF}} = \overline{\mathbf{x}}_f + \mathbf{X}'_f\mathbf{w}$. The calculation is similar to the one shown

above and is omitted here for that reason. After these manipulations, one arrives at

$$\mathbf{P}_a^{\mathrm{PF}} = \mathbf{X}_f \left(\mathbf{W} - \mathbf{w}\mathbf{w}^T\right) (\mathbf{X}_f)^T \tag{4.8}$$

which is exactly the same form as equation (4.6), only that now the ensemble matrix $\mathbf{X}_f$ is used directly. This implies that the central matrix $\mathbf{W} - \mathbf{w}\mathbf{w}^T$ implicitly subtracts the ensemble mean from a given ensemble matrix. The form (4.8) may be more convenient for some users as it renders a primary evaluation of the perturbation matrix unnecessary and therefore contributes to computational efficiency. For example, in a numerical implementation of the NLEAF, which requires to evaluate $N + 1$ of such covariances at each analysis step (see eq. 3.37), the alternative form is even more advantageous than equation (4.7) concerning computational effort. However, since our intention lies in deriving new analysis perturbations, the form of the covariance given in equation (4.7) is more suitable and will be used exclusively in the following considerations.

In summary, equations (4.4) and (4.7) constitute an efficient reformulation of the most important statistics that is usually derived from an PF analysis ensemble. The representation as matrix-vector products emphasizes the interpretation of the Bayesian mean as weighted linear combination and reduces computational costs as these computations can be implemented efficiently in most programming languages, without resorting to time-consuming loops. However, their usefulness even surpasses these points. The formulas offer the final clue to derive the NETF. In the following, these intermediate results are processed to generate an analysis ensemble which meets the aims defined in section 4.1.

4.3 Square root analysis scheme

The final intention is to obtain a new analysis ensemble $\mathbf{X}_a$, where the members are independently and identically distributed and the corresponding ensemble has mean and covariance that are equal to a Bayesian analysis. It is fairly easy to guarantee that the new ensemble exhibits the desired mean by shifting each ensemble member. Hence, the following three steps are applied to obtain an ensemble that simultaneously exhibits the desired mean and covariance:

1. Compute the prior perturbations from the prior ensemble via $\mathbf{X}'_f = \mathbf{X}_f - \overline{\mathbf{X}}_f$. By definition, the prior perturbations have zero mean, i.e., $\overline{\mathbf{X}}'_f = \mathbf{X}'_f \mathbf{1} = \mathbf{0}$.

2. Transform the prior perturbations into analysis perturbations such that they have the correct covariance and remain centered around zero, i.e., $\mathbf{X}'_a \mathbf{1} = \mathbf{0}$. The algorithm presented here employs a matrix square root technique for that purpose.

3. Re-center these perturbations such that the final ensemble has the correct mean.

4.3.1 Analysis perturbations

In order to create an ensemble which exhibits a pre-specified covariance, the methodology for deriving the ETKF, outlined in section 3.3.2, is utilized. The requirement for the analysis perturbations can be

expressed as follows,

$$\frac{1}{N-1}\mathbf{X}'_a\mathbf{X}'^T_a = \mathbf{P}^{\mathrm{PF}}_a \tag{4.9}$$

The key is to write down the covariance matrix based on the prior perturbations, which are accessible. For this purpose, we make use of the reformulated version of the Bayesian covariance in equation (4.7) and conveniently define the matrix

$$\mathbf{A} = \mathbf{W} - \mathbf{w}\mathbf{w}^T \tag{4.10}$$

such that

$$\frac{1}{N-1}\mathbf{X}'_a\mathbf{X}'^T_a = \frac{N}{N-1}\mathbf{X}'_f\mathbf{A}(\mathbf{X}'_f)^T \tag{4.11}$$

Comparing this with equation (3.17), we note that the analysis covariance has the same structure as in the ETKF, with $\mathbf{A}^{\mathrm{KF}}$ replaced by $\mathbf{A}$. This insight suggests that an analysis ensemble can be constructed by comparing left and right side of this equation,

$$\mathbf{X}'_a = \sqrt{N}\ \mathbf{X}'_f\mathbf{T} \quad \text{, where} \quad \mathbf{T} = \mathbf{A}^{1/2} \quad \text{such that} \quad \mathbf{T}(\mathbf{T})^T = \mathbf{A} \tag{4.12}$$

It is a trivial task to verify that this choice of $\mathbf{X}'_a$ fulfills equation (4.9). This shows that, as in the ETKF, the analysis ensemble perturbations evolve from the prior perturbations through a matrix transformation. The corresponding transform matrix $\mathbf{T}$ is computed as the matrix square root of the matrix $\mathbf{A}$ defined in equation (4.10). It is completely determined by the Bayesian weights, i.e., by the prior ensemble and the observation. For these reasons, the NETF can be classified as deterministic square root filter.

As $\mathbf{A}$ is a real, symmetric, positive semi-definite $N \times N$ matrix (see next subsection), it possesses a unique symmetric square root that can be computed through a SVD, $\mathbf{A} = \mathbf{U}\mathbf{S}\mathbf{U}^T$, such that $\mathbf{T} = \mathbf{U}\mathbf{S}^{1/2}\mathbf{U}^T$. The use of a symmetric square root is motivated by the experiences from ETKF variants (Lawson and Hansen 2004; Sun et al. 2009) and also guarantees that the ensemble perturbations remain centered at zero (Sakov and Oke 2008b).

4.3.1.1 Properties of A

As the matrix $\mathbf{A} = \mathbf{W} - \mathbf{w}\mathbf{w}^T$, the square of the transform matrix, is the central quantity within the NETF, it is necessary to investigate some of its properties, in particular concerning the existence of its square root. For this investigation, we assume all weights to be nonzero, which always holds if $p(\mathbf{y}|\mathbf{x})$ is Gaussian. Currently, it is unclear how a likelihood pdf with finite support that may assign zero weights would affect the NETF.

We first show that $\mathbf{A}$ is positive semidefinite. For this purpose, we make use of the fact that a symmetric, real and (weakly) diagonally dominant matrix with nonnegative diagonal entries is positive (semi)definite (Brouwer and Haemers 2012, p.30, Lemma 2.10.1). By definition, $\mathbf{A}$ is real and symmetric. The diagonal elements of $\mathbf{A}$ are $A_{nn} = w^n(1 - w^n)$, which are always nonnegative. With $A_{nm} = -w^n w^m$ for $n \neq m$, the sum of the absolute values of all off-diagonal elements in any matrix

row n is

$$\sum_{m=1,m\neq n}^{N} |A_{nm}| = w^n \sum_{m=1,m\neq n}^{N} w^m = w^n \left(\sum_{m=1}^{N} w^m - w^n \right) = w^n (1 - w^n)$$

Hence, in any row the sum of the absolute values of the off-diagonal elements equals the corresponding diagonal element. This proves that $\mathbf{A}$ also exhibits weak diagonal dominance, which finally allows us to establish its positive semidefiniteness. As a result, it possesses a unique symmetric square root $\mathbf{T}$ such that $\mathbf{T}(\mathbf{T})^T = \mathbf{A}$. Note that if one neglects an ensemble member to compute $\mathbf{A}$ in a $(N-1)$ dimensional subspace, e.g., the one with the smallest weight, it gets strictly diagonal dominant and therefore positive definite.

As $\mathbf{W}$ is a full-rank matrix and $\mathbf{w}\mathbf{w}^T$ has rank 1, the rank of $\mathbf{A}$ is $N-1$. This implies that one singular value vanishes and the matrix is singular. However, as $\mathbf{T}$ is not inverted, this does not represent a problem. The null space of $\mathbf{A}$ can be found by solving $\mathbf{A}\mathbf{z} = \mathbf{0} \Leftrightarrow \mathbf{W}\mathbf{z} = \mathbf{w}\mathbf{w}^T\mathbf{z}$ for $\mathbf{z}$. Evaluating both sides of this equation, we find that the constraint on any component of $\mathbf{z}$ is $z_n = \sum_m w^m z_m = \text{const}$. Hence, the null spaces of $\mathbf{A}$ and the transform matrix $\mathbf{T}$ consist only of the direction given by the vector $\mathbf{1}$. This property ensures that the analysis perturbations remain centered around zero, as $\mathbf{T}\mathbf{1} = \mathbf{0}$. Additionally, it explains that $\mathbf{A}$ implicitly subtracts the ensemble mean as noted before.

4.3.2 Analysis ensemble

Having derived a method to obtain analysis perturbations which exhibit the desired covariance, the remaining task is compute the final ensemble. From equation (4.4), we already know the sighted analysis mean. As mentioned at the beginning of section 4.3, the final step consists in re-centering the analysis perturbations by the mean vector. This can be conveniently written as $\mathbf{X}_a = \mathbf{X}'_a + \overline{\mathbf{X}}_a$. Here, $\overline{\mathbf{X}}_a = (\overline{\mathbf{x}}_a, \ldots, \overline{\mathbf{x}}_a)$ represents a $K \times N$ matrix with the analysis mean in each column. The covariance of the ensemble is invariant with respect to such a change in the location parameter because the perturbations remain unchanged.

The method suggested so far consists of first computing the perturbations and thereafter re-centering them around the correct mean. As mentioned by Nerger et al. (2012b) in context of the ETKF, both operations can be combined into one single update step in order to enhance computational efficiency,

$$\mathbf{X}_a = \overline{\mathbf{X}}_f + \mathbf{X}'_f \left(\overline{\mathbf{W}} + \sqrt{N}\,\mathbf{T} \right) \tag{4.13}$$

where $\overline{\mathbf{W}} = (\mathbf{w}, \ldots, \mathbf{w})$ is a matrix containing $\mathbf{w}$ in each column. The matrix $\overline{\mathbf{W}} + \sqrt{N}\,\mathbf{T}$ can be be interpreted as an enhanced transform matrix that updates the prior perturbations such that mean and covariance are simultaneously modified as desired.

The analysis algorithm summarized by equation (4.13) generates an ensemble with exactly the same mean and covariance as derived from Bayes' theorem, and therefore, it is referred to as the nonlinear ensemble transform filter (NETF). It applies the same paradigm as followed by the NLEAF (Lei and Bickel 2011) in a deterministic way, that is, without resorting to additional perturbations that introduce stochastic noise and might render the filter suboptimal for finite ensemble sizes. Whereas equation

(4.13) already represents the key component of the NETF, several implications and properties should be discussed before using it in an actual application. This will be the issue of the remainder of this chapter.

4.4 Random rotations

As already mentioned in section 3.3.2 for the ETKF, even though the symmetric square root of $\mathbf{A}$ is unique, the final transform matrix $\mathbf{T}$ of the NETF is not unique. An additional rotation of the ensemble in the space spanned by the ensemble members,

$$\tilde{\mathbf{T}} = \mathbf{T}\mathbf{\Lambda} \tag{4.14}$$

does not change the mean and covariance of the ensemble as long as $\mathbf{\Lambda}$ is orthogonal and satisfies $\mathbf{\Lambda 1} = \mathbf{1}$ (Livings et al. 2008). Hence, one can choose $\mathbf{\Lambda}$ as a random matrix with these properties. To generate such a random matrix, the "second order exact sampling" method of Pham (2001) is applicable. It was developed to obtain a random $N \times (N-1)$ rotation matrix, whereas we need a random $N \times N$ matrix, requiring a slight modification of their algorithm. Appendix C deals with technical aspects of the rotation matrix and also specifies explicit algorithms to generate such matrices efficiently.

For ETKFs, such an additional rotation has already been shown to potentially improve filter stability. The reason is that a purely deterministic update may enhance a situation where the ensemble is nearly collapsed, in the sense that most of the ensemble members are considerably close and the correct moments are enforced by a few strong outliers, as found by Lawson and Hansen (2004) and emphasized by Leeuwenburgh et al. (2005). The NETF derived here is particularly sensitive to this issue as its transform matrix is formed exclusively by the Bayesian weights, which are known to possibly have a large variance, especially in high-dimensional spaces. Such an ensemble is more likely to suffer from weight collapse in the successive analysis step. The random rotation suggested here counteracts this problem by generating a new ensemble which conserves the exact second-order statistics. This procedure comes at the price that higher-order moments are discarded and the analysis ensemble properties become Gaussian (Evensen 2009, p. 203). However, at the current stage it is unclear how the deterministic transformation $\mathbf{T}$ already affects higher order moments which are not considered in our scheme. As we will illustrate in Chapter 5, the experimental results confirm that augmenting the transform matrix with a random rotation strongly improves filter stability.

It should be mentioned that instead of using a random rotation matrix one may also apply a deterministic rotation that meets the requirements mentioned before. Possible choices are, for example, suggested by Nerger et al. (2012b) and Wang et al. (2004). In Chapter 5, we will show the impact of such deterministic rotations on the filter behavior as well.

4.5 Alternative ensemble transformation

The structure of the update mechanism derived in section 4.3.2 is very similar to the ETKF as it utilizes a right-sided transformation of the prior ensemble perturbation matrix $\mathbf{X}'_f$ to obtain new perturbations

$\mathbf{X}'_a$ with the desired covariance. The same request may also be met by an alternative transformation. In mathematical terms, we could attempt to use a left-sided multiplication,

$$\mathbf{X}'_a = \mathbf{T}_L \mathbf{X}'_f \tag{4.15}$$

which resembles the transformation appearing in the ensemble adjustment Kalman filter (EAKF, Anderson 2001). We suggest the following choice of a left-sided transform matrix

$$\mathbf{T}_L = \left(\mathbf{P}_a^{\mathrm{PF}}\right)^{1/2} \left(\mathbf{P}_f\right)^{-1/2} \tag{4.16}$$

It is easily verified that the analysis perturbations exhibit the desired second-order statistics:

- Their mean is $\mathbf{X}'_a \mathbf{1} = \mathbf{T}_L \mathbf{X}'_f \mathbf{1} = \mathbf{0}$, as the prior perturbations are centered at $\mathbf{0}$ by construction.
- Using the fact that covariance matrices and their square roots are symmetric, the empirical analysis covariance is

$$\begin{aligned} \mathbf{P}_a &= \frac{1}{N-1} \mathbf{X}'_a (\mathbf{X}'_a)^T = \frac{1}{N-1} \mathbf{T}_L \mathbf{X}'_f (\mathbf{X}'_f)^T \mathbf{T}_L^T \\ &= \left(\mathbf{P}_a^{\mathrm{PF}}\right)^{1/2} \left(\mathbf{P}_f\right)^{-1/2} \mathbf{P}_f \left(\mathbf{P}_f\right)^{-1/2} \left(\mathbf{P}_a^{\mathrm{PF}}\right)^{1/2} = \mathbf{P}_a^{\mathrm{PF}} \end{aligned}$$

Again, the update can be augmented by a suitable random rotation in ensemble space, as discussed in section 4.4,

$$\mathbf{X}_a = \overline{\mathbf{X}}_a + \mathbf{T}_L \mathbf{X}'_f \mathbf{\Lambda} \tag{4.17}$$

In this form, the filter variant is referred to as the NETF in state space (NETFiSS). When used with random rotations, the filter properties are expected to be identical to the NETF since a new ensemble with Gaussian characteristics is created. This question will be picked up in the experimental Chapter 5. However, the type of the ensemble transformation heavily affects the computational resources needed, as will be discussed next.

4.6 Computational complexity and implementation issues

Considering an application of the NETF in higher-dimensional systems, its computational efficiency is an important practical issue. However, also in problems of medium or intermediate size an application is often facilitated if the filter not only requires little programming efforts but also runs relatively fast. Furthermore, aiming at the principle applicability in larger-dimensional settings, it is of interest to judge its computational complexity in comparison with other ensemble filters.

4.6.1 Examination of individual steps

First, the main NETF computations are summarized with a focus on their numerical expenses, and suggestions for an efficient implementation are given. An explicit algorithm containing all equations and computational steps is provided in section 4.8.1.

- The ensemble perturbations have to be prepared, and all ensemble states have to be mapped into observations space.

- The Bayesian weights have to be evaluated, as in the PF. This step depends on the likelihood density. In case of Gaussian observations, the computation of the exponent in equation (3.26) using the innovation vectors is the most expensive part as the inverse of the observation error covariance matrix $\mathbf{R}$ is required. However, often it is constant over time, and hence, its inverse can be prepared in advance of the analysis cycles. Furthermore, given the relative high uncertainty about error correlations, often only the error variances are used, and $\mathbf{R}$ becomes diagonal, $R_{kl} = \sigma_l^2 \delta_{kl}$. Then, the weights can be computed by

$$w^n \propto \exp\left\{-\frac{1}{2}\sum_{l=1}^{L}\frac{(y_l - y_{f,l}^n)^2}{\sigma_l^2}\right\}$$

  which can be implemented very efficiently. In other applications, $\mathbf{R}$ may be block diagonal with each block representing a group of correlated observations (Hunt et al. 2007). Again, if the block size is small compared to the dimension of the matrix, the inversion can still be performed quickly.

- While matrix $\mathbf{A}$ is formed easily, given the weight vector $\mathbf{w}$, the next important step consists in finding its square root by a SVD. Most programming languages, such as `FORTRAN`, `C` or `R`, can be augmented by linear algebra packages. For example, LAPACK is designed for optimal performance on shared-memory vector and parallel processors (Anderson et al. 1999). It offers convenient subroutines to perform SVDs.

- The generation of random rotation matrices consumes some additional resources. Here, it should be possible to resort to a collection of pre-calculated random matrices since they only depend on ensemble size N.

- Once $\mathbf{T}$ and $\mathbf{\Lambda}$ are obtained, the final step consists in updating the ensemble states via a matrix multiplication in ensemble subspace.

This list reveals that, given a moderate ensemble size, the NETF does not exhibit any highly expensive operations or recursions. While these considerations refer to the standard NETF, the left-sided variant (NETFiSS) shows a distinct computational expense. It performs the update in state space as its transform matrix is of dimension $K \times K$, and it requires to compute the product of two matrix square roots of this size (of which one is an inverse matrix). Hence, not only two SVDs in state space have to performed, but also the inverse of the potentially ill-posed, low-rank matrix $\mathbf{P}_f^{1/2}$ has to be found. This filter variant is only computationally efficient if the state dimensionality is rather small compared to ensemble size, which is usually not the case.

4.6.2 Comparison to other filters

Its moderate computational cost also distinguishes the NETF from other PF-based methods (Andrieu et al. 2010; Doucet et al. 2001; van Leeuwen 2009) that employ clever algorithms but become very expensive even for low-dimensional problems. Furthermore, the NETF exhibits a simple, clearly-arranged design.

The operations performed by the NETF have the same structural form as in the ETKF, in particular concerning the computation of the transform matrix as a matrix square root and the update of the perturbations. The analysis is also performed in the N-dimensional subspace spanned by the ensemble members. In contrast to all EnKFs, our filter does not require to compute an inverse matrix. This avoids computational instabilities caused by considerably small singular values, which are sometimes neglected in ETKF implementations for that reason (Sakov et al. 2012). If a localized version of the NETF is applied (see below), the local analyses are independent and can be computed in parallel as for the ETKF. In summary, we can conclude that the computational expense is very similar for a given ensemble size. This was also confirmed by comparing the runtime of the filters in the experiments in Chapter 5.

As the NETF represents a deterministic realization of the paradigms employed by the NLEAF, it is important to compare their performance properties also from an economic point of view. The NLEAF, presented in section 3.7.2, exhibits a relatively high computational cost. For each of the N sets of perturbed observations, the weights of the full N-sized ensemble have to be re-computed, requiring N^2 weight computations, apart from the perturbation itself. Then, for the second-order version, it requires the evaluation of $N + 1$ matrix square roots of state covariance matrices at each analysis step, together with N matrix multiplications in state space (Lei and Bickel 2011, Sec. 3.c.). The first-order version is less expensive as it only adjusts each member after the weight computations. These update algorithms are feasible for small-sized problems but become very costly for systems of intermediate and high dimensionality, or for larger ensemble sizes. We conclude that, concerning computational complexity, among these nonlinear filters the NETF is the more efficient choice in larger-dimensional cases.

4.6.3 Subspace form

As for the PF, it may occur that in a certain analysis step some ensemble members have considerably low weights. Here, we suggest a method to neglect them and receive a full analysis ensemble with required first and second moments, originated from the final step in the singular evolutive interpolated KF (SEIK, Nerger et al. 2012b; Pham 2001).

Neglecting the members with low weights (e.g., $w^n < 10^{-3}$) nearly has no effect on the analysis mean and covariance in equations (4.4) and (4.7). After re-normalizing the weight vector of size $N' < N$, the matrix $\mathbf{A}$ as given in equation (4.10) and its square root $\mathbf{T}$ have an reduced dimension of $N' \times N'$. To regenerate a full ensemble of size N, we use a random rotation as shown in equation (4.14), $\tilde{\mathbf{T}} = \mathbf{T}\tilde{\mathbf{\Lambda}}^T$, only that we replace the random $N \times N$ matrix $\mathbf{\Lambda}$ with a $N \times N'$ matrix $\tilde{\mathbf{\Lambda}}^T$ by omitting $N - N'$ of the columns in the square matrix $\mathbf{\Lambda}$. If $\mathbf{\Lambda}\mathbf{\Lambda}^T = \mathbf{I}_N$ and $\mathbf{\Lambda}\mathbf{1} = \mathbf{1}$ hold, the column vectors of $\tilde{\mathbf{\Lambda}}^T$ are orthogonal to each other and to $(1, \ldots, 1)^T$, and therefore, mean and covariance are preserved.

Disregarding low-weight members in the analysis step could enhance an efficient computation of the matrix square root. However, in initial tests we found no clear differences concerning stability and performance. The subspace form may be important in case of non-Gaussian observation errors where ensemble members can actually achieve zero weights.

4.7 Supplements

So far, a basic version of the NETF was presented, which realizes the aims stated at the beginning of this chapter. It was also motivated that the random rotation should be an integral part of the algorithm in order to obtain useful analysis ensembles. As will be shown in Chapter 5, the filter presented in this form is capable of producing convincing results in the Lorenz 63 model as a first test case. However, one primary objective is the applicability in large-dimensional systems as typical in meteorology or geophysics. In section 3.4, extensions to render ensemble-based filters like the ETKF applicable for such problems were already discussed. Even though numerous ad-hoch techniques exist, they can be categorized as following:

- Localization techniques suppress spurious correlations and reduce the effective dimensionality of the assimilation problem.
- Inflation procedures try to counteract the tendency of the ensemble to underestimate the forecast uncertainty.

It is expected that these techniques can also improve the performance of the NETF. First, van Leeuwen (2009) suggested that localization might be a helpful tool to fight the curse of dimensionality inherent to PFs, as it strongly reduces the effective state dimensionality. However, it is argued that its application is a challenge for PFs since the weights become spatially dependent (Lei and Bickel 2011; van Leeuwen 2009). The NETF overcomes this problem because at each analysis step a new, equally-weighted ensemble is generated and local weights only exist intermediately. Second, the NETF analysis ensemble does not represent the true Bayesian analysis pdf but ”only” its first two moments. This ensemble is integrated in order to represent the prior pdf at the next observation time level, and it is potentially under-variant as some initial uncertainty may have been neglected by the second-order approximation in the previous analysis step.

In the following, we illustrate the consistent application of localization and inflation for the NETF. Along the way, it offers a solution to the problem of PF localization, an active topic in the literature within the last years (e.g., Cheng and Reich 2015; van Leeuwen 2009).

4.7.1 Localization

Localization has been introduced in detail in section 3.4.2. Covariance localization (CL) cannot be applied for the NETF, as it does not involve the prior covariance matrix. The main advantage of domain localization (DL), its generality, allows to perform local analyses with the NETF similar to the principles summarized by Hunt et al. (2007), computing a separate analysis for each model grid point

where only observations in a defined radius are considered. Additional to such a domain localization (Houtekamer and Mitchell 1998), the observational influence can be reduced with distance by multiplying the corresponding entries in $\mathbf{R}^{-1}$, which in our filter appears in $p(\mathbf{y}|\mathbf{x})$ in the case of Gaussian observations, with an appropriate weight function (observation localization, e.g., Kirchgessner et al. 2014; Miyoshi and Yamane 2007). The LETKF algorithm, given in section 3.4.3, can be adopted by only exchanging the LETKF analysis ensemble in step 2.f with the NETF analysis ensemble.

If localization is used, the random rotation has to be applied to the global ensemble in order to avoid inconsistent local random transformations, i.e., after the fully-deterministic local updates the global analysis is replaced by $\tilde{\mathbf{X}}_a = \overline{\mathbf{X}}_a^{\mathrm{PF}} + \mathbf{X}'_a\mathbf{\Lambda}$. Equivalently, one can use the same rotation matrix $\mathbf{\Lambda}$ in all local updates. The latter strategy may be more convenient to implement, see also section 4.8.1.

4.7.2 Inflation

As discussed in section 3.4.1, covariance inflation is usually beneficial to EnKFs to counteract the underestimation of the variance. Even though the NETF, in contrast to all EnKFs, does not directly work with the prior covariance matrix, an ensemble which underestimates the prior covariance is typically too tight, which in turn increases the probability of weight collapse as the particles are more likely to fall outside the region of significant probability. The NETF forecast ensemble is potentially under-variant because (1) it constitutes a finite-sized approximation of the prior pdf, and is therefore subject to the Monte Carlo error, (2) unconsidered model error terms during the forecast step hide uncertainty and (3) it arises from the integration of an analysis ensemble which does not fully represent the true analysis pdf because it only considers its first two moments. In conclusion, we can expect that inflation can also be beneficial to the NETF by increasing the forecast spread, as also observed for other PF-based algorithms (Frei and Künsch 2013a; Lei and Bickel 2011). It can be applied in the usual way before or after the analysis step by replacing the corresponding ensemble according to $\mathbf{X} \to \sqrt{\gamma}\,\mathbf{X}' + \overline{\mathbf{X}}$. This procedure procedure increases all entries of the state covariance matrix by a factor γ, typically slightly larger than 1 (Anderson and Anderson 1999).

It should be noted that even though inflation may be helpful, it also renders the filter suboptimal because it subjectively adjusts the forecast ensemble which is used to represent the prior pdf as used in Bayes' theorem. Therefore, it should be applied carefully, and it is expected that it is only required to make the NETF applicable for relatively small ensemble sizes. For more sophisticated applications, it may be useful to augment the algorithm by an adaptive inflation scheme as referred to in section 3.4.1.

4.8 Summary and final algorithm

The NETF is only concerned with the analysis step, converting any prior ensemble into an analysis ensemble by assimilating the current observation. It generates an equally-weighted ensemble whose empirical mean and covariance are identical to the Bayesian estimates, resulting from a nonparametric treatment of the forecast pdf. In the following, the NETF analysis step is summarized in an algorithmic form, including localization as discussed in section 3.4.2. This simplifies the implementation of the NETF into an existing ensemble DA system that works with domain localization.

4.8.1 Update algorithm

The algorithm given below considers a localized update. If localization is not required, one simply has to skip step 3 and ignore all distinctions between local and global quantities. We assume that the global state vector of dimension K is partitioned into a number of *local domains*, where each domain may for instance contain all model variables at one single grid point. For simplicity of the description, following Hunt et al. (2007), we mark all global quantities with an additional subscript $[g]$, i.e., we begin with the prior ensemble $\{\mathbf{x}^n_{f[g]}\}$ of size N, stored in the $K \times N$ ensemble matrix $\mathbf{X}_{f[g]}$, and the L-dimensional observation $\mathbf{y}_{[g]}$ with error covariance $\mathbf{R}_{[g]}$. All quantities referring to the current local domain have no additional subscript.

1. Compute the predicted observations by applying the observation operator to the prior ensemble, $\mathbf{y}^n_{f[g]} = \mathcal{H}(\mathbf{x}^n_{f[g]})$, and put the resulting vectors into a $L \times N$ ensemble matrix $\mathbf{Y}_{f[g]}$.

2. Prepare an appropriate orthogonal, mean-preserving random $N \times N$ rotation matrix $\mathbf{\Lambda}_{[g]}$ (see appendix C for an explicit algorithm).

3. For a local analysis, this step selects the data for the current local domain: Extract all rows of $\mathbf{X}_{f[g]}$ which belong to the current local domain to obtain the local ensemble states, $\mathbf{X}_f$. Select the entries of $\mathbf{y}_{[g]}$ which are to be considered for the local analysis (e.g., all observations within a specified localization radius) and the corresponding rows and columns of $\mathbf{R}_{[g]}$, which forms the local observation vector $\mathbf{y}$ and covariance matrix $\mathbf{R}$. If desired, multiply the entries of $\mathbf{R}^{-1}$ with an appropriate weight function to reduce the influence of more distant observations. Accordingly, choose the same rows from $\mathbf{Y}_{f[g]}$, forming $\mathbf{Y}_f$, which contains the ensemble's counterparts $\mathbf{y}^n_f$ of the localized observation $\mathbf{y}$.

   In case a global analysis is desired, simply set $\mathbf{X}_f = \mathbf{X}_{f[g]}$, $\mathbf{y} = \mathbf{y}_{[g]}$, $\mathbf{Y}_f = \mathbf{Y}_{f[g]}$ and $\mathbf{R} = \mathbf{R}_{[g]}$ in the following steps.

4. Calculate the Bayesian weights of the ensemble states using the observational likelihood density, normalize them and put them in the weight vector $\mathbf{w}$,

$$W^n \propto p(\mathbf{y}|\mathbf{x}^n_f) \quad , \quad w^n = \frac{W^n}{\sum_m W^m} \tag{4.18}$$

   For example, in case of Gaussian observation errors evaluate

$$W^n = \exp\left\{-\frac{1}{2}\left(\mathbf{y} - \mathbf{y}^n_f\right)^T \mathbf{R}^{-1}\left(\mathbf{y} - \mathbf{y}^n_f\right)\right\} \tag{4.19}$$

5. Compute the (local) analysis mean by

$$\overline{\mathbf{x}}_a = \overline{\mathbf{x}}_f + \mathbf{X}'_f \mathbf{w} \tag{4.20}$$

6. Form the matrix $\mathbf{A}$, with $\mathbf{W} = \mathrm{diag}(\mathbf{w})$, and compute the transform matrix:

$$\mathbf{T} = \mathbf{A}^{1/2} = \left(\mathbf{W} - \mathbf{w}\mathbf{w}^T\right)^{1/2} \tag{4.21}$$

   Use the symmetric square root by applying a singular value decomposition to $\mathbf{A}$.

7. Transform the prior ensemble perturbations into analysis perturbations by applying the transform matrix and random rotation,

$$\mathbf{X}'_a = \sqrt{N}\,\mathbf{X}'_f \mathbf{T} \mathbf{\Lambda}_{[g]} \tag{4.22}$$

8. Re-center the ensemble perturbations to obtain the (local) analysis ensemble:

$$\mathbf{X}_a = \overline{\mathbf{X}}_a + \mathbf{X}'_a \tag{4.23}$$

9. Having performed steps 3 to 8 for all local domains, aggregate the outputs of step 8 to form the final global analysis ensemble $\mathbf{X}_{a[g]}$.

4.8.2 Implementation advices

From this presentation, it becomes clear that the difference between a global and local update is only of organizational nature. The core steps 3 to 8 produce the analysis ensemble, given the observation and the prior ensemble as well as its corresponding representation in observation space. For the computations within these steps, it is irrelevant whether they represent global or local quantities. As a consequence, it is possible to outsource these steps into independent subroutines. This strategy also allows for a simple parallelization of the localized analysis step since the local updates can be performed independently.

In case an ensemble DA system using the ETKF is available, the NETF can directly be implemented by only exchanging the core routine that turns the (local) forecast ensemble into the (local) analysis ensemble. The forecast step is untouched, and all other analysis step routines that concern the selection of the local states and observations, the mapping into observation space and the aggregation of the local analyses, should remain identical as for the ETKF.

4.9 Nudged NETF

In section 3.7.1, is was shown that the PF can be adjusted by proposal densities that enable to nudge the particles towards the observation during the forecast phase, if a model error term is available. For the standard NETF, we exclusively concentrated on the analysis step to facilitate its application and to keep it general, as the nudging formulation depends on the forward model.
However, it is a nice feature of the NETF that it can be combined with a nudged forecast step with no additional effort. While the EWPF modifies the final proposal density before the next observation time level, one may also abandon the relatively complicated equivalent-weight step and proceed the nudged ensemble forecast until observation time. Then, in contrast to the standard NETF, the forecast ensemble $\mathbf{X}_f$ is not an equally-weighted ensemble anymore, but exhibits prior weights $\{\tilde{w}_f^n\}_{n=1\dots N}$, as specified by equation (3.30). Nevertheless, the NETF analysis can be performed as outlined in section 4.8.1, if the prior weights are considered in the computation of the intermediate weights in step 4 as follows:

$$W^n \propto \tilde{w}_f^n \, p(\mathbf{y}|\mathbf{x}_f^n) \tag{4.24}$$

The mathematical justification for this simple treatment has already been given when presenting the basic PF in section 3.6.2. In conclusion, the consideration of prior weights in the NETF is trivial and allows an easy combination with a nudged forecast step in case it is available anyway. This may also be beneficial to assess its relative performance as compared to the EWPF since the NETF offers a distinct, simplified formulation of the analysis step.

4.10 Conclusions

This chapter presented a new filter to create a second-order exact analysis ensemble. Various implications and properties of the suggested method were highlighted. The most important conclusions concerning the theoretical derivation and treatment of the NETF are as follows:

- Its analysis mean and covariance exactly match the Bayesian expectations, without any sampling noise.
- A new, equally-weighted analysis ensemble is created by adding increments to each member and applying a moment-preserving random rotation. Together with localization, the filter is expected to overcome the curse of dimensionality and should be applicable in larger dimensions.
- The update is computed in ensemble subspace, which is feasible also in high-dimensional systems.
- The algorithm is simple to apply as it is formally identical to the ETKF mechanism.

At the beginning of this chapter, a list of desired properties for the new filter was formulated (see section 4.1.2). It is now evident that the NETF meets all these requirements, at least from a theoretical point of view. The basic update equation (4.13) describes the transformation of the prior ensemble $\mathbf{X}'_f$. Despite its formal analogy to the ETKF, it is nonlinear as the weight vector $\mathbf{w}$, which determines the transformation, depends on the forecast ensemble in a nonlinear way through the likelihood density. These weights arise from the fully nonlinear, non-Gaussian Bayes' theorem.

In summary, the NETF represents an analysis algorithm with appealing properties for nonlinear, non-Gaussian DA. Consequently, as "the proof of the pudding is in the eating", the next step in its characterization consists in an application to actual DA problems.

Chapter 5

Empirical investigation of filter performance

The previous chapter presented the derivation of a novel analysis algorithm, the NETF, in the field of sequential, ensemble-based filtering. It represents a contribution to solve the nonlinear assimilation problem. The thorough analysis of its properties resulted in the conclusion that the filter offers a potential applicability to nonlinear and large-scale DA problems, which is particularly facilitated by its simple implementation that strongly resembles the ETKF.
The next step in the characterization of the new filter algorithm consists in its actual application to DA scenarios, beginning with typical test beds. This allows to confirm the theoretical considerations made in the previous chapter. Furthermore, we focus on comparing the performance of the NETF to other ensemble-based filters introduced in Chapter 3.

5.1 Preparations

In this chapter, we apply the NETF to a variety of DA problems that are characterized by different properties. We makes use of challenging, but simple systems in order to focus on the filter properties without influences of model shortcomings or other approximations.

5.1.1 Methodology

In all experiments, we perform so-called observation system simulation experiments (OSSEs, Masutani et al. 2010). Their advantages with respect to real-world applications are:

- The perfect-model environment and the availability of a reference, the truth, allow an unambiguous evaluation of the analysis results.
- The filter performance can be investigated without all the suboptimalities that affect real-world DA problems, as all assumptions that concern the external input to an ensemble-based filter, such as the observation error characteristics or the model specification, can be met exactly by

	NETF	ETKF	ETKFrot	EnKF	NLEAF1	NLEAF2
inspired by	PF	KF	KF	KF	PF	PF
update mechanism	det.	det.	det.	stoch.	stoch.	stoch.
random rotations	yes	no	yes	no	no	no
considered moments	2	2	2	2	1	2
dim(analysis space)	N	N	N	L	K	K

TABLE 5.1: Basic properties of the six ensemble filters used in the experiments.

construction. In real-world DA applications, the major implementation issue consists in tuning these poorly known variables in a desirable way.

- Assimilation settings, such as observation density or error, can be modified easily in order to investigate the filter's sensitivity.

5.1.2 Aims

There are two principle aims in this chapter. First, it is important to assess the performance properties of the NETF in situations that are characterized by different degrees of nonlinearity and dimensionality. This will not only allow to draw conclusions about the potential applicability to real-world problems but will also give some hints on implementation issues and limitations. Therefore, a particular focus is put on the behavior with small ensemble sizes. Second, we compare the NETF's performance with other ensemble-based filters, which will allow to judge its potential benefits that are expected from the theoretical investigation. Table 5.1 summarizes the properties of the filters used in this chapter.

5.1.3 Overview of the experiments

The experiments in this chapter comprise a variety of different models and scenarios. In order to keep track of their objectives, we first summarize the test cases to be investigated, along with their scientific relevance.

1. The Lorenz 63 system (L63) is a low-dimensional, yet strongly nonlinear and chaotic model. It allows to assess the benefits of the nonlinear formulation.

2. In addition, the Lorenz 96 and 2005 models (L96, L2005) exhibit a larger dimensionality. They are used to investigate whether the filter is able to deal simultaneously with nonlinearity and a larger state dimensionality.

3. The linear advection model (LA) sets up a typical environment where the ensemble size is much smaller than the dimensions of state and observation space.

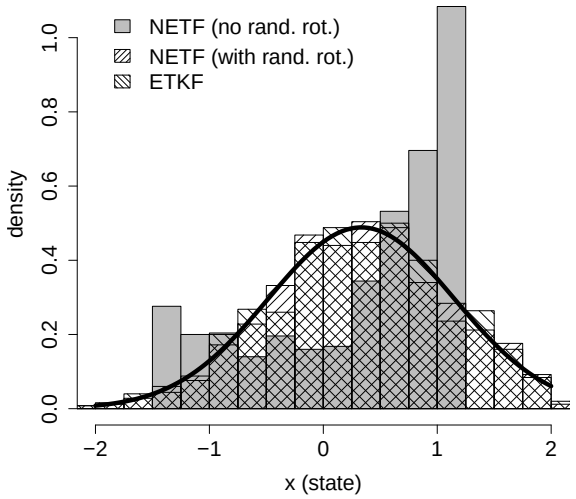


FIGURE 5.1: Empirical analysis distributions in the scalar example for the NETF without random rotation (gray), NETF with random rotation (shaded, ///) and ETKF (shaded, \\\\) with $N = 1000$ each. The latter two differ only by sampling variations. The black line shows the analytical solution.

5.2 Importance of the random rotations

In Chapter 4, the additional random ensemble transform was suggested, as the purely deterministic transform is potentially affected by a high variance of the Bayesian weights. Here, we perform a small test in order to investigate the role of the random rotation before turning to the dynamical models.

We demonstrate the single update of a scalar state variable x with Gaussian prior and likelihood densities, $\mathcal{N}(x; 1, 2)$ and $\mathcal{N}(x; 0, 1)$, respectively. The analytical solution, as given by the KF, is the Gaussian analysis pdf $\mathcal{N}(x; \frac{1}{3}, \frac{2}{3})$. An ensemble of $N = 1000$ forecast states is generated and updated by the ETKF and the NETF with and without random rotation. Figure 5.1 visualizes the resulting densities in form of histograms. As expected, the ETKF transform matrix (3.18), which is derived from the theoretical KF analysis covariance (2.44), produces a normally distributed ensemble. In contrast, the ensemble generated by the NETF without rotation clearly does not follow the desired analysis pdf, even though it exhibits the correct second-order statistics by construction. After the random rotation, the ensemble is Gaussian again. At the current stage, the impact of the purely-deterministic transform with (4.12) on higher-order moments is an open issue. However, already this simple example points out the importance of the random rotation for the NETF, since it creates a new, more consistent ensemble. This empirical insight will also be verified in a dynamical situation below.

5.3 Models and their setups

In all cases, we perform twin experiments, meaning that a reference trajectory, referred to as truth $\mathbf{x}_{tr}(t)$, is generated, from which synthetic observations are drawn by means of perturbation. The observation

errors are restricted to be non-correlated with constant variance, i.e., $\mathbf{R} = \sigma_{obs}^2 \mathbf{I}_L$, where L is the number of observed state components.

We compare the NETF with the ETKF variants (with and without random rotations) and with the first- and second-order versions of the NLEAF that rely on perturbed observations. In this context, we also include the classical EnKF, as defined in section 3.2.6, as the stochastic counterpart of the ETKF. Table 5.1 summarizes basic properties for all these filters. For each experimental run, all filters under investigation receive exactly the same observations, initial ensemble and, if needed, rotation matrices.

An ensemble filter's performance in nonlinear scenarios is quite sensitive to the chosen inflation, which is the most important tuning parameter (Sakov and Oke 2008b). In order to compare the filter results fairly, it is desirable to remove the dependency on inflation. We tuned the prior inflation factor γ for optimal performance (in terms of RMSE, see below) by varying $\sqrt{\gamma} \in \{1.00, 1.01, \ldots, 1.15\}$ for each experimental case and filter. While this simple but expensive procedure is sufficient for toy model tests, for real applications, more sophisticated methods such as adaptive inflation (e.g., Anderson 2007a; Luo and Hoteit 2013; Miyoshi 2011) may be more suitable.

The following setup issues are common to all Lorenz experiments: The truth is generated through the integration of an initial state for 100 nondimensional time units of the corresponding model, and the DA experiments are performed within this time range. The true initial state $\mathbf{x}_{tr}(0)$ itself is the outcome of a spin-up run over 30 time units. The initial ensembles are sampled around $\mathbf{x}_{tr}(0)$ (as, e.g., in Yang et al. 2012a), in order to minimize the effects of filter spin-up. They are generated by second-order exact sampling with a covariance of one tenth of the climatological one (extracted from the truth), see Appendix C for details.

5.3.1 Lorenz 1963 (L63)

The well-known L63 model (Lorenz 1963) is a three-dimensional system, $\mathbf{x} = (x, y, z)^T$, described by the ordinary differential equations

$$\begin{aligned} \dot{x} &= \sigma(y - x) \\ \dot{y} &= -xz + \rho x - y \\ \dot{z} &= xy - \beta z \end{aligned}$$

Because of its simple formulation but challenging dynamics, this model has often been applied as benchmark for DA experiments (e.g., Ahrens 1999; Miller et al. 1994; Sakov et al. 2012; Yang et al. 2012a). The system is solved with a fourth-order Runge-Kutta discretization scheme with time step $\Delta t = 0.01$ (therefore, 100 time units are equal to 10000 time steps), using the standard parameters $\sigma = 10$, $\rho = 28$ and $\beta = 8/3$. With this choice, the system exhibits strong nonlinearity that depends on the current flow. The spin-up run is initialized with $(-8, 8, 27)^T$. We use a partial observation system for x and y while the z component remains hidden. In addition to the variation of the ensemble size, the experiments differ in the time interval between two successive Gaussian observations, $\Delta t_{obs}/\Delta t \in \{10, 15, 20, 25\}$, and their error variance, $\sigma_{obs}^2 \in \{2, 4\}$.

5.3.2 Lorenz models of intermediate dimensionality

The L96 system (Lorenz 1996) roughly emulates a meteorological variable on a latitude

$$\dot{x}_k = (x_{k+1} - x_{k-2})x_{k-1} - x_k + F \quad , \quad k = 1 \dots K$$

It is widely applied as another typical nonlinear DA test bed (e.g., see Frei and Künsch 2013a; Kirchgessner et al. 2014; Lei and Bickel 2011; Nerger et al. 2012b). The system is applied on a periodic domain with $K = 80$ grid points (twice the usual dimensionality) so that $\mathbf{x} = (x_1, \dots, x_{80})^T$, and a forcing of $F = 8$ is used. It is solved with a fourth-order Runge-Kutta scheme with the typical time step $\Delta t = 0.05$ (therefore, 100 time units are equal to 2000 time steps). The spin-up run is initialized with $x_{K/2} = 1.001\, F$ and $x_k = F$ for $k \neq K/2$.

Because of the intermediate dimensionality of the system, we apply the filters in a localized form using local analyses together with observation localization, as outlined in section 4.7.1. Hence, for each grid point, only observations within the localization radius $r_{loc} = 5$ are considered, and we apply a fifth-order polynomial correlation function (Gaspari and Cohn 1999, eq. 4.10) to $\mathbf{R}^{-1}$ such that the observational weights smoothly decrease with distance. The observation error variance is set to $\sigma^2_{obs} = 1$ and the system is observed at all grid points with an odd index ($L = 40$) every other time step. This sparse-observation situation is a difficult setup for a DA system (Lei and Bickel 2011); particularly if one considers that the time between two observations corresponds to about 12 hours in the real atmosphere (Lorenz 1996). Furthermore, we create a slightly non-Gaussian observation scenario whose treatment is an advantage of the nonlinear filters. To that purpose, we sample the independent observation errors from a univariate Laplace (double exponential) density with zero mean and a scale parameter of $\sqrt{\sigma^2_{obs}/2}$ that ensures a density variance of σ^2_{obs} (see Appendix A). The KF-based filters implicitly assume the likelihood pdf to be a Gaussian density with the same variance, which in principle looks similar, but has slightly less weight near zero and in the tails.

