

---

# Joint state and parameter estimation to address model error in convective scale numerical weather prediction systems

Yvonne Muriel Ruckstuhl

---



München 2019



---

# **Joint state and parameter estimation to address model error in convective scale numerical weather prediction systems**

**Yvonne Muriel Ruckstuhl**

---

Dissertation  
an der Fakultät für Physik  
der Ludwig-Maximilians-Universität  
München

vorgelegt von  
Yvonne Muriel Ruckstuhl  
aus Genf, Schweiz

München, den 03 Mai 2019

Erstgutachter: Dr. Tijana Janjic

Zweitgutachter: Prof. Dr. Mark Wenig

Tag der mündlichen Prüfung: 19 Juni 2019

**Parts of this thesis are included in:**

**Ruckstuhl, Y.** and Janjic, T.: Parameter and state estimation with ensemble Kalman filter based algorithms for convectivescale applications. *Quarterly Journal of the Royal Meteorological Society*. 2018; 144: 826 841. <https://doi.org/10.1002/qj.3257>

Janjić, T., **Ruckstuhl, Y.**, Toint, P. L.: An algorithm for optimization with disjoint linear constraints and its application for predicting rain. *to be submitted*. 2019

**Ruckstuhl, Y.**, Janjić, T.: Combined state-parameter estimation with the LETKF in near operational setup for convective-scale weather forecasting. *to be submitted*. 2019



# Zusammenfassung

Numerische Wettervorhersagemodelle benötigen Anfangsbedingungen, um Wettervorhersagen zu erstellen. Diese Anfangsbedingungen werden durch einen als Datenassimilation bezeichneten Prozess berechnet, bei dem zuvor berechnete Modellzustände unter Verwendung neu gewonnener Beobachtungen der Atmosphäre aktualisiert werden. Das Datenassimilationssystem (KENDA) des Deutschen Wetterdienstes für regionale Vorhersagen basiert auf dem Ensemble Kalman Filter (EnKF), der unter der Annahme eines perfekten Modells im stochastischen Sinne und einer gaußverteilten Fehlerstatistik entworfen wurde. Da keine dieser Annahmen für operative Wettermodelle mit explizit modellierter Konvektion gilt, können Verbesserungen erzielt werden indem Methoden und Algorithmen mit schwächeren Annahmen entwickelt werden.

In dieser Arbeit untersuchen wir die Möglichkeit Modellfehler zu reduzieren indem wir unsichere statistische Modellparameter unter Verwendung von Datenassimilationstechniken stören und abschätzen. Insbesondere verwenden wir den Augmented-State-Ansatz, bei dem Parameter durch Beobachtungen über ihre Korrelation mit beobachteten Zustandsvariablen aktualisiert werden. Dieser Online-Ansatz bietet eine flexible und dennoch konsistente Möglichkeit Modellvariablen, die von den ausgewählten Parametern betroffen sind, besser an Beobachtungen anzupassen und gleichzeitig realisierbare Modellzustände sicherzustellen. Eine wesentliche Herausforderung besteht darin, eine Wahrscheinlichkeitsverteilung für die Parameter zu konstruieren, welche die Unsicherheit des anvisierten Modellfehlers widerspiegelt.

Wir zeigen im Rahmen eines operativen Modellsystems, dass die Darstellung von Wolken in COSMO-DE verbessert wird, wenn der Parameter für die zweidimensionale Rauigkeitslänge mit dem Ansatz des erweiterten Zustands geschätzt wird. Der betrachtete Modellfehler bezieht sich hier auf die Rauigkeitslänge selbst und die Oberflächenflüsse, welche die Einleitung der Konvektion beeinflussen. Die Wahrscheinlichkeitsdichtefunktion der Rauigkeitslänge, und schlussfolgernd der Modellfehler, der den Oberflächenflüssen entspricht, wird als gaußverteilt und mit einer bestimmten Kovarianzmatrix angenommen. Die Ergebnisse sind sehr empfindlich gegenüber der Wahl der Kovarianzmatrix und legen nahe, wie wichtig es ist die Oberflächenwindmessungen zu assimilieren.

Außerdem evaluieren wir zwei kürzlich entwickelte Modifikationen des EnKF, die entweder explizite Einschränkungen enthalten, wie die Massenerhaltung und Positivität des Niederschlags durch Lösen von eingeschränkten Minimierungsproblemen (QPEns), oder Momente höherer Ordnung, wie etwa die Schiefe (QF), einführen, um nicht-gaußverteilte

Fehlerstatistiken annehmen zu können. Wir zeigen in einem idealisierten Setup, dass die Schätzung der Parameter durch die Einführung von QF profitiert (auch für moderate Ensemblegrößen) und dass die QPEns den EnKF leistungsmäßig generell deutlich übertreffen. Um den hohen Rechenaufwand der QPEns zu reduzieren, schlagen wir zwei neue Ansätze vor. Einer der zwei vorgeschlagenen Algorithmen nutzt die Eigenschaften der zu lösenden Minimierungsprobleme aus, der Andere trainiert ein neuronales Netzwerk, um die durch die QPEns erzeugten Anfangsbedingungen aus den vom EnKF erzeugten Bedingungen zu reproduzieren.