In the L96 system, two neighboring grid points do not exhibit a significant correlation (Lorenz 2005). Here, localization is mainly applied as a dimension reduction technique, as discussed by Lei and Bickel (2011), since a filter solely based on the Bayesian weights is not able to work directly in high-dimensional spaces due to the curse of dimensionality. However, it may be more desirable to apply localization based on actual spatial dependence. For that purpose, we also investigate the filter behavior in model II of Lorenz (2005), which extends the L96 model by a spatial smoothing parameter κ,

$$\dot{x}_k = -[x_{k-2\kappa}][x_{k-\kappa}] + \Big[[x_{k-\kappa}]x_{k+\kappa}\Big] - x_k + F \quad , \quad k = 1 \dots K$$

where $[\cdot]$ denotes an average of the nearby grid points. Rainwater and Hunt (2013) applied this L2005 system to DA experiments, and we adopt their model design with $\kappa = 2$ (then, $[x_k] = (x_{k-1} + 2x_k + x_{k+1})/4$) and $F = 12$, resulting in a detectable correlation for up to five neighboring grid points. However, again we use a state dimensionality of $K = 80$. Except for an increased localization radius of $r_{loc} = 10$, the setup remains the same as that described for the L96 model.

5.3.3 Linear advection (LA)

In a final experiment, we test the filter in a simple model that is not characterized by its challenging dynamics but instead by a state dimensionality one order of magnitude larger than the just-presented Lorenz systems. This demand is met by the linear advection (LA) model on a one-dimensional grid with $K = 1000$, first used by Evensen (2004), that propagates the previous field to the right at each time step without changing its shape:

$$x_k(t+1) = x_{k-1}(t) \quad , \quad k = 1 \dots K \quad (\text{with } x_0 \equiv x_K)$$

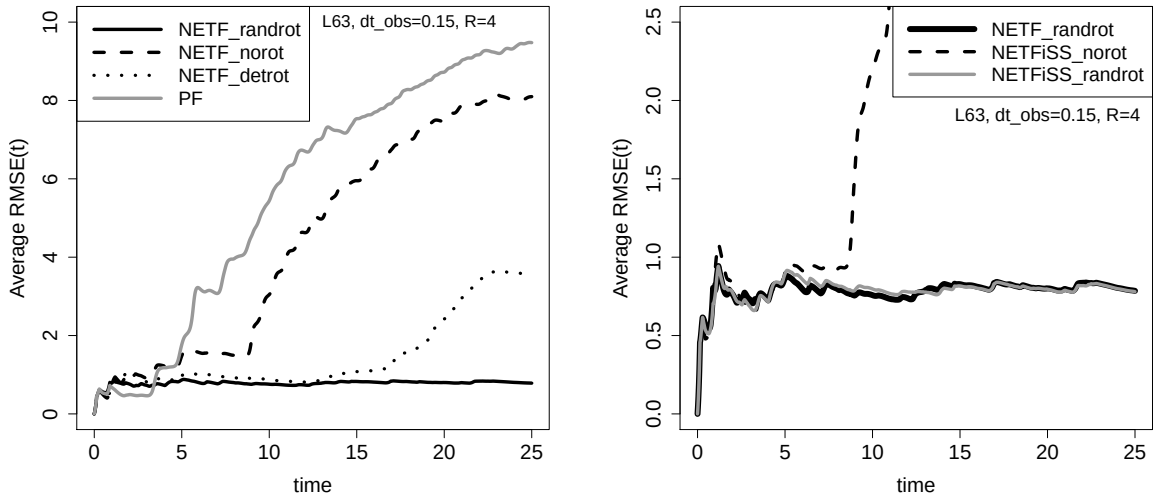
The experimental setup slightly differs from the Lorenz experiments: Following Sakov and Oke (2008b) and Evensen (2009), a climatological field is obtained by sampling a random Gaussian field whose correlation function is Gaussian with a spatial de-correlation length of 20 grid points and a variance of one. The true field as well as the initial ensemble fields are generated by adding additional random fields with this characteristics to the climatology. The integration time is $T = 250$, and a set of $L = 100$ Gaussian observations is created with $\sigma^2_{obs} = 1$ every five time steps at each tenth grid point. The localization radius equals twice the spatial length scale, $r_{loc} = 40$. No inflation is used within this linear scenario, where the analysis field should converge to the truth.

5.4 Evaluation

The main evaluation criterion is the root mean square error (RMSE). At all times t, the difference between the analysis mean and truth is quantified using $\mathrm{RMSE}(t) = \sqrt{\left|\left|\overline{\mathbf{x}}_a(t) - \mathbf{x}_{tr}(t)\right|\right|_2 / K}$ and then, the average over the entire period of interest is computed.

In some cases, we also assess the quality of the ensembles by computing the empirical ensemble spread via $\sigma_{ens} = \sqrt{\mathrm{trace}\left(\mathbf{P}_a\right)/K}$ as a measure of its uncertainty, which should on average be of similar magnitude as the RMSE (e.g., Hopson 2014; Palmer et al. 2005). The statistical reliability of the ensemble can be quantified by finding the 0.025 and 0.975 quantiles of the ensemble. For a well-calibrated ensemble, the truth should lie inside this interval in 95% of all cases. Finally, the continuous ranked probability score (CRPS, Gneiting et al. 2007) and ignorance score (CRIGN, Tödter and Ahrens 2012) are useful probabilistic measures that consider the entire ensemble distribution with respect to the truth. They are computed component-wise, and their purpose is to summarize the distance between the ensemble pdf and pdf of the verifying truth in probability space. While the CRPS applies a quadratic norm, the CRIGN quantifies the difference in information content and is more sensitive to cases where the truth lies close to or outside of the ensemble range boundaries.

The innovations, $\mathbf{d} = \mathbf{y} - \overline{\mathbf{Y}}_f$, allow the extraction of further diagnostics independent of the truth. Specifically, the expected innovation variance (Desroziers et al. 2005) can be expressed as the sum of the observation error variance and average prior ensemble variance in observation space, $(\sigma^{exp}_{inno})^2 = \overline{\mathrm{trace}\left(\mathbf{Y}'_f(\mathbf{Y}'_f)^T/(N-1)\right)}/L + \sigma^2_{obs}$. Following Anderson (2007a) and Compo et al. (2011), it should correspond to the actual innovation variance, $(\sigma_{inno})^2 = \overline{\mathbf{d}^T\mathbf{d}}/L$, in a well-calibrated DA system.



(a) NETFs vs. PF

(b) NETF vs. NETFiss

FIGURE 5.2: (a) Temporal evolution of average RMSEs for the NETF with rotations (full, black line), the NETF with deterministic rotations (dotted) as well the NETF without rotations (dashed) and the PF (gray). (b) Same, but for the NETF with rotations (full, black line), the NETFiSS without rotations (dashed) and the NETFiSS with random rotations (gray line). The latter uses the same rotation matrices as the NETF, resulting in an almost identical behavior. Note that the y-axis is scaled differently here.

5.5 Results and discussions

5.5.1 Lorenz 63 (L63)

The first experiment concentrates on some basic properties of the NETF, using $N = 100$ ensemble members in a setup with $\sigma^2_{obs} = 4$ and $\Delta t_{obs} = 15\Delta t$. In order to demonstrate the importance of the random rotation, we ran the filter (1) without augmented rotations, (2) with a deterministic, mean-preserving rotation matrix (Nerger et al. 2012b, eq. 24), and (3) with random rotation matrices as introduced in section 4.4. Figure 5.2a shows the temporal evolution of the average RMSEs for the first 2500 time steps. Initially, the basic PF (with universal resampling) shows the best results, but the strong increase in RMSE at $t > 4$ indicates filter collapse, which is expected in a fully deterministic system, as motivated in section 3.6. The NETF without rotations is not much better than the PF, as it diverges only a few assimilation cycles later. With a deterministic rotation, stability can be maintained for a longer period, but eventually the filter collapses as well. In contrast, the NETF with random rotations remains stable until the end of the assimilation window. We generated very similar results independently of the setup, and did not succeed in obtaining stable runs without rotations or with deterministic ones. Next, in Figure 5.2b we compare the standard NETF with the variant NETFiSS that employs a left-sided transformation, see section 4.5. Again, we notice that without random rotations, the filter quickly diverges, and in similar experiments, we did not obtain stable results over the long term, even with inflation tuning. However, if the NETFiSS uses the same random rotation matrices as the NETF, the performances are almost identical, as we expected from the theoretical considerations. As a consequence, in all further experiments we only refer to the standard NETF with random rotations, as summarized in section 4.8.1.

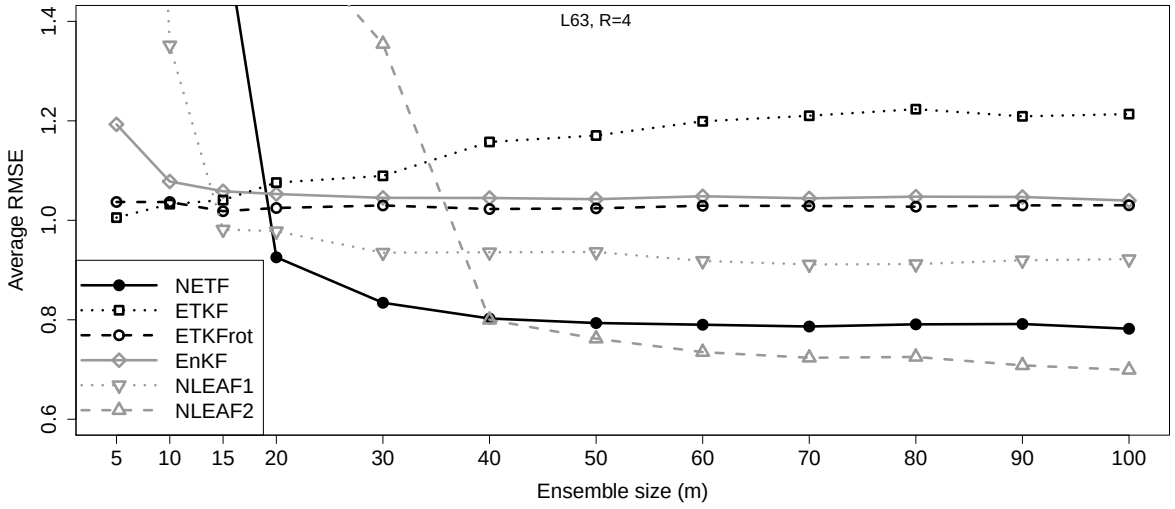


FIGURE 5.3: Average RMSEs in the L63 experiment with $\sigma^2_{obs} = 4$ and $\Delta t_{obs} = 15\Delta t$ for varying ensemble sizes. Shown are NETF (black line), ETKF (black, dotted), ETKFrot (black, dashed), EnKF (gray line), NLEAF1 (gray, dotted) and NLEAF2 (gray, dashed).

For practical reasons, it is important to investigate the filter performance with respect to ensemble size, so all experiments are run with $N \in \{5, 10, 15, 20, 30, \ldots, 100\}$. Figure 5.3 focusses on the required ensemble sizes to obtain reasonable RMSE results in the scenario with $\Delta t_{obs} = 15\Delta t$ and $\sigma^2_{obs} = 4$. While the ETKF with random rotations is only slightly better than the EnKF for $N \geq 20$, the ETKF without rotations degrades with ensemble size, probably because the probability of generating outliers increases (Anderson 2010; Lawson and Hansen 2004). The NETF's performance is in between the NLEAF1 and NLEAF2 for larger ensemble sizes. However, the NETF diverges only for $N \leq 15$ and delivers superior results even with just 20 members, whereas the NLEAF2 requires more than 40 ensemble members to outperform it, presumably because of the additional sampling noise.

For this scenario, Table 5.2 presents the additional evaluation measures introduced in section 5.4. Because of covariance inflation, the ensembles are not under-dispersive as their spread typically exceeds the RMSE slightly. This does not hold for the NLEAF1/2 at $N = 10$ and $N = 30$, respectively, as they require more members to work properly, in accordance with Figure 5.3. Apart from such exceptional cases, the innovation variance is usually of similar magnitude as the prior ensemble variance plus the observation error variance, indicating that inflation tuning leads to a reliable assimilation system. This is also confirmed by the fact that the truth is in between the 95% confidence interval of the ensemble close to 95% of all times, except for small ensemble sizes, and for the ETKF without rotations in general. The probabilistic scores CRPS and CRIGN emphasize that the NETF not only produces a better analysis mean than the KF-based filters at each time step, but also an ensemble distribution that accumulates more probability mass near the truth.

Finally, Figure 5.4 shows the optimal average RMSE for different temporal observation densities when the observation error variance is $\sigma^2_{obs} = 2$. Here, for each filter the smallest RMSE over all ensemble sizes has been chosen, based on the insight that large ensemble sizes can be detrimental, particularly to the ETKF. In general, the analysis quality decreases almost linearly with Δt_{obs}. The NETF clearly

ens. size	filter	RMSE	spread	σ_{inno}	σ_{inno}^{exp}	CRPS	CRIGN	p95
$N=10$	NETF	-	-	-	-	-	-	-
	ETKF	1.03	1.22	2.59	2.69	0.71	2.26	0.82
	ETKFrot	1.04	1.21	2.59	2.69	0.68	2.21	0.83
	EnKF	1.08	1.35	2.65	2.91	0.72	2.33	0.85
	NLEAF1	1.35	1.14	3.25	2.74	0.95	3.42	0.73
	NLEAF2	-	-	-	-	-	-	-
$N=30$	NETF	0.83	1.05	2.46	2.56	0.54	1.76	0.91
	ETKF	1.09	1.27	2.64	2.72	0.73	2.39	0.83
	ETKFrot	1.03	1.26	2.59	2.73	0.66	2.11	0.92
	EnKF	1.05	1.29	2.59	2.74	0.67	2.14	0.91
	NLEAF1	0.93	1.06	2.53	2.53	0.60	1.96	0.88
	NLEAF2	1.35	1.09	3.22	2.74	0.99	3.75	0.83
$N=50$	NETF	0.80	1.05	2.42	2.55	0.51	1.63	0.94
	ETKF	1.17	1.27	2.73	2.70	0.79	2.64	0.78
	ETKFrot	1.02	1.24	2.57	2.68	0.65	2.07	0.93
	EnKF	1.04	1.30	2.59	2.74	0.66	2.11	0.93
	NLEAF1	0.94	1.08	2.50	2.54	0.60	1.93	0.91
	NLEAF2	0.76	0.92	2.42	2.46	0.49	1.60	0.91
$N=100$	NETF	0.78	1.01	2.40	2.50	0.49	1.57	0.95
	ETKF	1.21	1.34	2.77	2.82	0.82	2.73	0.80
	ETKFrot	1.03	1.26	2.58	2.71	0.65	2.08	0.95
	EnKF	1.04	1.29	2.59	2.72	0.65	2.09	0.95
	NLEAF1	0.92	1.10	2.48	2.54	0.58	1.87	0.93
	NLEAF2	0.70	0.86	2.35	2.39	0.44	1.43	0.93

TABLE 5.2: Detailed evaluation of the L63 experiment with $\sigma_{obs}^2 = 4$ and $\Delta t_{obs} = 15\Delta t$. RMSE and spread are computed in state space. The innovation standard deviation is compared to the square root of the average prior ensemble variance plus the observation error variance. CRPS and CRIGN represent the average over all state variables, and the same holds for p_{95}, the relative frequency of the truth being within the 95% confidence interval of the ensemble. All values represent the respective time averages over five experimental runs. Empty entries indicate that no stable runs are achieved in this setup. For more details concerning the measures, see section 5.4.

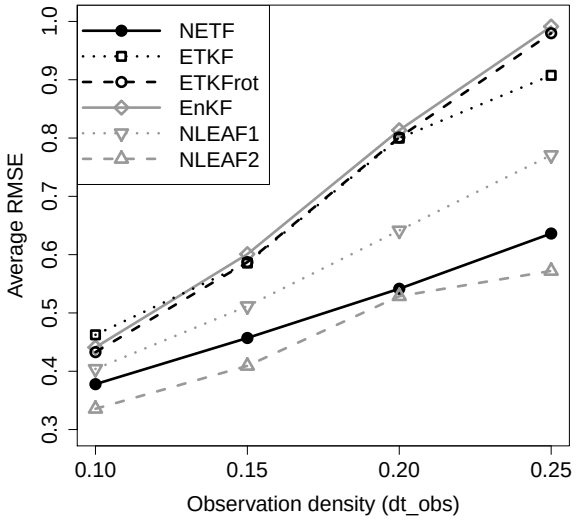


FIGURE 5.4: Minimal average RMSEs in the L63 experiment with $\sigma_{obs}^2 = 2$ for varying temporal observation densities. The smallest RMSE over all ensemble sizes has been chosen to determine the best achievable performance for each case and filter. Shown are NETF (black line), ETKF (black, dotted), ETKFrot (black, dashed), EnKF (gray line), NLEAF1 (gray, dotted) and NLEAF2 (gray, dashed).

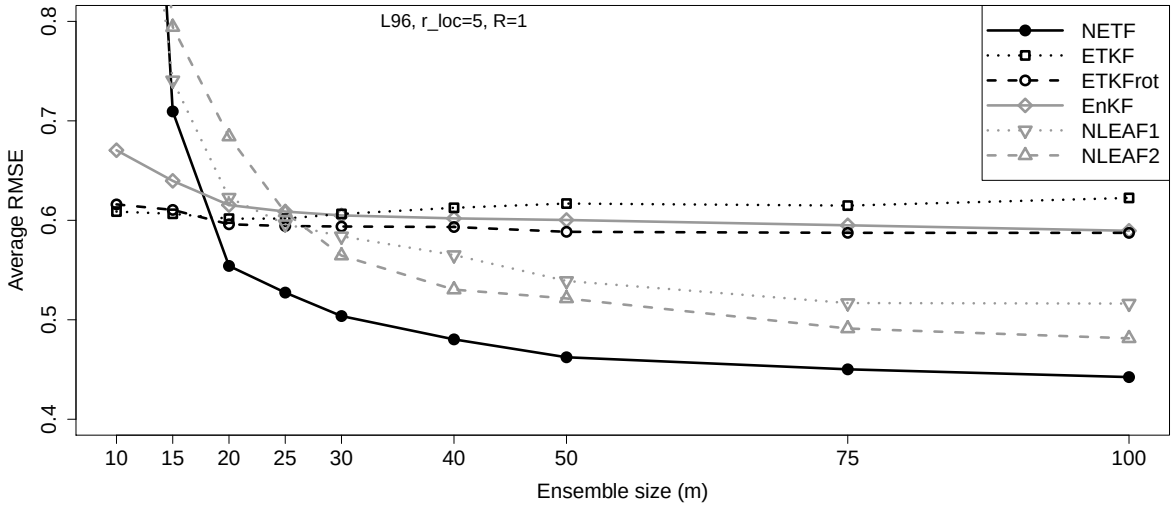


FIGURE 5.5: Result of the L96 experiment with double exponential observation errors with $\sigma^2_{obs} = 1$. Shown is the average RMSE for the six ensemble filters against ensemble size, i.e., NETF (black line), ETKF (black, dotted), ETKFrot (black, dashed), EnKF (gray line), NLEAF1 (gray, dotted) and NLEAF2 (gray, dashed).

outperforms the KF-based filters that here all give quite similar results, with the ETKFs being slightly better than their stochastic counterpart. In the most nonlinear scenario ($\Delta t_{obs} = 25\Delta t$) the ETKF without rotations shows the better performance due to its optimal behavior at very small ensemble sizes. The NETF performs similar to the second-order version NLEAF2, but is not able to improve upon it. Their relative improvement over the ETKF increases with larger observation distance, a sign of a better treatment of the rising nonlinear developments.

From the low-dimensional L63 experiments, we conclude that the NETF outperforms the KF-based algorithms because of its nonlinear ensemble update, as well as the NLEAF1 because it aims at second-order exactness. Nevertheless, it cannot completely achieve the quality of the NLEAF2 in the L63 environment. This can probably be explained by the fact that the random rotations tend to suppress higher-order moments that may implicitly be (at least partly) preserved by the stochastic NLEAF2 updates. A similar behavior has been pointed out in a comparison of the ETKF and EnKF by Lawson and Hansen (2004) who found the stochastic update to be more capable of dealing with strongly nonlinear situations. However, the NETF is still more efficient because the NLEAF2 requires larger ensemble sizes to achieve superior RMSE values.

5.5.2 Lorenz models of intermediate dimensionality

The L63 experiments confirmed that the NETF is able to create a stable and good analysis in the presence of strong nonlinearity. Regarding a potential application to real-world problems, its behavior in systems of larger dimensionality is an important issue.

The first experiment concerns the 80-dimensional L96 system with the setup described in section 5.3.2. Figure 5.5 summarizes the filter performances in terms of RMSE for various ensemble sizes. The

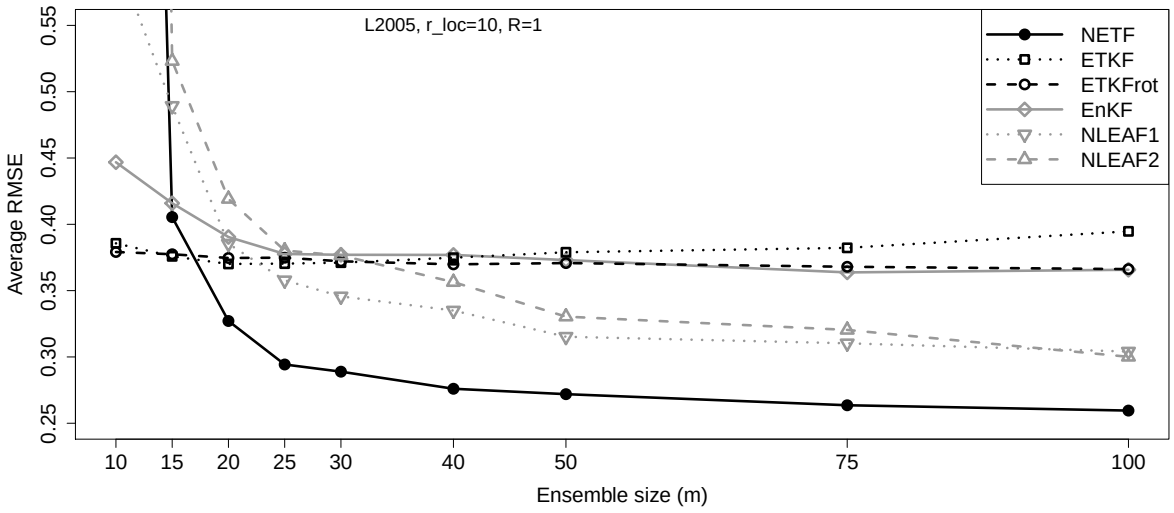


FIGURE 5.6: As Figure 5.5, but for the L2005 experiment with double exponential observation errors with $\sigma^2_{obs} = 1$.

KF-based filters achieve similar results, and it is noticeable that the ETKF without rotations performs best for $N = 20$ and then degrades again. With random rotations, the ETKF's performance is almost independent of ensemble size in this scenario, as also reported by Lei and Bickel (2011) for their experiments. The stochastic EnKF is only worse for small ensemble sizes. For $N \geq 30$, the NLEAFs outperform the KF-based methods, while the NETF consistently exhibits the best performance already with an ensemble size of 20 or more. This confirms that in such a nonlinear, non-Gaussian scenario of intermediate dimensionality, the suggested method performs well with localization, and the deterministic update mechanism seems to be more beneficial than its stochastic counterpart.

The next experiment concerns the L2005 model, which exhibits a distinct spatial structure compared to the L96 model. Figure 5.6 shows the analysis error with respect to ensemble size. Concerning the relative performances of the KF-based filters, the general structure is very similar to the L96 experiment. The most remarkable, seemingly counterintuitive difference is that here the NLEAF1 performs better than the NLEAF2, except when $N = 100$. Lei and Bickel (2011) do not show the NLEAF2 for their larger-dimensional experiments, hence, we cannot directly compare these findings to their results. A possible explanation could be revealed by the update formalism of the NLEAF2 (Lei and Bickel 2011, Eq. 3), which requires the estimation of an individual analysis covariance matrix for each ensemble member, based on each of the N perturbed observations. These low-rank approximations of the $K \times K$ covariance matrix with N members are subject to sampling error, and it seems that the stochastic errors of the perturbed observations amplify the errors in the estimation of the covariances, which may lead to these unexpected results in certain larger-dimensional cases. This hypothesis is supported by the fact that in the low-dimensional L63 experiments the NLEAF2 consistently outperformed the NLEAF1. Additionally, in the L2005 system, spatial correlations are more important than in the L96 system, hence a reliable estimation of the covariances is of greater relevance here. The NETF, which also focusses on the second-order statistics, does not suffer from this issue. Again, it exhibits the smallest

ens. size	filter	RMSE	spread	σ_{inno}	σ_{inno}^{exp}	CRPS	CRIGN	p95
$N=10$	NETF	-	-	-	-	-	-	-
	ETKF	0.39	0.38	1.09	1.09	0.22	0.71	0.80
	ETKFrot	0.38	0.40	1.09	1.10	0.21	0.69	0.84
	EnKF	0.45	0.44	1.11	1.10	0.25	0.84	0.81
	NLEAF1	0.60	0.50	1.18	1.12	0.34	1.16	0.75
	NLEAF2	-	-	-	-	-	-	-
$N=25$	NETF	0.29	0.27	1.06	1.04	0.16	0.53	0.89
	ETKF	0.37	0.39	1.08	1.09	0.20	0.66	0.90
	ETKFrot	0.37	0.36	1.08	1.08	0.21	0.66	0.87
	EnKF	0.38	0.44	1.08	1.09	0.21	0.68	0.93
	NLEAF1	0.36	0.37	1.08	1.07	0.20	0.64	0.91
	NLEAF2	0.38	0.39	1.09	1.08	0.21	0.69	0.89
$N=50$	NETF	0.27	0.29	1.05	1.05	0.15	0.47	0.93
	ETKF	0.38	0.38	1.09	1.09	0.21	0.70	0.89
	ETKFrot	0.37	0.39	1.08	1.09	0.20	0.65	0.91
	EnKF	0.37	0.42	1.08	1.08	0.20	0.66	0.94
	NLEAF1	0.32	0.34	1.06	1.06	0.17	0.55	0.94
	NLEAF2	0.33	0.32	1.07	1.05	0.18	0.58	0.91
$N=100$	NETF	0.26	0.28	1.05	1.05	0.14	0.44	0.95
	ETKF	0.39	0.42	1.09	1.10	0.22	0.75	0.88
	ETKFrot	0.37	0.37	1.08	1.08	0.20	0.64	0.91
	EnKF	0.37	0.43	1.08	1.09	0.20	0.65	0.96
	NLEAF1	0.30	0.35	1.06	1.06	0.16	0.53	0.96
	NLEAF2	0.30	0.33	1.06	1.05	0.16	0.52	0.95

TABLE 5.3: Detailed evaluation results as in Table 5.2 for the L2005 experiment with double exponential observation errors with $\sigma_{obs}^2 = 1$. For more details concerning the measures, see section 5.4.

analysis error for $N \geq 20$. We conclude that, particularly in larger-dimensional scenarios, the benefits of the deterministic update mechanism of the NETF become more apparent.

In order to strengthen these findings, Table 5.3 gives an overview of more diagnostic measures for $N \in \{10, 25, 50, 100\}$. As in the L63 case, the comparison of RMSE and spread as well as of innovation variance and expected innovation variance indicate that, thanks to the inflation procedure, the filters behave reasonably well in both state and observation space. Except for $N = 10$, all scores reveal that the NETF performs best. The CRIGN shows that in information-theoretical terms, the NETF ensembles also exhibit the closest match with the truth. The CRPS confirms this insight, but its differences are smaller. The reason is that the CRIGN is more sensitive to outliers, which is why it is also the best score for detecting the degradation of the ETKF without rotations for larger N, in contrast to the RMSE that only increases slightly. Furthermore, all scores are consistent with the finding that the NLEAF2 can only achieve the performance of the NLEAF1 for $N = 100$.

In the L96/L2005 experiments presented so far, the NETF and NLEAFs were able to compute their analysis based on the actual, non-Gaussian likelihood pdf while the ETKFs and EnKF could only use the specified covariance $\mathbf{R}$ in a Gaussian manner. A concluding L2005 experiment examines the impact of such a misspecification of the likelihood on the NETF. The setup remains identical as before, except that now *all* filters wrongly assume the likelihood to be Gaussian. Figure 5.7 shows the resulting RMSEs. While the performances of the ETKFs and EnKF are unchanged, the nonlinear filters now show less dominant behavior. It is not surprising that the approximation of non-Gaussian error distributions as Gaussian can lead to significant analysis errors (Fowler and van Leeuwen 2013). In particular, the

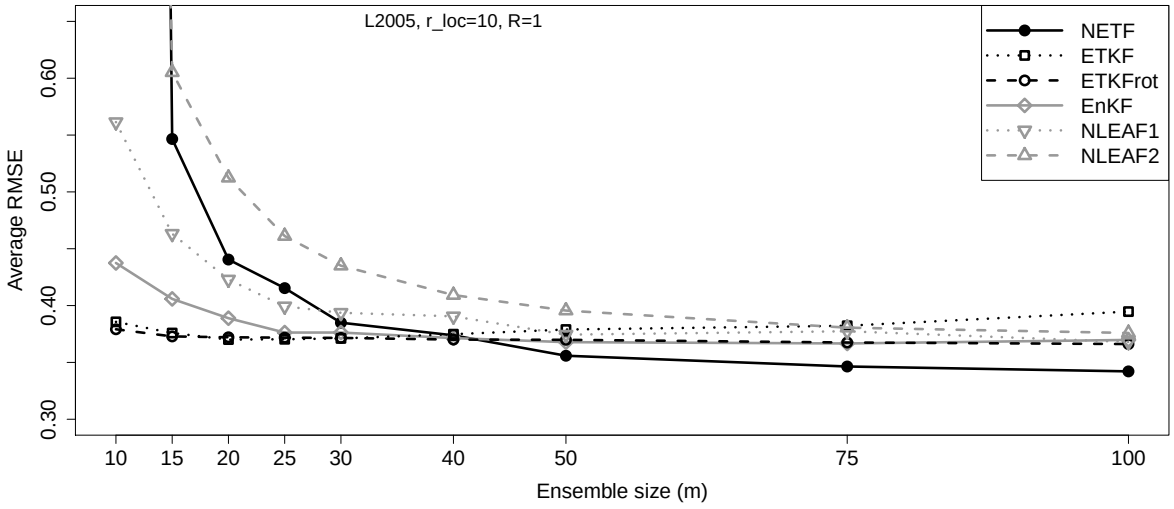


FIGURE 5.7: As Figure 5.6, except that in this L2005 experiment *all* filters wrongly assume the observation errors to be Gaussian, which affects the performance of the NETF and the NLEAFs.

NLEAFs are not able to outperform the EnKF and the ETKF anymore. Note that here the NLEAF1 always outperforms the NLEAF2. However, the NETF still exhibits a better analysis at least for $N \geq 50$, even though the improvement is not as pronounced as before. These results indicate that even if the NETF does not use the true likelihood pdf to find the analysis weights, it still has the potential to achieve superior results.

In summary, the experiments with intermediate Lorenz systems form a consistent picture. The NETF performs well already for an intermediate ensemble size and is able to outperform not only the ETKF and EnKF but also its stochastic counterpart, the NLEAF. We note that without random rotations the NETF diverges, as in the L63 cases. Based on the experiments performed in this work, we conclude that the additional random rotation is required to maintain filter stability by generating a new ensemble with Gaussian properties. However, it is possible that other experimental settings or implementations might lead to different experiences.

5.5.3 Linear advection (LA)

In this final experiment, the dimensions of ensemble ($N = 5 \ldots 50$), observations ($L = 100$), and state ($K = 1000$) differ by about one order of magnitude, representing a difficult setup for filters that rely only on the Bayesian weights. However, it approaches real-word situations that are usually characterized by $N \ll L \ll K$.

Figure 5.8 shows the RMSE of each filter resulting from simulations with various ensemble sizes. By definition, the chosen scenario represents a linear, Gaussian assimilation problem. Recalling the Gaussian assumption built into all KFs, it is not surprising that the ETKF, which utilizes this information in a very efficient manner, performs best. Thus, its RMSE can be seen as the lower limit for a given ensemble size. Nevertheless, the NETF, which cannot benefit from the Gaussianity of the situation as

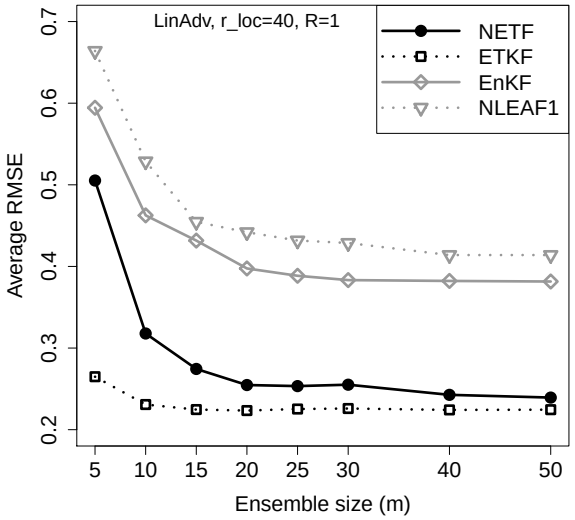


FIGURE 5.8: Results of the LA experiment: Average RMSE for NETF (black line), ETKF (black, dotted), NLEAF1 (gray, dotted) and EnKF (gray line) for varying ensemble sizes. Not shown are the NLEAF2, which performs slightly worse than the NLEAF1 in this scenario, and the ETKFrot, which behaves as the ETKF.

much as the ETKF because it only relies on the Bayesian weights, is stable and approaches the performance of the ETKF for an ensemble size of $N > 15$. In contrast, the stochastic filters (NLEAFs and EnKF) perform similarly in that they exhibit a considerably higher RMSE and do not converge to the minimum value. In this scenario, the perturbed observations introduce additional random noise of considerable magnitude and render these filters considerably suboptimal. In the L63 experiments, they helped to improve the analysis. However, the successive experiments have indicated that in situations where the main challenge is given by an increased dimensionality, the second-order exact filters (NETF and ETKF) show a more favorable behavior. Furthermore, by means of localization, it could be confirmed that the NETF promises a potential applicability in systems where the ensemble size is rather small compared to the dimensions of observation and state.

5.6 Additional aspects

This section briefly highlights two further aspects of nonlinear DA.

5.6.1 Parameter estimation

All experiments presented so far dealt with state estimation, i.e., finding the optimal value for *dynamic* quantities. Another possible application of DA methods with practical relevance is the estimation of parameters, which are usually *constant* model quantities (e.g., Aksoy et al. 2006). An illustrative example is the autoregressive model of first order, $x_{i+1} = \phi x_i + \epsilon_i$, where ϵ represents random noise and the parameter ϕ has to be estimated, given observations of the time series. This example already emphasizes

filter	RMSE (state)	MRD (σ) [%]	MRD (r) [%]	MRD (b) [%]
NETF	0.86	4.97	0.86	2.15
ETKF	1.71	12.68	1.86	4.33
ETKFrot	1.41	5.63	1.21	3.17
EnKF	1.44	5.74	1.35	3.15
NLEAF1	1.02	5.91	1.06	2.51
NLEAF2	0.83	5.81	1.02	2.54

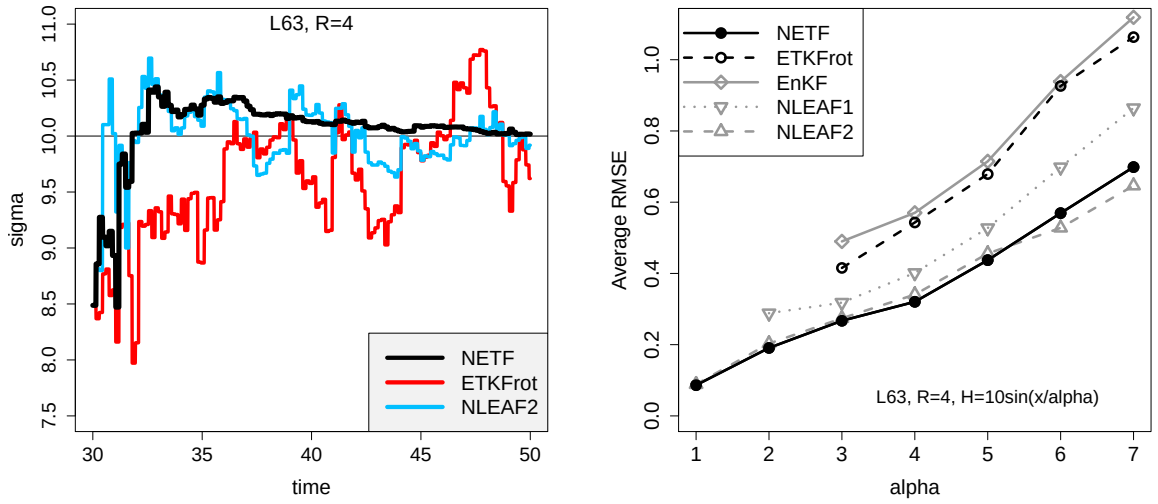
TABLE 5.4: Results for the L63 experiments with parameter estimation. MRD represents the mean relative deviation of the filter estimate with respect to the true value, averaged over all time steps. It is given in percent for all three L63 parameters. All values represent the average over ten repetitions.

that parameter estimation usually represents a nonlinear problem, even if the model dynamics itself is linear.

The general approach to include parameter estimation in DA is *state augmentation* (e.g., Evensen 2003), i.e., the parameters $\mathbf{p}$ are interpreted as time-dependent quantities which follow the "trivial" forward equation $\mathbf{p}_{i+1} = \mathbf{p}_i + \boldsymbol{\epsilon}_i$. The error term $\boldsymbol{\epsilon}$ allows to add random noise. Then, the analysis algorithms do not change if one identifies $\mathbf{z} = (\mathbf{x}, \mathbf{p})^T$ as generalized state vector $\mathbf{x}$. However, parameter estimation imposes some additional challenges to a DA system, apart from nonlinearity. For example, parameters are often restricted to a finite valid range, whereas the implicit assumption of Gaussian distributions allows parameter updates to non-permitted, potentially physically impossible values. It has been shown that, as expected, nonlinear filters can be advantageous to classical algorithms in the context of parameter estimation (e.g., Kivman 2003; Losa et al. 2003). However, it is argued that even though PFs may yield better estimates, the problem of filter divergence is often more pronounced, because particles with wrong parameter values may receive insignificant weights. This usually increases the required ensemble size.

Therefore, it is of interest to look at the performance of the NETF with respect to parameter estimation, as it aims at rendering unstable, fully-nonlinear filtering more stable by concentrating on the analysis mean and covariance only. The focus of this section is not on a full investigation of this issue, but rather on gaining a first insight. We apply the L63 testbed as described in section 5.3.1, with $\Delta t_{obs} = 15\Delta t$ and observations of x and z with $\sigma^2_{obs} = 4$. The assimilation window is shortened to 2000 time steps, as the parameter estimates should converge. The estimation targets are all three L63 parameters (σ, r, b), thus, the state dimension is doubled. The initial parameter ensembles are constructed by drawing $N = 100$ samples from uniform distributions within $[4; 13]$, $[21; 42]$ and $[1.9; 2.4]$ for σ, r and b, respectively. As the true values are 10, 28 and $\frac{8}{3}$, the initial knowledge is biased. Kivman (2003) performed a similar L63 experiment, comparing the PF and EnKF. However, he created an extremely nonlinear situation with very few observations, in which the EnKF fails. The PF required 1000 particles, but for σ, the most challenging parameter, the converged estimate still differed from the true value. In comparison, the situation here is already strongly nonlinear, as proven by the superiority of the nonlinear filters in section 5.5, but is still feasible even for the linear filters. Thus, it allows a more balanced judgement, as even in challenging real-world systems, such as the atmosphere, EnKFs are still able to deliver reasonable estimates.

The results are illustrated by the temporal evolution of the filter estimates for σ in Figure 5.9a. While the ETKF exhibits a rather slow convergence and high volatility, the NETF consistently and smoothly



(a) Parameter estimation

(b) Nonlinear $\mathcal{H}$

FIGURE 5.9: Additional aspects in the L63 experiment with $N = 100$, $\sigma^2_{obs} = 4$ and $\Delta t_{obs} = 15\Delta t$. (a) Evolution of the filter estimates in time, for the parameter $\sigma = 10$. Shown are NETF (black), ETKFrot (red) and NLEAF2 (blue). (b) RMSE results with the nonlinear observation operator for different values of the nonlinearity parameter α. Missing points indicate filter divergency, and the ETKF without rotations (not shown) does not work in any case.

approaches the true value. The NLEAF2 is also able to achieve this towards the end, but exhibits more variability, probably due to the perturbed observations. Table 5.4 gives further statistical information about the analysis errors. For the parameters, they are quantified by the relative absolute error, i.e., the absolute deviation to the truth divided by the true value. Due to the additional parameter uncertainty, the state RMSEs are increased compared to the corresponding reference case, see Table 5.2. While the nonlinear filters only perform slightly worse, the degradation is more pronounced for the EnKFs. The results in Table 5.4 confirm that now, in contrast to the case of pure state estimation in the L63 system, the NLEAF2 is not superior anymore. The NETF shows a similar RMSE for the state variables and is better for the parameters. While the perturbed observations were potentially helpful in estimating chaotic state variables in the low-dimensional system, they degrade the estimation of fixed parameters. In summary, it can be stated that the deterministic NETF is suitable for parameter estimation as well, compared to both the NLEAF and KF-based filters.

5.6.2 Nonlinear observation operator

In a concluding experiment, we intend to demonstrate the NETF's performance in another challenging situation. In all previous situations, the state vector was observed directly, at least partly. However, as discussed in section 2.2.3, in real-world applications the relation between model variables and observations can be more complicated and even nonlinear, as expressed by the operator $\mathcal{H}(\mathbf{x})$. The EnKFs map the ensemble members into observation space by the fully-nonlinear operator, see section 3.2.4. However, their analyses are still suboptimal as the update mechanisms originate from the KF update equation, which assumes a linear observation operator. This issue is valid for both deterministic or perturbed-observation versions. Additionally, the 3/4DVAR cost function (e.g., eq. 2.35) is not

quadratic anymore. Thus, local minima may appear, which poses a great challenge to minimization algorithms.

The treatment of nonlinear observations[1] is an active field of research. In 4DVAR, typically an iterative sequence of minimizations is employed, linearizing the predicted observations around the refined state estimates for a couple of times. A similar principle can be applied for EnKFs (Luo and Hoteit 2014b). In contrast, filters that are directly based on Bayes' theorem, such as the NETF of PFs, deal with nonlinear observations in a natural way. Only the likelihood pdf is needed, which is typically a function of the innovations, i.e., $p(\mathbf{y}|\mathbf{x}) = p(\mathbf{y} - \mathcal{H}(\mathbf{x}))$. This does not hold for the EWPF at the current stage, where the proposal densities rely on linear observations. Even though the Bayesian filters are able to fully consider the nonlinearity of $\mathcal{H}(\cdot)$, their application in such situations is more challenging as the required ensemble size to avoid filter divergency is usually larger. The reason is that due to the nonlinear mapping into observation space, where the likelihoods are evaluated, the relative weights of the particles can vary more strongly. Even small differences in the state variables may be amplified by the nonlinear mapping, and the relative weights are sensitive to small variations.

As for parameter estimation, this section serves as short investigation to test the principal ability of the NETF to deal with nonlinear observations. Again, the L63 experiment with $N = 100$, $\Delta t_{obs} = 15\Delta t$ and $\sigma^2_{obs} = 4$ is utilized. Observed variables are x and z, but the observation operator is $\mathcal{H}(x) = 10 \sin(x/\alpha)$ and the same for $\mathcal{H}(z)$. The period of the harmonic function is $2\pi\alpha$, thus, smaller values of α indicate a higher level of nonlinearity. The results of experiments with varying α are summarized in terms of average RMSE in Figure 5.9b. In general, for small values of α, $\mathcal{H}(\cdot)$ exhibits stronger nonlinearity, but is also more sensitive to changes in x or z. Then, the observations are better able to discriminate distinct states. In turn, they constrain the state more strongly, and this leads to lower RMSEs for small α, as visible in Figure 5.9b for all filters. The KF-based filters do not work for the most nonlinear cases ($\alpha = \{1, 2\}$). For $\alpha \geq 3$, their analysis errors are about twice as large as for the nonlinear filters, confirming the superiority of the latter. In contrast to the L63 experiment with linear observations, where the NLEAF2 outperformed the NETF (see Figure 5.3), they now perform similarly. In conclusion, the NETF is apparently able to deal with nonlinear observations in a desirable way.

5.7 Conclusions

We summarize the main insights gained from the series of experiments that applied the NETF to different models and setups:

- Despite being solely based on the Bayesian weights, the algorithm is capable of remaining stable even in difficult situations and higher-dimensional spaces. In the experiments conducted here, the random rotation after the ensemble transformation is required to maintain filter stability. This is probably reasoned in the effect of the deterministic transform with $\mathbf{T}$ on higher-order moments, which is an open issue at the current stage. For the ETKF, the influence of random rotations on the analysis quality depends on the experimental setup, such as ensemble size and model properties.

[1] For simplicity, we use the term "(non)linear observations" for observations that are related to the state by a (non)linear operator.

Thus, further research, from both theoretical and empirical point of view, about the deterministic transform and the impact of the random rotation step in the NETF is desirable.

- In nonlinear and non-Gaussian scenarios, the NETF is able to outperform the KF-based algorithms (ETKF and EnKF), as it does not rely on Gaussian assumptions. In all experiments, the ensemble sizes required to achieve superior results is fairly small, indicating that the update mechanism acts quite efficiently. However, it does not work properly with *very* small ensemble sizes such as five or ten, for which the ETKF may already achieve a sufficient performance.

- Domain localization, as suggested by Hunt et al. (2007) for the ETKF, works successfully also for the nonlinear filter formulation used by the NETF. It reduces the effective state dimensionality, which in turn contributes to avoid weight collapse in larger-dimensional systems. This allows to use the NETF with rather small ensemble sizes in these scenarios as well.

- In some cases, as in low-dimensional but strongly nonlinear scenarios, the perturbed observations used in the NLEAF can be beneficial, at least if the ensemble size is large enough to suppress the sampling errors. However, particularly in larger-dimensional problems, the NETF is able to outperform its stochastic counterpart because of its deterministic update mechanism. This also includes cases with parameter estimation and nonlinear observation operators.

We conclude from these initial experiments that the NETF, despite its rather simple design, appears to be a promising contribution to the field of nonlinear DA and may be suitable for applications in large-scale systems. This will be studied next.

Chapter 6

Application to a high-dimensional general circulation system

In the previous chapter, the NETF's performance was thoroughly investigated in a variety of systems with different dimensionalities and characteristics. Here, the results from an assimilation experiment in a general circulation model with considerably larger dimensionality are shown, which is a great challenge for a filter that only employs the Bayesian weights. The main focus is to demonstrate the applicability in such a large-scale system in order to assess the practical suitability of the new method. Before discussing the results, a comprehensive description of the experimental setup is given in order to fully understand the characteristics of the assimilation problem.

6.1 Prerequisites

6.1.1 Motivation

The performance of the NETF in the experiments of Chapter 5 consistently showed a favorable behavior, even in systems of larger dimensionality. Therefore, the next step consists in an application to a more realistic model that exhibits a dimensionality typical for geophysical systems. In the following, we motivate why such an experiment is an important contribution to current research, and why a large-scale ocean general circulation model (OGCM) represents a suitable system of interest.

6.1.1.1 High-dimensional nonlinear filtering

State estimation in high-dimensional environments, such as circulation systems, represents a huge challenge to assimilation algorithms. The main constraint is limited computational power, as already one single forward integration requires a large amount of resources and is usually parallelized on supercomputers. Variational algorithms require the repeated evaluations of model, tangent linear model and adjoint model in the minimization procedure. In ensemble-based algorithms, a multiple number of

forward integrations have to be performed. As of today, EnKF-based algorithms are the only practical techniques in the latter case. They work well with relatively small ensemble sizes (in the order of about hundred), given sufficient tuning of the filter details (Houtekamer et al. 2014). The linearity assumptions contained in the EnKFs is objectionable from a theoretical point of view, but it renders the filter quite robust for small ensemble sizes. Nonlinear methods, in particular most PF-based schemes, are theoretically appealing, however, they are not applicable to large-dimensional systems with such ensemble sizes (van Leeuwen 2009). Therefore, the applicability of nonlinear DA techniques in GCMs is a topic of great interest. As already mentioned in Chapter 4, the EWPF currently appears to be a promising approach and has been successfully applied to a stochastically-forced single-layer primitive equation model (Ades and van Leeuwen 2014). In contrast, the NETF is a much simpler technique from an implementation point of view, and it does not require stochastic models. Thus, the investigation of its applicability represents an important contribution to the challenge of enabling nonlinear high-dimensional filtering, considering the curse of dimensionality.

6.1.1.2 Ocean data assimilation

As motivated in the beginning, one major field of application for DA methods concerns the generation of analyses, which in turn are used to initialize forecasts. This is particularly relevant for weather prediction, where the predictability is limited to a few days. However, also for medium range forecasts, which aim at climatological predictions, the initial conditions can be of relevance, particularly for the slowly-varying components of the earth system. They possess a long-term memory which can influence the model simulation over a long time scale. It has been shown that for medium-range predictions the ocean plays a key role, particularly the deep layers (e.g., Brandt et al. 2011; Schleussner et al. 2014). While its dynamics is often neglected in weather predictions (using externally specified ocean surface conditions), a dynamic ocean is important on the medium-range scale, and the ocean initialization significantly influences seasonal and decadal predictions (e.g., Alves et al. 2004; Müller et al. 2012). Of course, another important application concerns the production of reanalyses (e.g., Balmaseda et al. 2013; Köhl 2014). They allow a homogenous reconstruction of the ocean state over many years, which is important for climatological investigations. Nowadays, climate predictions or hindcasts are often performed with coupled atmosphere-ocean models due to the relevance of the ocean on longer time scales, and its proper initialization is an issue of active research (Polkova et al. 2014). Therefore, ocean DA is an issue of interest for a broad field of research and users (e.g., Bennett 2002; Lermusiaux 2006), also in the atmospheric community.

6.1.2 Aims

The principal aim of this chapter is to investigate the applicability of the NETF in a high-dimensional and chaotic circulation model. These models are widely used in atmospheric and oceanographic forecasting, where DA represents a key aspect.

More specific, a filter's applicability is given if it successfully responds to the observational information such that it reduces the deviation from the truth consistently. In particular, it must not diverge during a sufficiently long assimilation window. Filter divergence, the major challenge for PF-based techniques,

means that the ensemble is not capable of reproducing the truth. Typically, its spread becomes too small and ceases to be a reliable measure of uncertainty. Consequently, the filter is not capable of responding to the observations and generating a reliable state estimate anymore. In this case, the filter can only reproduce the model climatology. In an OSSE, filter divergency can usually be diagnosed via the deviation from the truth which is of the same order as for the model climatology.

According to the definition, the analysis quality will be assessed by ensemble diagnostics and by an evaluation with respect to the truth. Some related questions will also be considered to gain more insight into the NETF's behavior:

- Can the NETF be used in the same generic form as presented in Chapter 4, or are further enhancements required?
- How successful is the application in quantitative terms? The answer requires to interpret the evolution of the analysis error.
- How strong are the requirements on applicability with respect to filter settings, in particular, ensemble size and localization?
- How do the performance properties compare with the ETKF as robust, established method?
- Are there particular sensitivities or aspects that influence the NETF's applicability?

In the following, the experimental setup to meet the aims and answer these questions is described.

6.2 The ocean model and its setup

For a successful application of DA algorithms, it is important to gain sufficient knowledge about the model at hand and its characteristics. First, the OGCM used for the assimilation experiment is briefly characterized. Next, we describe the specific model setup and its properties. This allows to appreciate the complexity of the assimilation problem that is constructed in this chapter.

The experiment applies a software framework for ocean modeling, the *Nucleus for European Modelling of the Ocean* (NEMO, Madec 2008) in version 3.3. In particular, the framework contains a primitive equation OGCM, which is based on the "océan parallélisé" model (OPA, Madec et al. 1998). The framework allows further extensions, for example, the inclusion of sea ice, biochemistry and other tracers besides temperature and salinity, or the usage of tangent linear and adjoint models. However, in this application, exclusively the primitive equation model is employed. NEMO is a state-of-the-art model and is used by several hundred researchers and institutions, focussing on various aspects of oceanographic research. It can also be applied in regional mode. A further strength is an interface that enables to couple NEMO to models of other components of the earth system, such as the atmosphere, aiming at a better simulation of the exchanged energy, momentum and mass fluxes (e.g., Akthar et al. 2014; Pham et al. 2014).

6.2.1 Characterization of NEMO

The following description gives a concise overview of the principal model characteristics. This naturally represents a rather simplified and limited point of view. For details, the reader is referred to the extensive NEMO reference book (Madec 2008), which also serves as basis for the presentation given here.

6.2.1.1 Primitive equations

The NEMO ocean engine solves the primitive equations that specify the ocean dynamics and thermodynamics in a numerical way. In principle, the primitive equations for the ocean are the Navier-Stokes equations, which represent Newton's law in fluid dynamics and describe the momentum tendencies, including the Coriolis force due to the earth's rotation. They are enhanced by conservation laws, so-called continuity equations, for fluid mass, salt content (salinity) and heat. This system of equations closely resembles the primitive equations used to model the earth's atmosphere, where one usually finds water vapor content, the atmospheric analog to salinity, and pressure or fluid density as prognostic variables besides the three-dimensional velocity field and temperature (e.g., Kalnay 2003).

In ocean modeling, some typical assumptions are imposed to simplify the equations (e.g., Griffies 2004). The hydrostatic hypothesis converts the vertical momentum equation into a diagnostic equation, relating pressure gradient and density. The Boussinesq hypothesis allows to neglect density fluctuations expect in the buoyancy force term. Additionally, water can safely be regarded as incompressible, and as a consequence, the velocity divergence always vanishes, which easily allows to diagnose the vertical velocity. The fluid density can be diagnosed from pressure, temperature and salinity via an equation of state. Therefore, the basic prognostic variables in NEMO are the horizontal velocity fields (U, V), temperature (T) and salinity (S). All fields depend on the three space coordinates and on time. Furthermore, in modern OGCMs, a free surface formulation is used, allowing a variable sea elevation. It can be described by a two-dimensional variable, the sea surface height (SSH), which is the time-dependent deviation from a fixed reference height. Its prognostic equation arises from a kinematic boundary condition for momentum. Then, high-frequent external gravity waves are also solution of the set of equations, which has to be taken into account by the time stepping scheme. In NEMO, they can be considered by a time-splitting formulation or filtered by an additional damping term in the momentum equation that conserves the slow barotropic Rossby waves (Roullet and Madec 2000).

6.2.1.2 Physical parameterizations

As in the atmosphere, small-scale features occurring on a sub-grid scale, such as turbulent fluxes or convection, have to be parameterized based on the large-scale fields in order to close the system of equations. An example is the diffusion of temperature and salinity which occurs on a molecular scale. They affect the large-scale circulation, but cannot be resolved by typical grid sizes. Due to the asymmetry induced by gravity, NEMO distinguishes between lateral and vertical physical parameterizations. In the vertical, the diffusive fluxes are derived by a second-order closure using the gradients of the large-scale fields together with associated eddy coefficients. The latter can either be set to a constant or computed in dependency on the current flow, e.g., using the Richardson number or even a full turbulent

closure model. As the hydrostatic hypotheses removes convective processes from the model solutions, they have to be parameterized as well in case the stratification is unstable. Finally, the velocity in the deepest layer is dragged by bottom friction, and the friction coefficient is either fixed or flow-dependent. In the horizontal, the eddies arising from baroclinic instabilities, inducing mesoscale turbulences, can be resolved explicitly if the grid resolution is sufficient. In the eddy-resolving case, NEMO offers various operators of different accuracy (in particular, second-order, also Laplacian, or fourth-order, also biharmonic or bi-Laplacian, schemes) to parameterize the lateral diffusion processes.

6.2.1.3 Boundary conditions

As the primitive equations are partial differential equations, appropriate conditions have to be specified at all domain boundaries at every time step. This concerns the momentum, mass and energy fluxes at the ocean interfaces to the bottom and the atmosphere as well as to the lateral boundaries. At the bottom, usually all fluxes are assumed to vanish. The upper boundary condition is expressed by short and long wave radiative fluxes, sensible and latent heat as well as the freshwater budget. The latter determines ocean salinity and volume and is given by precipitation and evaporation. Additionally, the horizontal components of the surface ocean stress are required in order to determine the exchange of momentum at the surface. These boundary conditions can directly be fed into the model, however, most of them are typically derived from atmospheric fields (such as air temperature, surface pressure, humidity, precipitation and wind speed) using adequate bulk formulations. As third option, the surface boundary conditions can also be specified by analytical formulas, allowing simplified investigations.
At lateral solid boundaries, such as coast lines, all mass and energy fluxes are set to zero, as well as the normal component of the velocity field. Its tangential component depends on the choice of a so-called slip boundary condition. River runoff can also be considered as a further source of freshwater influx. In order to facilitate regional NEMO applications, the user may also apply open boundary conditions within the ocean itself, using external boundary fields.

6.2.1.4 Space and time discretization

The discretization of the primitive equations in space is performed via second-order central finite differences on a curvilinear three-dimensional Arakawa C-type grid. Scalar fields, such as T and SSH, are evaluated at the grid center, while the discrete representations of the velocity fields (U, V, W) are defined on the grid interfaces. The vertical discretization has to consider the bathymetry, i.e., the bottom topography. Here, either a homogenous height coordinate (z) or a terrain-following height coordinate can be chosen, or a combination of both. At the top, the sea surface height quantifies the time-dependent elevation of the ocean.

NEMO applies a leap-frog time stepping scheme for all non-diffusive terms. As a three-level scheme, it contains one computational mode besides the physical one. The numerical noise can be suppressed by a time diffusion as offered by the Robert-Asselin time filter (Asselin 1972). However, as it degrades the second-order accuracy of the time integration (Williams 2009), a conservative variant of the filter is implemented (Leclair and Madec 2009). For the lateral diffusive and damping parts of the equations, a

forward time scheme is used. It is conditionally stable, depending on the choice of the mixing coefficients. In the vertical direction either a time-splitting technique or the implicit, unconditionally stable backward scheme is applied, owing to the comparably small grid distances ($\Delta z \ll \Delta x$).

6.2.1.5 Conclusion

In summary, NEMO represents a circulation model of high complexity that contains a state-of-the-art representation of ocean dynamics and physics. Atmospheric models for weather and climate prediction in principle solve the same system of equations and have to deal with similar challenges, for instance, regarding the sub-grid scale physical parameterizations (e.g., Achatz et al. 2013; Baldauf et al. 2011). Therefore, NEMO represents a suitable model to investigate whether the new filter possesses an applicability in nonlinear, high-dimensional settings as typical for weather and ocean predictions.

6.2.2 Model setup

It remains to define an explicit model setup that serves the aims of the chapter.

6.2.2.1 Double-gyre square basin configuration

The NEMO package (version 3.3) already contains some pre-defined configurations to guide new users through the orientation phase. Considering the aim of this chapter, we need a model setup that employs the nonlinear primitive equations which induce a chaotic flow. However, it should still be kept as simple as possible regarding the boundary conditions in order to focus on the challenge of large-scale DA rather than dealing with issues related to the model setup. Such questions are interesting as well from the DA point of view, but would interfere with the aim to investigate the principal filter properties. Therefore, we chose to work with a configuration family named "GYRE". It applies the NEMO model in a closed square basin using a simplified, analytical forcing formulation. It is designed to idealize the ocean circulation that is representative for the mid-latitudes, e.g., the Gulf stream in the North Atlantic. The forcing leads to a large-scale double gyre circulation which is enhanced by mesoscale eddies. The original GYRE configuration is based on the work of Lévy et al. (2010), who focus on the influence of parameterization schemes by employing a sub-mesoscale resolution. They apply seasonally varying forcing of heat and freshwater flux, solar radiation and wind stress. As described by Cosme et al. (2010), their setup can be further simplified by using constant wind forcing only, which still leads to the chaotic double-gyre circulation, but renders the circulation characteristics invariant in time, given sufficient spin-up. This type of configuration has been an established reference in oceanography for over 50 years, and has been proven to exhibit realistic circulation patterns (e.g., Carrier and Robinson 1962; Holland 1978; Provost and Verron 1987). We conclude that the wind-driven double gyre configuration constitutes a quasi-realistic full primitive equation system together with its complex physics. It can therefore be regarded as suitable environment to assess the NETF's performance in a nonlinear, high-dimensional setting. After describing the specific model settings, we will briefly discuss the principal characteristics of the flow.

6.2.2.2 Domain

The NEMO model is applied in a closed, rectangular basin in the North Atlantic, ranging from $-60°$W to $-30°$W and $24°$N to $44°$N. The horizontal grid employs a distance of $0.25°$. This resolution corresponds to an eddy-permitting setup. This discretization results in $n_x = 121$ grid points in zonal and $n_y = 81$ grid points in meridional direction, respectively. In the vertical, $n_z = 11$ layers with exponentially increasing thickness are defined by a generic z coordinate, see also Figure 6.2b. The first layer has a thickness of about 300 meters. No bathymetry needs to be specified, as the bottom is assumed to be flat and located in a depth of 5054 meters.

6.2.2.3 Settings for physics and dynamics

The model settings refer to the description in section 6.2.1, with much more details available in Madec (2008). For the time integration, the leap-frog scheme is used with a time step of $15\,min$. The augmented Robert-Asselin time filter applies a smoothing parameter of 0.1. External gravity waves are damped explicitly in the horizontal momentum equations according to Roullet and Madec (2000). As typical for the simplified GYRE configuration, the space variation of the lateral eddy coefficients is constrained to the horizontal, while the vertical ones are fixed. Here, the default values, $1.2 \cdot 10^{-4}\,m^2 s^{-1}$ and $1.2 \cdot 10^{-5}\,m^2 s^{-1}$ for momentum and temperature, respectively, are set. The parameterization of lateral mixing is realized by a biharmonic diffusion operator with an eddy coefficient of $-8 \cdot 10^{10}\,m^4 s^{-1}$ for both momentum and temperature. At the bottom, linear friction with a drag coefficient of $4 \cdot 10^{-4}\,ms^{-1}$ is prescribed, while the lateral boundaries are assumed to be frictionless.

6.2.2.4 Boundary conditions

As the NEMO configuration is applied in a closed domain, the bottom and the lateral boundaries act as fixed, solid walls by setting the corresponding field entries of all prognostic variables to zero (Madec 2008). No freshwater influx is considered. At the upper boundary, SSH is computed by a free surface formulation in its linearized form (Roullet and Madec 2000), as the typical elevation values are of order $1\,m$, which is much smaller than the first layer's thickness. At the surface, the heat and freshwater fluxes are also set to zero. Therefore, salinity, usually a prognostic variable in an ocean model, always remains constant and is disregarded in all following considerations. The observed ocean circulation, described below, is entirely forced by zonal wind. It is given by an analytical prescription of the zonal wind stress τ_x (in Ns^{-2}) that varies with latitude ϕ, but is constant in time t and for each longitude λ:

$$\tau_x(\phi, \lambda, t) = -\frac{1}{10} \cos\left(\frac{2\pi}{\Delta\phi}(\phi - \phi_1)\right) \quad [Ns^{-2}] \tag{6.1}$$

Here, $\phi_1 = 24°$ is the latitude at the southern boundary and $\Delta\phi = 20°$ is the latitude range of the domain. The stress is symmetric with respect to $\phi = 34°$ in the domain center, where the zonal wind directs to the East ($29° \leq \phi \leq 39°$). In the northern and southern part, the wind blows to the west.

6.2.3 Initial conditions and truth run

Having specified the model configuration and its setup, it remains to actually create a continuous state trajectory, which can be used for the assimilation experiments.

The model is initialized with an ocean at rest, i.e., the velocity fields are equal to zero at the beginning. Salinity is prescribed to a value of $35.5\,g\,kg^{-1}$ and remains constant. Each vertical ocean column is initialized by the same analytical temperature profile (in $^\circ C$), following Chassignet and Gent (1991).

$$T(x,y,z,t=0) = 25 + 24.05708\left(e^{-z/800m} - 1\right) \quad [^\circ C] \qquad (z \text{ in } m) \tag{6.2}$$

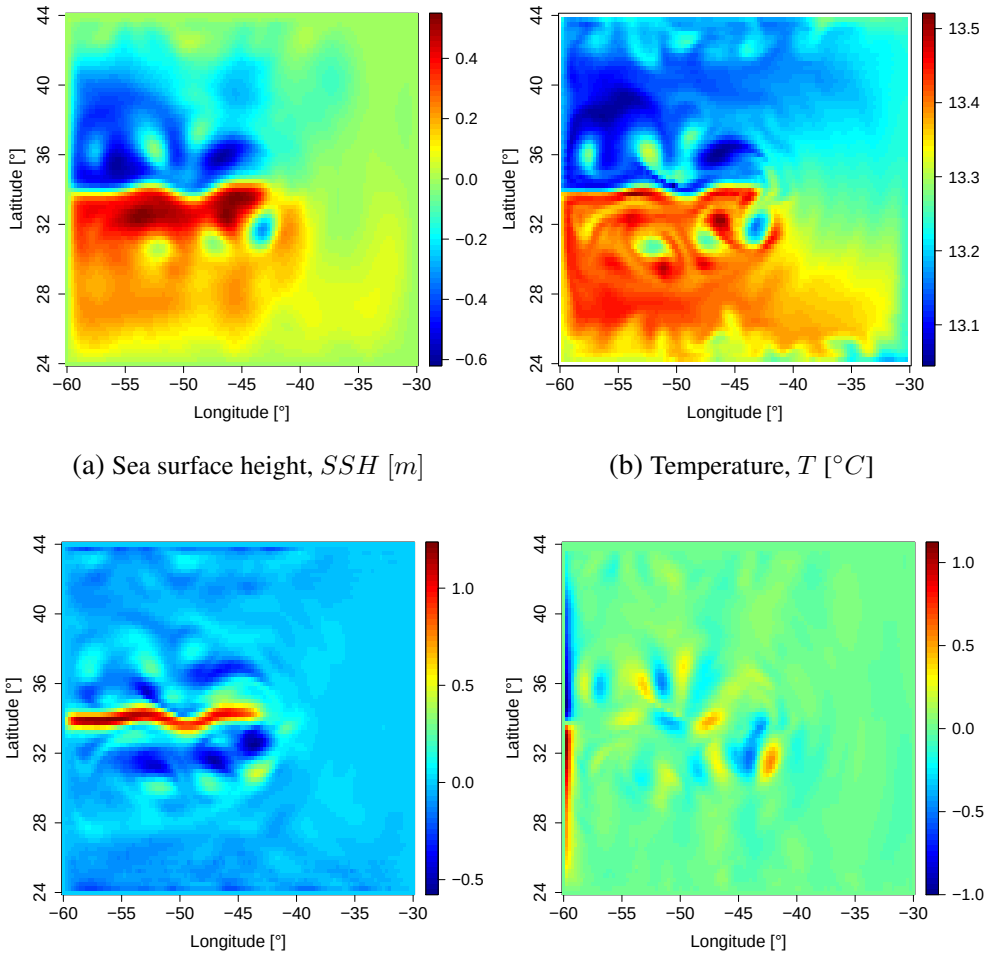
which corresponds to an exponential decrease of temperature with depth, a stratification typically observed in ocean climatologies.

After initialization, the model is integrated forward for 75 years. Here, one year is idealized to 360 days. The first 50 years are considered as spin-up phase towards reaching the model climatology. Afterwards, it is safe to state that a dynamic equilibrium of the model with respect to the applied forcing is reached (Cosme et al. 2010), which may for instance be diagnosed by the stable position of the central jet, as discussed below. The initial 50 years are disregarded. The actual DA experiment is performed in year 75, while the years 51-74 are utilized to estimate the model climatology, see also section 6.4.3.

6.2.4 Circulation characteristics

Before turning to the actual assimilation problem, it is useful to gain an overview of the dynamical structure of the wind-driven ocean. For this purpose, we refer to the true initial state in year 75. Figure 6.1 shows the surface fields. The large-scale double-gyre circulation, which intensifies at the western boundary, is directly visible in the SSH and T fields, Figures 6.1a and 6.1b, respectively. It is induced by the inhomogeneous zonal wind forcing described by equation (6.1). In the West, it leads to boundary currents, see Figure 6.1d. They support an eastward jet in the center, located around $\phi = 34^\circ$ in Figure 6.1c. However, the jet is subject to dynamic instabilities arising from the intrinsic properties of the strongly-nonlinear primitive equations. In consequence, a chaotic behavior can be observed that leads to a mesoscale flow dynamics besides the large-scale gyre circulation. It is characterized by eddies, which exhibit notable local differences in velocity (around $1\,m/s$), SSH (up to $1\,m$) and T (up to $0.5\,K$), and influence the large-scale flow (Holland 1978). Their horizontal range extends an order of about $100\,km$. These general circulation characteristics are similar to the ocean flow at the mid-latitudes, for example, in the Gulf stream in the North Atlantic (e.g., Kamenkovich et al. 2009; Thompson and Schmitz 1989; Willebrand et al. 2001).

This presentation only provides a rough sketch of the flow characteristics, as a more detailed investigation is beyond the scope of the work. For example, the influence of the eddies on the gyre circulation is an extensive field of research. The interested reader will find a large number of references in the oceanographic literature (e.g., Hecht and Smith 2008; Lévy et al. 2010; Schlax and Chelton 2008; Shen et al. 1999). Furthermore, the exact details of the model dynamics are not of primary relevance for



(a) Sea surface height, SSH $[m]$

(b) Temperature, T $[°C]$

(c) Zonal velocity, U $[m/s]$

(d) Meridional velocity, V $[m/s]$

FIGURE 6.1: Snapshots of the prognostic fields at the beginning of year 75 at the surface (SSH) or in the first layer (T, U, V), respectively. They visualize the double-gyre circulation with a central jet, meridional currents at the Western boundary and mesoscale eddies.

the investigation of the filter applicability. Concerning this aim, it is important to realize that the chosen configuration leads to a large-scale circulation pattern with chaotic mesoscale features that can be observed in the real ocean or atmosphere as well.

6.3 Description of the assimilation problem

So far, we have shown the model setup and the type of dynamics it generates. In order to finalize an experimental setup to which the NETF can be exposed, it remains to create an observation scenario that serves our purposes. Even though the DA experiment presented here utilizes artificial observations drawn from the truth run, the intention is to create a situation that resembles a typical ocean assimilation problem. Therefore, we refer to previous assimilation experiments that have already been performed within the NEMO-GYRE configuration described above. Cosme et al. (2010) investigated the performance of a square root smoother based on the KF, while Yan et al. (2014) compared different EnKF assimilation techniques. Both studies employed a sufficiently realistic observation scenario.

6.3.1 Observation scenario

Two distinct observation types of different spatial structure and density are considered. The observations are assimilated each second day, i.e., 192 model time steps define one analysis cycle. As the truth run covers one idealized year, 180 analysis steps are performed in total. The observations were provided by Y. Yan (University of Liège) and were also used in Yan et al. (2014). In the following, we give an overview to their background. This knowledge is not only useful to understand the assimilation characteristics, but it is also required for the implementation of the observation operators. The latter could be employed to reproduce new observations. However, taking a fixed observation data set better mimics a real assimilation situation, where only the filter settings, see section 6.4, can be modified.

6.3.1.1 Sea surface height

The first observation type takes into account that satellites are able to measure surface characteristics. Polar-orbiting satellites regularly re-visit the area of interest and generate measurements along the so-called tracks. Thus, these observations are rather sparse in space and time. In this context, *Envisat*, maintained by the European Space Agency (ESA), was a well-known satellite, active from 2002 to 2012. It flew at a height of $800\,km$ and exhibited a repeat cycle of 35 days in which 1002 passes were performed. Its primary objective was to supply remote sensing data concerning the Earth's environment, including atmosphere, land and ocean. For that purpose, the satellite carried ten different instruments. A radar altimeter allowed, among other functionalities, to estimate the sea surface height (ESA 2007). Validations with buoy measurements indicated the reliability of the retrievals (e.g., Durrant et al. 2009).

Here, the *Envisat* SSH observations are mimiced. First, for each assimilation window of two days, all satellite track locations falling into the model domain are extracted from the flight parameters of the year 2009. On average, about 150 observations can be considered per analysis time. Then, the true SSH field is interpolated to these locations via bilinear interpolation. In conjunction, this interpolation serves as the observation operator, as it relates the model variables to observation space. The actual observations were created by adding uncorrelated Gaussian noise with standard deviation of $0.03\,cm$ to the "true" observation, corresponding to the typical *Envisat* measurement error. However, in order to account for the additional representativity error introduced by the interpolation, the observation error matrix is chosen to be diagonal with a variance of $(0.06\,cm)^2$. In summary, this procedure ensures to obtain synthetic SSH observations with realistic spatial distribution and typical error statistics. The main simplification is given by the fact that we assume all observations to be valid at the analysis time, while in reality they are distributed in time as well. However, this approach has no influence on the principal filter performance, which is the main objective here, and is therefore justified from an algorithmic point of view.

6.3.1.2 Ocean temperature

SSH observations only concern the surface. Hence, other state variables, especially in the deeper layers, could only be updated through cross-correlations with SSH, but would not be constrained directly. Additionally, keeping the aim of this chapter in mind, it is useful to consider more observations.

This does not only create a situation which is more similar to weather prediction, but also renders the assimilation problem considerably more challenging for a filter such as the NETF, which is only based on the Bayesian weights. They are evaluated in observation space, and the observation dimension is mainly responsible for PF divergency (van Leeuwen 2009). Fortunately, even though observations for the atmosphere are more dense in space, also in the ocean an observation type exists which is able to increase the observed region significantly.

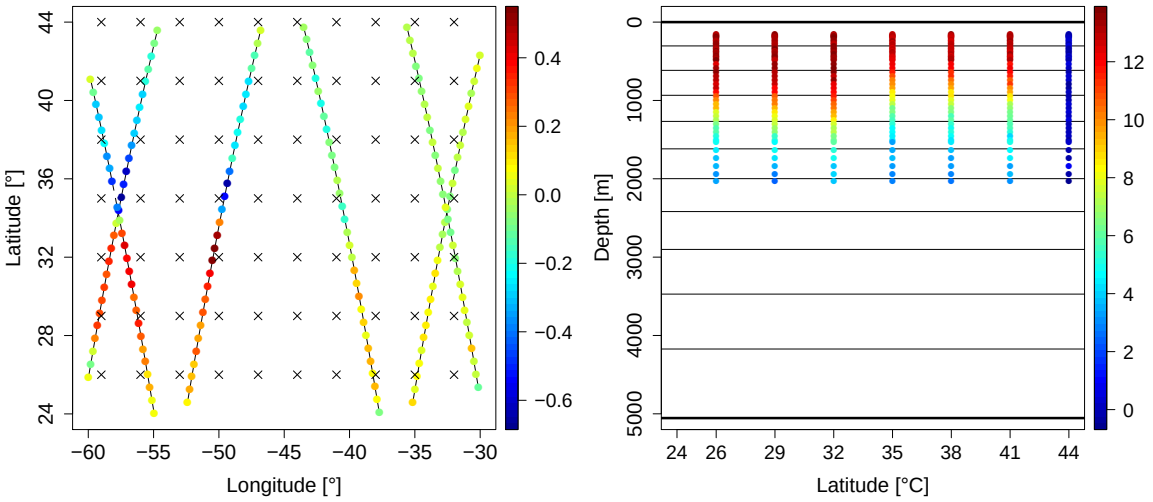
Vertical information can be gained with the help of profiling floats that are equipped with sensors to measure variables such as temperature, salinity or pressure up to a depth of about $2000\,m$. The best known large-scale realization is the *Argo* network, a global ocean observation system designed for research and climate monitoring (Carval et al. 2013), with currently about 3600 floats. The measurements, which are freely available (`http://www.argo.ucsd.edu`), have already been used in more than 1500 scientific publications and are also utilized operationally by weather centers as a valuable data source. For example, in ECWMF ocean analyses for seasonal forecasting and in ocean re-analyses the *Argo* data have a notable impact (Balmaseda et al. 2007, 2013).

The procedure to generate the synthetic temperature observations is very similar as described for SSH. The observations are taken to be valid at the analysis time, i.e., every other day. A further simplification compared to the real *Argo* network is that we neglect lateral movements of the profilers, but again, this has no principal influence on the filter performance and keeps the implementation effort manageable. It is partly considered by using a regular $3^\circ \times 3^\circ$ grid horizontally, and this grid is shifted by 1° at each analysis time to reflect the non-stationarity. The true model temperature is mapped to the *Argo* observation grid via trilinear interpolation, which serves as the observation operator. Then, the observations are simulated by adding uncorrelated Gaussian noise with a standard deviation of $0.3\,K$ to the interpolated field at each observation location. For temperature, no representativity error is assumed since the interpolation introduces only minor errors due to the relatively small spatial gradients in the temperature fields (Yan et al. 2014). Vertically, 46 observation levels are considered, which correspond to the *Argo* profiles and reach to a depth of about $2000\,m$. The *Argo* levels above the first layer's center ($150\,m$ depth) are discarded because their observation operator would require an extrapolation of the model field. Furthermore, using more of these highly dense observations near the surface would increase the risk of overfitting in the relatively thick first layer ($300\,m$) of this idealized NEMO configuration. In real applications, the vertical grid resolution near the surface is typically much finer, allowing to consider the dense near-surface observations as well.

Figure 6.2 visualizes the observation characteristics on day 8, i.e., at the fourth analysis step. Figure 6.2a is a horizontal snapshot, showing both the *Envisat* locations and the *Argo* network used at this time. The SSH observations along the satellite tracks are indicated by their color. Figure 6.2b gives an example of the temperature profiles (46 values each) along the $\lambda = -50^\circ$ line and their location in the vertical NEMO grid.

6.3.2 Discussion

The global state vector consists of all prognostic variables of NEMO except salinity, i.e., T, U, V (three-dimensional fields) and SSH (two-dimensional). Therefore, the global state dimensionality is



(a) SSH $[m]$ (b) T $[°C]$

FIGURE 6.2: Observation characteristics on day 8: (a) The horizontal domain is shown, together with the *Argo* profiler locations (crosses) and the synthetic SSH observations (colored) on the *Envisat* tracks (thin lines). (b) The vertical grid of 11 layers is visualized, and embedded are the artificial *Argo* temperature profiles along the $\lambda = -50°$ longitude line. Note that at $\phi = 44°$, the true temperature field is zero due to the lateral boundary conditions.

$3 \cdot (121 \cdot 81 \cdot 11) + 121 \cdot 81 = 333234$, which is several orders of magnitude larger than in the experiments conducted in Chapter 5. Furthermore, NEMO is not an idealized "toy model". It solves the same set of complex physical equations that drive atmospheric and oceanic systems. Therefore, it is suited to assess the NETF's performance in a nonlinear, high-dimensional and more realistic environment.

One may compare this experiment to the setup used by Ades and van Leeuwen (2014) (AvL14 in the following), who applied the EWPF to a primitive equation model. They also used a simplified, wind-driven ocean system, but with reduced gravity as they focussed on the possible generation of gravity waves. The principal differences to our setup are as follows:

- The NEMO model used here is entirely deterministic, i.e., no stochastic model errors are added. The EWPF does not work in a deterministic system.
- The ocean in AvL14 has a single layer and thus, the state dimensionality is about one order of magnitude smaller than in our three-dimensional experiment. Furthermore, AvL14 confine to the momentum equations, while here the full set of the primitive equations is active, including the thermodynamical ones, which allows to use temperature observations as well. Additionally, the observation scenario is more realistic.
- As will be elaborated in section 6.4.3, the experiment shown here is initialized by a climatological ensemble, while AvL14 perturb the true initial state.

6.4 Assimilation setup

This section describes the application of the filter to the introduced assimilation scenario.

6.4.1 Technical implementation

The experiments shown in Chapter 5 were conducted in a newly-created assimilation framework, written in `R`.[1] This environment is convenient and clearly-arranged for DA problems of small and intermediate dimensions. However, with complex, large-scale models, the assimilation procedure should be performed in a more efficient programming language. The NEMO code is written in `FORTRAN`.

A major advantage of ensemble-based filter techniques is the fact that they can be separated into forecast and analysis steps. The latter are entirely generic and independent of the model at hand, and the model itself does not have to be modified at all. Therefore, it is recommendable to refer to a predefined environment that contains optimized implementations of the analysis algorithms. Then, the remaining task for the user is to implement the interface to the model, along with model-specific settings. The best known framework is the DA research test bed (DART, Anderson et al. 2009). An alternative is the Parallel Data Assimilation Framework (PDAF, Nerger and Hiller 2013), developed at AWI Bremerhaven (Nerger et al. 2005). This tool was chosen for three reasons: (1) It offers a more natural access to the filter core routines, which is useful for implementing a new filter, (2) the author of this thesis has already gained experiences with PDAF in the development of a land surface DA system, and (3) the spatial adjacency facilitates the support and has already led to successful collaborations.

The standard PDAF package[2] contains implementations of analysis routines for the EnKF, ETKF, SEIK and SEEK, with focus on computational efficiency. Additionally, the ensemble subspace transform Kalman filter (ESTKF, Nerger et al. 2012b), an efficient unification of ETKF and SEIK, is included. All deterministic filter variants can be used in conjunction with domain localization. PDAF relies on parallelization via Message Passing Interface (MPI, MPI Forum 2012). There are two options to create an assimilation environment with PDAF. First, it can be used in an offline mode. Then, the model writes the forecast ensemble to an output file, which is read by an individual PDAF program. It transforms the former into an analysis ensemble, which is then put back into the model to initialize the next forecast step. Alternatively, the calls to the PDAF interfaces can be implemented directly into the model time stepping routine. Even though this variant requires a slight modification of the model code, it avoids the input/output writings, as they can strongly degrade the computational efficiency for large-scale models.

6.4.1.1 Filter implementation

The NETF was implemented into PDAF in collaboration with P. Kirchgessner (AWI Bremerhaven). For reasons of efficiency, mainly LAPACK routines were used to carry out the matrix operations. It is planned to include the NETF and its localized variant in an upcoming official PDAF release. In principal, a fairly small amount of work was necessary by copying the LETKF routines first. In PDAF, the LETKF is organized by an *"update"* routine. It first maps the global ensemble states into observation space. Then, a loop over all local domains is performed. For each local domain, the local forecast ensemble is extracted, together with the localized observation. Next, the filter core routine, *"analysis"*, is called that transforms the ensemble into a local analysis ensemble, using the corresponding localized

[1] It is envisaged to publish the filter core routines as R package in future.

[2] PDAF version 1.10, released in October 2013, freely available from `http://pdaf.awi.de/`.

observation. Therefore, the implementation of the NETF only required to write a new *"analysis"* routine, following the algorithm presented in section 4.8.1. All operations to select the local ensembles and observations do not change and hence, the *"update"* routine remains as for the LETKF. Additionally, the random rotation is performed by default. In order to use a nudged NETF, see section 4.9, it is sufficient to read the prior weights from the nudging step into the *"analysis"* routine.

The implementation was thoroughly verified. First, we externalized the *"analysis"* routine and compared the output with the `R` function, given identical input data. Then, it was tested in Lorenz 96 experiments, where the performance was in good agreement with the results presented in Chapter 5. These experiences confirm that the NETF can be directly implemented into an existing DA system that employs domain localization.

6.4.1.2 Analysis step: user routines

Even though the ensemble transformation itself is entirely generic, some user routines had to be written in order to adapt the generic variables (such as the ensemble matrices $\mathbf{X}$ and $\mathbf{Y}$ or observation vector $\mathbf{y}$) to the specific assimilation problem. The following list summarizes these routines, which also gives an overview of the technical organization of the analysis step. The ensemble transform itself, as just described, is located at step 4.

1. Observations: The current observations (SSH and T) have to be read from files and put into an observation vector $\mathbf{y}$.
2. Observation operator: The connection of the observation and state vector has to be specified. Here, either a bilinear (SSH) or trilinear interpolation (T) of the corresponding model field is performed.
3. Localization: The local state has to be extracted from the global state vector. Additionally, the observations considered for a local domain are extracted from $\mathbf{y}$, using the predefined localization radius (see also below).
4. Observation error: During the analysis step, the (localized) matrix $\mathbf{R}$ is required. It is implicitly formulated by the result when being multiplied by another matrix or vector, which occurs in both NETF and ETKF. Here, also the weighting of observations by their distance is performed.
5. Globalization: The global state is re-constructed from the local analysis vectors.
6. Diagnostics: In order to monitor the performance, the computation of ensemble spread and RMSE with respect to the truth (see below) are directly implemented into PDAF.
7. Output: The ensemble, its mean state and the diagnostics are written to a NetCDF file.

These user routines have mainly been prepared by P. Kirchgessner (AWI Bremerhaven) during previous work with NEMO. Some routines were revised or enhanced in mutual work.

The analysis step is carried out serially for the ease of this experimental implementation. If more efficiency is required, it can be parallelized as well. The local domains could be partitioned such that

each processor computes the local updates for one region, also known as domain decomposition. A different option is to use OpenMP to parallelize the loop over the local domains directly, which is included in PDAF version 1.11 (Dec. 2014).

6.4.1.3 Forecast step: coupling to NEMO

The forecast step simply requires to run NEMO for each of the N ensemble members, initialized by the analysis fields produced by the PDAF routines. It is easily parallelized by MPI, as the ensemble integration consists of independent processes by definition. Thus, each process performs an individual NEMO integration. Communication is only required at the beginning and at the end of the forecast step. The model itself is not parallelized, but for larger domains, an additional domain decomposition could be applied during the forecast step as well.

In our experiment, NEMO is coupled to PDAF in online mode. The PDAF library, which contains the filter core routines, is linked to the program at compilation time. The following modifications in the NEMO source code were required:

1. Initialization of MPI and its communicators.
2. Initialization of PDAF, defining the assimilation setup and its variables. Most of them are read via namelist parameters. Additionally, for each process, the model fields are overwritten by the initial ensemble member to be used for the assimilation experiment.
3. Assimilation: At the end of each model time step, the routine *"assimilate"* is called. If an observation is available at the current time step, the following actions are performed:
   (a) Ensemble collection: The forecast states are vectors which have to be constructed from the multidimensional model fields (SSH, T, U, V). This has to be done for each member, and then, all ensemble states are gathered on one processor core.
   (b) Analysis: While the analysis is computed, the other processors remain idle.
   (c) Ensemble distribution: For each member, the model fields are reconstructed from the analysis state vector. Then, each processor core receives one model state such that it can proceed with the forward integration.

6.4.2 Localization and inflation

As discussed in section 3.4, ensemble-based filters show suboptimal behavior related to the necessity of using rather small ensemble sizes in large-scale systems. For the EnKFs, this is usually counteracted by localization and inflation techniques. As demonstrated in chapters 4 and 5, these considerations are valid similarly for the NETF.

Domain localization is applied as outlined in Chapter 4. Here, a local domain is defined by all state variables in a single, vertical ocean column. Hence, the local state dimensionality is always $3n_z + 1 = 34$ for each of the $n_x \cdot n_y = 9801$ local domains. Localization is therefore only performed in

Statistics	only SSH	only T	combined
global observation dimension	145	3128	3273
maximum local observation dimension	16	184	199
average local observation dimension	7	90	94
number of local domains with observations	6262	9654	9742
number of local domains without observations	3539	147	59

TABLE 6.1: Information about the observations and localization. All statistical values are temporal averages over the 180 analysis steps. The total number of local domains is 9801, and this table is valid for a localization radius of $2.5°$.

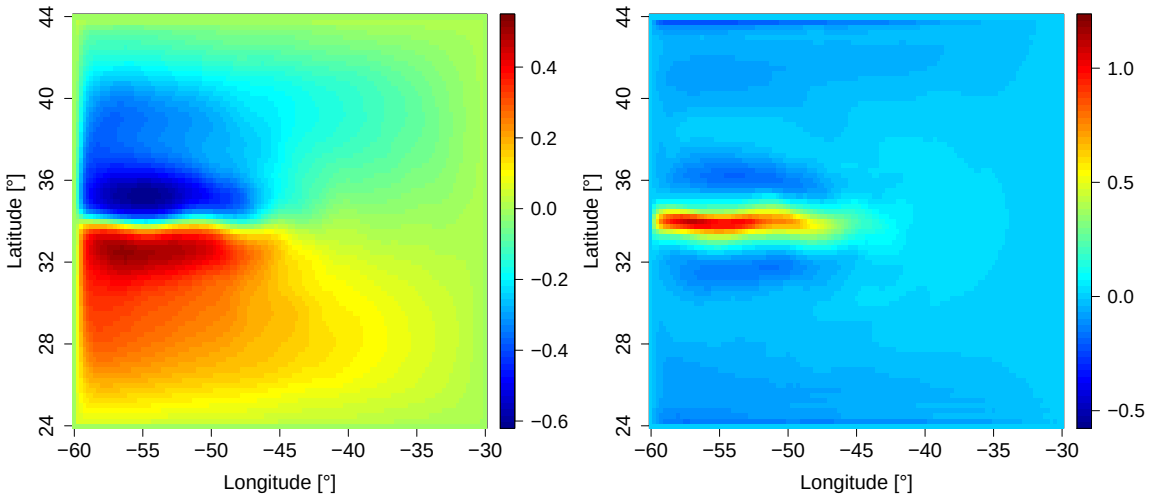
the horizontal directions. During one analysis step, each ocean column is updated separately using only observations within the localization radius. As in chapter 5, we apply the fifth-order polynomial correlation function (Gaspari and Cohn 1999) to reduce the influence of distant observations. It should be noted that the NETF weights become spatially dependent, but as they are only used intermediately, localization represents no challenge for the NETF, in contrast to PFs in general. A localization radius of $2.5°$ is set, roughly corresponding to $250\,km$. Thus, all observations within circle of $5°$ diameter around the current local domain are considered for the local analysis. This value arises from experiences in prior LETKF experiments in this setting (P. Kirchgessner, personal communication) and is in agreement with the setup of Cosme et al. (2010). Furthermore, it is stated that the spatial correlation values of SSH becomes close to zero already at a distance of less than $200\,km$ (Yan 2013, Fig. 3.4). Therefore, the chosen localization radius is also consistent with the statistical properties of the system that arise from the physical model equations.