Obwohl wir zeigen, dass der Modellfehler durch das Schätzen geeigneter Modellparameter selbst in einem operativen Setup erheblich reduziert wird, diskutieren wir weitere Ansätze und mögliche Untersuchungen, die potenziell zu zusätzlichen großen Leistungsvorteilen führen können.

# Abstract

Numerical weather prediction models need initial conditions to produce weather forecasts. These initial conditions are computed through a process called data assimilation, where previously computed model states are updated using newly obtained observations of the atmosphere. The data assimilation system (KENDA) employed at the German Weather Service for regional forecasts is based on the Ensemble Kalman Filter (EnKF), which was designed under the assumption of a perfect model in a stochastic sense and Gaussian error statistics. As neither of these assumptions is valid for operational convection permitting weather prediction models, improvement can be gained by developing methods and algorithms for which these assumptions can be relaxed.

In this thesis we investigate the feasibility of addressing model error by perturbing and estimating uncertain static model parameters using data assimilation techniques. In particular we use the augmented state approach, where parameters are updated by observations via their correlation with observed state variables. This online approach offers a flexible, yet consistent way to better fit model variables affected by the chosen parameters to observations, while ensuring feasible model states. A key challenge is to design the probability distribution of the parameters, which should reflect the uncertainty of the targeted model error.

We show in an operational setup that the representation of clouds in COSMO-DE is improved if the two dimensional roughness length parameter is estimated with the augmented state approach. Here, the targeted model error is the roughness length itself and the surface fluxes, which influence the initiation of convection. The probability density function of the roughness length, and by extension the model error corresponding to the surface fluxes, is assumed Gaussian with a certain covariance matrix. The results are highly sensitive to the choice of covariance matrix, and strongly suggest the importance of assimilating surface wind measurements.

In addition we evaluate two recently developed modifications of the EnKF that either explicitly incorporate constraints such as mass conservation and positivity of precipitation by solving constrained minimization problems (QPEnS), or introduce higher order moments such as skewness (QF) to deal with non-Gaussian error statistics. We show in an idealized setup that the estimation of parameters benefits from the QF (even for moderate ensemble sizes) and that the QPEnS generally outperforms the EnKF significantly. To reduce the high computational costs of the QPEnS we propose a new algorithm that exploits properties of the minimization problems that need to be solved. We also explore a different approach

where we train a neural network to reproduce the initial conditions generated by the QPEs from those generated by the EnKF.

Besides the encouraging finding that even in a near operational setup model error is significantly reduced by estimating appropriate model parameters, we provide various suggestions for further research that can lead to further improvements.

# Contents

<b>Zusammenfassung</b>	<b>vii</b>
<b>Abstract</b>	<b>ix</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Operational data assimilation algorithms . . . . .	2
1.2 General challenges for convective scale DA . . . . .	5
1.2.1 Observation related challenges . . . . .	5
1.2.2 Model related challenges . . . . .	6
1.2.3 Method related challenges . . . . .	8
1.3 Scientific Goal . . . . .	10
1.3.1 Approach . . . . .	10
1.3.2 Challenges of parameter estimation . . . . .	10
1.4 Outline . . . . .	11
<b>2 Mathematical background of ensemble methods</b>	<b>13</b>
2.1 Ensemble Kalman filters . . . . .	13
2.1.1 Kalman Filter . . . . .	13
2.1.2 Stochastic EnKF . . . . .	15
2.1.3 ETKF . . . . .	16
2.2 Localization . . . . .	18
2.3 Augmented state parameter estimation . . . . .	18
<b>3 Evaluating data assimilation algorithms in an idealized setup</b>	<b>21</b>
3.1 Data assimilation algorithms . . . . .	22
3.1.1 Quadratic Programming Ensemble . . . . .	22
3.1.2 Quadratic Filter . . . . .	23
3.2 Augmented state parameter estimation . . . . .	24
3.2.1 Localization in parameter space . . . . .	24
3.2.2 Dynamical model for parameters . . . . .	26
3.3 Experiment setup . . . . .	26
3.4 Results . . . . .	28
3.4.1 State estimation . . . . .	29