Table 6.1 provides some statistics of the observation dimensions from a global and local point of view. On average, at each analysis time 3273 observations, by the majority temperature data, are assimilated. The average local observation dimension is nearly 100, but it may reach 200, and almost all local domains have access to observations. In other words, the likelihood weights are evaluated in a 100-dimensional subspace, on average. Even though this is significantly smaller than the global observation dimension, it is still huge considering that the curse of dimensionality demands ensemble sizes of $10^6 - 10^{11}$ in such dimensions (Snyder et al. 2008). The region of significant probability mass is extremely small, and hence, filter divergency can only be avoided if all particles are sufficiently close to the truth. Additionally, the weight evaluation is performed 9801 times at each analysis step. The fact that the filter does not diverge, in spite of about only hundred members, confirms that the generation of a new ensemble at each analysis step works very well for the NETF.

Concerning inflation, we restrict to a factor γ which is fixed in time and space, as for the experiments in Chapter 5. At each analysis step, it is applied to the prior ensemble. As discussed in section 4.7.2, it could be used for the analysis ensemble as well. For the main experiment[3], γ is set to $1/0.975 \approx 1.025$. It should be kept in mind that inflation constitutes the main tuning factor in an ensemble-based DA system (Houtekamer et al. 2005; Sakov and Oke 2008b).

Using a fixed localization radius and inflation factor corresponds to the most simple techniques. Of course, more sophisticated approaches are conceivable, as discussed in section 3.4. Even though most of them have been developed for EnKFs, they could be applied to the NETF as well due to its similar

[3]PDAF uses the *forgetting factor*, the inverse of the inflation factor γ (Nerger et al. 2012b).



(a) SSH $[m]$ (b) U $[m/s]$

FIGURE 6.3: Ensemble mean state at initial time, for (a) SSH and (b) U (first layer). The mesoscale circulations average out over the initial ensemble. For comparison, the true initial fields are shown in Figure 6.1 (with identical color legend).

update formalism. Adaptive methods are capable of considering flow-dependent and inhomogeneous localization radii and inflation factors. For example, in this NEMO application, it would already be more realistic to apply different inflation factors for the different variables. One might also choose different localization radii for the two observation types, or localization functions that are not isotropic. However, the aim of this study is not to perform an extensive tuning to gain the best result possible, but rather to demonstrate the principal applicability of the basic algorithm. Regarding this aim, the restriction to the basic techniques is not only sufficient, but also more constructive, as the results reflect the pure filter behavior instead of the tuning efforts. In a potential future real-world application of the NETF, it is recommendable to further enhance the performance by using more advanced approaches.

6.4.3 Initial ensemble

Any ensemble-based filter requires an initialization at the beginning of the assimilation window. While this is a rather trivial task in small test models, it becomes an important issue in large-scale problems, similarly to the specification of the background error covariance matrix in variational techniques. In principle, the initial ensemble should represent the best knowledge about the system at the initial time t_0 and its error covariance structure. Furthermore, in large-scale circulation models it should be taken care that the ensemble states are balanced in order to avoid an initial shock at the beginning, or even the generation of gravity waves (Ades and van Leeuwen 2014). Therefore, the initial ensemble is generated in a realistic way. Within the years 51-74, i.e., after spin-up and prior to the experiment year, a state is sampled every other month. This results in a collection of 143 states, considered as a climatological sample of the system. Its covariance matrix contains the dominant variances and error directions.

The initial ensemble is generated with second-order exact sampling to gain an optimal low-rank representation of the climatological error covariance; see Appendix C for more details on the procedure. The

initial members contain mesoscale features. However, their mean state only represents the large-scale double-gyre circulation and the central jet, as visualized in Figure 6.3. Thus, this choice of an initial ensemble represents a challenge for the DA system, as only very limited flow-dependent information is available at the beginning. In real applications, usually more prior information than climatology is available, for example from the output of previous analysis cycles, which might already be fairly close to the truth. However, we restrict to the more difficult case, since a useful filter should not only be able to keep track of the truth, but also to find it at first. Nevertheless, the true state should be within the range of a consistent initial ensemble. Therefore, we implemented the option to inflate the *initial* spread and used a factor of 1.25. The influence of the initial ensemble on the filter performance will be investigated in more detail in section 6.6.1.

The ensemble size is set to $N = 120$ due to the availability of six dual socket nodes with ten cores each. Thus, the ensemble size is of the same order as in the experiments of Chapter 5, while the state dimension is strongly increased. This allows to verify whether the ensemble size required by the NETF increases with dimension, as for the PF, or if it really scales better, as the results in Chapter 5 indicate. The good scaling is a major advantage of the EnKFs which deliver reliable results with roughly hundred members for a wide range of dimensions, i.e., for Lorenz to NWP models.

6.4.4 Summary of the experimental setup

Table 6.2 summarizes the filter settings and observation properties for a quick overview.

Setup variable	Value	Explanation / comment
filter	NETF	localized, with random rotation
model	NEMO	same as for truth run (deterministic)
state dimension K	333234	U, V, T, SSH on the $121 \times 81 \times 11$ grid
observations	SSH, T	*Envisat* and *Argo*, simulated from truth
observation operator $\mathcal{H}$	lin. interpol.	bi/tri-linear interpolation to observation grids
observations error variance	$(0.06m)^2$, $(0.3K)^2$	for SSH and T ; $\mathbf{R}$ is diagonal
observation dimension L	≈ 3273	slightly varies in time (see Sec. 6.5)
assimilation window	360 days	year 75 of the transient NEMO run
no. of analysis steps	180	every 2 days
ensemble size N	120	120 cores used in the forecast step
inflation factor γ	1.025	applied to the prior ensemble
no. of local domains	9801	each vertical ocean column
localization radius r_{loc}	2.5°	in horizontal direction (isotropic)
localization type	DL with OL	weighting by fifth-order polynomial
initial ensemble	climatological	second-order exact sampling from 143 model samples of years 51-74
initial inflation	1.25	for the initial covariance only

TABLE 6.2: Standard setup for the assimilation experiment. Not included are the NEMO model settings, which are described in section 6.2.2.

6.5 Results and discussion

Even though ensemble-based filter algorithms are in principle model-independent, their usage and the interpretation of their performance requires a thorough knowledge about the background of the assimilation problem, its design and the properties of the model at hand. This information was provided in the previous sections, and it now allows to focus on the actual results of the assimilation run.

6.5.1 Evaluation measures

The main evaluation criterium for the NETF's performance is, as in Chapter 5, the root mean square error (RMSE) of the ensemble mean at each analysis time level $j \in 0, 1, 2, \ldots, J = 180$. It is computed separately for each model variable according to

$$\mathrm{RMSE}_j(X) = \sqrt{\frac{\sum_{gp} \left(\overline{X_j} - X_{j,true}\right)^2}{\dim(X)}}$$

Here, X stands for each of the four different model fields (T, U, V, SSH) and $\overline{X_j} = \frac{1}{N}\sum_{n=1}^{N} X_j^n$ represents the ensemble mean field. $\dim(X)$ is the dimension of the field, i.e., 107811 for T, U, V and 9801 for SSH, and $\sum_{gp}$ indicates summation over all model grid points. Thus, the term within the square root equals the field-averaged squared error per grid point. The RMSE can be computed for both the forecast or analysis ensemble. Their difference is related to the analysis increments and allows to assess the impact of the current observations. The RMSE allows a general judgment of the analysis quality. As an overall measure, it may not capture all performance details, but a low RMSE represents a necessary condition for a successful analysis. Ensemble-based filters offer an *estimated* RMSE by the field-averaged ensemble spread,

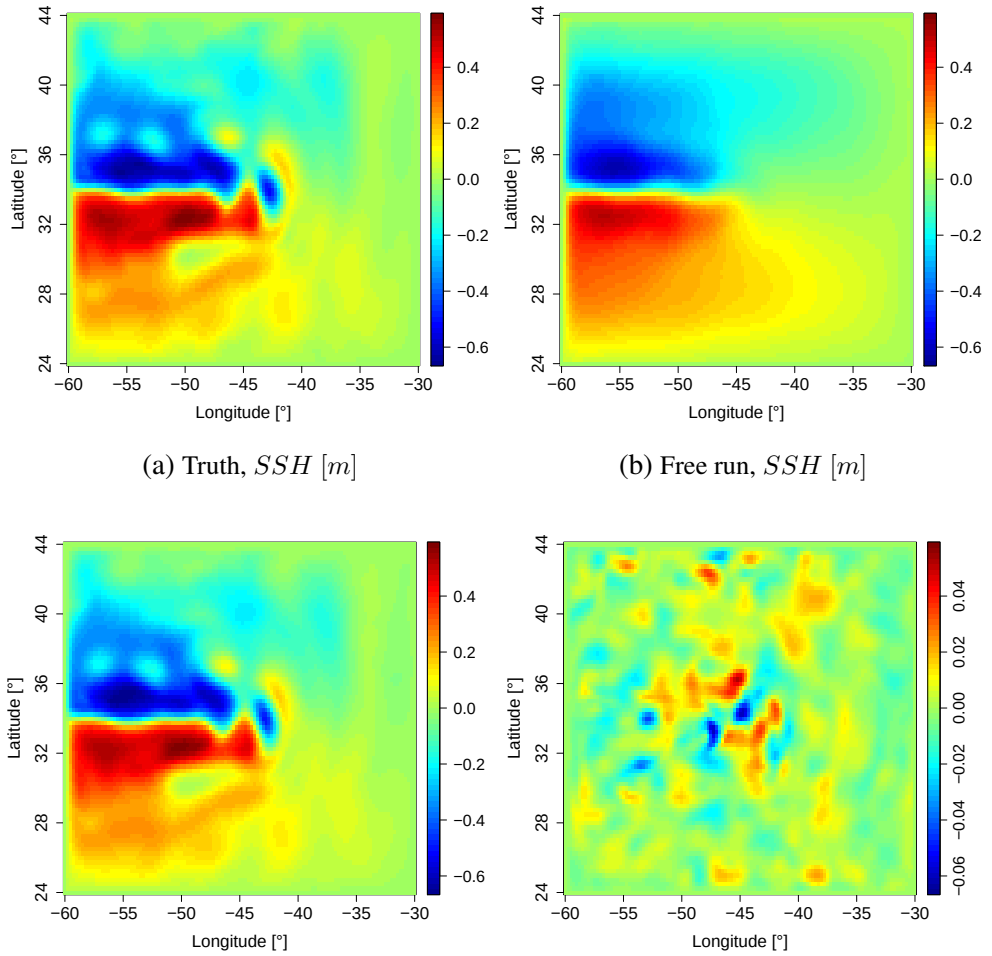
$$\mathrm{SPREAD}_j(X) = \sqrt{\frac{\sum_{gp} \left[\frac{1}{N-1}\sum_{n=1}^{N}(X_j^n - \overline{X_j})^2\right]}{\dim(X)}}$$

This measure can be evaluated independently from the truth. In a well-calibrated DA system, it should be of similar magnitude as the RMSE, since then the ensemble distribution allows a reliable diagnostics of the filter uncertainty (e.g., Hopson 2014).

For comparative reasons, it is useful to include the so-called *free run*. Here, the same initial ensemble as used in the DA experiments is integrated throughout the assimilation window, except that it does not consider the observations. With this design, the free run constitutes a reference that allows to reveal the impact of assimilating the observational data into the model.

6.5.2 Qualitative evaluation

As discussed in section 6.4.3, the initial ensemble contains no specific information about the true flow at the initial time. Thus, without additional constraints, the free ensemble can only deliver climatological information of the system during the forecast phase, i.e., the large-scale double-gyre circulation



(a) Truth, SSH $[m]$

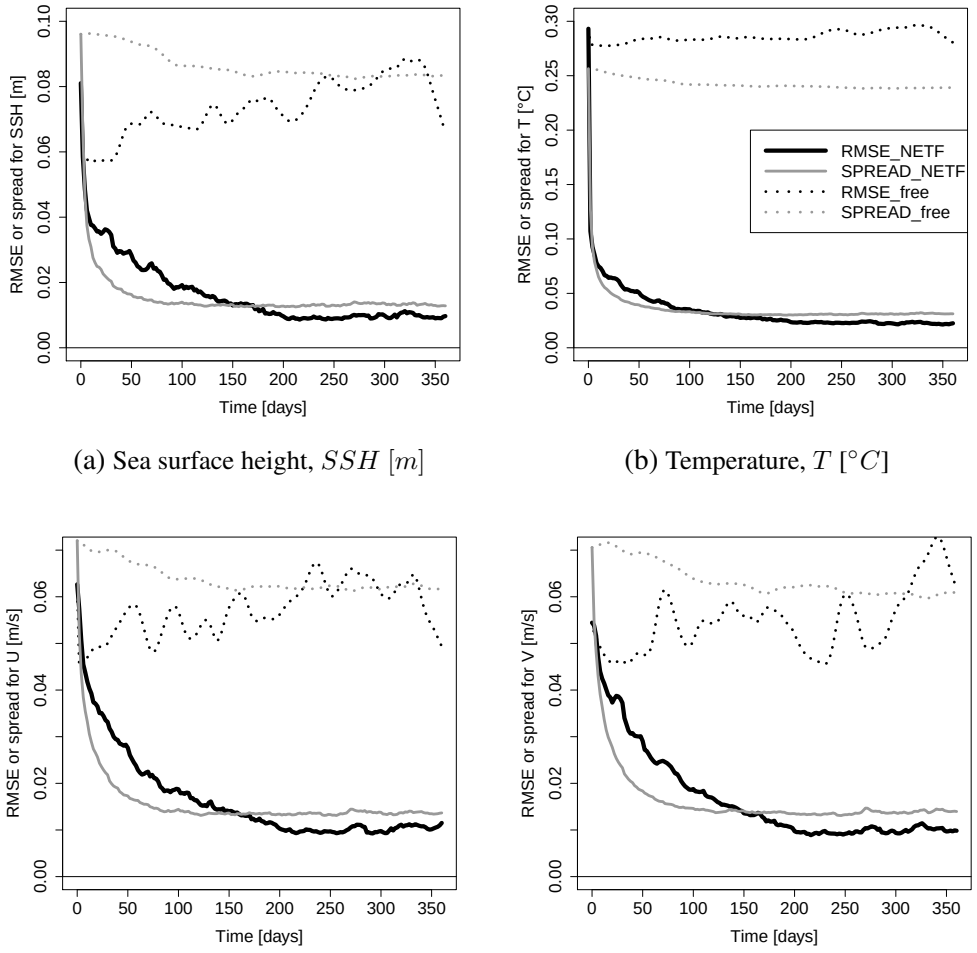
(b) Free run, SSH $[m]$

(c) NETF, SSH $[m]$

(d) NETF - truth, SSH $[m]$

FIGURE 6.4: Results for SSH at day 260. Shown are the fields from (a) truth, (b) free run (ensemble mean), and (c) NETF analysis ensemble mean. The field in (d) is the difference between analysis and truth. Note that in (d) the color scale is enlarged by a factor of 10, as otherwise analysis and truth can be barely distinguished.

created by the wind forcing. However, it is not able to resolve the mesoscale patterns, which are of chaotic nature and average out over the free ensemble. These expectations are confirmed by Figure 6.4, showing the SSH fields at day 260. In principal, the free ensemble mean, Figure 6.4b, has not changed compared to the initial time, revealed by a comparison with Figure 6.3a. In contrast, the NETF analysis at that time, Figure 6.4c, yields an estimate that closely resembles the true circulation including the mesoscale features, as shown in Figure 6.4a. The differences, plotted in Figure 6.4d, are only of minor magnitude and mainly of statistical nature. The filter is close to perfect tracking, only in the most active region around $\lambda = -45°$ and $\phi = 34°$ small mismatches in the magnitude of the mesoscale perturbations are visible. These qualitative findings are similarly valid for all other variables as well, including the unobserved velocity fields. They already confirm that the NETF works successfully in this high-dimensional scenario because it is able to reproduce the truth.



(a) Sea surface height, SSH $[m]$

(b) Temperature, T $[°C]$

(c) Zonal velocity, U $[m/s]$

(d) Meridional velocity, V $[m/s]$

FIGURE 6.5: Temporal evolution of the evaluation measures for the NETF (solid lines) and free run (dotted lines) for all control variables, i.e., (a) SSH, (b) T, (c) U and (d) V. While the RMSE is drawn in black, the spread is in gray. The legend in (b) is valid for all panels.

6.5.3 Quantitative evaluation

Having proven the applicability of the NETF in a qualitative way, it remains to strengthen the results from a quantitative point of view. Figure 6.5 shows the temporal evolution of the analysis RMSE, together with the average ensemble spread, for each control variable. The corresponding results for the free run are also added as reference. As already revealed, the free ensemble can only output a climatological estimate and does not improve upon the initial time. The fact that its error and spread remain of similar magnitude confirms that the initial ensemble is a good representation of the model climatology. Its error and spread measures for the dynamical variables (U, V, SSH) exhibit a higher variability due to the sub-scale variability of the true flow. The RMSEs of the NETF clearly indicate that it works successfully, as its analyses only exhibit small errors. This reflects the insight that the NETF is able to resolve the mesoscale patterns of the true flow. For all variables, the initial error is reduced nearly monotonously until convergency is reached. This spin-up phase is explained by the fact that the initial ensemble does not contain information about the mesoscale circulation (Yang et al. 2012b) and will be discussed shortly after. Beyond spin-up, the error remains approximately constant, which shows

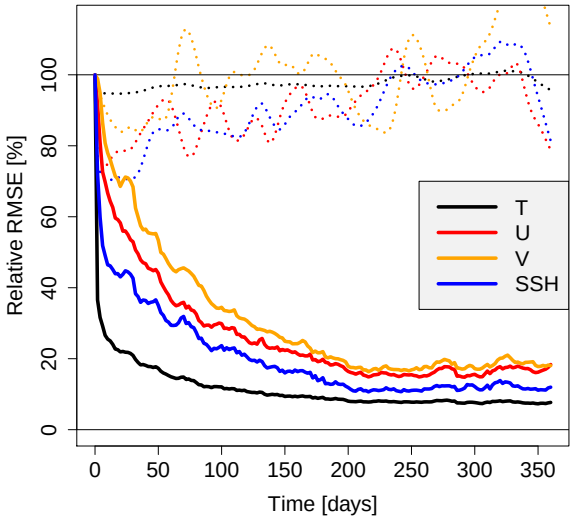


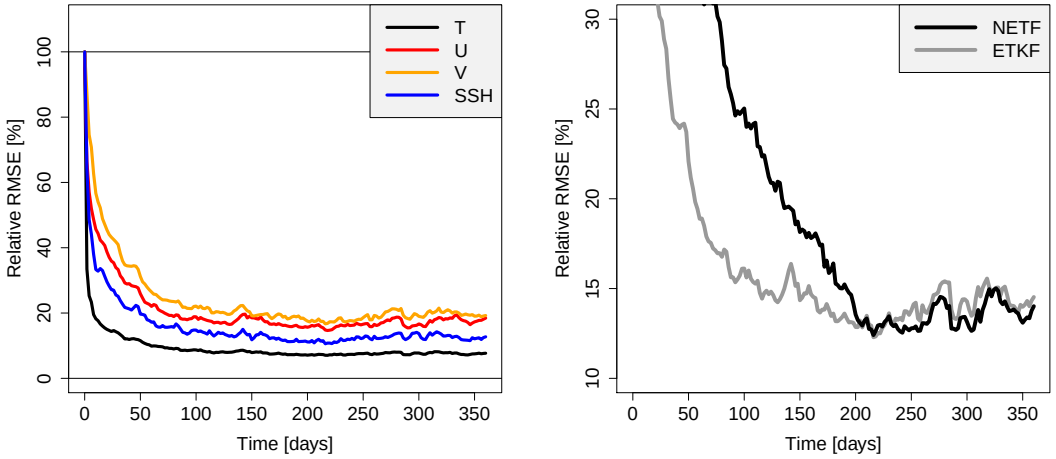
FIGURE 6.6: Relative error plot. The solid lines represents the relative error (RMSE normalized by its initial value) of NETF analysis for T (black), U (red), V (orange) and SSH (blue). The dotted lines refer to the free run.

that the filter ensemble is in a quasi-stationary balance constrained by the observations. This is further confirmed by the ensemble spread. After the spin-up phase, it is also nearly constant, and is typically slightly larger than the RMSE. Therefore, the ensemble distributions are also consistent with the truth in a statistical sense. This confirms that the inflation factor is well chosen. If it was too large, the RMSE would increase again after the spin-up phase. In contrast, too little inflation would increase the spin-up time, or even cause the filter to collapse.

For a better comparison of the performances concerning the error reduction, the RMSEs are also normalized by their values at the initial time (day 0), at which they are equivalent for the NETF and free run. Figure 6.6 depicts the temporal evolution of these *relative errors*. For temperature, the strongest error reduction is observed. This is easily explained by the fact that at least the upper temperature is constrained by the majority of the observations. Nevertheless, a minimal relative error of only about 7.5% can be achieved, indicating that the deeper layers are affected positively as well. This will be a topic for further exploration in section 6.6. SSH, also an observed variable, achieves the second-best minimal relative error, with about 10.6% beyond spin-up. Even though the velocity fields are not observed, they can also be estimated well, with minimal relative errors of 14.8% and 16.3%, respectively. U exhibits slightly smaller errors, presumably because it is, at least close to the surface, constrained by the fixed wind forcing. The good results for U and V confirm that the filter is able to generate correct analysis increments for all variables. It is emphasized that, in the NETF, the increments are directly based on the Bayesian weights which in turn are evaluated from the high-dimensional probability distributions.

6.5.4 Comparison to the ETKF

Even though the NETF produces consistent and appealing filter results, the spin-up phase appears to take about 200 days, as visible in Figure 6.6. This is also revealed by a comparison with the ETKF, which we applied with the same initial ensemble and setup. The ETKF results for the relative errors are shown



(a) ETKF (b) ETKF vs. NETF

FIGURE 6.7: Results for the ETKF. (a) Relative analysis errors as in Fig. 6.6, but for the ETKF. (b) The relative error, averaged over all prognostic variables, is shown for the NETF (black) and ETKF (gray).

in Figure 6.7a. The filter results towards the end of the assimilation window (days 200-360) are very similar for ETKF and NETF. Figure 6.7b compares the relative RMSEs, averaged over all four variables. It confirms that the ETKF requires only slightly more than hundred days to reach convergency. Its advantage in the spin-up phase is particularly apparent for the less constrained dynamical variables U, V and SSH. However, towards the end of the year, the NETF performs slightly better. The difference is not really significant, but at least consistent, which might be a hint that the nonlinear approach could offer some minor benefits. However, in this setup probably not much improvement over the ETKF is possible, as least not without more elaborate tuning, as discussed in section 6.4.2, because the general flow is strongly constrained by the fixed wind forcing.

The extended spin-up time is explained by revisiting Figure 6.5. During the spin-up phase, the spread decreases faster than the RMSE for all variables. The ensembles exhibit too little spread, and therefore, the NETF analyses do not consider the observations sufficiently. In principle, this could be resolved by increasing the inflation factor. However, once convergency is reached, this would be detrimental to the filter and the RMSE would rise again. Hence, an adaptive inflation might be helpful to provide more inflation in the beginning, but less in the convergency phase. For the ETKF, more elaborate methods to accelerate the spin-up have been proposed (Kalnay and Yang 2010). From a more general point of view, these findings also indicate that the initial ensemble is potentially of great relevance for the NETF. Therefore, its influence will be further investigated in section 6.6.1.

In summary, the quantitative analysis of the overall performance of the NETF reveals that the new method is actually applicable for nonlinear, high-dimensional DA, despite the fact that the assimilation is entirely based on the non-parametric Bayes' theorem. The NETF is not only able to perform a stable analysis in a high-dimensional problem, but its performance is also in good accordance with the principal filter behavior one would expect for the given experimental setup. Its analyses offer a strong error reduction which shows that the observations are used efficiently, and it is potentially able to improve the estimates of the ETKF, an established method. These results have been obtained with the generic NETF as derived in Chapter 4, no further adaptions are required in this large-scale application.

Statistics	**Climatology**	$T\ [°C]$	$SSH\ [m]$	$U\ [m/s]$	$V\ [m/s]$
RMSE	years 65-74	0.3336	0.0775	0.0641	0.0546
	years 51-74	0.2931	0.0810	0.0627	0.0544
spread	years 65-74	0.1151	0.0662	0.0527	0.0524
	years 51-74	0.2056	0.0772	0.0583	0.0571
bias	years 65-74	0.0613	-0.0008	0.0001	0.0000
	years 51-74	-0.0141	0.0001	0.0001	0.0000

TABLE 6.3: Basic properties of the climatological samples which are used to generate the initial ensemble, evaluated for each individual variable. RMSE and bias are computed for the climatological mean state with respect to the true initial state in year 75.

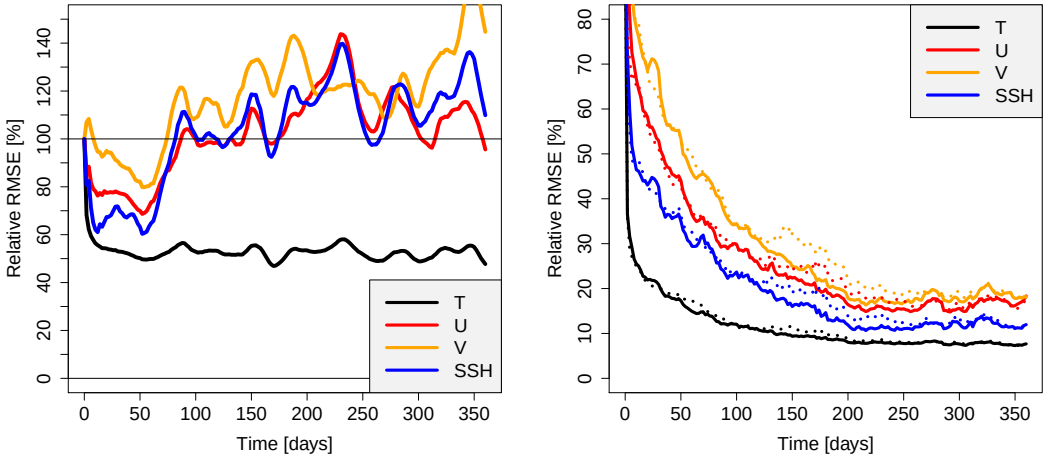
6.6 Investigation of selected aspects

In the previous section, it was demonstrated that the NETF is applicable to a large-scale DA problem. Thus, the principal aim of this chapter could be accomplished. This section highlights some additional aspects, aiming at a deepened view on the filter behavior.

6.6.1 Filter initialization

The main experiment in section 6.5 has revealed that the NETF yields good results, but requires a relatively long spin-up time. An explanation is that the NETF is presumably more sensitive to the specification of the initial ensemble than the ETKF. This can be understood well by the fact that the NETF update is entirely based on the Bayesian weights. Given the climatological initialization here, at the first analysis step many members are located far away from the observation. Therefore, many weights are insignificant and the new second-order statistics can only be estimated from a few members with significant weights, which degrades the quality of these estimates. Consequently, the analysis ensemble will be generated with these poor estimates, which decelerates the convergency rate as the increments are only small. To verify this hypothesis, the average *effective ensemble size*, $\left(\sum_n (w^n)^2\right)^{-1}$, is computed. It equals N if all members have equal weight and functions as descriptive measure of the variance of the weights (Stordal et al. 2011). In the first analysis step, the effective ensemble size is only about twenty, while it increases to about hundred later on. These numbers support the theory that in the beginning most members do not contribute much information. Thus, it takes a rather long time to equilibrate the ensemble. These theoretical considerations reveal that the NETF is relatively sensitive to the initial ensemble. The findings are further supported by the following observations:

1. In a first test case, the climatology of only the prior ten years has been used to generate the initial ensemble. In this case, the NETF does not work and shows signs of filter divergency. It exhibits small spreads, but the RMSEs are not much improved compared to the initial time, see Figure 6.8a, expect for temperature, where a slight impact is visible. A basic analysis of the initial ensemble, as shown in Table 6.3, reveals the problem. The deviations from the true initial state, as measured by the RMSE, are of similar magnitude for both climatologies. However, in comparison, the spread in temperature of the 10-years-climatology is very small, only about one third of the RMSE. Hence, the initial ensemble is distributed in a way that it does not properly cover the



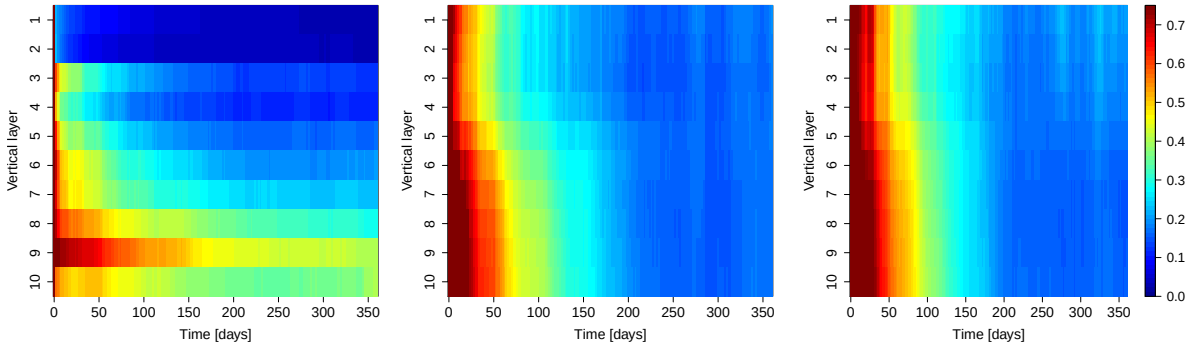
(a) Using the 10-years climatology (b) Using the 24-years climatology

FIGURE 6.8: Experiments on filter initialization for the NETF. Shown are always the relative RMSEs, as in Fig. 6.6 for all variables. (a) If the 10-years climatology is used to create the initial ensemble, the filter diverges. (b) Initialization with the 24-years climatology. Here, the solid lines refer to an initial inflation of 1.25, and the dotted to 1.0.

truth, which is confirmed by the bias. For the other variables, the situation is less problematic, but the weights are computed in observation space, which is mainly populated by temperature. Therefore, at the first analysis step, most particles receive insignificant relative weights, which explains the failure of the filter in this case. In contrast, the 24-years-climatology exhibits more consistent statistics. Even though the spread is still too small, its difference to the RMSE is less pronounced, and the bias is removed. Using this climatology, the NETF works properly as shown in section 6.5. This confirms the importance of the choice of the initial ensemble, as already discussed from a theoretical point of view.

2. Based on the insight that the initial ensemble might be too narrow, the initial inflation is a simple, yet practical way to increase the initial spread which allows more members to be in the range of the observations. This can help to obtain more significant weights in the first steps. Without initial inflation, the NETF still works, but the spin-up time is slightly increased, see Figure 6.8b. Inflation cannot be used to overcome the long spin-up phase in principal. However, in other cases, the initial inflation might more important to avoid divergency, see also section 6.6.4.

The EnKFs do not suffer from this issue as much due to their distinct update structure. The increments are determined by the Kalman gain, which itself is computed from ensemble perturbations alone. The actual positions of the members are not relevant. The adjustment of the ensemble members is proportional to the innovation, see equation (3.12). Thus, members that exhibit a large distance from the observation are pulled towards it more strongly. This is contrary to PF-based methods, where these members receive insignificant relative weights and are not able to contribute to the analysis.
It becomes clear that, in our experiment, the climatological initial ensemble represents a major challenge for the NETF. Nevertheless, it still is able to reconstruct the truth. In realistic situations, usually more initial information is available. For instance, often the analysis ensemble of the previous assimilation cycle can be utilized. It is usually fairly close to the truth and may include information about



(a) temperature, T (b) zonal velocity, U (c) meridional velocity, V

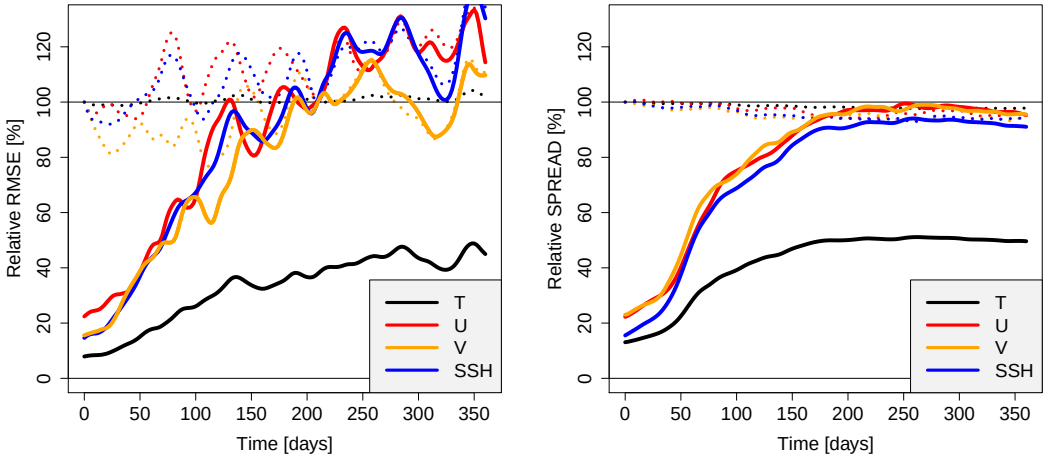
FIGURE 6.9: Temporal evolution of the relative RMSEs for each vertical layer for (a) T, (b) U and (c) V. At each time, the relative RMSE equals the RMSE divided by the corresponding RMSE at initial time. The legend bar in (c) is valid for all panels.

the mesoscale flow, which strongly reduces the spin-up phase (Kalnay and Yang 2010). This issue is expected to be of similar relevance for the EWPF. In the large-scale EWPF experiments published so far (Ades and van Leeuwen 2014; van Leeuwen and Ades 2013), the initial ensembles were generated by perturbing the true initial state. This strongly ensures reasonable weights in the first analysis steps. Thus, the initialization might represent a challenge for the EWPF not yet revealed. Furthermore, the true initial state is not available in realistic DA applications.

6.6.2 Layer-wise evaluation

As this chapter deals with the principal applicability of the NETF, the evaluation measures are usually averaged over the whole field for each variable. A multi-faced evaluation would be beyond the scope of this experiment, but it might be of interest to highlight one aspect. A major challenge in ocean DA concerns the estimation of deeper layers which exhibit a slow variability, particularly for temperature, and are usually unobserved, even if *Argo* floaters are present (Zhang and Rosati 2010). Therefore, a layer-wise evaluation is performed for T, U and V. Figure 6.9 shows the temporal evolution of the relative RMSEs for each layer. Here, the normalization of each RMSE time series has been performed by the corresponding value at the initial time. Thus, the relative RMSEs quantify the error reduction in each layer.

The results confirm the expectation that in principle (1) the spin-up time for the state estimation increases with depth, and (2) in the upper layers, the error can be reduced more strongly. Furthermore, the spin-up time depends on the variable. The upper layers of T show the quickest convergency rate. The analysis results for U are, at least in the upper layers, slightly better than for V, concerning both spin-up time and minimal relative error. This was already noted in section 6.5.3, and could be related to the surface forcing, which constrains the zonal velocity more than V. The mentioned difficulty in estimating the deep ocean temperature can be seen for the ninth T layer in Figure 6.9a. In general, deeper layers are expected to exhibit larger relative RMSEs. However, our NEMO configuration applies a closed boundary at the bottom which fixes the temperature in layer 11 (see section 6.2.2). This strongly constrains the energy flux into the tenth layer from below and explains the surprising observation that the tenth layer shows smaller RMSEs for T. More advanced configurations usually rely on a no-flux



(a) RMSE (b) SPREAD

FIGURE 6.10: Forecast skill in year 76, i.e., after the assimilation year, as given by (a) relative RMSE and (b) relative ensemble spread. The solid lines refer to the evolution of the final NETF analysis ensemble, while the dotted lines refer to the free ensemble. For each case and variable, the normalization is performed using the corresponding statistics of the free run at the initial time of the year.

boundary condition that allows varying bottom temperatures, which is more appropriate for real ocean DA applications.

In summary, the layered evaluation shows that the NETF is able to improve the estimates in the deeper layers, and the characteristics are as expected. It also corresponds to the ETKF results which only differ in the lengths of the spin-up times (not shown), as already discussed.

6.6.3 Impact on predictions

The results shown so far indicate that the assimilation of observations into the ocean model can lead to significant error reductions for all prognostic variables. Thus, past states can be estimated reliably, which is useful for example for re-analyses. As motivated in Chapter 1, the outcome of DA is often utilized to initialize forecasts, and it is expected that an optimized analysis also allows better predictions. Here, we briefly evaluate the usefulness of the NETF analysis from a forecasting point of view by looking at the predictability properties of the model. As the characteristics of the large-scale circulation are nearly stationary in our setup, the predictability limit does not vary much in time as well. Thus, an ensemble integration within the year 76 is performed. It is initialized by the final NETF analysis ensemble of year 75 (day 360). As reference, the free run is also extended to the year 76.

The results in terms of relative RMSEs are shown in Figure 6.10a. As expected, the RMSEs increase in time as the forecast is not constrained by observations anymore. This loss of forecast skill is particularly evident for the dynamical variables, i.e., U, V and SSH. After about 125 days, their relative errors exceed 80%, and after about 200 days, the skill is similar as for the free, climatological ensemble. In contrast, the thermodynamic variable, T, behaves differently. Its forecast skill degrades slower and does not reach the climatological no-skill level within this year. This can be related to the relatively slow variability of T, which we already noticed for the estimation of deeper layers in section 6.6.2, even

in comparison with U and V. In consequence, the medium-range climatology for T may significantly differ from a long-term climatology. This well-known property of ocean circulation (e.g., Palter et al. 2014; Schleussner et al. 2014) is also reflected in our simplified ocean configuration, as revealed by the differences in the 24-years- and 10-years-climatology, which were already discussed in section 6.6.1. Therefore, some predictive skill is preserved for T within the year 76. The temporal error evolution can further be used to obtain a quantitative estimate of predictability. For the first hundred days, the error e is modeled by an exponential growth, i.e., $e(t) \propto \exp\left(\frac{t}{t_d/\ln(2)}\right)$. Then, the error doubling time t_d can be derived empirically from the slope of a linear fit of its logarithm. This procedure yields values between 50 and 60 days for all variables.

The relative ensemble spreads, shown in Figure 6.10b, indicate the increasing uncertainty of the forecast, as they are principally consistent with the RMSEs. However, after the ensembles are equilibrated, their spreads appear to decrease very slightly. This is already visible for the free ensemble in year 75, see Figure 6.5. This marginal loss of spread might be reasoned in our NEMO configuration with a closed domain and fixed forcing. However, a proper identification and explanation of this issue would require to investigate an extended multi-year ensemble forecast, which is beyond the scope of this chapter.

In summary, even this simplified ocean system allows to conclude that an analysis transfers into predictive skill for slightly more than hundred days, concerning the dynamics. After that, the mesoscale patterns cannot be resolved anymore and the chaotic behavior is dominant. However, the improvement of the ocean's stratification, as given by temperature, positively affects the forecasts for much longer time scales, at least in comparison with the long-term climatology. In a wider context, this can be beneficial for climate predictions (Brandt et al. 2011; Schleussner et al. 2014), as ocean temperature determines the lower boundary condition for atmospheric models. These results underline the usefulness of the temperature observations in order to improve the ocean's energy budget (Kröger et al. 2012).

6.6.4 Other issues

Finally, we shortly state some additional findings that are not presented in detail as they are of minor relevance here. Nevertheless, they might be interesting to know with regard to future applications.

6.6.4.1 Ensemble size

In all experiments, an ensemble size of $N = 120$ yields good results. With 140 members, the filter behavior is quite similar and exhibits minor differences only, such as a slightly shortened spin-up phase. With hundred members, the filter still performs well, but it requires to increase the initial inflation to 1.5 to avoid divergency in the beginning. These findings confirm that the applicability in high dimensions is not restricted by an unrealistic ensemble size. Only about hundred members give good results in very different systems where the dimensions vary over many orders of magnitudes. At the current stage, it appears that the NETF's applicability is primarily determined by a consistent representation of the initial ensemble rather than by ensemble size.

6.6.4.2 Sparse observations

In another side experiment, only the SSH observations are assimilated. This intends to examine whether the NETF is capable of dealing with sparse observations that constrain the system only weakly, see Table 6.1. The NETF remains stable (with an increased inflation factor of 1.1) and exhibits relative RMSEs of roughly 30% (SSH) to 50 % (U, V), while for T ($> 80\%$) only very weak improvements are achieved. These results suggest that the observations are assimilated successfully, but their density is too low to sufficiently resolve the mesoscale patterns. Hence, this experimental setup probably does not allow the true flow to be tracked as well as with the additional profiler information. This interpretation is supported by the ETKF results, which are similar. These findings emphasize the importance of the vertical profiles to constrain the model, particularly concerning the ocean's stratification and heat budget, as also reflected in state-of-the-art ocean DA applications (Balmaseda et al. 2013).

6.6.4.3 Application of inflation

As mentioned in section 6.4.2, either the forecast or analysis ensemble can be inflated, and we have always restricted to prior inflation. An experiment with analysis inflation has been performed as well, and the resulting performance shows no significant differences to the main experiment. We conclude that the filter is not sensitive to the way inflation is implemented, probably because the increased spread of an inflated analysis ensemble yields a similarly inflated prior ensemble at the next analysis step. However, this conclusion might change in systems with different dynamics. For the ETKF, it is not yet clarified which inflation type is preferable in general (Rainwater and Hunt 2013). Anyway, inflation in the NETF should be applied to the global ensemble. In cases with sparse observations, the unobserved local domains should be inflated as well. Otherwise the local ensembles might exhibit too little spread when an observation is available at a future analysis step, and might fail to assimilate it correctly.

6.7 Final discussion and conclusions

In this chapter, the NETF was applied to a large-scale general circulation model. The empirical results clearly show that the NETF behaves stable in this scenario. Additionally, all results indicate a preferable filter behavior. Even if initialized with climatological information only, the NETF analysis converges to the truth and keeps track of it, including the mesoscale circulation patterns.

Therefore, the suggested filter is actually applicable in high-dimensional DA problems with a typical ensemble size of around hundred. This represents an important result in the context of nonlinear, high-dimensional filtering, which is a field of rapid progress at the current stage. Ades and van Leeuwen (2014) demonstrated that the EWPF can be applied in high-dimensional cases as well, using a nudged forecast step in a stochastic model. The results shown here demonstrate that even in a deterministic model a generic filter algorithm solely based on the Bayesian weights can still be successful. The implementation of the NETF is extremely simple and corresponds to changing a few lines of code in case an ETKF system is already available. In particular, no model-specific adjustments to the generic algorithm are required, and, as the ETKF, the filter is only tuned by inflation and localization.

The detailed evaluation of the NETF has further revealed that it performs reasonably well in the high-dimensional scenario. The deviation from the truth can be reduced strongly as compared to a free run, and once the filter has converged, RMSE and ensemble spread are in good agreement. These findings hold for all prognostic fields, also the unobserved ones. In comparison with the ETKF, the filter achieves a very similar performance beyond the spin-up phase, which is a remarkable result considering the simple formulation of the NETF. Probably, a major improvement over the ETKF is not possible in this scenario. We also found that the NETF is sensitive to the initial ensemble. If it is not properly specified, many members may obtain insignificant weights. This insight is important for further applications of PF-based methods, including the EWPF. Nevertheless, we demonstrated that even with very limited initial information the NETF is successful, even though it requires an extended spin-up time.

As the ensemble size has not been increased compared to the low- and medium-range applications in Chapter 5, it is safe to conclude that the NETF is able to overcome the curse of dimensionality. This is reasoned in (1) localization, which decreases the effective dimension and (2) the ensemble transform, which puts an increment to each ensemble member such that the ensemble is equally-weighted after the analysis. In contrast to the EWPF, the NETF does not attempt to consider the full analysis pdf, but only its second-order statistics. Of course, some limitations remain. The sensitivity to the initial ensemble shows that the applicability is not as unconditional as for the ETKF, which is a very robust method by construction. Furthermore, the localization radius should be chosen in dependence on the density of the observation network such that the local observation dimension is not too large. However, in our experiment the NETF is able to deal with a more than hundred-dimensional local observation space dimension, which already corresponds to a quite large number of local observations.

A side benefit of the experiments conducted here is that the results emphasize the usefulness of ocean DA, particularly if vertical profiles are available. It was illustrated that the analysis is able to improve subsequent predictions compared to the long-term climatology. This is of great relevance for seasonal or medium-range climate prediction, where the energy fluxes exchanged between ocean and atmosphere are an important model component (Schleussner et al. 2014). Particularly temperature exhibits a long-term memory, and its estimation shows the quickest convergency. For instance, temperature determines the ocean heat content, a crucial component for decadal predictions (e.g., Kröger et al. 2012; Müller et al. 2012), but also for climate diagnostics. Hence, the *Argo* profiles are expected to be highly beneficial for any ocean DA system.

The successful demonstration of large-scale applicability induces a variety of aspects that are worth to be investigated in the future. First, it would be interesting to compare the performance of the NETF and the EWPF, necessarily with a stochastic model. Then, the NETF could also make use of the nudging forecast step, as explained in section 4.9. This would allow to assess the impact of the more complex EWPF analysis step in contrast to the simple NETF analysis, as both output an equally-weighted ensemble. Second, it would be important to apply the NETF to another large-scale situation, perhaps a system where nonlinear developments have a stronger impact. In our scenario, the mesoscale disturbances do not alter the large-scale flow in principal, as it is constrained by the analytical forcing. In fully-realistic GCM applications, the large-scale flow is not stationary in time either. Then, the NETF might offer a stronger potential benefit over the ETKF. For instance, the results in this chapter motivate the application to an atmospheric system, which exhibits some additional challenges, such as higher spatial and temporal variability and a dense, heterogeneous observation network.

Chapter 7

Conclusions

The final chapter summarizes this thesis and the main conclusions drawn from the theoretical and empirical results. Even though conclusions were already discussed in the separate chapters, they are merged here and linked to the scientific aims stated in Chapter 1. Finally, an outlook is given to offer possible extensions of the research conducted within this work.

7.1 Summary and key conclusions

In principle, the thesis consists of three major parts, as outlined in Chapter 1. In the following, the main findings and conclusions are summarized.
The first part contains a review of data assimilation (DA) techniques, focussing on ensemble-based filters.

- A general DA framework has been formulated in probabilistic terms. It is emphasized that this point of view is able to capture all aspects inherent in DA, in particular, the uncertainties connected with models and data. All objective, explicit DA algorithms can be described within this framework, including the new filter. Sequential filters partition DA into an iteration of forecast and analysis steps, where the latter are determined by Bayes' theorem.
- The ensemble Kalman filter (EnKF, Evensen 1994) is based on the KF and proposes a nonlinear Monte Carlo solution to the forecast step in sequential algorithms. This idea renders the EnKF applicable even to large-scale problems with a limited ensemble size. Nowadays, the ensemble transform KF (ETKF, Hunt et al. 2007) is a standard tool in DA. It yields robust and reasonable results, if enhanced by localization and inflation. However, its analysis step relies on the KF equations which imply a Gaussian forecast pdf. Therefore, in nonlinear systems, its analysis remains suboptimal even though the forecast ensemble contains nonlinear features.
- The ultimate goal in DA is a nonlinear, non-Gaussian filter that makes no assumptions about the form of the prior pdf. The particle filter (PF, Gordon et al. 1993) in principle satisfies this request, but the required ensemble size increases exponentially with the dimension of the problem. Therefore, active research is performed to render nonlinear filters applicable to large-scale systems.

  - The equivalent weights PF (EWPF, van Leeuwen 2010) utilizes the great freedom that is offered by the introduction of proposal densities. They allow to draw particles closer to the observations during the forecast step and to generate an equally-weighted ensemble at the analysis step. In initial experiments, the filter proved to be applicable also in large-scale systems (Ades and van Leeuwen 2014). However, it requires a stochastic model and the algorithm is rather complex. For instance, the nudging step has to be adapted to the model at hand, which leads to additional computational expenses and currently relies on linear observation operators with Gaussian error statistics.
  - The nonlinear ensemble adjustment filter (NLEAF, Lei and Bickel 2011) adds a stochastic increment to each ensemble member at analysis time such that its mean and covariance approximately meet the Bayesian expectations. This approach is more generic and does not require model error terms. However, as for the original EnKF, the perturbed observations introduce additional stochastic errors, particularly if the ensemble size is relatively small as typical in large-scale situations. Additionally, the algorithm is computationally very expensive.

The second part builds on the insights gained from the review in the first part, and derives the new nonlinear ensemble transform filter (NETF). The NETF follows a similar concept as the NLEAF, but avoids its drawbacks. It generates a new analysis ensemble, focussing on the first two moments of the analysis pdf. The primary advantages with respect to other ensemble-based filters are as following:

- The generation of a new, equally-weighted ensemble increases filter stability compared to PFs, particularly, if augmented random rotations are used.
- The update mechanism is deterministic, i.e., the first-and second-order statistics of the analysis ensemble exactly match the nonlinear Bayesian expectations. Thus, the analysis step does not introduce biases or sampling errors. This also holds if an additional, moment-preserving random rotation is used. Resampling, as required in PFs, is avoided entirely.
- The algorithm updates the ensemble perturbations by $\mathbf{X}'_a = \sqrt{N}\,\mathbf{X}'_f\mathbf{T}$, which is formally identical to the ETKF. The only difference lies in the transform matrix $\mathbf{T}$ (see equation 4.21), which is derived from the fully-nonlinear Bayesian weights, avoiding a Gaussian assumption. Therefore, the NETF also works in ensemble subspace and is computationally efficient. As the Bayesian weights are only used intermediately, in contrast to PFs the analysis can be localized by domain localization, as in the ETKF.
- The NETF only acts on the analysis step and is of generic nature. It also works with deterministic models, and there are no restrictions on the observation operator or the form of the likelihood pdf. It can be implemented directly into an existing ETKF system with little effort.

Therefore, the NETF has appealing properties from a theoretical point of view and can be considered as a promising alternative method for nonlinear filtering problems.

Consequently, the third part of this thesis deals with applications of the NETF in order to confirm its usefulness. First, the NETF is applied to typical DA test beds to assess its performance properties. The main conclusions from these experiments are as following:

- The NETF works successfully in deterministic systems of different dimensions and degrees of nonlinearity, even with small ensemble sizes. This confirms that the deterministic generation of a new ensemble, together with localization, strongly improves filter stability.
- In nonlinear scenarios, the NETF clearly outperforms the EnKFs. Therefore, as expected by its construction, the NETF is potentially more suitable to deal with highly nonlinear situations. This becomes even more evident in the presence of nonlinear observation operators or non-Gaussian likelihood pdfs that may quickly appear for real observations (Anderson 2010).
- The NETF also outperforms the NLEAF in all systems of larger dimensionality. Hence, the NETF is not only computationally more efficient, but its deterministic design is also more beneficial in terms of analysis quality.

Finally, motivated by these results, the NETF is applied to a more realistic assimilation problem in an ocean general circulation model (NEMO) with a state dimension of about $3.3 \cdot 10^5$. The assimilation of temperature profiles and sea surface heights mimics realistic observation networks. The system is characterized by its wind-driven large-scale dynamics, a double-gyre circulation as occurs in the North Atlantic, and mesoscale eddies that express its chaotic nature. This complex application leads to further valuable conclusions:

- The NETF does not only remain stable in this setting, but also shows a desirable performance and is able to converge to the truth and keep track of it, including the small-scale features. Thus, it can be applied to realistic circulation systems in its generic form with little implementation effort, and the model does not need to be adjusted. Therefore, the NETF is applicable to high-dimensional problems even though it only relies on the Bayesian weights, and it is able to overcome to curse of dimensionality. This represents an important contribution to the progress in nonlinear DA. To the knowledge of the author, so far, a similar successful large-scale application could only be achieved with the much more complex EWPF, which is restricted to stochastic models.
- The experiment further reveals that the NETF strongly reduces the analysis error and that the ensembles are statistically consistent. Therefore, it offers potential benefits over established methods such as the ETKF. Based on the results of this thesis, this probably concerns situations characterized by strong nonlinearity, but it has to be elaborated in further large-scale applications.
- The results indicate that the NETF is sensitive to its initialization. An inconsistent initial ensemble, e.g., with too little spread, results in many members with insignificant weights in the first analysis step and might cause the filter to collapse. Furthermore, this issue increases the spin-up phase in case the initial ensemble only contains little information about the true flow, a problem that is likely to concern the EWPF as well. In future applications, it should be ensured to specify consistent initial ensembles that include the truth and contain as much knowledge about the state as possible. Alternatively, a practical solution to accelerate the spin-up might consist in using the ETKF for the initial phase only and thereafter switching to the nonlinear filter.
- In this experiment, the observation dimension is of order $\mathcal{O}(10^3)$. It is mainly achieved by the profilers and is already relatively large for ocean DA, where most observation types concern the surface only. For the localized NETF, a decisive quantity for stability is the local observation

space dimension, where the local weights are evaluated. In this setup, the NETF works fine with a realistic localization radius of $2.5°$, leading to a hundred-dimensional local observation space (on average). Even though this is already a huge dimension for a PF-based algorithm with only about hundred members considering the curse of dimensionality (Snyder et al. 2008), it probably also represents the main practical limitation of the filter as the localization radius has to be chosen depending on the observation network. For instance, in current atmospheric DA, the variety of observation types can lead to a relatively large observation vector, particularly if many satellite retrievals are included as characteristic of an operational context. In turn, the local observation vector may be significantly larger as well. This issue could be counteracted by reducing the localization radius, but it also renders the filter suboptimal if too many observations are excluded by too tight localization (Nerger et al. 2014). Thus, the application with very dense observation networks may still represent a future challenge and needs to be assessed empirically.

In summary, the results shown in this work form a consistent picture. A simple and generic algorithm for nonlinear DA with appealing theoretical properties is proposed. The empirical experiments demonstrate that the NETF is able to outperform other ensemble-based filters and that it is applicable in high-dimensional systems in its general formulation.

7.2 Outlook

The NETF derived in this work is a fundamental algorithm of generic nature. Its potential usefulness could be demonstrated in various situations. Based on these results, numerous possibilities for future research arise. Here, some suggestions are outlined, but they are probably not exhaustive. In principle, follow-up research could be performed in two directions. The first direction concerns theoretical issues:

1. In Chapter 4, a thorough investigation of mathematical filter properties was conducted. However, some questions still remain open. First, at the current stage, the effect of the purely deterministic transform with matrix $\mathbf{T}$, see equation (4.21), on the prior ensemble is unclear. Even though is creates an ensemble with correct mean and covariance, it appears that higher-order moments are affected in an undesirable way. This renders the random rotation mandatory, which increases filter stability but also destroys potential higher-order moments of the analysis ensemble. It was shown that $\mathbf{T}$ is ill-conditioned, which is not problematic as it is not inverted, however, it may be a hint to answering the question. Perhaps it might be possible to derive another deterministic transform matrix with better condition.

   Second, like the other ensemble filters tested in these experiments, the NETF concentrates on the first two moments of the analysis pdf. In principle, third or higher-order moments could also be derived using equation (4.1). This raises the question whether ensembles of higher-order exactness may be generated in a similar way, which could be a direction for further research. In the work of Miller et al. (1994), the consideration of third and fourth moments with an extended KF resulted in major improvements in L63 experiments. Hence, a first step could concern the development of a higher-order EnKF or ETKF before using the Bayesian estimates. However, such a filter may be limited by the fact that the Monte Carlo estimates of higher-order moments

require a rather large ensemble size to be reliable, and by the computational expenses needed to evaluate higher-order moments in larger-dimensional spaces.

2. This work concentrates on the filtering problem. As shown in Chapter 2, it is also possible to describe a smoother solution to DA. In fact, all EnKFs can be turned into smoothers by using the sequential smoother formulation. In principle, they work by applying the analysis weights of a certain time to past filter estimates in order to update them by the future observations (Evensen and van Leeuwen 2000; Posselt and Bishop 2012). The KF-based smoothers implicitly rely on the temporal correlations between model states and make Gaussian assumptions for the joint-in-time pdfs. Nonlinear or particle smoothers exist as well (Andrieu et al. 2010; Kitagawa 1996), but are subject to the same problems as PFs. Therefore, it would be interesting to develop a nonlinear smoother based on the NETF that could be used for reanalysis purposes or similar objectives. As the update equation of the NETF is formally identical to the ETKF, this should be rather straightforward. It can be expected that only the smoother time lag appears as additional tuning factor (Nerger et al. 2014).

The second direction of future research concerns further applications of the NETF in order to gain more knowledge about performance properties other than those discussed in this thesis.

1. In this work, the NETF was applied in OSSEs, using synthetic observations, in order to concentrate on the filter behavior. With real observation data, the impact of the filter algorithm can be concealed by sub-optimal issues such as model or observation biases or the specification of the observation error covariances. Therefore, the NETF should be applied with empirical data. In particular, the sensitivity of the NETF towards these issues could be investigated, which is another important question regarding its practical applicability. However, these problems are probably similar for other PF-based techniques such as the EWPF.

2. The applications shown in this work confirmed a reasonable behavior of the NETF in scenarios of different complexity, also in the presence of nonlinear observation operators and non-Gaussian likelihood densities. Though the results in Chapter 5 indicate benefits in these situations as compared to the ETKF, there is still room for further explorations.
   In particular, a further large-scale application with different circulation characteristics and a time-dependent forcing could continue the work presented in Chapter 6. This might reveal potential benefits over linear methods such as the ETKF. For instance, an atmospheric circulation system would be appropriate, as it could demonstrate how the NETF deals with a circulation of increased variability and a potentially denser observation network, compared to the ocean experiment presented in this work. As discussed above, it is an open question whether the NETF is able to work with an even larger local observation dimension than in the NEMO experiment. In this context, a reasonable first step might employ an atmospheric system with rather sparse observations. For example, the Twentieth Century Reanalysis (20CR, Compo et al. 2011) applied an EnKF to obtain a global reanalysis including uncertainty information from 1871-2008, assimilating only surface pressure observations. Nevertheless, the results are of good quality, even compared to other reanalyses that use upper air and satellite data in the second half of the twentieth century. This also holds for upper air circulations and the representation of low-frequent variability beyond the

synoptic scale (Compo et al. 2011). These insights prove that the comparably sparse observations are able to constrain the atmospheric circulation well, because surface pressure determines the barotropic flow through geostrophy and its tendency is related to the vertically integrated mass flux divergence (Whitaker et al. 2009). Thus, a similarly successful application could be anticipated with the NETF. Later on, this application could be extended by gradually increasing the observation dimension, e.g., by including other observation networks or even satellite retrievals. This would represent a useful practical application that simultaneously allows to explore the NETF's limitations in atmospheric DA.

3. As discussed in this work, the NETF is a nonlinear filter that, in contrast to the EWPF, concentrates on the first two moments of the analysis pdf only, which allows to use deterministic models and renders the algorithm much simpler. Thus, a comparison of both methods should be performed to understand the impact of the complex proposal densities in the EWPF. As the EWPF requires a stochastic model to perform the nudged forecast step, within such an experiment one should also run the NETF with nudging, which is possible without changing the algorithm (see Chapter 4). The results might give valuable hints on the importance of considering higher-order moments in nonlinear, high-dimensional DA. For low-dimensional systems, this question could be answered in a simplified way by comparing the NETF to the PF directly.

4. Another aspect could concern the re-utilization of the final analysis ensemble for ensemble predictions, as already briefly initiated in Chapter 6. Wang and Bishop (2003) showed that an ETKF-based initialization scheme creates better ensemble forecasts than classical alternatives such as singular vectors or breeding, particularly if enhanced by rotations (Wang et al. 2004). In this context, it would be interesting to further investigate the impact of a nonlinear analysis scheme on successive ensemble forecasts (see also Ades 2013, Ch. 7.3), potentially enhanced by random rotations as discussed in this work.

Appendix A

Mathematical background

This appendix contains some background to the various mathematical methods used throughout this work. It may serve as a quick reference for the techniques applied in the main text. It does not represent a detailed, self-contained treatment of the different topics, for which standard text books can be consulted (e.g., Doucet et al. 2001; Horn and Johnson 1985; Lütkepohl 1996; Robert and Casella 2004; Wilks 2011).

A.1 Matrix Algebra

A.1.1 Transpose and adjoint

$\mathbb{X}$ and $\mathbb{Y}$ are two vector spaces, and $\mathbf{A}$ is a linear operator that maps from $\mathbb{X} \to \mathbb{Y}$. Its adjoint $\mathbf{A}^\dagger$ is an operator mapping from $\mathbb{Y} \to \mathbb{X}$ and is defined by the following identify:

$$(\mathbf{A}\mathbf{x}) \circ \mathbf{y} = \mathbf{x} \circ (\mathbf{A}^\dagger \mathbf{y}) \qquad \forall \quad \mathbf{x} \in \mathbb{X} \wedge \mathbf{y} \in \mathbb{Y} \tag{A.1}$$

Here, $\circ$ denotes an inner product defined in the vector spaces. If the linear operator is represented as matrix and the inner product is the standard scalar product of vectors, its adjoint is equal to its complex conjugate transpose, i.e., $A^\dagger_{ij} = A^*_{ji}$. In the DA context, the relevant operators are typically real, which will be assumed from now on. Then, the adjoint reduces to the transpose, $\mathbf{A}^\dagger = \mathbf{A}^T$. However, the operators are usually not represented by an explicit matrix rather than complex computer code. Thus, the adjoint is a new piece of code, called the adjoint model. The following identities are useful:

$$(\mathbf{A}^T)^T = \mathbf{A} \tag{A.2}$$

$$(\lambda\mathbf{A})^T = \lambda\mathbf{A}^T \qquad \forall \quad \lambda \in \mathbb{R} \tag{A.3}$$

$$(\mathbf{A}+\mathbf{B})^T = \mathbf{A}^T + \mathbf{B}^T \tag{A.4}$$

$$(\mathbf{A}\mathbf{B})^T = \mathbf{B}^T\mathbf{A}^T \tag{A.5}$$

$$(\mathbf{A}^{-1})^T = (\mathbf{A}^T)^{-1} \quad \text{, if } \mathbf{A} \text{ is invertible (see below)} \tag{A.6}$$

If a square matrix equals its transpose, $\mathbf{A} = \mathbf{A}^T$, it is called *symmetric* (or, in general, hermitian).

A.1.2 Inverse matrix

The inverse matrix of a square matrix $\mathbf{A}$, denoted $\mathbf{A}^{-1}$, is defined by $\mathbf{A}^{-1}\mathbf{A} = \mathbf{A}\mathbf{A}^{-1} = \mathbf{I}$. In general, an inverse matrix does not necessarily exist. If it exists, $\mathbf{A}$ is called nonsingular or invertible, and $\mathbf{A}^{-1}$ is unique. A matrix is singular if and only if its determinant vanishes, which again is equivalent to its columns being linearly independent. The inverse can be computed by numerous techniques, e.g., by an eigendecomposition (see below).
If the inverse of $\mathbf{A}$ equals its transpose $\mathbf{A}^T$, it follows that $\mathbf{A}\mathbf{A}^T = \mathbf{A}^T\mathbf{A} = \mathbf{I}$. Then, the columns of $\mathbf{A}$ are mutually orthonormal and $\mathbf{A}$ is called *orthogonal* (or unitary, in the complex case). The determinant of an orthogonal matrix is always 1 or -1.

A.1.3 Trace

The trace of a square matrix is a scalar function that yields the sum of its diagonal elements, $\mathrm{tr}(\mathbf{A}) = \sum_i A_{ii}$. The trace is additive, i.e., $\mathrm{tr}(\mathbf{A} + \mathbf{B}) = \mathrm{tr}(\mathbf{A}) + \mathrm{tr}(\mathbf{B})$.
If the matrix $\mathbf{A}$ has evolved from manipulations (matrix sums, multiplications) of the matrix $\mathbf{K}$, the trace of $\mathbf{A}$ in general depends on all entries of $\mathbf{K}$. The partial derivatives of the trace with respect to all entries K_{ij} can be collected into a matrix by defining:

$$\left(\frac{\partial \,\mathrm{tr}(\mathbf{A})}{\partial \mathbf{K}}\right)_{ij} \equiv \frac{\partial \,\mathrm{tr}(\mathbf{A})}{\partial K_{ij}}$$

This allows to derive some identities that are needed in the statistical derivation of the KF analysis equation in section B.2:

1. $\mathrm{tr}(\mathbf{K}\mathbf{A}) = \sum_i (\mathbf{K}\mathbf{A})_{ii} = \sum_{ik} K_{ik}A_{ki} \quad \Rightarrow \quad \frac{\partial \mathrm{tr}(\mathbf{K}\mathbf{A})}{\partial \mathbf{K}} = \mathbf{A}^T$
2. $\mathrm{tr}(\mathbf{A}\mathbf{K}^T) = \sum_i (\mathbf{A}\mathbf{K}^T)_{ii} = \sum_{ik} A_{ik}K_{ik} \quad \Rightarrow \quad \frac{\partial \mathrm{tr}(\mathbf{A}\mathbf{K}^T)}{\partial \mathbf{K}} = \mathbf{A}$
3. $\mathrm{tr}(\mathbf{K}\mathbf{A}\mathbf{K}^T) = \sum_i (\mathbf{K}\mathbf{A}\mathbf{K}^T)_{ii} = \sum_{ijk} K_{ik}A_{kj}K_{ij} \quad \Rightarrow \quad \frac{\partial \mathrm{tr}(\mathbf{K}\mathbf{A}\mathbf{K}^T)}{\partial \mathbf{K}} = \mathbf{K}(\mathbf{A} + \mathbf{A}^T)$

A.1.4 Matrix Decompositions

For many applications, it is useful to decompose a matrix by representing it in a different basis. Here, only the two variants that are relevant for this thesis are shown. Other methods, such as the Cholesky or LU decomposition, are omitted.

A.1.4.1 Eigenvalue Decomposition

If an $N \times N$ square matrix $\mathbf{A}$ (or, in general, a linear operator) does not change the direction of a vector $\mathbf{e}$, this vector is an eigenvector of $\mathbf{A}$:

$$\mathbf{A}\mathbf{e} = \lambda\mathbf{e}$$

The unique multiple value λ is the eigenvalue. It can be calculated from the roots of the characteristic polynom, $p(\lambda) = \det(\mathbf{A} - \lambda \mathbf{I}_N)$, showing that each $N \times N$ square matrix has N (in general complex) eigenvalues λ_n, which may be of algebraic multiplicity. The corresponding eigenvectors are only defined up to a multiple. Usually, the eigenvectors are normalized such that $||\mathbf{e}_n|| = 1$.

The eigendecomposition theorem states that any square matrix can be factorized as following, named the eigenvalue decomposition (EVD):

$$\mathbf{A} = \mathbf{U}\mathbf{E}\mathbf{U}^{-1} \tag{A.7}$$

Here, $\mathbf{E}$ is a diagonal matrix containing the eigenvalues λ_n, and $\mathbf{U} = \Big(\mathbf{e}_1, \ldots, \mathbf{e}_N\Big)$ is the matrix composed from the N linearly independent eigenvectors, which form the basis of the eigenspace of $\mathbf{A}$. The EVD is, among other applications, useful in finding any matrix power of $\mathbf{A}$. It is simply given by $\mathbf{A}^p = \mathbf{U}\mathbf{E}^p\mathbf{U}^{-1}$, where $\mathbf{E}^p$ is easily computed. Of particular interest are the inverse matrix ($p = -1$) and the matrix square root ($p = 1/2$).

An important special case occurs if $\mathbf{A}$ is symmetric (hermitian in the complex case). Then, its eigenvalues and eigenvectors are real and the latter can be chosen to form an orthonormal basis in $\mathbb{R}^N$. Thus, the matrix of eigenvectors $\mathbf{U}$ becomes orthogonal and the EVD simplifies to $\mathbf{A} = \mathbf{U}\mathbf{E}\mathbf{U}^T$.

A.1.4.2 Singular value decomposition

The singular value decomposition (SVD) generalizes the EVD to a rectangular, real $M \times N$ matrix $\mathbf{A}$,

$$\mathbf{A} = \mathbf{U}\mathbf{S}\mathbf{V}^T \tag{A.8}$$

If a square matrix $\mathbf{A}$ is real, symmetric and positive definite, its SVD and EVD are identical. Its components are further described:

- $\mathbf{S}$ is a nonnegative $M \times N$ diagonal matrix, and its diagonal entries are the r singular values in descending order,

  $$\mathrm{diag}(\mathbf{S}) = \Big(\sigma_1, \ldots, \sigma_r, 0 \ldots 0\Big)$$

  where $\sigma_n \geq \sigma_{n-1} > 0$ and $r \leq \min(M, N)$.
- $\mathbf{U}$ is a real $M \times M$ orthogonal matrix. Its column vectors $\mathbf{u}_n$ are the left-singular vectors of $\mathbf{A}$.
- $\mathbf{V}$ is a real $N \times N$ orthogonal matrix. Its column vectors $\mathbf{v}_n$ are the left-singular vectors of $\mathbf{A}$.

These names originate in the following connection of the singular vectors and values:

$$\mathbf{A}\mathbf{v}_n = \sigma_n \mathbf{u}_n \quad , \quad \mathbf{A}^T\mathbf{u}_n = \sigma_n \mathbf{v}_n \qquad \text{for} \quad n = 1, \ldots, \min(M, N)$$

The SVD can be used to calculate the pseudo-inverse of a rectangular matrix, given by

$$\tilde{\mathbf{A}}^{-1} = \mathbf{V}\mathbf{S}^{-1}\mathbf{U}^T$$

where the diagonal matrix $\mathbf{S}^{-1}$ exhibits the entries $\frac{1}{\sigma_n}$ for $n \leq r$, and zero everywhere else.

A.2 Vector analysis

A.2.1 Nonlinear operators: Jacobian and adjoint

Consider a nonlinear operator $\mathcal{F}(\mathbf{x})$ which maps from vector space $\mathbb{X}$ to $\mathbb{Y}$, with dimensions K and L. In the DA context, this may be either the observation operator, projecting into observation space, or the model operator, propagating the state one time step ahead (in this case, $\mathbb{Y}$ is equal to $\mathbb{X}$). Mathematically, the operator can be represented as a vectorial function

$$\mathcal{F} : \mathbb{X} \to \mathbb{Y} \quad , \quad \mathcal{F}(\mathbf{x}) = \mathbf{y} \quad \text{where} \quad \mathcal{F}(\mathbf{x}) \equiv \Big(F_1(\mathbf{x}), \ldots, F_L(\mathbf{x})\Big)^T \tag{A.9}$$

In DA, such a function is usually not available explicitly, and the operator is rather represented by complex computer code which takes the vector $\mathbf{x}$ as input and outputs $\mathbf{y}$. Each subroutine, or even each line of the code, can be regarded as an individual operator. Hence, $\mathcal{F}$ can be written as a sequence of M nonlinear functions f_m which are consecutively applied to the input vector:

$$\mathcal{F}(\mathbf{x}) = f_M(f_{M-1}(\cdots(f_1(\mathbf{x})))) \tag{A.10}$$

Another example for such a factorization is the application of the model operator for several time steps, as used in Chapter 2, see equation (2.2). Next, is is applied to obtain the theoretical basis for the construction of the tangent linear and adjoint model.

The *Jacobian* $\mathbf{F}$ can be considered as the derivative of the operator $\mathcal{F}$. In principal, it is a $L \times K$ matrix that contains all possible partial derivatives:

$$\mathbf{F} = \frac{\partial \mathcal{F}}{\partial \mathbf{x}} \quad \text{with} \quad F_{lk} = \frac{\partial F_l}{\partial x_k} \quad (k = 1 \ldots K,\, l = 1 \ldots L) \tag{A.11}$$

In most practical applications, $\mathbf{F}$ cannot be obtained in this form due to $\mathcal{F}$ being a complex operator. Exploiting equation (A.10), each function f_m can be linearized to get the *tangent linear model* (TLM)

$$\mathbf{F}\,\mathbf{x} = \prod_{m=M}^{1} \mathbf{F}_m\,\mathbf{x} \tag{A.12}$$

Thus, it can be applied successively as the full model, using the linearized basic functions. In praxis, the TLM is obtained by linearizing each line of code.

For real operators, the adjoint model is $\mathbf{F}^T$, i.e., the transpose of the TLM:

$$\mathbf{F}^T\mathbf{x} = \prod_{m=1}^{M} \mathbf{F}_m^T\mathbf{x} \tag{A.13}$$

Thus, applying the adjoint model to a vector $\mathbf{x}$ reverses the order the sub-functions are applied. If $\mathcal{F}$ is the model operator, this can be interpreted as an integration backward in time. As the TLM, the adjoint code can obtained line-by-line.

A.2.2 Scalar functions: Gradient and Hessian

Assume a nonlinear scalar function in a vector space $\mathbb{X}$, like the cost function $J(\mathbf{x})$ ($\mathbb{X} \rightarrow \mathbb{R}$). Its gradient is a vector containing the partial derivatives in the corresponding directions and hence, it points in the direction of the steepest ascent:

$$\nabla J = \left(\frac{\partial J}{\partial x_1}, \ldots, \frac{\partial J}{\partial x_k}, \ldots, \frac{\partial J}{\partial x_K} \right)^T \tag{A.14}$$

The Hessian $\mathbf{J}''$ is a symmetric $K \times K$ matrix storing all second derivatives,

$$J''_{kl} = \frac{\partial^2 J}{\partial x_k \partial x_l} \quad (k, l = 1 \ldots K) \tag{A.15}$$

Thus, it contains information about the curvature of $J(\mathbf{x})$.

A.2.3 Delta distribution

The delta distribution $\delta(\mathbf{x} - \mathbf{x}_0)$ can be regarded as a function which vanishes everywhere expect at a single point $\mathbf{x} = \mathbf{x}_0$, where it is infinitely large, with the additional condition of a finite integrability:

$$\int \delta(\mathbf{x} - \mathbf{x}_0)\, d\mathbf{x} = 1 \tag{A.16}$$

Formally, the delta function can be obtained as limiting case of a regular function, for example, using a Gaussian pdf with vanishing standard deviation (see also section A.3.5):

$$\delta(\mathbf{x} - \mathbf{x}_0) = \lim_{\sigma \rightarrow 0} \mathcal{N}(\mathbf{x}; \mathbf{x}_0, \mathbf{P}) \tag{A.17}$$

where the covariance matrix $\mathbf{P}$ is given as diagonal matrix with all entries being equal to σ^2. Therefore, the delta function can be interpreted as the formal representation of the pdf in a deterministic case. As it vanishes almost everywhere, it filters the specific value from any function. This leads to an important *convolution* property:

$$f(\mathbf{x})\delta(\mathbf{x} - \mathbf{x}_0) = f(\mathbf{x}_0)\delta(\mathbf{x} - \mathbf{x}_0) \quad \Rightarrow \quad \int f(\mathbf{x})\delta(\mathbf{x} - \mathbf{x}_0) d\mathbf{x} = f(\mathbf{x}_0) \tag{A.18}$$

which is of high relevance when dealing with particle filters, see Chapters 3 and 4.

A.3 Stochastics in vector spaces

A key aspect for the description and derivation of DA methods is their probabilistic interpretation. This section summarizes the basic concepts of stochastics required for this thesis.

A.3.1 Random variables

A random variable can take different values under identical conditions. If the set of possible values is finite, it is classified as a discrete random variable, otherwise, it is a continuous random variable. In the DA context, usually continuous random variables are dealt with, e.g., the temperature at a certain location. Furthermore, a random variable does not have to be scalar but can also represent a collection of individual random variables, expressed by a vector.

More specifically, let the vector space $\mathbb{X}$ be of finite dimensionality K, and $\mathbf{x} = \left(x_1, \ldots, x_K\right)^T \in \mathbb{X}$ is a multivariate random variable. A common example in DA is the state space, containing all possible realizations of the dynamical system at a specific time. In most cases, the true state of the system is unknown and only estimates are available, for example, a first guess or an analysis. Consequently, this inherent uncertainty must be dealt with by describing the system's state in stochastic terms. The knowledge about the state is contained in a probability density function (pdf) $p(\mathbf{x})$, which is a real, nonnegative and normalized scalar function:

$$p : \mathbb{X} \rightarrow \mathbb{R} \quad , \quad p(\mathbf{x}) \geq 0 \; \forall \; \mathbf{x} \in \mathbb{X} \quad , \quad \int_{\mathbb{X}} p(\mathbf{x}) \, d\mathbf{x} = 1 \tag{A.19}$$

The interpretation is that the integral of the pdf over a finite state space volume $A \subset \mathbb{X}$,

$$\int_A p(\mathbf{x}) \, d\mathbf{x} = P(\mathbf{x} \in A) \tag{A.20}$$

equals the probability of finding the system in this region. If the state is known perfectly, the delta function still allows it to be described by a pdf, see section A.2.3.

A.3.2 Combination of random variables

Let $\mathbf{x}$ and $\mathbf{y}$ be two random variables, for example, the state and observation vector. Another common example are the state vectors at different times. A full description of both variables and their relations is given by the joint pdf, $p(\mathbf{x}, \mathbf{y})$. It contains the probability density values for all possible combinations. Again, the integral $\int \int p(\mathbf{x}, \mathbf{y}) d\mathbf{x} d\mathbf{y}$ over finite regions equals the probability of finding both random variables simultaneously in the corresponding regions. The marginal pdf describes the overall behavior of one single random variable without considering the second one. Mathematically, it is obtained by integrating out the other variable,

$$p(\mathbf{x}) = \int p(\mathbf{x}, \mathbf{y}) \, d\mathbf{y} \tag{A.21}$$

The conditional pdf $p(\mathbf{x}|\mathbf{y})$ describes the distribution of $\mathbf{x}$ for a given realization of $\mathbf{y}$. The joint pdf can be factorized into marginal and conditional pdfs in two distinct ways,

$$p(\mathbf{x}, \mathbf{y}) = p(\mathbf{x})p(\mathbf{y}|\mathbf{x}) = p(\mathbf{y})p(\mathbf{x}|\mathbf{y}) \tag{A.22}$$

If the random variables are stochastically independent, the joint pdf reduces to the product of the marginal ones, $p(\mathbf{x}, \mathbf{y}) = p(\mathbf{x})p(\mathbf{y})$. A consequence from the equality of both factorizations in equation (A.22) is *Bayes' theorem*, which reveals the way to compute a conditional pdf:

$$p(\mathbf{x}|\mathbf{y}) = \frac{p(\mathbf{x}, \mathbf{y})}{p(\mathbf{y})} = \frac{p(\mathbf{x})p(\mathbf{y}|\mathbf{x})}{\int p(\mathbf{x})p(\mathbf{y}|\mathbf{x})\, d\mathbf{x}} \tag{A.23}$$

Despite its formal simplicity, Bayes' theorem has a major relevance in almost any field of theory or application due to its generality. Usually, $p(\mathbf{x})$ is interpreted as the *prior* information since it is an unconditional pdf. Furthermore, a realization of the random variable $\mathbf{y}$, given $\mathbf{x}$ has occurred, may be available. This dependency is formally described by the *likelihood* pdf $p(\mathbf{y}|\mathbf{x})$. Then, the conditional pdf $p(\mathbf{x}|\mathbf{y})$ represents the *posteriori* pdf that incorporates this additional information. Thus, Bayes theorem tells how to update the knowledge about an uncertain variable, as represented by its pdf, with a new piece of information.

The rules shown here can easily be extended to more than two random variables by combining a subset of them into a single random variable. For example,

$$p(\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3) = p(\mathbf{x}_1)p(\mathbf{x}_2, \mathbf{x}_3|\mathbf{x}_1) = p(\mathbf{x}_1)p(\mathbf{x}_2|\mathbf{x}_1)p(\mathbf{x}_3|\mathbf{x}_1, \mathbf{x}_2) \tag{A.24}$$

A.3.3 Expected values

The *expected value* of any function of a random variable $\mathbf{x}$, $f(\mathbf{x})$, is given by an integration over its pdf, which can be interpreted as weighting the function with the corresponding density:

$$\langle f(\mathbf{x}) \rangle = \int f(\mathbf{x})p(\mathbf{x})\, d\mathbf{x} \tag{A.25}$$

The expected value of the random variable itself can be computed by simply setting $f(\mathbf{x}) = \mathbf{x}$, which is also the first moment of the pdf. The second moment is usually expressed by the covariance matrix $\mathbf{P}$. It contains the covariances between all components of $\mathbf{x}$,

$$\boldsymbol{\mu} = \langle \mathbf{x} \rangle = \int f(\mathbf{x})\mathbf{x}\, d\mathbf{x} \tag{A.26}$$

$$P_{kl} = \langle (x_k - \langle x_k \rangle)(x_l - \langle x_l \rangle) \rangle = \langle x_k x_l \rangle - \langle x_k \rangle \langle x_l \rangle \tag{A.27}$$

Compactly, the covariance matrix can be written as an outer vector product

$$\mathbf{P} = \mathrm{cov}(\mathbf{x}) = \left\langle (\mathbf{x} - \langle \mathbf{x} \rangle)(\mathbf{x} - \langle \mathbf{x} \rangle)^T \right\rangle \tag{A.28}$$

The diagonal of the covariance matrix contains the variances of all components, i.e., $P_{kk} = \sigma_k^2$ where σ_k is the standard deviation of x_k. By definition, the covariance matrix is real and symmetric. Furthermore, it is always positive definite. The correlation between two components is their covariance normalized

by their standard deviations:

$$\mathrm{cor}(x_k, x_l) = \frac{P_{kl}}{\sqrt{P_{kk}P_{ll}}} = \frac{\sigma_{kl}^2}{\sigma_k \sigma_l} \tag{A.29}$$

It always falls in the range $[-1, 1]$, where the extreme values correspond to maximal negative and positive correlation. A value of zero indicates no correlation, which must not be confused with stochastic independency.
Higher-order moments could be derived by the same mechanism, using equation (A.25). However, in higher-dimensional spaces they are often not feasible.

Finally, an important property of the second-order statistics of any distribution is shown. Let $\mathbf{z} = \mathbf{L}\mathbf{x} + \mathbf{a}$ be a linear transform of the random variable $\mathbf{x}$. Then, the expected mean and covariance of the transformed variable $\mathbf{z}$ are given by

$$\langle \mathbf{z} \rangle = \mathbf{L}\langle \mathbf{x} \rangle + \mathbf{a} \quad ; \quad \mathrm{cov}(\mathbf{z}) = \mathbf{L}\,\mathrm{cov}(\mathbf{x})\,\mathbf{L}^T = \mathbf{L}\mathbf{P}\mathbf{L}^T \tag{A.30}$$

These results directly follow from the definition (A.25).

A.3.4 Monte Carlo estimators

In many applications, it is not possible to evaluate the integral (A.25) because the dimensionality is too high, or the pdf $p(\mathbf{x})$ might even not be known. The Monte Carlo method approximates integrals using a finite number of random samples. A number of N independent samples $\{\mathbf{x}^n\}$, distributed according to $q(\mathbf{x})$, called the *proposal density*, can be used to approximately compute the integral. Mathematically, this means that the density is replaced by a mixture delta distribution,

$$\mathbf{x}^n \sim q(\mathbf{x}) \quad \Rightarrow \quad q(\mathbf{x}) \approx \frac{1}{N}\sum_{n=1}^{N} \delta(\mathbf{x} - \mathbf{x}^n) \tag{A.31}$$

Using this representation in equation (A.25), the expected value of any function can be approximated by an empirical mean:

$$\langle f(\mathbf{x}) \rangle = \int \frac{f(\mathbf{x})p(\mathbf{x})}{q(\mathbf{x})} q(\mathbf{x})\, d\mathbf{x} \approx \overline{f(\mathbf{x})} = \frac{1}{N}\sum_n \frac{p(\mathbf{x}^n)}{q(\mathbf{x}^n)}\, f(\mathbf{x}^n) \tag{A.32}$$

The weight factor $p(\mathbf{x}^n)/q(\mathbf{x}^n)$ accounts for the fact that the proposal density is different from $p(\mathbf{x})$. An important special case occurs if the samples $\mathbf{x}^n$ already follow the target density, i.e., $q(\mathbf{x}) \equiv p(\mathbf{x})$. Then, the expected value of $f(\mathbf{x}) = \mathbf{x}$ is simply represented by their mean, $\overline{\mathbf{x}} = \frac{1}{N}\sum_n \mathbf{x}^n$. Furthermore, the empirical covariance gets

$$\mathbf{P} = \frac{1}{N-1}\sum_n (\mathbf{x}^n - \overline{\mathbf{x}})(\mathbf{x}^n - \overline{\mathbf{x}})^T \tag{A.33}$$

Here, typically the coefficient $N-1$ is used instead of N to obtain an unbiased estimate (Wilks 2011). It is convenient to collect the deviations into the columns of a perturbation matrix. Then, the empirical

covariance can be calculated with one matrix multiplication:

$$\mathbf{X}' = \left(\mathbf{x}^1 - \overline{\mathbf{x}}, \ldots, \mathbf{x}^N - \overline{\mathbf{x}}\right) \quad \Rightarrow \quad \mathbf{P} = \frac{1}{N-1}\mathbf{X}'\mathbf{X}'^T \tag{A.34}$$

Alternatively, the perturbation matrix itself can also be obtained by a matrix multiplication, $\mathbf{X}' = \mathbf{X}\mathbf{T}$, where $\mathbf{T} = \mathbf{I}_N - \frac{1}{N}\mathbf{a}\mathbf{a}^T$ and $\mathbf{a} = (1, \ldots, 1)^T$.

A.3.5 Analytical distributions

So far, the pdf $p(\mathbf{x})$ appeared in a generic form. Depending on the situation, a parameterization of the pdf, i.e., a representation in functional form, can be useful. For a scalar random variable, a large variety of analytical distributions with different properties exists. Some of them can be generalized to vector variables. Here, only the densities used within this thesis are shown.

A.3.5.1 Gaussian distribution

The most important analytical distribution is the multivariate normal (MVN). It generalizes the univariate normal density, $p(x) = \frac{1}{\sqrt{2\pi}\sigma}\exp\{-(x-\mu)^2/(2\sigma^2)\}$, with mean μ and standard deviation σ. The practical relevance of the Gaussian density is based on the central limit theorem (see below) and the fact that it is uniquely characterized by its first two moments, which are (in a K-dimensional vector space) the mean vector $\boldsymbol{\mu}$ and the $K \times K$ covariance matrix $\mathbf{P}$. The functional form of the MVN is given by

$$p(\mathbf{x}) = \frac{1}{\sqrt{(2\pi)^K \det(\mathbf{P})}} \exp\left\{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^T\mathbf{P}^{-1}(\mathbf{x}-\boldsymbol{\mu})\right\} \equiv \mathcal{N}(\mathbf{x}; \boldsymbol{\mu}, \mathbf{P}) \tag{A.35}$$

The notation $\mathcal{N}(\mathbf{x}; \boldsymbol{\mu}, \mathbf{P})$ refers to the pdf's functional form $p(\mathbf{x})$, and the first entry specifies the dependent variable, here $\mathbf{x}$. In contrast, the notation $\mathbf{x} \sim \mathcal{N}(\boldsymbol{\mu}, \mathbf{P})$ states that $\mathbf{x}$ is distributed according to a MVN with mean $\boldsymbol{\mu}$ and covariance $\mathbf{P}$.

The MVN exhibits some important properties:

- A normal distribution is conserved under a linear transformation of the random variable. If $\mathbf{x} \sim \mathcal{N}(\boldsymbol{\mu}, \mathbf{P})$ and a linear transform is given by $\mathbf{z} = \mathbf{L}\mathbf{x} + \mathbf{a}$, then $\mathbf{z} \sim \mathcal{N}(\mathbf{L}\boldsymbol{\mu} + \mathbf{a}, \mathbf{L}\mathbf{P}\mathbf{L}^T)$, see equation (A.30).
- In particular, the transformation $\mathbf{z} = \mathbf{P}^{-1/2}(\mathbf{x} - \boldsymbol{\mu})$ standardizes this random variable, since then $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_K)$.
- Central limit theorem: The distribution of the sum $\sum_{n=1}^N \mathbf{x}_n$ of N independent random variables $\mathbf{x}_n$, which itself may follow arbitrary distributions, approximates a Gaussian density for $N \to \infty$.
- The sum of N independent Gaussian random variables yields another Gaussian random variable, even for finite N. If $\mathbf{x}_n \sim \mathcal{N}(\boldsymbol{\mu}_n, \mathbf{P}_n)$, then $\sum_{n=1}^N \mathbf{x}_n \sim \mathcal{N}\left(\sum_{n=1}^N \boldsymbol{\mu}_n, \sum_{n=1}^N \mathbf{P}_n\right)$. In other words, the means and covariances are simply added.

- All marginal distributions of the MVN are also Gaussian distributions. This can be seen by choosing a linear transformation $\mathbf{L}$ that selects a subset of the components of $\mathbf{x}$.

A sample of the MVN can be obtained by first transforming K uniformly distributed random numbers into K normally distributed numbers, using, for example, the Box-Muller transform, and collecting them into a vector $\tilde{\mathbf{x}} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_K)$. Finally, the transformation $\mathbf{x} = \boldsymbol{\mu} + \mathbf{P}^{1/2}\tilde{\mathbf{x}}$ is executed, such that $\mathbf{x} \sim \mathcal{N}(\boldsymbol{\mu}, \mathbf{P})$. This procedue requires to compute the matrix square root of the covariance $\mathbf{P}$.

A.3.5.2 Laplace distribution

The scalar Laplace distribution is the generalization of the exponential distribution ($p(x) = \lambda e^{-\lambda x}$), which is only defined for $x \in [0, \infty)$, to the full real domain. Therefore, it is also referred to as the *double exponential* distribution. Its functional form is given by $p(x) = \frac{1}{2\lambda} \exp\{-|x - \mu|/\lambda\}$, where λ is called the scale parameter. The mean of this Laplace density is μ and its variance is $\sigma^2 = 2\lambda^2$. In principal, the Laplace density has (for the same mean and variance) similar characteristics as the Gaussian density, as it is also symmetric around μ and shows a monotonic, exponential decrease with distance from the mean, with $p(x) \to 0$ for $x \to \pm\infty$. The Laplace density initially decreases more strongly, but exhibits fatter tails than the Gaussian density.

An easy extension to the multidimensional case is possible if the components are supposed to be independent. Then, the joint pdf is simply given by the product of the corresponding scalar pdfs:

$$p(\mathbf{x}) = \frac{1}{2^K \prod_{k=1}^{K} \lambda_k} \exp\left\{ -\sum_{k=1}^{K} \frac{|x_k - \mu_k|}{\lambda_k} \right\} \tag{A.36}$$

Consequently, this density exhibits a mean of $\boldsymbol{\mu} = (\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_K)^T$ and a diagonal covariance matrix with variances $2\lambda_1^2, \ldots, 2\lambda_K^2$. More general multivariate extensions that allow mutually dependent components exist (e.g., Eltoft et al. 2006), but are not used within this thesis.

In order to sample from the scalar Laplace density, its inverse cumulative distribution is utilized (Kotz et al. 2001). Given random samples u drawn from the uniform distribution $\mathcal{U}(-1/2, +1/2)$, the following transformation yields Laplacian samples with mean μ and scale parameter λ:

$$x = \mu - \lambda \cdot \operatorname{sgn}(u) \ln\left(1 - 2|u|\right) \tag{A.37}$$

Hence, a sample of the multivariate variant (A.36) is easily generated by producing K independent scalar Laplacian samples.

Appendix B

Kalman filter: Derivations and identities

This appendix contains mathematical details concerning the KF, which is an integral component of the EnKF and ETKF. In section 2.6, the KF is motivated within the probabilistic framework established in Chapter 2. First, the rather technical computation of the analysis pdf is presented. Second, an alternative to the probabilistic derivation of the KF is shown by taking a statistical point of view. This entirely different approach offers additional insight into the mechanism of the Kalman filter. Then, the connection of the KF with 3DVAR is established in a formal way. Finally, some identities are proven that allow different representation of the KF analysis equations.

B.1 Derivation of the analysis pdf

In section 2.6.3, the KF analysis step is derived within the probabilistic framework. Specifically, the task consists in performing the multiplication of the prior Gaussian pdf and the Gaussian likelihood pdf according to Bayes' theorem. The corresponding equation (2.41) is, using a shortened notation,

$$p(\mathbf{x}|\mathbf{y}) \propto p(\mathbf{x}|\mathbf{y})p(\mathbf{y}|\mathbf{x}) = \mathcal{N}(\mathbf{x}; \mathbf{x}_f, \mathbf{P}_f) \cdot \mathcal{N}(\mathbf{y}; \mathcal{H}(\mathbf{x}), \mathbf{R}) \tag{B.1}$$

After plugging in the respective Gaussian parameterizations, it is convenient to work with the negative logarithm of the analysis pdf. It can be considered as the *cost function*, as in 3DVAR:

$$J(\mathbf{x}) = -\ln\left(p(\mathbf{x}|\mathbf{y})\right) = \frac{1}{2}(\mathbf{x}-\mathbf{x}_f)^T\mathbf{P}_f^{-1}(\mathbf{x}-\mathbf{x}_f) + \frac{1}{2}(\mathbf{y}-\mathcal{H}(\mathbf{x}))^T\mathbf{R}^{-1}\left(\mathbf{y}-\mathcal{H}(\mathbf{x})\right) + c$$

c is a constant, which can be safely neglected as it only corresponds to the normalization factor of the pdf. The aim is to transform this equation into a single quadratic form. First, the observation operator is linearized around the prior estimate, using its Jacobian matrix $\mathbf{H}$,

$$\mathcal{H}(\mathbf{x}) \approx \mathcal{H}(\mathbf{x}_f) + \mathbf{H}(\mathbf{x}-\mathbf{x}_f) \tag{B.2}$$

Furthermore, the abbreviations $\mathbf{a} \equiv \mathbf{x}-\mathbf{x}_f$ and $\mathbf{b} = \mathbf{y}-\mathcal{H}(\mathbf{x}_f)$ are used. Note that while $\mathbf{a}$ depends on the variable $\mathbf{x}$, $\mathbf{b}$ represents a constant with respect to the function $J(\mathbf{x})$. Using equation (B.2), the

cost function is now written as

$$2J(\mathbf{x}) = \mathbf{a}^T\mathbf{P}_f^{-1}\mathbf{a} + (\mathbf{b}-\mathbf{H}\mathbf{a})^T\mathbf{R}^{-1}(\mathbf{b}-\mathbf{H}\mathbf{a})$$

Next, the quadratic forms are expanded and sorted

$$2J(\mathbf{x}) = \mathbf{a}^T\left(\mathbf{P}_f^{-1}+\mathbf{H}^T\mathbf{R}^{-1}\mathbf{H}\right)\mathbf{a} - \mathbf{a}^T\mathbf{H}^T\mathbf{R}^{-1}\mathbf{b} - \mathbf{b}^T\mathbf{R}^{-1}\mathbf{H}\mathbf{a} + \underbrace{\mathbf{b}^T\mathbf{R}^{-1}\mathbf{b}}_{=\text{const}}$$

Again, the constant term is neglected. The matrices $\mathbf{P}_a$ and $\mathbf{K}$ are defined as following, and some of their combinations are computed, given that $\mathbf{P}_a$ is symmetric.

$$\begin{aligned}\mathbf{P}_a &= \left(\mathbf{P}_f^{-1}+\mathbf{H}^T\mathbf{R}^{-1}\mathbf{H}\right)^{-1} \quad , \quad \mathbf{K} = \mathbf{P}_a\mathbf{H}^T\mathbf{R}^{-1} \Rightarrow \mathbf{K}^T = \mathbf{R}^{-1}\mathbf{H}\mathbf{P}_a \\ \Rightarrow \quad \mathbf{P}_a^{-1}\mathbf{K} &= \mathbf{H}^T\mathbf{R}^{-1} \quad \text{and} \quad \mathbf{K}^T\mathbf{P}_a^{-1} = \mathbf{R}^{-1}\mathbf{H}\end{aligned}$$

These terms allow a convenient reformulation of the cost function. A properly-chosen constant term is further added to complete the quadratic form:

$$\begin{aligned}2J(\mathbf{x}) &= \mathbf{a}^T\mathbf{P}_a^{-1}\mathbf{a} - \mathbf{a}^T\mathbf{P}_a^{-1}\mathbf{K}\mathbf{b} - \mathbf{b}^T\mathbf{K}^T\mathbf{P}_a^{-1}\mathbf{a} + \mathbf{b}^T\mathbf{K}^T\mathbf{P}_a^{-1}\mathbf{K}\mathbf{b} \\ \Rightarrow \quad J(\mathbf{x}) &= \frac{1}{2}(\mathbf{a}-\mathbf{K}\mathbf{b})^T\mathbf{P}_a^{-1}(\mathbf{a}-\mathbf{K}\mathbf{b})\end{aligned}$$

Resetting $\mathbf{a}$ and $\mathbf{b}$, it follows that the posteriori pdf can we written as

$$p(\mathbf{x}|\mathbf{y}) \propto \exp\left\{\frac{1}{2}(\mathbf{x}-\mathbf{x}_a)^T\mathbf{P}_a^{-1}(\mathbf{x}-\mathbf{x}_a)\right\} \propto \mathcal{N}(\mathbf{x};\mathbf{x}_a,\mathbf{P}_a) \tag{B.3}$$

$$\text{where} \quad \mathbf{x}_a = \mathbf{x}_f + \mathbf{K}(\mathbf{y}-\mathcal{H}(\mathbf{x}_f)) \quad \text{and} \quad \mathbf{P}_a = \left(\mathbf{P}_f^{-1}+\mathbf{H}^T\mathbf{R}^{-1}\mathbf{H}\right)^{-1} \tag{B.4}$$

$$\mathbf{K} = \mathbf{P}_a\mathbf{H}^T\mathbf{R}^{-1} \tag{B.5}$$

Hence, the analysis pdf is again a Gaussian distribution defined by the analysis moments $\mathbf{x}_a$ and $\mathbf{P}_a$.

B.2 Statistical derivation of the KF

This section shows an alternative derivation of the KF forecast and analysis step.

B.2.1 Prerequisites

In this section, $\mathbf{x}_i$ refers to the true state at time t_i. Of course, the true states or observations are not known. Only *estimates* about the true states are available, which are indicated by a hat, and specific realizations of the measurement. In the statistical approach, the inherent uncertainty is accounted for by allowing a random deviation from the true value, the *error*. Even though the error itself is unknown as well, assumptions about the *error statistics* can be made, that is, properties of the underlying error distribution which could only be verified by a large set of realizations. In specific terms, any estimate

is partitioned as following,

$$\hat{\mathbf{x}} = \mathbf{x} + \mathbf{e} \tag{B.6}$$

$\mathbf{e}$ represents the error. Both terms on the right hand side of the equation are not accessible. However, it is assumed that the expected mean of the error distribution, $\langle \mathbf{e} \rangle$, and its covariance, $\langle \mathbf{e}\, \mathbf{e}^T \rangle$, are known (see also appendix A). Usually, the mean error is assumed to be zero. In other words, it is expected that, on average, the state estimates and the observations are unbiased, i.e., there is no systematic difference from the truth.
In general, the KF aims at finding an optimal estimate of the system's state such that the total error variance, which is represented by the trace of the error covariance matrix, is minimized.

B.2.2 Forecast step

The forward transition by one time step, from t_{i-1} to t_i, is considered. The time t_{i-1} may represent a previous observation time, in which case $\hat{\mathbf{x}}_{i-1}$ represents the analysis estimate. However, it may also represent the outcome of a previous forecast step, since usually, several model time steps are performed between two subsequent observations. Here, the time indices of the model operator and tangent linear model are dropped as they are obvious from the context, i.e., $\mathcal{M}_{i-1\to i}(\cdot) \to \mathcal{M}(\cdot)$ and $\mathbf{M}_{i-1\to i} \to \mathbf{M}$. The following assumptions are posed to constrain the basic problem:

- The previous, unbiased state estimate is known, together with its error covariance.

$$\hat{\mathbf{x}}_{i-1} = \mathbf{x}_{i-1} + \mathbf{e}_{i-1} \quad \text{with} \quad \langle \mathbf{e}_{i-1} \rangle = \mathbf{0} \quad \text{and} \quad \langle \mathbf{e}_{i-1} \mathbf{e}_{i-1}^T \rangle = \hat{\mathbf{P}}_{i-1} \tag{B.7}$$

- The model propagates the state, and the model error captures the deviation from the truth:

$$\mathbf{x}_i = \mathcal{M}(\mathbf{x}_{i-1}) + \boldsymbol{\eta}_{i-1} \quad \text{with} \quad \langle \boldsymbol{\eta}_{i-1} \rangle = \mathbf{0} \quad \text{and} \quad \langle \boldsymbol{\eta}_{i-1} \boldsymbol{\eta}_{i-1}^T \rangle = \mathbf{Q}_{i-1} \tag{B.8}$$

- State and model error are uncorrelated, i.e., $\langle \mathbf{e}_{i-1} \boldsymbol{\eta}_{i-1}^T \rangle = \langle \boldsymbol{\eta}_{i-1} \mathbf{e}_{i-1}^T \rangle = \mathbf{0}$.
- The aim is to find an estimators for the state and covariance at time t_i,

$$\hat{\mathbf{x}}_i = \mathbf{x}_i + \mathbf{e}_i \quad \text{with} \quad \langle \mathbf{e}_i \rangle = \mathbf{0} \quad \text{and} \quad \langle \mathbf{e}_i \mathbf{e}_i^T \rangle = \hat{\mathbf{P}}_i \tag{B.9}$$

Taking the expectation of $\mathbf{x}_i$ in equation (B.8) already reveals that the unbiasedness of the model operator leads to $\hat{\mathbf{x}}_i = \mathcal{M}(\hat{\mathbf{x}}_{i-1})$. Consequently, the direct integration of the deterministic model yields the best state estimate. It remains to find its error covariance. Hence, an expression for the new error $\mathbf{e}_i$ is required, based on model error and previous error,

$$\mathbf{e}_i = \hat{\mathbf{x}}_i - \mathbf{x}_i = \mathcal{M}(\hat{\mathbf{x}}_{i-1}) - \mathbf{x}_i \tag{B.10}$$

Additionally, the model operator is linearized around the (unknown) true state $\mathbf{x}_{i-1}$,

$$\mathcal{M}(\hat{\mathbf{x}}_{i-1}) \approx \mathcal{M}(\mathbf{x}_{i-1}) + \mathbf{M}\,(\hat{\mathbf{x}}_{i-1} - \mathbf{x}_{i-1}) = \mathbf{x}_i - \boldsymbol{\eta}_{i-1} + \mathbf{M}\,\mathbf{e}_{i-1} \tag{B.11}$$

where $\mathbf{M}$ is the tangent linear model (see appendix A). Plugging this back into equation (B.10) yields an approximate expression of the error without an explicit reference to the true state:

$$\mathbf{e}_i = \mathbf{M}\,\mathbf{e}_{i-1} - \boldsymbol{\eta}_{i-1} \quad \Rightarrow \quad \mathbf{e}_i^T = \mathbf{e}_{i-1}^T\mathbf{M}^T - \boldsymbol{\eta}_{i-1}^T \tag{B.12}$$

The calculation of its covariance is now straightforward:

$$\begin{aligned} \hat{\mathbf{P}}_i &= \left\langle \mathbf{e}_i\mathbf{e}_i^T \right\rangle = \left\langle \left(\mathbf{M}\,\mathbf{e}_{i-1} - \boldsymbol{\eta}_{i-1}\right)\left(\mathbf{e}_{i-1}^T\mathbf{M}^T - \boldsymbol{\eta}_{i-1}^T\right) \right\rangle & \text{(B.13)} \\ &= \mathbf{M}\left\langle \mathbf{e}_{i-1}\mathbf{e}_{i-1}^T \right\rangle\mathbf{M}^T + \left\langle \boldsymbol{\eta}_{i-1}\boldsymbol{\eta}_{i-1}^T \right\rangle = \mathbf{M}\hat{\mathbf{P}}_{i-1}\mathbf{M}^T + \mathbf{Q}_{i-1} & \text{(B.14)} \end{aligned}$$

The vanishing correlation between analysis and model error allows to remove all corresponding cross-covariances. The final equation shows how the error covariance matrix is approximately propagated during the forecast step. The direct application of this equation requires both the storage and the integration of all elements of the covariance matrix, which is prohibitive for large-scale DA. In Chapter 2, the issues arising from the covariance update equation are discussed in more detail.

B.2.3 Analysis step

The analysis step is derived at an arbitrary observation time $t_{i(j)}$, $j \in \{1, \ldots, J\}$. The time index is dropped here for clarity, as all quantities are valid at the same time. The input to the KF analysis step are a prior state estimate, $\mathbf{x}_f := \tilde{\hat{\mathbf{x}}}_{i(j)}$, and a prior estimate of the error covariance, $\mathbf{P}_f := \tilde{\hat{\mathbf{P}}}_{i(j)}$. Usually, they result from a preceding forecast step. The aim is to find the posteriori estimate, the analysis $\mathbf{x}_a := \hat{\mathbf{x}}_{i(j)}$, together with its error covariance $\mathbf{P}_a := \hat{\mathbf{P}}_{i(j)}$. The following list summarizes all quantities and the underlying assumptions that are needed.

- The prior $\mathbf{x}_f$ is assumed to be an unbiased estimate of the true state $\mathbf{x}$:

$$\mathbf{x}_f = \mathbf{x} + \mathbf{e}_f \quad \text{with} \quad \langle \mathbf{e}_f \rangle = \mathbf{0} \quad \text{and} \quad \langle \mathbf{e}_f\mathbf{e}_f^T \rangle = \mathbf{P}_f \tag{B.15}$$

- The observation error $\mathbf{e}_o$ contains both measurement and representativeness error. The measurement $\mathbf{y}$ is an unbiased realization of the true observation and is characterized by an error covariance $\mathbf{R}$. Additionally, the observation operator is supposed to be unbiased.

$$\mathbf{y} = \mathcal{H}(\mathbf{x}) + \mathbf{e}_o \quad \text{with} \quad \langle \mathbf{e}_o \rangle = \mathbf{0} \quad \text{and} \quad \langle \mathbf{e}_o\mathbf{e}_o^T \rangle = \mathbf{R} \tag{B.16}$$

- Observation and state errors are uncorrelated, i.e., $\langle \mathbf{e}_o\mathbf{e}_f^T \rangle = \langle \mathbf{e}_f\mathbf{e}_o^T \rangle = \mathbf{0}$.
- The aim is to find an unbiased analysis and its error covariance.

$$\mathbf{x}_a = \mathbf{x} + \mathbf{e}_a \quad \text{with} \quad \langle \mathbf{e}_a \rangle = \mathbf{0} \quad \text{and} \quad \langle \mathbf{e}_a\mathbf{e}_a^T \rangle = \mathbf{P}_a \tag{B.17}$$

First, the analysis is expressed as a linear combination of all available information,

$$\mathbf{x}_a = \mathbf{A}_1\mathbf{x}_f + \mathbf{A}_2\mathcal{H}(\mathbf{x}_f) + \mathbf{A}_3\mathbf{y} + \mathbf{A}_4 \tag{B.18}$$

Here, $\mathbf{A}_1, \ldots, \mathbf{A}_4$ are yet unknown matrices of appropriate dimensions. Taking the expectation of this equation leads to the condition

$$\mathbf{x} = \mathbf{A}_1\mathbf{x} + \mathbf{A}_2\mathcal{H}(\mathbf{x}) + \mathbf{A}_3\mathcal{H}(\mathbf{x}) + \mathbf{A}_4 \tag{B.19}$$

This equation has to hold for an arbitrary $\mathbf{x}$. Therefore, a comparison of both sides shows that $\mathbf{A}_4$ has to vanish ($\mathbf{A}_4 = \mathbf{0}$), $\mathbf{A}_1 = \mathbf{I}$ and $\mathbf{A}_2 = -\mathbf{A}_3$. Renaming $\mathbf{A}_3 \equiv \mathbf{K}$ allows a convenient reformulation of the linear update equation (B.18):

$$\mathbf{x}_a = \mathbf{x}_f - \mathbf{K}\mathcal{H}(\mathbf{x}_f) + \mathbf{K}\mathbf{y} = \mathbf{x}_f + \mathbf{K}\left(\mathbf{y} - \mathcal{H}(\mathbf{x}_f)\right) \tag{B.20}$$

The second term is the *increment* which is added to the first guess. It remains to find the matrix $\mathbf{K}$, the *Kalman gain*, which is applied to the difference between the actual and the predicted observation, called the *innovation*. The regularization imposed to determine the Kalman gain is that the analysis should have a minimal total variance. Hence, first an equation for the analysis error $\mathbf{e}_a$ is needed. The nonlinear observation operator can be dealt with by a linearization around the true state,

$$\mathcal{H}(\mathbf{x}_f) \approx \mathcal{H}(\mathbf{x}) + \mathbf{H}(\mathbf{x}_f - \mathbf{x}) = \mathcal{H}(\mathbf{x}) + \mathbf{H}\mathbf{e}_f \tag{B.21}$$

with the tangent linear operator $\mathbf{H}$. This approximation is used in the update equation (B.20), and the true state $\mathbf{x}$ is subtracted on both sides to obtain the analysis error $\mathbf{e}_a$ and its transpose:

$$\begin{aligned}
\mathbf{x}_a - \mathbf{x} &= \mathbf{x}_f - \mathbf{x} + \mathbf{K}\left(\mathcal{H}(\mathbf{x}) + \mathbf{e}_o - \mathcal{H}(\mathbf{x}) - \mathbf{H}\mathbf{e}_f\right) && \text{(B.22)}\\
\mathbf{e}_a &= \mathbf{e}_f + \mathbf{K}\left(\mathbf{e}_o - \mathbf{H}\mathbf{e}_f\right) = (\mathbf{I} - \mathbf{K}\mathbf{H})\mathbf{e}_f + \mathbf{K}\mathbf{e}_o && \text{(B.23)}\\
\Rightarrow \quad \mathbf{e}_a^T &= \mathbf{e}_f^T(\mathbf{I} - \mathbf{H}^T\mathbf{K}^T) + \mathbf{e}_o^T\mathbf{K}^T && \text{(B.24)}
\end{aligned}$$

This allows to evaluate the analysis error covariance. In the following, the cross-terms vanish since observation error and prior error are uncorrelated by assumption:

$$\begin{aligned}
\mathbf{P}_a &= \left\langle \mathbf{e}_a\mathbf{e}_a^T \right\rangle = \left\langle \left[(\mathbf{I} - \mathbf{K}\mathbf{H})\mathbf{e}_f + \mathbf{K}\mathbf{e}_o\right]\left[\mathbf{e}_f^T(\mathbf{I} - \mathbf{H}^T\mathbf{K}^T) + \mathbf{e}_o^T\mathbf{K}^T\right] \right\rangle && \text{(B.25)}\\
&= (\mathbf{I} - \mathbf{K}\mathbf{H})\left\langle \mathbf{e}_f\mathbf{e}_f^T \right\rangle(\mathbf{I} - \mathbf{H}^T\mathbf{K}^T) + \mathbf{K}\left\langle \mathbf{e}_o\mathbf{e}_o^T \right\rangle\mathbf{K}^T && \text{(B.26)}\\
&= (\mathbf{I} - \mathbf{K}\mathbf{H})\mathbf{P}_f(\mathbf{I} - \mathbf{H}^T\mathbf{K}^T) + \mathbf{K}\mathbf{R}\mathbf{K}^T && \text{(B.27)}
\end{aligned}$$

Expanding the terms leads to

$$\mathbf{P}_a = \mathbf{P}_f - \mathbf{K}\mathbf{H}\mathbf{P}_f - \mathbf{P}_f\mathbf{H}^T\mathbf{K}^T + \mathbf{K}(\mathbf{H}\mathbf{P}_f\mathbf{H}^T + \mathbf{R})\mathbf{K}^T \tag{B.28}$$

The aim is to find $\mathbf{K}$ such that it minimizes the total analysis variance, i.e., the trace of $\mathbf{P}_a$. Hence, the derivative of the trace with respect to $\mathbf{K}$, which can be derived using the identities shown in appendix A.1.3, has to be the null matrix. Additionally, covariance matrices are symmetric, i.e., they are equal to their transposes.

$$\begin{aligned}
\frac{\partial \mathrm{tr}(\mathbf{P}_a)}{\partial \mathbf{K}} &= -(\mathbf{H}\mathbf{P}_f)^T - \mathbf{P}_f\mathbf{H}^T + \mathbf{K}\left[\mathbf{H}\mathbf{P}_f\mathbf{H}^T + \mathbf{R} + (\mathbf{H}\mathbf{P}_f\mathbf{H}^T)^T + \mathbf{R}^T\right] && \text{(B.29)}\\
\mathbf{0} &= 2\mathbf{K}(\mathbf{H}\mathbf{P}_f\mathbf{H}^T + \mathbf{R}) - 2\mathbf{P}_f\mathbf{H}^T && \text{(B.30)}
\end{aligned}$$

This linear equation can be easily reversed to solve for the Kalman gain:

$$\mathbf{K} = \mathbf{P}_f\mathbf{H}^T\left(\mathbf{H}\mathbf{P}_f\mathbf{H}^T + \mathbf{R}\right)^{-1} \tag{B.31}$$

When this optimal Kalman gain is used in equation (B.28), the last two terms cancel out,

$$\mathbf{P}_a = \mathbf{P}_f - \mathbf{K}\mathbf{H}\mathbf{P}_f - \mathbf{P}_f\mathbf{H}^T\mathbf{K}^T + \underbrace{\mathbf{K}(\mathbf{H}\mathbf{P}_f\mathbf{H}^T + \mathbf{R})}_{=\mathbf{P}_f\mathbf{H}^T}\mathbf{K}^T \tag{B.32}$$

$$\Rightarrow \quad \mathbf{P}_a = (\mathbf{I} - \mathbf{K}\mathbf{H})\,\mathbf{P}_f \tag{B.33}$$

In summary, equations (B.20), (B.31) and (B.33) constitute the final update equations, as they indicate how to adjust the prior mean and covariance. They are identical to the equations derived in Chapter 2 within the probabilistic framework if Gaussian densities are assumed. In the derivation shown here, no parametric assumption about the error distribution has been made, however, it is required to know their second-order statistics. Hence, the Gaussian density represents the natural parameterization of this information.

B.2.4 Summary and discussion

The KF equations, that are also the fundament of the EnKF or ETKF, have been derived from a statistical point of view by making some basic assumptions about the error distributions. The KF determines the analysis that minimizes the total error variance. Thus, given the Gaussian assumption, the minimum variance estimator equals both the maximum likelihood and least squares estimators. The latter follows from the equality of the 3DVAR and KF analysis (see below). However, the KF also explicitly estimates the covariances, which is useful for diagnostic reasons. Nevertheless, it has some drawbacks, particularly concerning its practical computation in larger-dimensional systems, as discussed more elaborately in Chapter 2.
This non-probabilistic derivation has revealed some further properties inherently built into the KF that are not directly revealed in the probabilistic approach:

- The KF requires the prior estimates, the model, the observation operator and the observation itself to be unbiased. In other words, no systematic deviations from the truth are assumed, which in reality is hard to ensure.
- Particularly the unbiasedness of the model is important, as only one single state is integrated in time. However, in a nonlinear system, a forward integration of the best estimate does not necessarily deliver the best estimate at the next time step.
- A linearized model integration is used to advance the covariance matrix in time. The validity of the linear approximation decreases with the nonlinearity of the model and the length of the forecast step. This may lead to a poor estimate of the prior covariance matrix, and the analysis is quite sensitive to the specification of $\mathbf{P}_f$ when forming the Kalman gain.

B.3 KF analysis step and 3DVAR

In section 2.6.3, it is noticed that the 3DVAR cost function equals the negative logarithm of Bayes' theorem, which also appears in the derivation of the KF analysis step. It therefore remains to establish formally the connection between the variational and the sequential approach.
For this purpose, the 3DVAR cost function as given by equation (2.30), which is usually minimized numerically, is now minimized analytically by setting its gradient, equation (2.31), equal to zero. The 3DVAR background state and its error covariance can be identified with the corresponding prior estimates that are the input to the KF analysis step. That is, here $\mathbf{x}_b$ and $\mathbf{B}$ are the analog to $\mathbf{x}_f$ and $\mathbf{P}_f$, respectively, which should be kept in mind when interpreting the results below. The minimization problem is

$$\min_{\mathbf{x}} \quad \left\{ \frac{1}{2}(\mathbf{x}-\mathbf{x}_b)^T\mathbf{B}^{-1}(\mathbf{x}-\mathbf{x}_b) + \frac{1}{2}(\mathbf{y}-\mathcal{H}(\mathbf{x}))^T\mathbf{R}^{-1}(\mathbf{y}-\mathcal{H}(\mathbf{x})) \right\} \tag{B.34}$$

$$\Rightarrow \qquad \mathbf{B}^{-1}(\mathbf{x}-\mathbf{x}_b) + \mathbf{H}^T\mathbf{R}^{-1}\left(\mathcal{H}(\mathbf{x})-\mathbf{y}\right) = \mathbf{0} \tag{B.35}$$

The observation operator is linearized around the background state,

$$\mathcal{H}(\mathbf{x}) \approx \mathcal{H}(\mathbf{x}_b) + \mathbf{H}(\mathbf{x}-\mathbf{x}_b) \tag{B.36}$$

This allows to proceed with equation (B.35), and the solution is denoted $\mathbf{x}_a$:

$$\begin{aligned}
& \mathbf{B}^{-1}(\mathbf{x}_a-\mathbf{x}_b) + \mathbf{H}^T\mathbf{R}^{-1}\left(\mathcal{H}(\mathbf{x}_b) + \mathbf{H}(\mathbf{x}_a-\mathbf{x}_b) - \mathbf{y}\right) = \mathbf{0} \\
\Leftrightarrow \quad & \left(\mathbf{B}^{-1} + \mathbf{H}^T\mathbf{R}^{-1}\mathbf{H}\right)(\mathbf{x}_a-\mathbf{x}_b) = \mathbf{H}^T\mathbf{R}^{-1}\left(\mathbf{y}-\mathcal{H}(\mathbf{x}_b)\right) \\
\Leftrightarrow \quad & \mathbf{x}_a = \mathbf{x}_b + \underbrace{\left(\mathbf{B}^{-1} + \mathbf{H}^T\mathbf{R}^{-1}\mathbf{H}\right)^{-1}\mathbf{H}^T\mathbf{R}^{-1}}_{\equiv\tilde{\mathbf{K}}}\left(\mathbf{y}-\mathcal{H}(\mathbf{x}_b)\right)
\end{aligned}$$

The resulting update equation has a structure that resembles the KF solution as in equation (B.4), as the increment is determined by a weight matrix $\tilde{\mathbf{K}}$ acting on the innovation vector. Even though this matrix looks different at first glance, the identity to $\mathbf{K}$ can be shown, see section B.4.1. Thus, if the 3DVAR background estimates are associated with the prior moments of the KF analysis step, both methods provide, despite their entirely different mechanisms, exactly the same analysis mean and covariance. Given the Gaussian assumption, the Bayesian solution therefore equals the least-squares estimator.

B.4 Identities

Here, it is shown that the different representations for the Kalman gain and the analysis covariance derived above actually are identical.

B.4.1 Kalman gain matrix

The two alternative formulations are given by equations (B.31) and (B.5). In order to distinguish the different expressions here, the latter is denoted as $\tilde{\mathbf{K}}$.

$$\mathbf{K} = \mathbf{P}_f\mathbf{H}^T\left(\mathbf{H}\mathbf{P}_f\mathbf{H}^T + \mathbf{R}\right)^{-1} \tag{B.37}$$

$$\tilde{\mathbf{K}} = \left(\mathbf{P}_f^{-1} + \mathbf{H}^T\mathbf{R}^{-1}\mathbf{H}\right)^{-1}\mathbf{H}^T\mathbf{R}^{-1} \tag{B.38}$$

In order to show their equality, some matrix manipulations are applied to $\tilde{\mathbf{K}}$:

$$\begin{aligned}\left(\mathbf{P}_f^{-1} + \mathbf{H}^T\mathbf{R}^{-1}\mathbf{H}\right)\tilde{\mathbf{K}} &= \mathbf{H}^T\mathbf{R}^{-1} = \mathbf{H}^T\mathbf{R}^{-1}\left(\mathbf{H}\mathbf{P}_f\mathbf{H}^T + \mathbf{R}\right)\left(\mathbf{H}\mathbf{P}_f\mathbf{H}^T + \mathbf{R}\right)^{-1} \\ &= \left(\mathbf{H}^T\mathbf{R}^{-1}\mathbf{H}\mathbf{P}_f\mathbf{H}^T + \mathbf{P}_f^{-1}\mathbf{P}_f\mathbf{H}^T\right)\left(\mathbf{H}\mathbf{P}_f\mathbf{H}^T + \mathbf{R}\right)^{-1} \\ &= \left(\mathbf{H}^T\mathbf{R}^{-1}\mathbf{H} + \mathbf{P}_f^{-1}\right)\mathbf{P}_f\mathbf{H}^T\left(\mathbf{H}\mathbf{P}_f\mathbf{H}^T + \mathbf{R}\right)^{-1}\end{aligned}$$

Now, the matrix $\left(\mathbf{H}^T\mathbf{R}^{-1}\mathbf{H} + \mathbf{P}_f^{-1}\right)$ is removed on both sides and the identity is confirmed:

$$\tilde{\mathbf{K}} = \left(\mathbf{P}_f^{-1} + \mathbf{H}^T\mathbf{R}^{-1}\mathbf{H}\right)^{-1}\mathbf{H}^T\mathbf{R}^{-1} = \mathbf{P}_f\mathbf{H}^T\left(\mathbf{H}\mathbf{P}_f\mathbf{H}^T + \mathbf{R}\right)^{-1} = \mathbf{K} \tag{B.39}$$

While an inverse matrix is computed for $\mathbf{K}$ in observation space, the inversion needed for $\tilde{\mathbf{K}}$ is performed in state space. Hence, for large-scale applications, the form $\mathbf{K}$ as in equation (B.37) is computationally more efficient. The identity (B.39) derived here is actually a variant of the Sherman-Morrison-Woodbury formula (Kalnay 2003, p. 195).

B.4.2 Analysis covariance matrix

The two alternative formulations are given by equations (B.33) and (B.4), denoting the latter as $\tilde{\mathbf{P}}_a$:

$$\mathbf{P}_a = (\mathbf{I} - \mathbf{K}\mathbf{H})\mathbf{P}_f \tag{B.40}$$

$$\tilde{\mathbf{P}}_a = \left(\mathbf{P}_f^{-1} + \mathbf{H}^T\mathbf{R}^{-1}\mathbf{H}\right)^{-1} \tag{B.41}$$

These equations can be reformulated as follows,

$$\mathbf{K}\mathbf{H} = \mathbf{I} - \mathbf{P}_a\mathbf{P}_f^{-1} \tag{B.42}$$

$$\mathbf{H}^T\mathbf{R}^{-1}\mathbf{H} = \tilde{\mathbf{P}}_a^{-1} - \mathbf{P}_f^{-1} \tag{B.43}$$

Next, equation (B.5) is multiplied by $\mathbf{H}$ (to the left) and then, equations (B.42) and (B.43) are applied:

$$\mathbf{K}\mathbf{H} = \tilde{\mathbf{P}}_a\mathbf{H}^T\mathbf{R}^{-1}\mathbf{H} \tag{B.44}$$

$$\Leftrightarrow \quad \mathbf{I} - \mathbf{P}_a\mathbf{P}_f^{-1} = \tilde{\mathbf{P}}_a(\tilde{\mathbf{P}}_a^{-1} - \mathbf{P}_f^{-1}) \tag{B.45}$$

$$\Leftrightarrow \quad \mathbf{P}_a = \tilde{\mathbf{P}}_a \tag{B.46}$$

This confirms that both representations of the KF analysis covariance matrix are identical.

Appendix C

Second-order exact sampling with random rotations

Second-order exact sampling provides a simple, yet efficient framework to generate rotation matrices or perturbations exhibiting a specified covariance. In the context of ensemble square root filters, rotations in ensemble space, particularly that of random nature, appear because the transform matrix to generate the analysis perturbations is not unique. Furthermore, the framework allows for a practical method to generate an ensemble at time t_0, which is required to initialize any ensemble-based DA scheme.
This appendix first reviews the usage of rotation matrices in DA and then presents explicit algorithms for the generation of random and deterministic rotation matrices. These algorithms are used for the experiments in chapters 5 and 6. Then, the equations used to generate the initial ensembles in these experiments are shown. The presentation here has the purpose to facilitate other numerical implementations of the NETF since the random rotation has been proven to be an essential component of the filter.

C.1 Background

In ensemble square root filters, such as the ETKF and NETF, the key equation transforms the forecast perturbations into analysis perturbations,

$$\mathbf{X}'_a = \mathbf{X}'_f \mathbf{T} \tag{C.1}$$

The transform matrix $\mathbf{T}$ (of size $N \times N$) is designed such that the empirical analysis covariance, $\mathbf{P}_a = \mathbf{X}'_a(\mathbf{X}'_a)^T/(N-1) = \mathbf{X}'_f \mathbf{T}\mathbf{T}^T(\mathbf{X}'_f)^T$, exactly yields a specified term. The transformation can be enhanced by an additional $N \times N$ matrix $\mathbf{\Lambda}$, the so-called rotation matrix. The ensemble covariance gets

$$\mathbf{P}_a = \frac{1}{N-1}\mathbf{X}'_a(\mathbf{X}'_a)^T = \frac{1}{N-1}\mathbf{X}'_f \mathbf{T}\mathbf{\Lambda}\mathbf{\Lambda}^T\mathbf{T}^T(\mathbf{X}'_f)^T \tag{C.2}$$

Thus, $\mathbf{P}_a$ is unchanged if the rotation matrix exhibits $\mathbf{\Lambda}\mathbf{\Lambda}^T = \mathbf{I}_N$, i.e., it has to be orthogonal (see Appendix A). Additionally, the ensemble perturbations should still be centered around zero, which can be expressed as condition

$$\mathbf{X}'_a\mathbf{1} = \mathbf{X}'_f\mathbf{T}\mathbf{\Lambda}\mathbf{1} = \mathbf{0} \tag{C.3}$$

As $\mathbf{X}'_f\mathbf{1} = \mathbf{1}$ is given by construction, and $\mathbf{T}\mathbf{1} = \mathbf{1}$ (Sakov and Oke 2008b) or $\mathbf{T}\mathbf{1} = \mathbf{0}$ (see section 4.3.1), the vector $\mathbf{1}$ has to be an eigenvector of $\mathbf{\Lambda}$ in order to conserve the ensemble mean, i.e., $\mathbf{\Lambda}\mathbf{1} = \alpha\mathbf{1}$. As $\mathbf{\Lambda}$ is orthogonal, the eigenvalue is necessarily one, since an orthogonal matrix conserves as vector's norm: $|\alpha\mathbf{1}| = \mathbf{\Lambda}^T\mathbf{1}^T\mathbf{\Lambda}\mathbf{1} = \mathbf{1}^T\mathbf{1} = |\mathbf{1}|$ implies $\alpha = 1$. In summary, the additional matrix $\mathbf{\Lambda}$ can be chosen arbitrarily as long as it is orthogonal and $\mathbf{1}$ is included in its eigenspace. Hence, it is possible to use a random matrix that meets these two requirements, but there are also some deterministic matrices that can be applied. It is emphasized that the application of such a rotation changes the analysis ensemble, but its mean and covariance are conserved. Several studies pointed out that the ensemble exhibits a Gaussian characteristics after rotation (Evensen 2009), if the rotation matrices shown next are used.

C.2 Generation of random rotation matrices

This section shows an explicit method to generate a suitable random matrix, based on the algorithm summarized by Nerger et al. (2012b). Additionally, it is proven that the resulting matrix actually is orthogonal and mean-preserving. In mathematical terms, the aim is to generate a matrix that exhibits $\mathbf{\Lambda}^T\mathbf{\Lambda} = \mathbf{\Lambda}\mathbf{\Lambda}^T = \mathbf{I}$ and $\mathbf{\Lambda}\mathbf{1} = \mathbf{1}$. It is shown how to generate both random and deterministic matrices with these properties. These algorithms are also implemented similarly in the parallel DA framework (PDAF, see Chapter 6).

C.2.1 Householder matrix

In this appendix, the matrix $\mathbf{H}$ does *not* refer to the tangent linear observation operator but to a *Householder matrix*. A Householder matrix associated with a vector $\mathbf{a}$ of length n and unit norm ($\mathbf{a}^T\mathbf{a} = \sum_{m=1}^{n} a_m^2 = 1$) is a square matrix of size $n \times n$ given by

$$\mathbf{H}(\mathbf{a}) = \mathbf{I}_n - \frac{1}{|a_n| + 1}\mathbf{a}^s(\mathbf{a}^s)^T \tag{C.4}$$

Here, $\mathbf{a}^s$ is equal to $\mathbf{a}$, except for the last element a_n^s, which is replaced according to

$$a_n^s = a_n + \mathrm{sgn}(a_n) = \mathrm{sgn}(a_n)(|a_n| + 1) = \begin{cases} |a_n| + 1 & \text{for } a_n > 0 \\ -(|a_n| + 1) & \text{for } a_n < 0 \end{cases}$$

The $\mathrm{sgn}(\cdot)$ function outputs $+1$ or -1, depending on the sign of its argument. The norm of $\mathbf{a}^s$ can be simplified to $(\mathbf{a}^s)^T\mathbf{a}^s = \sum_{m=1}^{n-1} a_m^2 + (a_n + \mathrm{sgn}(a_n))^2 = \sum_{m=1}^{n} a_m^2 + 2a_n\mathrm{sgn}(a_n) + 1 = 2(1 + |a_n|)$.[1]

[1] $a_n\mathrm{sgn}(a_n) = |a_n|$ can be verified by looking at all three possible cases ($a_n > 0$, $a_n < 0$, $a_n = 0$).

It is directly visible that the Householder matrix is symmetric. Thus,

$$\mathbf{H}\mathbf{H}^T = \mathbf{H}^2 = \mathbf{I}_n - \frac{2}{|a_n|+1}\mathbf{a}^s(\mathbf{a}^s)^T + \frac{1}{(|a_n|+1)^2}\mathbf{a}^s \underbrace{(\mathbf{a}^s)^T\mathbf{a}^s}_{=2(1+|a_n|)}(\mathbf{a}^s)^T = \mathbf{I}_n$$

which proves that it is also orthogonal. Another important property is revealed by a closer investigation of the matrix' last column:

$$\text{last col. of } \mathbf{H} = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ 1 \end{pmatrix} - \frac{1}{|a_n|+1}\begin{pmatrix} a_1 \\ \vdots \\ a_{n-1} \\ a_n + \text{sgn}(a_n) \end{pmatrix} \cdot (a_n + \text{sgn}(a_n)) = -\text{sgn}(a_n) \cdot \mathbf{a}$$

Hence, the last column is either $\pm\mathbf{a}$. As the orthogonality of $\mathbf{H}$ implies that all of its columns are orthogonal to each other, it follows that the first $n-1$ columns of the Householder matrix defined by equation (C.4) are orthogonal to $\mathbf{a}$.

C.2.2 $N \times (N-1)$ rotation matrix

The first algorithm generates a random, mean-preserving $N \times (N-1)$ matrix $\mathbf{\Omega}$. Under the name "second-order exact sampling", Pham (2001) introduced this technique in more detail.

1. Set $\mathbf{\Omega}_1$ as a 1×1 matrix whose entry is given by either $+1$ or -1 with equal probability.

2. Compose the $n \times n$ matrix $\mathbf{\Omega}_n$ recursively for $n = 2 \dots (N-1)$ as following:

$$\mathbf{\Omega}_n = \begin{pmatrix} \tilde{\mathbf{H}}(\mathbf{a}_n)\mathbf{\Omega}_{n-1} & \mathbf{a}_n \end{pmatrix} \tag{C.5}$$

   Here, $\mathbf{a}_n$ is a random vector on the unit sphere in n dimensions, which is obtained by drawing n samples from the univariate standard normal density, collecting them into a vector $\mathbf{a}_n$ and normalizing it such that $||\mathbf{a}_n|| = 1$. $\tilde{\mathbf{H}}$ represents the first $n-1$ columns of the Householder matrix $\mathbf{H}$.

3. Construct the final matrix via

$$\mathbf{\Omega} = \tilde{\mathbf{H}}(\mathbf{a}_N)\mathbf{\Omega}_{N-1} \tag{C.6}$$

   where $\mathbf{a}_N = \mathbf{1}/\sqrt{N}$.

If in contrast a deterministic $N \times (N-1)$ rotation matrix is needed, a simple choice (Nerger et al. 2012b) is given by the first $N-1$ columns of $\mathbf{H}(\mathbf{a}_N)$, i.e, $\tilde{\mathbf{H}}(\mathbf{a}_N)$. Other choices of deterministic rotation matrices can be found in the literature (e.g., Hoteit et al. 2002; Wang et al. 2004).

C.2.3 $N \times N$ rotation matrix

The following steps transform the random $N \times (N-1)$ matrix $\mathbf{\Omega}$ into a proper $N \times N$ rotation matrix $\mathbf{\Lambda}$. For this purpose, the suggestion of Sakov and Oke (2008b) can be utilized. It is summarized by the following algorithm:

1. Use the algorithm presented above, but omit the last step, resulting in an orthogonal $(N-1) \times (N-1)$ matrix $\mathbf{\Omega} \equiv \mathbf{\Omega}_{N-1}$. Note that one further application of this algorithm would indeed generate an orthogonal random $N \times N$ matrix, but it would not have the mean-preserving property required here.

2. Create an arbitrary orthogonal $N \times N$ matrix $\mathbf{B} = (\mathbf{b}_1, \ldots, \mathbf{b}_N)$ having $\mathbf{b}_1 = 1/\sqrt{N}(1, \ldots, 1)^T$ as first basis vector. An easy method to generate such a matrix is to calculate the full Householder matrix associated with this vector, $\mathbf{H}(\mathbf{b}_1)$, which has $-\mathbf{b}_1$ as its last column. Then, one simply constructs $\mathbf{B}$ by combining the column vector $\mathbf{b}_1$ and the first $(N-1)$ columns of $\mathbf{H}(\mathbf{b}_1)$,

$$\mathbf{B} = \begin{pmatrix} \mathbf{b}_1 & \tilde{\mathbf{H}}(\mathbf{b}_1) \end{pmatrix} \tag{C.7}$$

   Note that the interchange of columns and reversal of sign of the first column does not affect the orthogonality of the matrix.

3. Compose the random rotation matrix in the basis $\mathbf{B}$ as

$$\mathbf{\Lambda}_\mathbf{B} = \begin{pmatrix} 1 & 0 \\ 0 & \mathbf{\Omega} \end{pmatrix} \tag{C.8}$$

4. Perform the inverse basis transformation to obtain the final random rotation matrix:

$$\mathbf{\Lambda} = \mathbf{B}\mathbf{\Lambda}_\mathbf{B}\mathbf{B}^T \tag{C.9}$$

An alternative for the first step is to use the orthogonal matrix $\mathbf{V}^T$ from the singular value decomposition of a random $(N-1) \times (N-1)$ matrix $\mathbf{\Omega}$, i.e., $\mathbf{\Omega} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$ (Evensen 2004).

If in contrast a deterministic $N \times N$ rotation matrix is needed, step 1 is replaced by simply specifying the Householder matrix associated with the vector $\mathbf{a}_{N-1} = \mathbf{1}/\sqrt{N-1}$ (of length $N-1$), i.e., $\mathbf{\Omega} \equiv \mathbf{H}(\mathbf{a}_{N-1})$. Then, steps 2 to 4 are carried out as described above.

C.2.4 Proofs

For the following proofs, we assume that the $N \times (N-1)$ matrix $\mathbf{\Omega}$ is orthogonal and mean-preserving, i.e., $\mathbf{\Omega}^T\mathbf{\Omega} = \mathbf{I}$ and $\mathbf{\Omega}\mathbf{1} = \mathbf{1}$ hold, as shown in detail by Pham (2001).

First, we deal with the orthogonality of $\mathbf{\Lambda}$. Equation (C.9) directly yields $\mathbf{\Lambda}^T = \mathbf{B}\mathbf{\Lambda}_\mathbf{B}^T\mathbf{B}^T$. Since $\mathbf{B}$ is orthogonal, $\mathbf{\Lambda}\mathbf{\Lambda}^T = \mathbf{B}\mathbf{\Lambda}_\mathbf{B}\mathbf{\Lambda}_\mathbf{B}^T\mathbf{B}^T$ follows. Equation (C.8) reveals that the orthogonality of $\mathbf{\Omega}$ implies that $\mathbf{\Lambda}_\mathbf{B}$ is orthogonal, and consequently, $\mathbf{\Lambda}\mathbf{\Lambda}^T = \mathbf{I}$.

Second, we show the mean preservation property. $\mathbf{\Lambda 1} = \mathbf{B}\mathbf{\Lambda}_\mathbf{B}^T\mathbf{B}^T\mathbf{1} = \mathbf{1}$ can be shown by successive evaluation of the matrix-vector products. First, $\mathbf{B}^T\mathbf{1} = (\sqrt{N}\,, \mathbf{0})^T$ because the first column vector of $\mathbf{B}$ is $\mathbf{1}/\sqrt{N}$, and due to its orthogonality, all other column vectors are orthogonal to $\mathbf{1}$. Second, the matrix $\mathbf{\Lambda_B}$ leaves $(\sqrt{N}\,, \mathbf{0})^T$ unchanged due to its special form (C.8). Finally, $\mathbf{B} \cdot (\sqrt{N}\,, \mathbf{0})^T = \mathbf{1}$ because all entries in its first column are equal to $1/\sqrt{N}$.

C.3 Ensemble generation via EOF decomposition

In the previous section, a class of random matrix transformations was shown that does not alter the first two moments if applied to an ensemble matrix. In combination with principal component analysis (PCA, Wilks 2011), the same technique can be used to construct an ensemble that exhibits a given mean and covariance, for example, to initialize the DA cycle. This section provides the details to generate suitable ensemble perturbations $\mathbf{X}'_0$ that have zero row sums. Then, the correct ensemble mean is easily achieved by adding the desired mean state to the perturbations.

C.3.1 Decomposition of the covariance matrix

First, assume that the ensemble to be generated is supposed to have a specified covariance $\mathbf{P}_0$. An eigenvalue decomposition, $\mathbf{P}_0 = \mathbf{U}\mathbf{E}\mathbf{U}^T$, supplies its eigenvectors (in $\mathbf{U}$) and eigenvalues (in $\mathbf{E}$). An ensemble of size N may now be constructed via

$$\mathbf{X}'_0 = \sqrt{N-1}\,\mathbf{U}\mathbf{E}^{1/2}\mathbf{\Omega}^T$$

Usually, when $N \leq K$, $\mathbf{\Omega}$ is a $N \times (N-1)$ random rotation matrix, and only the first $N-1$ eigenvectors and eigenvalues are taken to from $\mathbf{U}$ and $\mathbf{E}$. In this case, the empirical ensemble covariance is a low-rank $(N-1)$ approximation of the input covariance $\mathbf{P}_0$, which can be easily verified by computing $\frac{1}{N-1}\mathbf{X}'_0(\mathbf{X}'_0)^T$. Otherwise, if $N \geq K+1$, only the first K columns of $\mathbf{\Omega}$ are required, using all eigenvectors and values, which yields an ensemble of full rank.

C.3.2 Principal component analysis

However, in large-scale applications it is unlikely that a covariance matrix $\mathbf{P}_0$ can be specified explicitly. Here, a more practical approach is offered based on PCA. Suppose there are M model states available, stored in $\mathbf{X}_c$, for example from past model simulations. They may represent a climatological sample of the system and carry information about its error covariance structure. As an advantage, such states are not only easily available also in real-world problems, but they also represent balanced states of the system. Therefore, it is desirable to utilize them for the generation of a new ensemble of size $N \leq M+1$. Simply choosing N states from the M-sized sample would be a valid approach, but in this case the ensemble would only approach the desired covariance and mean for $N \to M$. Instead, the second order exact sampling framework is applied as follows.

The empirical covariance of the M states is $\mathbf{P}_0 = \frac{1}{M-1}\mathbf{X}'_c{\mathbf{X}'_c}^T$, but it is never calculated explicitly. Instead, a singular value decomposition (SVD) is applied to the perturbation matrix, yielding $\mathbf{X}'_c = \mathbf{U}\mathbf{S}\mathbf{V}^T$. For convenience, $\tilde{\mathbf{S}} = \frac{1}{\sqrt{M-1}}\mathbf{S}$ is defined as the diagonal matrix that contains the scaled singular values. Then, due the orthogonality of $\mathbf{V}$, the covariance can be written as $\mathbf{P}_0 = \mathbf{U}\tilde{\mathbf{S}}^2\mathbf{U}^T$. The singular vectors contained in $\mathbf{U}$ are also called the empirical orthogonal functions (EOFs). They define the main error directions, and the squared singular values in $\tilde{\mathbf{S}}^2$ are the corresponding variances, stored in descending order. Accordingly, the EOFs in $\mathbf{U}$ are stored in order of decreasing relevance. Finally, the ensemble perturbations are computed from the leading EOFs by

$$\mathbf{X}'_0 = \sqrt{N-1}\,\mathbf{U}\tilde{\mathbf{S}}\mathbf{\Omega}^T \tag{C.10}$$

In equation (C.10), the $N-1$ leading singular values and vectors have to be chosen. Again, $\mathbf{\Omega}$ is a $N \times (N-1)$ random rotation matrix. It can easily be verified that the empirical covariance of the ensemble $\mathbf{X}'_0$ approximates $\mathbf{P}_0$, even though it is a low-rank approximation. However, the ensemble is optimal in the sense that its empirical covariance exhibits the smallest deviation from the original matrix $\mathbf{P}_0$, as measured by the Frobenius norm (Lermusiaux and Robinson 1999).

The procedure outlined here guarantees that the most significant error directions are considered in the new ensemble. However, in some applications it may be important to include as well error directions which exhibit less strength, but whose variability is desired to be contained in the ensemble. For example, if the state vector contains variables of different scales (e.g., temperature in $Kelvin$ and sea surface height in $meter$), the leading error directions are determined by the temperature variances. This issue can be treated by a form of multivariate normalization (Lermusiaux and Robinson 1999). First, the corresponding rows of $\mathbf{X}'_c$ are devided by the standard deviation of the whole field. Hence, the state values are now normalized to a common scale. After the SVD, the singular vectors are rescaled by multiplying the same rows by these standard deviations again. Then, the ensemble is generated as in equation (C.10).

Bibliography

Achatz, U., U. Löbl, and S. I. Dolaptchiev, 2013: Fluctuation-dissipation supplemented by nonlinearity: A climate-dependent sub-grid-scale parameterization in low-order climate models. *J. Atmos. Sci.*, **70**, 1833–1846.

Ades, M., 2013: Data assimilation in highly nonlinear systems. Ph.D. Thesis, University of Reading, 178 pp.

Ades, M. and P. J. van Leeuwen, 2013: An exploration of the equivalent weights particle filter. *Quart. J. Roy. Meteor. Soc.*, **139**, 820–840.

Ades, M. and P. J. van Leeuwen, 2014: The effect of the equivalent-weights particle filter on dynamical balance in a primitive equation model. *Mon. Wea. Rev.*, doi:10.1175/MWR-D-14-00050.1.

Ahrens, B., 1999: Variational data assimilation for a Lorenz model using a non-standard genetic algorithm. *Meteor. Atmos. Phys.*, **70**, 227–238.

Aksoy, A., F. Zhang, and J. W. Nielsen-Gammon, 2006: Ensemble-based simultaneous state and parameter estimation in a two-dimensional sea-breeze model. *Mon. Wea. Rev.*, **134**, 2951–2970.

Akthar, N., J. Brauch, A. Dobler, K. Béranger, and B. Ahrens, 2014: Medicanes in an ocean–atmosphere coupled regional climate model. *Nat. Hazards Earth Syst. Sci*, **14**, 2189–2201.

Alves, O., M. A. Balmaseda, D. Anderson, and T. Stockdale, 2004: Sensitivity of dynamical seasonal forecasts to ocean initial conditions. *Quart. J. Roy. Meteor. Soc.*, **130**, 647–667.

Anderson, E., et al., 1999: *LAPACK Users' Guide*. 3d ed., Society for Industrial and Applied Mathematics, Philadelphia, PA.

Anderson, J. L., 1996: A method for producing and evaluating probabilistic forecasts from ensemble model integrations. *J. Climate*, **9**.

Anderson, J. L., 2001: An ensemble adjustment Kalman filter for data assimilation. *Mon. Wea. Rev.*, **129**, 2884–2903.

Anderson, J. L., 2007a: An adaptive covariance inflation error correction algorithm for ensemble filters. *Tellus*, **59A**, 210–224.

Anderson, J. L., 2007b: Exploring the need for localization in ensemble data assimilation using a hierarchical ensemble filter. *Physica D*, **230**, 99–111.

Anderson, J. L., 2009: Spatially and temporally varying adaptive covariance inflation for ensemble filters. *Tellus A*, **61**, 72–83.

Anderson, J. L., 2010: A non-Gaussian ensemble filter update for data assimilation. *Mon. Wea. Rev.*, **138**, 4186–4198.

Anderson, J. L., 2012: Localization and sampling error correction in ensemble Kalman filter data assimilation. *Mon. Wea. Rev.*, **140**, 2359–2371.

Anderson, J. L. and S. L. Anderson, 1999: A Monte Carlo implementation of the nonlinear filtering problem to produce ensemble assimilations and forecasts. *Mon. Wea. Rev.*, **127**, 2741–2758.

Anderson, J. L., T. Hoar, K. Raeder, H. Lui, and N. Collins, 2009: The Data Assimilation Research Testbed: A community facility. *Bull. Amer. Meteor. Soc.*, **90**, 1283–1296.

Andrews, A., 1968: A square root formulation of the Kalman covariance equations. *AIAA J.*, **6 (6)**, 1165–1168.

Andrieu, C., A. Doucet, and R. Holenstein, 2010: Particle Markov chain Monte Carlo methods. *J. Roy. Stat. Soc.*, **72**, 269–342.

Asselin, R., 1972: Frequency filter for time integrations. *Mon. Wea. Rev.*, **100**, 487–490.

Aster, R. C., B. Borchers, and C. H. Thurber, 2005: *Parameter Estimation and Inverse Problems*. Elsevier Academic Press, 301 pp.

Atkins, E., M. Morzfeld, and A. J. Chorin, 2013: Implicit particle methods and their connection with variational data assimilation. *Mon. Wea. Rev.*, **141**, 1786–1803.

Baldauf, M., A. Seifert, J. Förstner, D. Majewski, and M. Raschendorfer, 2011: Operational convective-scale numerical weather prediction with the COSMO model: Description and sensitivities. *Mon. Wea. Rev.*, **139**, 3887–3905.

Balmaseda, M. A., D. Anderson, and A. Vidard, 2007: Impact of Argo on analyses of the global ocean. *Geophys. Res. Lett.*, **34**, doi:10.1029/2007GL030452.

Balmaseda, M. A., K. Morgensen, and A. T. Weaver, 2013: Evaluation of the ECMWF ocean reanalysis system ORAS4. *Quart. J. Roy. Meteor. Soc.*, **139**, 1132–1161.

Bengtsson, L., et al., 2007: The need for a dynamical climate reanalysis. *Bull. Amer. Meteor. Soc.*, **88**, 495–501.

Bengtsson, T., P. Bickel, and B. Li, 2008: Curse-of-dimensionality revisited: Collapse of the particle filter in very large scale systems. *Probability and Statistics: Essays in Honor of David A. Freedman*, **2**, 316–334, doi:10.1214/193940307000000518.

Bennett, A. F., 2002: *Inverse Modeling of the Ocean and Atmosphere*. Cambridge University Press, 256 pp.

Bishop, C. H., B. J. Etherton, and S. J. Majumdar, 2001: Adaptive sampling with the ensemble transform Kalman filter. Part I: Theoretical aspects. *Mon. Wea. Rev.*, **129**, 420–436.

Bishop, C. H. and D. Hodyss, 2009: Ensemble covariances adaptively localized with ECO-RAP. Part I: Tests on simple error models. *Tellus A*, **61**, 84–96.

Bocquet, M., C. A. Pires, and L. Wu, 2010: Beyond Gaussian statistical modeling in geophysical data assimilation. *Mon. Wea. Rev.*, **138**, 2997–3023.

Bollmeyer, C. and A. Hense, 2014: Inverse modeling of energy transports and budgets of the atmosphere. *Clim. Dyn.*, **43**, 829–844.

Bonavita, M., L. Isaksen, and E. Holm, 2012: On the use of EDA background error variances in the ECMWF 4D-var. *Quart. J. Roy. Meteor. Soc.*, **138**, 1540–1559.

Bonavita, M., L. Torrisi, and F. Marcucci, 2008: The ensemble Kalman filter in an operational regional NWP system: Preliminary results with real observations. *Quart. J. Roy. Meteor. Soc.*, **134**, 1733–1744.

Bonavita, M., L. Torrisi, and F. Marcucci, 2010: Ensemble data assimilation with the CNMCA regional forecasting system. *Quart. J. Roy. Meteor. Soc.*, **136**, 132–145.

Bouttier, F., 1994: A dynamical estimation of the forecast error covariances in an assimilation system. *Mon. Wea. Rev.*, **122**, 2376–2390.

Brandt, P., A. Funk, V. Hormann, M. Dengler, R. J. Greatbatch, and J. M. Toole, 2011: Interannual atmospheric variability forced by the deep equatorial Atlantic Ocean. *Nature*, **473**, 497–500.

Bratseth, A. M., 1986: Statistical interpolation by means of successive corrections. *Tellus A*, **38**, 439–447.

Brouwer, A. E. and W. E. Haemers, 2012: *Spectra of Graphs*. Springer, 263 pp.

Buehner, M., 2005: Ensemble-derived stationary and flow-dependent background error covariances: Evaluation in a quasi-operational NWP setting. *Quart. J. Roy. Meteor. Soc.*, **131**, 1013–1043.

Buehner, M., P. L. Houtekamer, C. Charette, H. L. Mitchell, and B. He, 2010: Intercomparison of variational data assimilation and the ensemble Kalman filter for global deterministic NWP. Part I: Description and single-observation experiments. *Mon. Wea. Rev.*, **138**, 1550–1566.

Buehner, M., J. Morneau, and C. Charette, 2013: Four-dimensional ensemble-variational data assimilation for global deterministic weather prediction. *Nonlin. Processes Geophys.*, **20**, 669–682.

Buizza, R., M. Leutbecher, and L. Isaksen, 2008: Potential use of an ensemble of analyses in the ECMWF ensemble prediction system. *Quart. J. Roy. Meteor. Soc.*, **134**, 2051–2066.

Burgers, G., P. J. van Leeuwen, and G. Evensen, 1998: Analysis scheme in the ensemble Kalman filter. *Mon. Wea. Rev.*, **126**, 1719–1724.

Carrier, G. F. and A. R. Robinson, 1962: On the theory of the wind-driven ocean circulation. *J. Fluid. Mech.*, **12**, 49–80.

Carval, T., et al., 2013: *ARGO user's manual Version 3.0*. ARGO data managament, URL http://www.argodatamgt.org/content/download/17799/115797/file/argo-dm-user-manual-v3.0.pdf.

Caya, A., J. Sun, and C. Snyder, 2005: A comparison between the 4DVAR and the ensemble Kalman filter techniques for radar data assimilation. *Mon. Wea. Rev.*, **133**, 3081–3094.

Chassignet, E. P. and P. R. Gent, 1991: The influence of boundary conditions on midlatitude jet separation in ocean numerical models. *J. Phys. Ocean.*, **21**, 1290–1299.

Cheng, Y. and S. Reich, 2015: A McKean optimal transportation perspective on Feynman-Kac formulae with application to data assimilation. *Frontiers in Applied Dynamical Systems*, URL http://arxiv.org/abs/1311.6300.

Chiodo, G. and L. Haimberger, 2010: Interannual changes in mass consistent energy budgets from ERA-Interim and satellite data. *J. Geophys. Res.*, **115 (D02112)**, doi:10.1029/2009JD012049.

Chorin, A. J., M. Morzfeld, and X. Tu, 2010: Implicit particle filters for data assimilation. *Commun. Appl. Math. Comput. Sci.*, **6**, 221–240.

Clayton, A. M., A. C. Lorenc, and D. M. Barker, 2013: Operational implementation of a hybrid ensemble/4D-Var global data assimilation system at the Met Office. *Quart. J. Roy. Meteor. Soc.*, **139**, 1445–1461.

Cohn, S. E., 1993: Dynamics of short term univariate forecast error covariances. *Mon. Wea. Rev.*, **121**, 3123–3149.

Compo, G. P., et al., 2011: The twentieth century reanalysis project. *Quart. J. Roy. Meteor. Soc.*, **137**, 1–28.

Cosme, E., J.-M. Brankart, J. Verron, P. Brasseur, and M. Krysta, 2010: Implementation of a reduced rank square-root smoother for high resolution ocean data assimilation. *Ocean Modelling*, **33**, 87–100.

Courtier, P., E. Andersson, and W. Heckley, 1998: The ECMWF implementation of three-dimensional variational assimilation (3D-Var). I: Formulation. *Quart. J. Roy. Meteor. Soc.*, **124**, 1783–1807.

Courtier, P., J.-N. Thépaut, and A. Hollingsworth, 1994: A strategy for operational implementations of 4D-Var, using an incremental approach. *Quart. J. Roy. Meteor. Soc.*, **120**, 1367–1387.

Cressman, G. P., 1959: An operational objective analysis scheme. *Mon. Wea. Rev.*, **87**, 367–374.

Daley, R., 1992: The effect of serially correlated observation and model error on atmospheric data assimilation. *Mon. Wea. Rev.*, **120**, 164–177.

Dee, D. P., et al., 2011: The ERA-Interim reanalysis: configuration and performance of the data assimilation system. *Quart. J. Roy. Meteor. Soc.*, **137**, 553–597.

Derber, J. C. and F. Bouttier, 1999: A reformulation of the background error covariance in the ECMWF global data assimilation system. *Tellus A*, **51**, 195–221.

Desroziers, G., L. Berre, B. Chapnik, and P. Poli, 2005: Diagnosis of observation, background and analysis-error statistics in observation space. *Quart. J. Roy. Meteor. Soc.*, **131**, 3385–3396.

Doucet, A., N. de Freitas, and N. Gordon, 2001: *Sequential Monte-Carlo Methods in Practice*. Springer-Verlag, 581 pp.

Durrant, T. H., D. J. M. Greenslade, and I. Simmonds, 2009: Validation of Jason-1 and Envisat remotely sensed wave heights. *J. Atmos. Oceanic Technol.*, **26**, 123–134.

Eltoft, T., T. Kim, and T.-W. Lee, 2006: On the multivariate Laplace distribution. *Signal Processing Letters, IEEE*, 300–303, doi:10.1109/LSP.2006.870353.

Epstein, E. S., 1969: Stochastic dynamic prediction. *Tellus*, **21 (6)**, 739–759.

ESA, 2007: *ENVISAT RA2/MWR Product Handbook, Issue 2.2*. European Space Agency.

Evensen, G., 1992: Using the extended Kalman filter with a multilayer quasi-geostrophic ocean model. *J. Geoph. Res.*, **97 (C11)**, 17 905–17 924.

Evensen, G., 1994: Sequential data assimilation with a nonlinear quasi-geostrophic model using Monte Carlo methods to forecast error statistics. *J. Geoph. Res.*, **99**, 10 143–10 162.

Evensen, G., 2003: The ensemble Kalman filter: theoretical formulation and practical implementation. *Ocean Dyn.*, **53**, 343–367.

Evensen, G., 2004: Sampling strategies and square root analysis schemes for the EnKF. *Ocean Dyn.*, **54**, 539–560.

Evensen, G., 2009: *Data Assimilation: The Ensemble Kalman Filter*. Springer, 279 pp.

Evensen, G. and P. J. van Leeuwen, 1996: Assimilation of Geosat altimeter data for the Agulhas current using the ensemble Kalman filter with a quasi-geostrophic model. *Mon. Wea. Rev.*, **124**, 85–96.

Evensen, G. and P. J. van Leeuwen, 2000: An ensemble Kalman smoother for nonlinear dynamics. *Mon. Wea. Rev.*, **128**, 1852–1867.

Fairbairn, D., S. R. Pring, A. C. Lorenc, and I. Roulstone, 2013: A comparison of 4DVar with ensemble data assimilation methods. *Quart. J. Roy. Meteor. Soc.*, **140**, 281–294.

Feynman, R. P., 1982: Simulating physics with computers. *Int. J. Theor. Physics*, **21**, 467–488.

Fisher, M., 1998: Minimization algorithms for variational data assimilation. *Proc. ECMWF Seminar on recent developments in data assimilation for atmosphere and ocean, 7–11 Sept. 1998*, 364–385.

Fisher, M., 2003: Background error covariance modelling. *Proc. ECMWF Seminar on recent developments in data assimilation for atmosphere and ocean, 8–12 Sept. 2003*, ECMWF, 45–64.

Fisher, M., M. Leutbecher, and G. A. Kelly, 2005: On the equivalence between Kalman smoothing and weak-constraint four-dimensional variational assimilation. *Quart. J. Roy. Meteor. Soc.*, **131**, 3235–3246.

Fowler, A. and P. J. van Leeuwen, 2013: Observation impact in data assimilation: the effect of non-Gaussian observation error. *Tellus*, **65A (20035)**.

Frei, M. and H. R. Künsch, 2013a: Bridging the ensemble Kalman and particle filters. *Biometrika*, **100**, 781–800.

Frei, M. and H. R. Künsch, 2013b: Mixture ensemble Kalman filters. *Comp. Statist. Data Anal.*, **58**, 127–138.

Gaspari, G. and S. E. Cohn, 1999: Construction of correlation functions in two and three dimensions. *Quart. J. Roy. Meteor. Soc.*, **125**, 723–757.

Gneiting, T., F. Balabdaoui, and A. E. Raftery, 2007: Probabilistic forecasts, calibration and sharpness. *J. Roy. Statist. Soc.*, **69**, 243–268.

Gordon, N., D. J. Salmond, and A. F. M. Smith, 1993: Novel approach to nonlinear/non-Gaussian Bayesian state estimation. *IEE Proceedings F (Radar and Signal Processing)*, **140**, 107–113.

Greybush, S. J., E. Kalnay, T. Miyoshi, K. Ide, and B. R. Hunt, 2011: Balance and ensemble Kalman filter localization techniques. *Mon. Wea. Rev.*, **139**, 511–522.

Griffies, S., 2004: *Fundamentals of ocean climate models*. Princeton University Press, 434 pp.

Haimberger, L., 2007: Homogenization of radiosonde temperature time series using innovation statistics. *J. Climate*, **20**, 1377–1403.

Haimberger, L., C. Tavolato, and S. Sperka, 2012: Homogenization of the global radiosonde temperature dataset through combined comparison with reanalysis background series and neighboring stations. *J. Climate*, **25**, 8108–8131.

Hamill, T. M. and C. Snyder, 2000: A hybrid ensemble Kalman filter-3d variational analysis scheme. *Mon. Wea. Rev.*, **128**, 2905–2919.

Hamill, T. M. and J. S. Whitaker, 2011: What constrains spread growth in forecasts initialized from ensemble Kalman filters? *Mon. Wea. Rev.*, **139**, 117–131.

Hastings, W. K., 1970: Monte Carlo sampling methods using Markov chains and their applications. *Biometrika*, **57 (1)**, 97–109.

Hecht, M. W. and R. D. Smith, 2008: *Toward a Physical Understanding of the North Atlantic: A Review of Model Studies in an Eddying Regime*. in: Ocean Modeling in an Eddying Regime (eds M. W. Hecht and H. Hasumi), American Geophysical Union, Washington, D. C.

Hoke, J. E. and R. A. Anthes, 1976: The initialization of numerical models by a dynamic initialization technique. *Mon. Wea. Rev.*, **104**, 1551–1556.

Holland, W. R., 1978: The role of mesoscale eddies in the general circulation of the ocean - numerical experiments using a wind-driven quasi-geostrophic model. *J. Phys. Ocean.*, **8**, 363–392.

Hopson, T. M., 2014: Assessing the ensemble spread-error relationship. *Mon. Wea. Rev.*, **142**, 1125–1142.

Horn, R. A. and C. R. Johnson, 1985: *Matrix Analysis*. Cambridge University Press, 561 pp.

Hoteit, I., X. Luo, and D.-T. Pham, 2012: Particle Kalman filtering: A nonlinear Bayesian framework for ensemble Kalman filters. *Mon. Wea. Rev.*, **140**, 528–542.

Hoteit, I., D.-T. Pham, and J. Blum, 2002: A simplified reduced order Kalman filtering and application to altimetric data assimilation in tropical pacific. *J. Mar. Syst.*, **36**, 101–127.

Hoteit, I., D.-T. Pham, G. Triantafyllou, and G. Korres, 2008: A new approximate solution of the optimal nonlinear filter for data assimilation in meteorology and oceanography. *Mon. Wea. Rev.*, **136**, 317–334.

Houtekamer, P. L., B. He, and H. L. Mitchell, 2014: Parallel implementation of an ensemble Kalman filter. *Mon. Wea. Rev.*, **142**, 1163–1182.

Houtekamer, P. L. and H. L. Mitchell, 1998: Data assimilation using an ensemble Kalman filter technique. *Mon. Wea. Rev.*, **126**, 796–811.

Houtekamer, P. L. and H. L. Mitchell, 2001: A sequential ensemble Kalman filter for atmospheric data assimilation. *Mon. Wea. Rev.*, **129**, 123–137.

Houtekamer, P. L. and H. L. Mitchell, 2005: Ensemble Kalman filtering. *Quart. J. Roy. Meteor. Soc.*, **131**, 3269–3289.

Houtekamer, P. L. and H. L. Mitchell, 2009: Model error representation in an operational ensemble Kalman filter. *Mon. Wea. Rev.*, **137**, 2126–2143.

Houtekamer, P. L., H. L. Mitchell, G. Pellerin, M. Buehner, M. Charron, L. Spacek, and B. Hansen, 2005: Atmospheric data assimilation with an ensemble Kalman filter: Results with real observations. *Mon. Wea. Rev.*, **133**, 604–620.

Hunt, B. R., E. Kostelich, and I. Szunyogh, 2007: Efficient data assimilation for spatiotemporal chaos: A local ensemble transform Kalman filter. *Physica D*, **230**, 112–126.

Ide, K., P. Courtier, M. Ghil, and A. C. Lorenc, 1997: Unified notation for data assimilation: Operational, sequential and variational. *J. Meteor. Soc. Japan*, **75 (1B)**, 181–189.

Janjić, T., L. Nerger, A. Albertella, J. Schröter, and S. Skachko, 2011: On domain localization in ensemble-based Kalman filter algorithms. *Mon. Wea. Rev.*, **139**, 2046–2060.

Jazwinski, A. H., 1970: *Stochastic Processes and Filtering Theory*. Academic Press, San Diego, CA.

Jolliffe, I. T. and D. B. Stephenson, 2011: *Forecast Verification: A Practitioner's Guide in Atmospheric Science*. John Wiley & Sons, 292 pp.

Kalman, R., 1960: A new approach to linear filtering and prediction problems. *J. Basic Eng.*, **82**, 35–45.

Kalman, R. and R. Bucy, 1961: New results in linear filtering and prediction theory. *J. Basic Eng.*, **83D**, 95–108.

Kalnay, E., 2003: *Atmospheric modeling, data assimilation and predictability*. Cambridge University Press, 341 pp.

Kalnay, E., H. Li, T. Miyoshi, S.-C. Yang, and J. Ballabrera-Poy, 2007: 4D-Var or ensemble Kalman filter? *Tellus*, **59A**, 748–773.

Kalnay, E. and S.-C. Yang, 2010: Accelerating the spin-up of ensemble kalman filtering. *Quart. J. Roy. Meteor. Soc.*, **136**, 1644–1651.

Kamenkovich, I., P. Berloff, and J. Pedlosky, 2009: Role of eddy forcing in the dynamics of multiple zonal jets in the North Atlantic. *J. Phys. Ocean.*, **39**, 1361–1379.

Kirchgessner, P., L. Nerger, and A. Bunse-Gerstner, 2014: On the choice of an optimal localization radius in ensemble Kalman filter methods. *Mon. Wea. Rev.*, **142**, 2165–2175.

Kitagawa, G., 1996: Monte Carlo filter and smoother for non-Gaussian nonlinear state space models. *J. Comput. Graph. Stat.*, **5**, 1–25.

Kivman, G. A., 2003: Sequential parameter estimation for stochastic systems. *Nonlin. Processes Geophys.*, **10**, 253–259.

Köhl, A., 2014: Evaluation of the GECCO2 ocean synthesis: Transports of volume, heat and freshwater in the Atlantic. *Quart. J. Roy. Meteor. Soc.*, doi:10.1002/qj.2347.

Kotz, S., T. Kozubowski, and K. Podgorski, 2001: *The Laplace Distribution and Generalizations*. Springer Science & Business Media, 349 pp.

Kröger, J., W. A. Müller, and J. S. von Storch, 2012: Impact of different ocean reanalyses on decadal climate prediction. *Clim. Dyn.*, **39**, 795–810.

Lawson, W. G. and J. A. Hansen, 2004: Implications of stochastic and deterministic filters as ensemble-based data assimilation methods in varying regimes of error growth. *Mon. Wea. Rev.*, **132**, 1966–1981.

Leclair, M. and G. Madec, 2009: A conservative leap-frog time stepping method. *Ocean Modelling*, **30**, 88–94.

LeDimet, F.-X. and O. Talagrand, 1986: Variational algorithms for analysis and assimilation of meteorological observations: Theoretical aspects. *Tellus A*, **36**, 97–110.

Leeuwenburgh, O., G. Evensen, and L. Bertino, 2005: The impact of ensemble filter definition on the assimilation of temperature profiles in the tropical Pacific. *Quart. J. Roy. Meteor. Soc.*, **131**, 3291–3300.

Lei, J. and P. Bickel, 2009: Ensemble filtering for high dimensional non-linear state space models, URL http://www.stat.berkeley.edu/~bickel/LeiB09_NLEAF.pdf, unpublished manuscript, UC Berkeley.

Lei, J. and P. Bickel, 2011: A moment matching ensemble filter for nonlinear non-Gaussian data assimilation. *Mon. Wea. Rev.*, **139**, 3964–3973.

Lei, J., P. Bickel, and C. Snyder, 2010: Comparison of ensemble Kalman filters under non-Gaussianity. *Mon. Wea. Rev.*, **138**, 1293–1306.

Lermusiaux, P. F. J., 2006: Uncertainty estimation and prediction for interdisciplinary ocean dynamics. *J. Comput. Phys.*, **217**, 176–199.

Lermusiaux, P. F. J. and A. R. Robinson, 1999: Data assimilation via error subspace statistical estimation. Part I: Theory and schemes. *Mon. Wea. Rev.*, **127**, 1385–1407.

Lévy, M., P. Klein, A.-M. Tréguier, D. Iovino, G. Madec, S. Masson, and K. Takahashi, 2010: Modifications of gyre circulation by sub-mesoscale physics. *Ocean Modelling*, **34**, 1–15.

Li, H., E. Kalnay, and T. Miyoshi, 2009: Simultaneous estimation of covariance inflation and observation errors within an ensemble Kalman filter. *Quart. J. Roy. Meteor. Soc.*, **135**, 523–533.

Li, Z. and I. M. Navon, 2001: Optimality of variational data assimilation and its relationship with the Kalman filter and smoother. *Quart. J. Roy. Meteor. Soc.*, **127**, 661–683.

Livings, D. M., S. L. Dance, and N. K. Nichols, 2008: Unbiased ensemble square root filters. *Physica D*, **237**, 1021–1028.

Lorenc, A. C., 1981: A global three-dimensional multivariate statistical interpolation scheme. *Mon. Wea. Rev.*, **109**, 701–721.

Lorenc, A. C., 1986: Analysis methods for numerical weather prediction. *Quart. J. Roy. Meteor. Soc.*, **112**, 1177–1194.

Lorenc, A. C., 2003: The potential of the ensemble Kalman filter for NWP - a comparison with 4D-Var. *Quart. J. Roy. Meteor. Soc.*, **129**, 3183–3203, doi:10.1256/qj.02.132.

Lorenc, A. C., N. E. Bowler, A. M. Clayton, S. R. Pring, and D. Fairbairn, 2014: Comparison of hybrid-4DEnVar and hybrid-4DVar data assimilation methods for global NWP. *Mon. Wea. Rev.*, doi: 10.1175/MWR-D-14-00195.1.

Lorenz, E. N., 1963: Deterministic nonperiodic flow. *J. Atmos. Sci.*, **20**, 130–141.

Lorenz, E. N., 1996: Predictability - a problem partly solved. *Proc. Seminar on Predictability*, Reading, Berkshire, UK, ECMWF, 1–18.

Lorenz, E. N., 2005: Designing chaotic models. *J. Atmos. Sci.*, **62**, 1574–1587.

Losa, S. N., G. A. Kivman, J. Schröter, and M. Wenzel, 2003: Sequential weak constraint parameter estimation in an ecosystem model. *J. Mar. Syst.*, **43**, 31–49.

Luo, X. and I. Hoteit, 2013: Covariance inflation in the ensemble Kalman filter: A residual nudging perspective and some implications. *Mon. Wea. Rev.*, **141**, 3360–3368.

Luo, X. and I. Hoteit, 2014a: Efficient particle filtering through residual nudging. *Quart. J. Roy. Meteor. Soc.*, **140**, 557–572.

Luo, X. and I. Hoteit, 2014b: Ensemble Kalman filtering with residual nudging: An extension to state estimation problems with nonlinear observation operators. *Mon. Wea. Rev.*, **142**, 3696–3712.

Lütkepohl, H., 1996: *Handbook of Matrices*. John Wiley & Sons Ltd.

Lynch, P., 1992: Richardson's barotropic forecast: A reappraisal. *Bull. Amer. Meteor. Soc.*, **73**, 35–47.

Madec, G., 2008: Nemo ocean engine. Tech. Rep. 27, Note du Pôle de modélisation, Institut Pierre-Simon Laplace (IPSL), 357 pp.

Madec, G., P. Delecluse, M. Imbard, and C. Lévy, 1998: OPA 8.1 ocean general circulation model reference manual. Tech. Rep. 11, Note du Pôle de modélisation, Institut Pierre-Simon Laplace (IPSL), 91 pp.

Masutani, M., et al., 2010: *Data Assimilation - Making Sense of Observations*, chap. Observing System Simulation Experiments, 647–681. Springer.

Maybeck, P. S., 1982: *Stochastic Models, Estimation, and Control*, Vol. 1. Academic Press.

Metropolis, N., A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller, 1953: Equation of state calculations by fast computing machines. *J. Chem. Phys.*, **21**, 1087–1092.

Metropolis, N. and S. Ulam, 1949: The Monte Carlo method. *J. Am. Statist. Assoc.*, **44 (247)**, 335–341.

Miller, R. N., E. F. Carter, and S. T. Blue, 1999: Data assimilation into nonlinear stochastic models. *Tellus*, **51A**, 167–194.

Miller, R. N., M. Ghil, and F. Gauthiez, 1994: Advanced data assimilation in strongly nonlinear dynamical systems. *J. Atmos. Sci.*, **51**, 1037–1056.

Miyoshi, T., 2011: The Gaussian approach to adaptive covariance inflation and its implementation with the local ensemble transform Kalman filter. *Mon. Wea. Rev.*, **139**, 1519–1535.

Miyoshi, T., E. Kalnay, and H. Li, 2013: Estimating and including observation-error correlations in data assimilation. *Inverse Problems in Science and Engineering*, **21**, 387–398.

Miyoshi, T., Y. Sato, and T. Kadowaki, 2010: Ensemble Kalman filter and 4D-var intercomparison with the japanese operational global analysis and prediction system. *Mon. Wea. Rev.*, **138**, 2846–2866.

Miyoshi, T. and S. Yamane, 2007: Local ensemble transform Kalman filtering with an AGCM at a T159/l48 resolution. *Mon. Wea. Rev.*, **135**, 3841–3861.

Molteni, F., R. Buizza, T. Palmer, and T. Petroliagis, 1996: The ECMWF ensemble prediction system: Methodology and validation. *Quart. J. Roy. Meteor. Soc.*, **122**, 73–119.

Morzfeld, M. and A. J. Chorin, 2012: Implicit particle filtering for models with partial noise, and an application to geomagnetic data assimilation. *Nonlin. Processes Geophys.*, **19**, 365–382.

MPI Forum, 2012: *MPI: A Message-Passing Interface Standard, Version 3.0*. High Performance Computing Center Stuttgart (HLRS), 852 pp.

Müller, W. A., et al., 2012: Forecast skill of multi-year seasonal means in the decadal prediction system of the Max Planck Institute for Meteorology. *Geophys. Res. Lett.*, **39 (22)**, doi: 10.1029/2012GL053326.

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

Nerger, L. and W. Hiller, 2013: Software for ensemble-based data assimilation systems - implementation strategies and scalability. *Computers and Geosciences*, **55**, 110–118.

Nerger, L., W. Hiller, and J. Schröter, 2005: A comparison of error subspace Kalman filters. *Tellus*, **57A**, 715–735.

Nerger, L., T. Janjić, J. Schröter, and W. Hiller, 2012a: A regulated localization scheme for ensemble-based Kalman filters. *Quart. J. Roy. Meteor. Soc.*, **138**, 802–812.

Nerger, L., T. Janjić, J. Schröter, and W. Hiller, 2012b: A unification of ensemble square root Kalman filters. *Mon. Wea. Rev.*, **140**, 2335–2345.

Nerger, L., S. Schulte, and A. Bunse-Gerstner, 2014: On the influence of model nonlinearity and localization on ensemble Kalman smoothing. *Quart. J. Roy. Meteor. Soc.*, **140**, 2249–2259.

Ott, E., et al., 2004: A local ensemble Kalman filter for atmospheric data assimilation. *Tellus*, **56A**, 415–428.

Palmer, T., R. Buizza, R. Hagedorn, A. Lawrence, M. Leutbecher, and L. Smith, 2005: Ensemble prediction: A pedagogical perspective. *ECMWF Newsletter*, **106**, 10–17.

Palter, J. B., S. M. Griffies, B. L. Samuels, E. D. Galbraith, A. Gnanadesikan, and A. Klocker, 2014: The deep ocean buoyancy budget and its temporal variability. *J. Climate*, **27**, 551–573.

Papadakis, N., E. Mémin, A. Cuzol, and N. Gengembre, 2010: Data assimilation with the weighted ensemble Kalman filter. *Tellus*, **62A**, 673–697.

Penny, S. G., 2014: The hybrid local ensemble transform Kalman filter. *Mon. Wea. Rev.*, **142**, 2139–2149.

Pham, D. T., 2001: Stochastic methods for sequential data assimilation in strongly nonlinear systems. *Mon. Wea. Rev.*, **129**, 1194–1207.

Pham, T. V., J. Brauch, C. Dieterich, B. Früh, and B. Ahrens, 2014: New coupled atmosphere-ocean-ice system COSMO-CLM/NEMO: On the air temperature sensitivity on the north and baltic seas. *Oceanologia*, **56**, 167–189.

Pitt, M. K. and N. Shephard, 1999: Filtering via simulation: Auxiliary particle filters. *J. Am. Statist. Assoc.*, **94 (446)**, 590–599.

Polkova, I., A. Köhl, and D. Stammer, 2014: Impact of initialization procedures on the predictive skill of a coupled ocean–atmosphere model. *Clim. Dyn.*, **42**, 3151–3169.

Posselt, D. J. and C. H. Bishop, 2012: Nonlinear parameter estimation: Comparison of an ensemble Kalman smoother with a Markov chain Monte Carlo algorithm. *Mon. Wea. Rev.*, **140**, 1957–1974.

Provost, C. L. and J. Verron, 1987: Wind-driven mid-latitude circulation – transition to barotropic instability. *Dyn. Atm. Ocean.*, **11**, 175–201.

Rabier, F., H. Järvinen, E. Klinker, J.-F. Mahfouf, and A. Simmons, 2000: The ECMWF operational implementation of four-dimensional variational assimilation. I: Experimental results with simplified physics. *Quart. J. Roy. Meteor. Soc.*, **126**, 1143–1170.

Rabier, F., et al., 1998: The ECMWF implementation of three-dimensional variational assimilation (3D-Var). II: Structure functions. *Quart. J. Roy. Meteor. Soc.*, **124**, 1809–1829.

Rainwater, S. and B. R. Hunt, 2013: Ensemble data assimilation with an adjusted forecast spread. *Tellus A*, **65**, 19 929, doi:10.3402/tellusa.v65i0.19929.

Reich, H., A. Rhodin, and C. Schraff, 2011: LETKF for the nonhydrostatic regional model COSMO-DE. *COSMO Newsletter*, **11**, 27–31.

Reich, S., 2013: A nonparametric ensemble transform method for Bayesian inference. *SIAM J. Sci. Comput.*, **34**, A2013–A2024.

Reichle, R. H., D. B. McLaughlin, and D. Entekhabi, 2002: Hydrologic data assimilation with the ensemble Kalman filter. *Mon. Wea. Rev.*, **130**, 103–114.

Richardson, D., J. Bidlot, L. Ferranti, T. Haiden, T. Hewson, M. Janousek, F. Prates, and F. Vitart, 2013: Evaluation of ECMWF forecasts, including 2012–2013 upgrades. Tech. Rep. 710, ECMWF, Reading, Berkshire, UK.

Robert, C. and R. Casella, 2004: *Monte Carlo Statistical Methods*. Springer, 649 pp.

Roullet, G. and G. Madec, 2000: Salt conservation, free surface and varying levels: A new formulation of for ocean general circulation models. *J. Geoph. Res.*, **105**, 23 972–23 942.

Sakov, P. and L. Bertino, 2011: Relation between two common localisation methods for the EnKF. *Comput. Geosci.*, **15**, 225–237.

Sakov, P. and P. R. Oke, 2008a: A deterministic formulation of the ensemble Kalman filter: An alternative to ensemble square root filters. *Tellus*, **60A**, 361–371.

Sakov, P. and P. R. Oke, 2008b: Implications of the form of the ensemble transformation in the ensemble square root filters. *Mon. Wea. Rev.*, **136**, 1042–1053.

Sakov, P., D. S. Oliver, and L. Bertino, 2012: An iterative EnKF for strongly nonlinear systems. *Mon. Wea. Rev.*, **140**, 1988–2004.

Schlax, M. G. and D. B. Chelton, 2008: The influence of mesoscale eddies on the detection of quasi-zonal jets in the ocean. *Geophys. Res. Lett.*, **35 (L24602)**.

Schleussner, C. F., J. Runge, J. Lehmann, and A. Levermann, 2014: The role of the north atlantic overturning and deep ocean for multi-decadal global-mean-temperature variability. *Earth Syst. Dynam.*, **5**, 103–115.

Schmidt, S. F., 1981: The Kalman filter - Its recognition and development for aerospace applications. *J. Guidance and Control*, **4 (1)**, doi:10.2514/3.19713.

Shen, J., T. T. Medjo, and S. Wang, 1999: On a wind-driven, double-gyre, quasi-geostrophic ocean model: Numerical simulations and structural analysis. *J. Comp. Phys.*, **155**, 387–409.

Smith, G. L., S. F. Schmidt, and L. A. McGee, 1962: Application of statistical filter theory to the optimal estimation of position and velocity on board a circumlunar vehicle. Tech. rep., National Aeronautics and Space Administration.

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

Sorenson, H. W., 1970: Least-squares estimation: from Gauss to Kalman. *IEEE Spectrum*, **7**, 63–68.

Stordal, A. S., H. A. Karlsen, G. Nævdal, H. J. Skaug, and B. Vallès, 2011: Bridging the ensemble Kalman filter and particle filters: the adaptive Gaussian mixture filter. *Comput. Geosci.*, **15**, 293–305.

Stratonovich, R. L., 1960: Application of the Markov processes theory to optimal filtering. *Radio Engineering and Electronic Physics*, **5**, 1–19.

Sun, A. Y., A. Morris, and S. Mohanty, 2009: Comparison of deterministic ensemble Kalman filters for assimilating hydrogeological data. *Adv. Water Resour.*, **32**, 280–292.

Szunyogh, I., E. J. Kostelich, G. Gyarmati, E. Kalnay, B. R. Hunt, E. Ott, E. Satterfield, and J. A. Yorke, 2008: A local ensemble transform Kalman filter data assimilation system for the NCEP global model. *Tellus A*, **60**, 113–130.

Talagrand, O. and P. Courtier, 1987: Variational assimilation of meteorological observations with the adjoint vorticity equation. I: Theory. *Quart. J. Roy. Meteor. Soc.*, **113**, 1311–1328.

Tarantola, A., 2005: *Inverse Problem Theory and Methods for Model Parameter Estimation*. Siam, 342 pp.

Thompson, J. D. and W. J. Schmitz, 1989: A limited-area model of the Gulf Stream: Design, initial experiments, and model data intercomparison. *J. Phys. Ocean.*, **19**, 791–814.

Tierney, L., 1994: Markov chains for exploring posterior distributions. *Annals of Statistics*, **22**, 1701–1762.

Tikhonov, A. N. and V. Arsenin, 1977: *Solution of ill-posed problems*. Winston and Sons, Washington, DC, USA.

Tippett, M. K., J. L. Anderson, C. H. Bishop, T. Hamill, and J. S. Whitaker, 2003: Ensemble square root filters. *Mon. Wea. Rev.*, **131**, 1485–1490.

Tödter, J., 2011: New aspects of information theory in probabilistic forecast verification. M.Sc. Thesis, Institute for Theoretical Physics, Goethe University, 128 pp., Frankfurt/Main.

Tödter, J. and B. Ahrens, 2012: Generalization of the ignorance score: Continuous ranked version and its decomposition. *Mon. Wea. Rev.*, **140**, 2005–2017.

Tödter, J. and B. Ahrens, 2015: A second-order exact ensemble square root filter for nonlinear data assimilation. *Mon. Wea. Rev.*, **143**, doi:10.1175/MWR-D-14-00108.1.

Toth, Z. and E. Kalnay, 1997: Ensemble forecasting at NCEP and the breeding method. *Mon. Wea. Rev.*, **125**, 3297–3319.

Tremolet, Y., 2004: Diagnostics of linear and incremental approximations in 4D-var. *Quart. J. Roy. Meteor. Soc.*, **130**, 2233–2251.

Tremolet, Y., 2006: Accounting for an imperfect model in 4D-var. *Quart. J. Roy. Meteor. Soc.*, **132**, 2483–2504.

Tremolet, Y., 2007: Model-error estimation in 4D-var. *Quart. J. Roy. Meteor. Soc.*, **133**, 1267–1280.

van Leeuwen, P. J., 2009: Particle filtering in geophysical systems. *Mon. Wea. Rev.*, **137**, 4089–4114.

van Leeuwen, P. J., 2010: Nonlinear data assimilation in geosciences: An extremely efficient particle filter. *Quart. J. Roy. Meteor. Soc.*, **136**, 1991–1999.

van Leeuwen, P. J. and M. Ades, 2013: Efficient fully nonlinear data assimilation for geophysical fluid dynamics. *Computers and Geosci.*, **55**, 16–27.

van Leeuwen, P. J. and G. Evensen, 1996: Data assimilation and inverse methods in terms of a probabilistic formulation. *Mon. Wea. Rev.*, **124**, 2898–2913.

Wang, X., 2010: Incorporating ensemble covariance in the gridpoint statistical interpolation variational minimization: A mathematical framework. *Mon. Wea. Rev.*, **138**, 2990–2995.

Wang, X. and C. H. Bishop, 2003: A comparison of breeding and ensemble transform Kalman filter ensemble forecast schemes. *J. Atmos. Sci.*, **60**, 1140–1158.

Wang, X., C. H. Bishop, and S. J. Julier, 2004: Which is better, an ensemble of positive–negative pairs or a centered spherical simplex ensemble? *Mon. Wea. Rev.*, **132**, 1590–1605.

Wang, X., D. Parrish, D. Kleist, and J. S. Whitaker, 2013: GSI 3Dvar-based ensemble–variational hybrid data assimilation for NCEP global forecast system: Single-resolution experiments. *Mon. Wea. Rev.*, **141**, 4098–4117.

Whitaker, J. S., G. P. Compo, and J.-N. Thépaut, 2009: A comparison of variational and ensemble-based data assimilation systems for reanalysis of sparse observations. *Mon. Wea. Rev.*, **137**, 1991–1999.

Whitaker, J. S. and T. Hamill, 2012: Evaluating methods to account for system errors in ensemble data assimilation. *Mon. Wea. Rev.*, **140**, 3078–3089.

Whitaker, J. S. and T. M. Hamill, 2002: Ensemble data assimilation without perturbed observations. *Mon. Wea. Rev.*, **130**, 1913–1924.

Whitaker, J. S., T. M. Hamill, X. Wei, Y. Song, and Z. Toth, 2008: Ensemble data assimilation with the NCEP global forecast system. *Mon. Wea. Rev.*, **136**, 463–482.

Wikle, C. K. and L. M. Berliner, 2007: A Bayesian tutorial for data assimilation. *Physica D*, **230**, 1–16.

Wilks, D. S., 2011: *Statistical methods in the atmospheric sciences*. Academic Press, 704 pp.

Willebrand, J., et al., 2001: Circulation characteristics in three eddy-permitting models of the North Atlantic. *Progr. Oceanography*, **48**, 123–161.

Williams, P. D., 2009: A proposed modification to the Robert–Asselin time filter. *Mon. Wea. Rev.*, **137**, 2538–2546.

Wu, X., W. Li, G. Han, S. Zhang, and X. Wang, 2014: A compensatory approach of the fixed localization in EnKF. *Mon. Wea. Rev.*, **142**, 3713–3733.

Xiong, X., I. M. Navon, and B. Uzunoglu, 2006: A note on the particle filter with posterior Gaussian resampling. *Tellus*, **58A**, 456–460.

Yaman, F., V. G. Yakhno, and R. Potthast, 2013: Recent theory and applications on inverse problems. *Mathematical Problems in Engineering*, **2013 (Article ID 303154)**, doi:10.1155/2013/303154.

Yan, Y., 2013: Medium case benchmark report. Stochastic Assimilation for the Next Generation Ocean Model Applications, EU FP7 SPACE-2011-1 project 283580, University of Liège, Belgium.

Yan, Y., A. Barth, and J. M. Beckers, 2014: Comparison of different assimilation schemes in a sequential Kalman filter assimilation system. *Ocean Modelling*, **73**, 123–137.

Yang, S.-C., E. Kalnay, and B. Hunt, 2012a: Handling nonlinearity in an ensemble Kalman filter: Experiments with the three-variable Lorenz model. *Mon. Wea. Rev.*, **140**, 2628–2646.

Yang, S.-C., E. Kalnay, and T. Miyoshi, 2012b: Accelerating the EnKF spinup for typhoon assimilation and prediction. *Wea. Forecasting*, **27**, 878–897.

Zhang, F., C. Snyder, and J. Sun, 2004: Impacts of initial estimate and observation availability on convective-scale data assimilation with an ensemble Kalman filter. *Mon. Wea. Rev.*, **132**, 1238–1253.

Zhang, M., F. Zhang, X.-Y. Huang, and X. Zhang, 2011: Intercomparison of an ensemble Kalman filter with three- and four-dimensional variational data assimilation methods in a limited-area model over the month of June 2003. *Mon. Wea. Rev.*, **139**, 566–572.

Zhang, S. and A. Rosati, 2010: An inflated ensemble filter for ocean data assimilation with a biased coupled GCM. *Mon. Wea. Rev.*, **138**, 3905–3931.

Zupanski, D., 1997: A general weak constraint applicable to operational 4DVAR data assimilation systems. *Mon. Wea. Rev.*, **125**, 2274–2292.

Acknowledgements

The acknowledgments have been removed for the online version of this thesis.

Supplement: Zusammenfassung

Dieser Abschnitt enthält eine Zusammenfassung der vorliegenden Arbeit. Hierzu werden die wesentlichen Ergebnisse und Aussagen der Kapitel 1 bis 7 wiedergegeben.

1. Hintergrund, Motivation und Ziele

Verlässliche Prognosen werden benötigt, um bessere Entscheidungen für zukünftiges Handeln treffen zu können. Ein wichtiges Beispiel sind Wetter- und Klimaprognosen. Diese werden nicht nur im persönlichen Alltag genutzt, sondern bilden auch die Grundlage für wirtschaftliche oder politische Maßnahmen, etwa in Bezug auf Extremwetterereignisse oder Klimaerwärmung. Hierfür werden komplexe, hochdimensionale Modelle verwendet, um die physikalischen Prozesse in der Atmosphäre, den Meeren und der Landoberfläche der Erde quantitativ nachzubilden. Jede Prognoserechnung startet mit dem sogenannten *Anfangszustand*. Da die Atmosphäre ein stark nichtlineares, chaotisches System ist, können anfänglich kleine Fehler sehr schnell wachsen. Diese weithin als "Schmetterlingseffekt" bekannte Eigenschaft ist der Hauptgrund für die begrenzte Vorhersagbarkeit des Wetters. Die gute Spezifizierung des Anfangszustands stellt somit eine große Herausforderung dar, die den Aufwand der eigentlichen Prognose oft sogar übertrifft. Hierfür wird auf die inverse Modellierung, auch als *Datenassimilation* (DA) bekannt, zurückgegriffen, wobei das Modell mit empirischen Beobachtungen kombiniert wird. Die resultierende Zustandsschätzung ist zudem für Reanalysen wichtig, etwa um konsistente Klimatologien abzuleiten oder um systematische Fehler von Modellen und Beobachtungen zu identifizieren.

Die große Herausforderung der DA in geophysikalischen Systemen besteht in der hohen Dimensionalität des Zustandsraums und der nur geringen Kenntnisse über die involvierten Wahrscheinlichkeitsverteilungen, was viele analytische und numerische Verfahren bereits ausschließt. In operativen Anwendungen wird zumeist auf Variationsverfahren (4DVAR) zurückgegriffen. Eine Alternative stellen sequentielle Filter dar, die Vorhersage- und Analyseschritte iterieren, um Beobachtungen der Reihe nach zu assimilieren. Der Vorteil liegt in der einfacheren Implementierung, da der eigentliche Analysealgorithmus in der Regel modellunabhängig ist. Einen Meilenstein in dieser Richtung stellt der *Ensemble Kalman Filter* (EnKF) dar. Er wendet im Vorhersageschritt ein Monte-Carlo-Verfahren an, die Ensembleprognose. Deterministische Varianten, insbesondere der *Ensemble Transform Kalman Filter* (ETKF) werden mittlerweile auch für großskalige Probleme wie die globale Wettervorhersage eingesetzt.

Jedoch beruhen sowohl 4DVAR als auch der EnKF auf einer Normalverteilungs- oder Gauß'schen Annahme, was in nichtlinearen Systemen im Allgemeinen zu systematischen Fehlern führt. *Partikelfilter*

(PF) bieten zwar eine vollständig nichtlineare und nicht-Gauß'sche Lösung, sind jedoch in deterministischen Systemen oder höherdimensionalen Räumen nicht anwendbar. Die direkte Anwendung von Bayes' Theorem im Beobachtungsraum führt in der Regel zu einer großen Varianz der resultierenden Partikelgewichte, wodurch eine exponentiell mit der Dimension wachsende Ensemblegröße nötig ist, um Filterdivergenz zu verhindern. Eine aussichtsreiche neuere Entwicklung ist der EWPF, der die Partikel so kontrolliert, dass sie nahezu gleiche Gewichte (*equivalent weights*, EW) erhalten. Er erfordert jedoch stochastische Modelle und eine entsprechende Manipulation des Vorhersageschrittes, und ist damit nicht mehr modellunabhängig. Ein anderer approximativer Vorschlag, der *Nonlinear Ensemble Adjustment Filter* (NLEAF), ist zwar von allgemeiner Natur, verliert jedoch durch Zufallsfehler und rechenintensive Algorithmen seine Anwendbarkeit auf hochdimensionalen Probleme.

Das Ziel dieser Arbeit besteht daher darin, einen neuen, nichtlinearen Filter zu entwickeln, der auch in hohen Dimensionen anwendbar ist, zugleich aber ähnlich einfach wie der ETKF anwendbar ist. Daher werden drei Arbeitsschritte definiert. Zunächst soll ein konsistenter Rückblick auf DA-Verfahren den aktuellen Stand der Forschung aufzeigen und somit identifizieren, wo Weiterentwicklungen möglich sind. Dies soll in der Folge zur Herleitung und Untersuchung eines neuen nichtlinearen, ensemble-basierten Filters (NETF) verwendet werden. Danach soll dessen potentieller praktischer Nutzen empirisch demonstriert werden. Hierzu werden zunächst einfache Systeme verwendet, um die Filtereigenschaften zu untersuchen und Vergleiche mit anderen ensemble-basierten Filtern durchzuführen. Zum Abschluss erfolgt die Kopplung des NETF mit einem großskaligen Ozeanmodell mit einem realistischen Beobachtungsszenario, um die Anwendbarkeit in hochdimensionalen Problemen zu untersuchen.

2. Datenassimilation: Grundlagen und klassische Methoden

DA besteht aus zwei Komponenten. Das *Modell* löst die dem System zugrunde liegenden Gleichungen und liefert somit zu jedem Zeitschritt einen vollständigen Zustand. *Beobachtungen* hingegen entsprechen zwar Messungen der realen Welt, sind aber räumlich und zeitlich irregulär und hängen oft nur indirekt mit den Zustandsvariablen zusammen. Die Verknüpfung von Modell und Beobachtungen erfolgt durch einen *Beobachtungsoperator*, der vom Zustands- in den Beobachtungsraum transformiert. Da sowohl Modellzustände als auch Beobachtungen mit Unsicherheiten behaftet sind, erfolgt die Beschreibung von DA mithilfe von Wahrscheinlichkeitsverteilungen. In Kapitel 2 wird eine konsistente probabilistische Rahmenbeschreibung vorgestellt, dessen Grundlage *Bayes' Theorem* darstellt. Aufgrund der großen Dimensionen der Wahrscheinlichkeitsräume ist eine direkte Berechnung der Analyseverteilung jedoch nicht möglich. Da Modellzustände gewöhnlich eine Markov-Kette erster Ordnung bilden, lässt sich das Problem sequentiell lösen, indem Beobachtungen der Reihe nach assimiliert werden. Während *Glätter* stets alle Beobachtungen in einem Zeitfenster berücksichtigen, hängen die Schätzungen eines *Filters* nur von vergangenen Beobachtungen ab.

Die klassische Lösung des DA-Problems besteht in der Annahme von Normalverteilungen für Zustände und Beobachtungen. Wird dies auf den allgemeinen Glätter angewendet, ergibt sich die Variationsmethode (4DVAR), bei der eine komplexe quadratische Kostenfunktion minimiert werden muss. Die Lösung ist die Trajektorie, welche die kleinsten quadratischen Abweichungen zu einer ersten Schätzung

sowie den Beobachtungen aufweist. Die Minimierung erfolgt mit einem geeigneten numerischen Verfahren, wobei das tangential lineare und das adjungierte Modell erforderlich sind.
Wird die Gauß'sche Annahme hingegen im sequentiellen Filter gemacht, ergibt sich der KF als Lösung. Sein Analyseschritt besteht in einer linearen Modifikation des Schätzwertes und einer Reduktion der Varianz als Maß der Schätzunsicherheit. Während der KF für lineare Systeme die optimale Lösung darstellt und äquivalent zu 4DVAR ist, muss im allgemeinen Fall auf das linearisierte Modell zurückgegriffen werden, um die Kovarianz in der Zeit zu integrieren. Dies ist nicht nur äußerst rechenintensiv, sondern kann in nichtlinearen Systemen auch zu signifikanten Fehlern führen. Daher wird in operativen Anwendungen oft 4DVAR vorgezogen, dessen Implementierung und Umsetzung jedoch sehr aufwendig ist. Zudem liefert 4DVAR in nichtlinearen Situationen ebenfalls nicht die optimale Lösung.

3. Ensemble-basierte Methoden in der Datenassimilation

In Kapitel 3 werden neuere Entwicklungen hinsichtlich des nichtlinearen Filterns betrachtet, die durch die Ensembletechnik möglich wurden. Eine Integration der Wahrscheinlichkeitsverteilung in der Zeit ist nicht möglich, da diese weder hinreichend bekannt ist, noch die zugrunde Fokker-Planck-Gleichung in der Praxis lösbar ist. Ensembleprognosen bieten jedoch eine praktikable *Monte-Carlo*-Lösung. Hierzu wird eine endliche Zahl von unabhängig und identisch verteilten Zuständen mit dem vollständigen, nichtlinearen Modell integriert. Die Ensembleverteilung lässt sich zu jedem Zeitpunkt als Stichprobe aus der unbekannten Wahrscheinlichkeitsverteilung interpretieren.

Die Anwendung der Ensembletechnik auf den Vorhersageschritt des KF führt zum EnKF. Der Mittelwert und die Kovarianz der Vorhersageverteilung, die in den Analyseschritt eingehen, werden dann direkt aus dem Ensemble geschätzt. Damit wird die explizite, linearisierte Integration der Kovarianzmatrix des KF umgangen. Da im klassischen EnKF die Beobachtung stochastisch gestört werden muss, ergeben sich Zufallsfehler, die insbesondere bei den typischerweise kleinen Ensembles geophysikalischer Anwendungen nicht vernachlässigbar sind. Daher wurden deterministische EnKF entwickelt, die das Vorhersageensemble so transformieren, dass Mittelwert und Kovarianz des Analyseensembles exakt den KF-Gleichungen entsprechen. Da der EnKF die Kovarianz durch eine Matrix von niedrigem Rang approximiert, müssen in der Praxis einige Ergänzungen vorgenommen werden. Durch *Inflation* wird der Tendenz zur Unterschätzung der Unsicherheit begegnet, und die räumliche Lokalisierung der Analyse sorgt dafür, dass zufällige, fälschliche Korrelationen unterdrückt werden. Zusammen mit diesen Erweiterungen ist der EnKF mit 4DVAR wettbewerbsfähig, und lässt sich auch für großskalige Systeme, wie etwa in der numerischen Wetterprognose, einsetzen. Hierzu wird oft der lokalisierte ETKF als Referenz verwendet. Sein größter Vorteil liegt in der relativ einfachen Anwendbarkeit, da der Analysealgorithmus prinzipiell unabhängig vom Modell ist. Im operativen Bereich werden zudem 4DVAR und Ensemblemethoden zu sogenannten Hybridsystemen verbunden.

Gleichwohl ist die EnKF-Analyse meist suboptimal, da die von den linearen KF-Gleichungen implizierte Normalverteilung in nichtlinearen Systemen nicht erfüllt ist. Dies führt zu systematischen Fehlern in der Analyse. Der PF bietet eine vollständig nichtlineare, nicht-Gauß'sche Monte-Carlo-Lösung des DA-Problems. Insbesondere ist keine parametrische Annahme über die Vorhersageverteilung erforderlich. Im Analyseschritt wird jedes Ensemblemitglied (Partikel) entsprechend seiner

Relevanz, gemessen durch die Beobachtungsverteilung, gewichtet. Aus theoretischer Sicht ist der PF zwar sehr ansprechend, seine Anwendung wird jedoch durch den "Fluch der Dimensionalität" sehr erschwert. Ist die Ensemblegröße relativ klein, wird die effektive Ensemblegröße bei jedem Analyseschritt verringert, da viele Partikel zu kleine Gewichte erhalten. Beispielsweise sind bereits mindestens 10^6 Partikel nötig, damit der PF bei 100 Beobachtungen nicht divergiert. Damit ist der PF, und viele seiner Varianten, für typische hochdimensionale Probleme nicht anwendbar.
Um diese Divergenz zu vermeiden, müssen die Partikel nicht nur hinreichend nahe an der Beobachtung liegen, sondern zudem möglichst gleiche Gewichte aufweisen. Der EWPF versucht, diese Forderungen mithilfe von sogenannten Vorschlagsverteilungen umzusetzen. Dabei wird im Vorhersageschritt ein zusätzlicher Term zu den Modellgleichungen gefügt, der die Partikel näher zur nächsten Beobachtung zieht (*Nudging*). Zudem wird der letzte Vorhersageschritt vor der Beobachtung so modifiziert, dass im Analyseschritt fast alle Partikel das nahezu gleiche Gewicht erhalten. Der EWPF benötigt somit eine adäquate Formulierung des Modellfehlers, um die Freiheit der Vorschlagsverteilungen nutzen zu können, und erfordert zudem eine dem Modell angepasste Implementierung des Vorhersageschritts. Eine Alternative stellt der NLEAF dar, der nur Mittelwert und Kovarianz der Bayes'schen Analyseverteilung berücksichtigt. Er benötigt zwar kein stochastisches Modell, jedoch ist der stochastische Analysealgorithmus sehr rechenintensiv und führt zu ungewollten Zufallsfehlern. Beide Aspekte erschweren eine Anwendung mit kleinen Ensembles in hohen Dimensionen.

4. Herleitung und Eigenschaften des neuen Filters

Basierend auf den zuvor gewonnen Erkenntnissen wird in Kapitel 4 der neue *Nonlinear Ensemble Transform Filter* (NETF) hergeleitet und untersucht. Er baut auf der grundsätzlichen Idee des NLEAF auf, ist jedoch auch für hohe Dimensionen geeignet. Zur Vorbereitung werden zunächst die Monte-Carlo-Schätzer für Mittelwert und Kovarianz der Analyse in eine effiziente Matrixschreibweise umformuliert. Daraus lässt sich eine positiv semidefinite Transformationsmatrix ableiten, die aus den Partikelgewichten gebildet wird. Diese Matrix transformiert das Vorhersageensemble in ein Analyseensemble aus gleich verteilten Zuständen, so dass die ersten beiden Momente exakt den nicht-parametrischen Monte-Carlo-Schätzern entsprechen. Dadurch werden die systematischen Analysefehler des EnKF reduziert. Die Transformation wird zudem noch um eine geeignete Zufallsrotation im Ensembleunterraum ergänzt. Diese erhält den exakten Mittelwert und die Kovarianz des Ensembles.

Der neue Algorithmus ist von allgemeiner Natur und modellunabhängig, da nur der Analyseschritt modifiziert wird. Formal ähnelt er zudem dem ETKF und ist genauso einfach zu implementieren. Insbesondere lässt sich die Analyse analog zum ETKF lokalisieren. Dies erhöht die Stabilität des Filters, da die effektive Dimensionalität des Beobachtungsraums reduziert wird, und ermöglicht somit die Anwendbarkeit in großskaligen Systemen. Eine Analyse der Komplexität des Filters ergibt, dass sein Rechenaufwand dem des ETKF entspricht, und er somit effizient ist. Abschließend wird gezeigt, dass sich der NETF auf triviale Weise mit einem Nudging-Vorhersageschritt verknüpfen lässt, was in stochastischen Systemen von zusätzlichem Vorteil sein kann.

5. Empirische Untersuchung der Filtereigenschaften

Der nächste Schritt, beschrieben in Kapitel 5, besteht in der empirischen Charakterisierung des Filters durch sein Verhalten in DA-Experimenten mit einfachen Modellen unterschiedlicher Dimensionalität und Nichtlinearität. In allen Fällen wird zunächst eine ”wahre” Modelltrajektorie erzeugt. Aus dieser werden gestörte Beobachtungen simuliert, die wiederum dem Filter zur Assimilation zur Verfügung gestellt werden. Die Leistung des Filters kann nun u.a. durch die Abweichungen des Ensemblemittels von der ”Wahrheit” (Analysefehler) sowie der Standardabweichung des Ensembles (Spread) beurteilt werden. Zunächst wird anhand eines skalaren Beispiels demonstriert, dass die zusätzliche Zufallsrotation erforderlich ist, um ein statistisch konsistentes Ensemble zu erzeugen.

Die Anwendung im chaotischen Lorenz-63-Modell zeigt, dass der NETF aufgrund seiner nichtlinearen Formulierung bereits für moderate Ensemblegrößen deutlich bessere Ergebnisse als die EnKF liefert. Eine Ausnahme stellen sehr kleine Ensemblegrößen wie 5 oder 10 dar, bei denen der ETKF bereits sehr effizient sein kann. Der NETF kann in diesen Experimenten jedoch nicht den NLEAF übertreffen, da sich der Zufallsfehler durch die gestörten Beobachtungen in diesem niedrigdimensionalen Fall noch nicht nachteilig auswirkt. Zudem wird demonstriert, dass der NETF grundsätzlich auch in Situationen mit Parameterschätzung und nichtlinearem Beobachtungsoperator überlegene Ergebnisse liefert.
Anschließend werden die Filter in lokalisierter Form in den 80-dimensionalen Lorenz-96- und Lorenz-2005-Modellen getestet. Hier liefert der NETF ebenfalls vielversprechende Ergebnisse, was zeigt, dass die Lokalisierung auch für einen nichtlinearen Filter funktioniert und ihn für höhere Dimensionen anwendbar macht. In diesen Fällen kann die Leistung des NETF nicht nur die KF-basierten Filter übertreffen, sondern auch den NLEAF. Das zeigt, dass der deterministische Analysealgorithmus in höherdimensionalen Situationen von Vorteil ist, da er nicht nur effizienter ist, sondern auch Zufallsfehler vermeidet, die bei den typischerweise kleinen Ensemblegrößen nicht vernachlässigbar sind. Außerdem ergeben die Experimente, dass der NETF auch bei nicht-Gauß’schen Beobachtungsverteilungen Vorteile bringt, da diese von den EnKF nicht direkt berücksichtigt werden können.
In einem abschließenden Experiment wird mit einem einfachen Modell der linearen Advektion eine Situation betrachtet, in der die Zustandsdimension (1000) deutlich größer ist als die Dimensionen von Beobachtung (200) bzw. Ensemble (5...50), was einem realen, großskaligen DA-Problem näher kommt. Hierbei stellt der ETKF per Konstruktion zwar den bestmöglichen Ensemblefilter dar, der NETF kann aber auch hier die Leistung des ETKF annähernd erreichen. Das bestätigt, dass der NETF Probleme hoher Dimensionalität auch mit relativ kleinen Ensembles lösen kann.

Die konsistenten empirischen Ergebnisse lassen somit den Schluss zu, dass der NETF ein gutes Potential für die nichtlineare Datenassimilation aufweist. Die effiziente, deterministische Formulierung des Analyseschritts ist insbesondere in höheren Dimensionen von Vorteil, und durch Lokalisierung bleibt der Filter stabil, obwohl er ausschließlich auf den Bayes’schen Gewichten basiert.

6. Anwendung auf ein hochdimensionales Zirkulationssystem

Im letzten Kapitel 6 wird die tatsächliche Anwendbarkeit des neuen Filters in Problemen geophysikalischer Dimension demonstriert. Hierzu wird das Ozeanmodell NEMO verwendet. Es löst die

primitiven Gleichungen, welche die Dynamik und Thermodynamik der Meereszirkulation beschreiben. Subgitterskalige Prozesse werden durch physikalische Parametrisierungen approximiert, etwa für konvektive und turbulente Flüsse. Die atmosphärische Zirkulation wird in NWP-Systemen grundsätzlich durch einen sehr ähnlichen Satz an Gleichungen und Parametrisierungen modelliert. Zudem spielt die Ozeanassimilation eine wichtige Rolle zur Initialisierung von saisonalen und mittelfristigen Klimaprognosen sowie zur Klimadiagnostik. So bestimmt etwa die Meeresoberflächentemperatur auf längere Sicht maßgeblich die Energieflüsse zwischen Ozean und Atmosphäre. Somit wird ein modernes, sehr komplexes Zirkulationsmodell verwendet, um den NETF in einem realistischen Szenario zu testen.
Um die Filterleistung weitgehend zeitunabhängig betrachten zu können, wird eine vereinfachte NEMO-Konfiguration verwendet. Das Modellgebiet besteht aus $121 \times 81 = 9801$ horizontalen Gitterpunkten sowie 11 vertikalen Schichten. Ein zeitlich konstanter zonaler Windantrieb sorgt für eine großskalige Doppelwirbelzirkulation mit einem zentralen Jet. Charakteristisch sind mesoskalige Stromwirbel (Eddies), welche die chaotische Natur des Systems belegen. Die von dieser Konfiguration erzeugte Zirkulation ist grundsätzlich auch im Nordatlantik beobachten ist.

Die prognostischen Zustandsvariablen in NEMO sind Temperatur (T), Meereshöhe (SSH) sowie zonale (U) und meridionale Fluidgeschwindigkeit (V). Der daraus gebildete Zustandsraum hat eine Dimension von 333234, was belegt, dass es sich um ein hochdimensionales Assimilationsproblem handelt. Zudem wird ein realistisches Beobachtungsszenario nachgebildet, indem sowohl Envisat-Satellitenbeobachtungen der Meereshöhe sowie Temperaturprofile des Argo-Netzwerks simuliert werden. Das Assimilationsexperiment wird im Jahr 75 einer transienten Modellsimulation durchgeführt, wobei alle zwei Tage etwa 3270 Beobachtungen assimiliert werden. Zur technischen Realisierung wurde der NETF zunächst in das Parallel DA Framework (PDAF) implementiert und anschließend NEMO mit PDAF gekoppelt. Hierbei bestätigte sich, dass die Implementierung des ETKF direkt übernommen werden kann. Einzig die Analyseroutine, die das (lokale) Vorhersageensemble in ein (lokales) Analyseensemble transformiert, wurde gemäß dem in Kapitel 4 erläuterten Algorithmus modifiziert. Um die lokale Dimensionalität zu reduzieren, werden die 9801 Ozeansäulen in jedem Analyseschritt unabhängig voneinander aktualisiert, wobei der horizontale Lokalisierungsradius auf $2.5°$ gesetzt ist. Im Mittel werden dadurch pro Säule etwa 100 Beobachtungen, maximal jedoch sogar 200, verwendet. Ein Anfangsensemble von 120 Zuständen wird aus einer 24-jährigen Modellklimatologie gebildet. Diese gibt nur die großskalige Zirkulation wieder, enthält jedoch keine Informationen über die mesoskaligen Eddies zu Beginn des Assimilationsfensters, und stellt damit eine große Herausforderung an den Filter dar. In realen Situationen liegen meist mehr Informationen vor, etwa durch vorangegangene Analysen.

Ein Vergleich der Analysefelder mit den "wahren" Feldern zeigt qualitativ, dass der NETF die tatsächliche Zirkulation, inklusive der mesoskaligen Muster, grundsätzlich sehr gut wiedergibt. Die Leistung des NETF wird weiterhin quantitativ beurteilt. Der Analysefehler wird kontinuierlich reduziert, bis ein Gleichgewicht erreicht ist, in dem der Fehler nur noch zwischen 7-17 % (je nach Variable) gegenüber dem einer freien Ensembleintegration beträgt. Erwartungsgemäß können für die Temperatur und SSH die besten Ergebnisse erzielt werden, aber auch die unbeobachteten Geschwindigkeitsfehler werden deutlich verbessert. Die NETF-Ensembles sind zudem statistisch konsistent, was sich an der analogen Reduzierung des Spread zeigt. Der NETF erweist somit auch in dieser anspruchsvollen Anwendung als praktikable Methode.

Im Vergleich mit dem ETKF zeigt sich, dass die Analyse des NETF nach Erreichen der Konvergenz etwas besser ist, was an der nichtlinearen Formulierung liegen könnte. Jedoch benötigt der NETF deutlich länger, um die Konvergenz zu erreichen. Das kann auf eine Sensitivität des Filters gegenüber dem Anfangsensemble zurückgeführt werden. Im Gegensatz zum ETKF können Ensemblezustände, die einen großen Abstand zu der Beobachtung aufweisen, nicht zur Analyse beitragen. Der EWPF wurde in bisherigen Publikationen nur um den wahren Anfangszustand initialisiert, hier wird eine ähnliche Problematik erwartet. Die konsistente Spezifizierung des Anfangsensembles stellt somit eine wichtige Voraussetzung für nichtlineare Filter dar.

Abschließend werden noch einige ausgewählte Aspekte untersucht, um die Nützlichkeit des NETF zu belegen. Beispielhaft sei hier die weiterführende Simulation im Jahr 76 genannt, die mit dem finalen Analyseensemble aus dem Jahr 75 initialisiert wird. Erwartungsgemäß steigt der Vorhersagefehler in dem chaotischen System mit der Zeit an, bis die Modellklimatologie erreicht ist. Die typische Zeitskala der Fehlerverdopplung kann empirisch auf etwa 50-60 Tage geschätzt werden. Für die Temperatur hingegen bleibt der Fehler bei einem niedrigeren Wert, was auf die langsame Variabilität in tiefen Schichten zurückzuführen ist. Auch wenn das Experiment eine simplifizierte Ozeankonfiguration verwendet, ist es dennoch in der Lage, die Bedeutung der vertikalen Temperaturbeobachtungen für die Ozeananalyse zu unterstreichen. So bildet die Assimilation der Argo-Profile beim ECMWF einen wichtigen Stützpfeiler für die Initialisierung von saisonalen Vorhersagen.

Der erfolgreiche Test des NETF in einem großskaligen Zirkulationssystem zeigt somit, dass der Filter für die hochdimensionale DA anwendbar ist, selbst wenn anfänglich keine Informationen über den wahren Zustand vorliegen. In sämtlichen betrachteten Punkten entspricht das Verhalten des NETF den Anforderungen an eine nützliche Assimilationsmethode.

7. Schlussfolgerungen und Ausblick

Kapitel 7 fasst die wesentlichen in dieser Arbeit erzielten Ergebnisse zusammen und erlaubt daraus resultierende Schlussfolgerungen.

Im Rahmen eines Rückblicks auf klassische Analysemethoden sowie neuere Entwicklungen im Bereich von ensemble-basierten Filtern wurde zunächst der aktuelle Stand der Forschung identifiziert. Ensemble-basierte Filter wie der EnKF werden mittlerweile zwar erfolgreich eingesetzt, liefern aber aufgrund der Gauß'schen Annahme im Analyseschritt nicht die optimale Lösung. Nichtlineare Methoden, basierend auf dem PF, können hingegen in größeren Dimensionen nicht verwendet werden. Der EWPF bietet zwar ein großes Potential, ist jedoch in deterministischen Systemen nicht anwendbar und erfordert eine adäquate Manipulation des Vorhersageschritts.

Der hier entwickelte NETF stellt eine alternative nichtlineare Technik dar, die nur auf den Bayes'schen Partikelgewichten beruht. Im Gegensatz zum EWPF wird jedoch nicht die gesamte Analyseverteilung berücksichtigt. Der einfach zu implementierende Algorithmus betrifft ausschließlich den Analyseschritt, bei dem mithilfe einer Matrixtransformation ein neues Ensemble erzeugt wird, dessen Mittelwert und Kovarianz exakt den nichtlinearen Schätzern entsprechen. Die Analyse kann wie im ETKF lokalisiert werden, was eine Anwendung in großskaligen Systemen erlaubt, und auch der Rechenwand ist sehr

ähnlich. Der NETF ist ein generischer Algorithmus, da der Vorhersageschritt nicht modifiziert wird, und ist daher trotz seiner nichtlinearen Formulierung einfach zu implementieren.

Die Anwendungen in einem Spektrum unterschiedlicher Assimilationsexperimente zeigen, dass der NETF auch mit relativ kleinen Ensembles stabil und erfolgreich arbeitet. In nichtlinearen, chaotischen Systemen liefert er nicht nur bessere Ergebnisse als die EnKF, sondern bietet in höherdimensionalen Problemen auch deutliche Vorteile gegenüber dem NLEAF. Dieser stellt eine stochastische Implementierung dar, die jedoch sehr rechenintensiv ist und bei den typischen, kleinen Ensemblegrößen signifikanten Zufallsfehlern unterliegt. Die Anwendbarkeit des NETF in hochdimensionalen System kann abschließend in einem komplexen Ozeanzirkulationsmodell mit einem realistischen Beobachtungsszenario demonstriert werden. Auch hier liefert der Filter stabile Resultate, konsistente Ensembles und Analysefelder, welche die wahre Zirkulation sehr gut wiedergeben. Dabei wird festgestellt, dass die Spezifizierung des Anfangsensembles die größte Herausforderung für nichtlineare Filter darstellt. In der Praxis kann es daher sinnvoll sein, bis zum Erreichen der Konvergenz einen ETKF zu verwenden, und dann erst zu einem nichtlinearen Filter zu wechseln.

Sämtliche Experimente bestätigen somit konsistent, dass der NETF einen praktikablen Filter für die nichtlineare DA darstellt. Die Transformation des Vorhersageensembles sorgt gemeinsam mit der Lokalisierung der Analyse dafür, dass der Filter auch in hochdimensionalen Problemen mit realistischen Ensemblegrößen stabil bleibt und korrekte Analysen produziert. Somit kann insbesondere der "Fluch der Dimensionalität", die größte Herausforderung für nichtlineare DA-Methoden, prinzipiell und auf einfache Weise umgegangen werden.

Da der NETF einen allgemeinen Algorithmus darstellt und die in dieser Arbeit gezeigten Ergebnisse vielversprechend sind, bietet sich eine Vielzahl möglicher weiterführender Arbeiten an. Auf theoretischer Seite könnten beispielsweise weitere Eigenschaften der Transformationsmatrix untersucht werden, insbesondere der Effekt auf höhere Momente ist derzeit noch ungeklärt. Außerdem könnte der NETF zu einem sequentiellen Glätter weiterentwickelt werden, in Analogie zum ETKS.
Auf praktischer Seite wären, nachdem die prinzipielle Anwendbarkeit auch in hohen Dimension gezeigt worden ist, weitere Anwendungen aufschlussreich. Zum einen könnten weitere Experimente in nichtlinearen Zirkulationsmodellen etwaige Vorteile gegenüber den EnKF besser aufdecken, zum anderen ließen sich die Grenzen des NETF feststellen. Beispielsweise würde sich eine Anwendung in einem atmosphärischen Zirkulationsmodell anbieten. Diese könnte aufzeigen, wie sich der Filter bei einer potentiell großen Dimension des Beobachtungsraums durch die Vielzahl an verfügbaren Beobachtungen, im Vergleich zum Ozean, verhält und ob Lokalisierung in diesem Fall ausreichend ist, um Stabilität zu gewährleisten. Ein Vorteil des NETF gegenüber dem EWPF ist seine leichtere Anwendbarkeit, die auch deterministische Systeme zulässt. In einer Anwendung mit einem großskaligen, stochastischen Modell hingegen könnten NETF und EWPF verglichen werden, was Erkenntnisse über die Relevanz höherer Momente in der nichtlinearen, hochdimensionalen DA erlauben würde. Eine Anwendung in einem Assimilationsproblem mit echten Daten wäre zudem interessant. Hierbei könnte untersucht werden, wie empfindlich der NETF gegenüber systematischen Fehler in Modellen oder Beobachtungen ist.

Publications, proceedings and miscellaneous

Refereed publications

Tödter, J. and B. Ahrens, 2012: Generalization of the ignorance score: Continuous ranked version and its decomposition. *Mon. Wea. Rev.*, **140**, 2005–2017.

Wacker, U., J. Fiebig, J. Tödter, B. R. Schöne, A. Bahr, O. Friedrich, T. Tütken, E. Gischler, and M. M. Joachimski, 2014: Empirical calibration of the clumped isotope paleothermometer using calcites of various origins. *Geochimica et Cosmochimica Acta*, **141**, 127–144.

Tödter, J. and B. Ahrens, 2015: A second-order exact ensemble square root filter for nonlinear data assimilation. *Mon. Wea. Rev.*, **143**, doi:10.1175/MWR-D-14-00108.1. *(accepted)*

Tödter, J., P. Kirchgessner, L. Nerger and B. Ahrens, 2015: Large-scale data assimilation with a simple nonlinear filter. *(in preparation, to be submitted to Mon. Wea. Rev.)*

Conference proceedings

Tödter, J. and B. Ahrens, 2010: Informationstheorie bei der Evaluation von Vorhersagen. *DACH Meteorologentagung*, Bonn, Germany. (Poster presentation)

Tödter, J. and B. Ahrens, 2012: Generalization of information-based concepts in forecast verification. *EGU General Assembly*, Vienna, Austria. (Poster presentation)

Tödter, J., M. Pieroth and B. Ahrens, 2012: Coupled systems assimilation. *3rd International Conference on Earth System Modelling*, Hamburg, Germany. (Poster presentation)

Tödter, J., M. Pieroth and B. Ahrens, 2012: Data assimilation with coupled sub-systems. *International Symposium on Data Assimilation*, Offenbach, Germany. (Poster presentation)

Tödter, J. and B. Ahrens, 2013: Land surface data assimilation with TERRA. *COSMO/CLM User Seminar*, Offenbach, Germany. (Oral presentation)

Tödter, J. and B. Ahrens, 2013: Land surface data assimilation with TERRA. *Patterns in Soil-Vegetation-Atmosphere-Systems: Monitoring, Modelling & Data Assimilation*, Bonn, Germany. (Oral presentation)

Tödter, J. and B. Ahrens, 2013: Merging Classical and Information-Based Forecast Evaluation. *EGU General Assembly*, Vienna, Austria. (Oral presentation)

Tödter, J. and B. Ahrens, 2013: Land surface assimilation in a climate context. *International workshop on seasonal to decadal prediction*, Toulouse, France. (Poster presentation)

Tödter, J. and B. Ahrens, 2013: Land surface assimilation in a climate context. *Data Assimilation Summer School*, Reading, UK. (Poster presentation)

Tödter, J. and B. Ahrens, 2014: The ensemble transform particle filter. *International Symposium on Data Assimilation*, Munich, Germany. (Poster presentation)

Tödter, J. and B. Ahrens, 2014: New initializations of TERRA for climate simulations. *COSMO/CLM User Seminar*, Offenbach, Germany. (Poster presentation)

Tödter, J. and B. Ahrens, 2014: The ensemble transform particle filter. *COSMO/CLM User Seminar*, Offenbach, Germany. (Poster presentation)

Kothe, S., J. Tödter, and B. Ahrens, 2014: Soil moisture assimilation in the framework of regional decadal climate predictability. *Eumetsat satellite soil moisture validation and application workshop*, Amsterdam, Netherlands. (Poster presentation)

Tödter, J., S. Kothe, and B. Ahrens, 2014: Land surface initialization in a climate context. *CLM Community Assembly*, Frankfurt, Germany. (Oral presentation)

Tödter, J., M. Breil, T. Engel, A. Paxian, H.-J. Panitz, D. Sein, M. Warscher, B. Ahrens, A. Fink, D. Jacob, H. Kunstmann, H. Paeth, 2015: Decadal predictability of West African monsoon and impact of improved initial and boundary conditions: land surface. *MiKlip Status Seminar*, Offenbach, Germany. (Poster presentation)

Tödter, J., P. Kirchgessner, and B. Ahrens, 2015: A simple new filter for nonlinear high-dimensional data assimilation. *EGU General Assembly*, Vienna, Austria.

Kirchgessner, P., J. Tödter, and L. Nerger, 2015: A comparison of linear and non-linear data assimilation methods using the NEMO ocean model. *EGU General Assembly*, Vienna, Austria.

Tödter, J., S. Kothe and B. Ahrens, 2015: Ensemble-based, large-scale land surface data assimilation with TERRA and satellite data. *COSMO/CLM User Seminar*, Offenbach, Germany.

Theses

Tödter, J., 2008: Introduction to Markov Chain Monte Carlo Methods Applied on Ising Spin Systems. B.Sc. Thesis, Institute for Theoretical Physics, Goethe University, Frankfurt/Main.

Tödter, J., 2011: New aspects of Information Theory in Probabilistic Forecast Verification. M.Sc. Thesis, Institute for Theoretical Physics, Goethe University, Frankfurt/Main.

Tödter, J., 2015: Derivation and Characterization of a New Filter for Nonlinear High-Dimensional Data Assimilation. PhD Thesis, Institute for Atmospheric and Environmental Sciences, Goethe University, Frankfurt/Main.

Non-refereed publications

Tödter, J., S. Kothe and B. Ahrens, 2015: Inverse Modellierung zur Verbesserung von Vorhersagen. *Hochleistungsrechnen in Hessen*, Hessisches Kompetenzzentrum für Hochleistungsrechnen. *(to appear)*.

Internally-refereed articles and reports

Tödter, J., and B. Ahrens, 2013: Decadal predictability of West African monsoon and impact of improved initial and boundary conditions. *Progress report 2012*, MiKlip project.

Marotzke, J., et al., 2014: MiKlip - a National Research Project on Decadal Climate Prediction. *MiKlip milestone report*, MiKlip project.

Tödter, J., and B. Ahrens, 2014: Decadal predictability of West African monsoon and impact of improved initial and boundary conditions. *Progress report 2013*, MiKlip project.

Students (supervision)

Pieroth, M., 2012: Datenassimilationsexperimente in einem gekoppelten Lorenzsystem. B.Sc. Thesis, Institute for Atmospheric and Environmental Sciences, Goethe University, Frankfurt/Main.

Leps, N., 2013: Soil temperature data assimilation experiments. B.Sc. Thesis, Institute for Atmospheric and Environmental Sciences, Goethe University, Frankfurt/Main.

Reitz, T., 2014: Data assimilation experiments of soil moisture with an ensemble transform Kalman filter. B.Sc. Thesis, Institute for Atmospheric and Environmental Sciences, Goethe University, Frankfurt/Main.

Pieroth, M., 2014: Non-Gaussian forecast skill in ensemble prediction systems. M.Sc. Thesis, Institute for Atmospheric and Environmental Sciences, Goethe University, Frankfurt/Main. (Co-advisor)

External research and training

- 02/2012: Background and application of COSMO/CLM (training), DWD, Langen, Germany.
- 04-05/2012: Data assimilation and the use of satellite data (training), ECMWF, Reading, UK.
- 04/2013: High Performance Computing Hessen (workshop), Technical University Darmstadt, Germany.
- 07/2013: Data Assimilation & Inverse Problems (Summer school/Creative workshop), University of Reading, Reading, UK.
- 03/2014: Parallelization with MPI and OpenMP (training), Gutenberg University, Mainz, Germany.
- 11/2014: Nonlinear ocean data assimilation (invited visiting fellowship), Alfred-Wegener-Institut, Bremerhaven, Germany.

Reviewer activity

- *Monthly Weather Review* (Editor: Dr. J. Anderson, National Center for Atmospheric Research, Boulder, USA)
- *Meteorology and Atmospheric Physics* (Editor: Prof. Dr. B. Ahrens, Goethe University, Frankfurt/Main)