3.4.2	Parameter estimation . . . . .	31
3.5	Conclusion . . . . .	43
<b>4</b>	<b>Estimation of roughness lengths in COSMO-KENDA</b>	<b>45</b>
4.1	COSMO . . . . .	46
4.1.1	Parameterizations . . . . .	46
4.1.2	Subgrid-scale turbulence parametrization . . . . .	48
4.1.3	Surface Fluxes . . . . .	48
4.1.4	Representation of roughness length . . . . .	50
4.2	Observations . . . . .	50
4.2.1	Conventional observations . . . . .	50
4.2.2	Radar data . . . . .	51
4.2.3	Satellite data . . . . .	53
4.3	KENDA . . . . .	53
4.3.1	Localization . . . . .	54
4.3.2	Spread inflation techniques . . . . .	54
4.3.3	Observation errors . . . . .	55
4.3.4	Implementation notes . . . . .	55
4.4	Experiment setup . . . . .	56
4.4.1	Dynamical model for roughness length . . . . .	56
4.4.2	Weather situation . . . . .	57
4.5	Results . . . . .	59
4.5.1	Roughness length evolution . . . . .	59
4.5.2	Verification of atmospheric fields against observations . . . . .	64
4.6	Conclusion . . . . .	72
4.7	Critical look and future work . . . . .	78
4.7.1	Heat versus momentum flux . . . . .	78
4.7.2	Spurious correlations . . . . .	78
4.7.3	Tuning . . . . .	79
4.7.4	Resampling . . . . .	79
4.7.5	Scale separation . . . . .	79
4.7.6	QPEnS and QF . . . . .	80
<b>5</b>	<b>Reducing computational costs for the QPEnS</b>	<b>81</b>
5.1	QP for disjoint constraints . . . . .	81
5.1.1	Problem . . . . .	81
5.1.2	Background . . . . .	82
5.1.3	The new algorithms . . . . .	84
5.1.4	Performance summary . . . . .	90
5.2	Neural Network to replace QPEnS . . . . .	91
5.2.1	What is a neural network? . . . . .	91
5.2.2	Convolutional neural network . . . . .	91
5.2.3	Training a neural network . . . . .	93

---

5.2.4	Experiment setup . . . . .	94
5.2.5	Results . . . . .	95
5.3	Conclusion . . . . .	97
<b>6</b>	<b>Conclusion</b>	<b>101</b>
<b>A</b>	<b>Verification scores</b>	<b>105</b>
A.1	Bias . . . . .	105
A.2	Root Mean Squared Error (RMSE) . . . . .	105
A.3	Spread . . . . .	106
A.4	Fractions Skill Score (FSS) . . . . .	106
A.5	False Alarm Ratio (FAR) and Equitable Threat Score (ETS) . . . . .	106
<b>B</b>	<b>Proof of positive definite localized error covariance matrix</b>	<b>109</b>
<b>C</b>	<b>KKT conditions</b>	<b>111</b>
C.0.1	Equality constrained optimization . . . . .	111
C.0.2	Inequality constrained optimization . . . . .	112
	<b>List of Abbreviations</b>	<b>115</b>
	<b>Bibliography</b>	<b>128</b>
	<b>Acknowledgement</b>	<b>129</b>



# Chapter 1

## Introduction

Throughout history, weather has strongly effected human life. From choice of clothing, to agricultural impacts, to natural disasters caused by floods and storms, even to influencing the outcome of important political wars (Ackinson, 1973). Knowing the weather ahead of time can therefore save lives and prevent severe economic losses. For millennia people have tried to predict the weather. In ancient times observation by eye was the driving mechanism. Experts would recognize recurring cloud patterns and predict their evolution based on experience. This knowledge remains valuable today, and is relied upon by many people all over the world. Nevertheless, the field of meteorology continued to develop and was brought to a new level with the invention of the electric telegraph. This facilitated the gathering of local weather reports, which were then summarized in the form of weather maps to track the evolution of atmospheric patterns, eventually leading to primitive, yet effective synoptic scale weather forecasting (Frisinger, 1978). It was not until the early 1900's that Abbe (1901) suggested to describe atmospheric flow in terms of mathematical equations. A practical formulation of his work followed by Bjerknes (1904), in the form of an initial value problem. However, this weather prediction approach became only feasible in the second half of the century when computers began to become available. In 1922 Lewis Fry Richardson led an early attempt to manually solve the numerical equations for a six hour forecast over only two points in central Europe, but after 6 weeks of computation time, the predicted change in surface pressure was unrealistic by two orders of magnitude (Richardson, 1922). This large error was caused by an imbalance in the pressure and wind velocity fields that were used as initial conditions for the initial value problem. In contrast to humans, computers could process large amounts of data to generate proper initial conditions and subsequently provide numerical solutions to the initial value problem. This marked the beginning of numerical weather prediction (Lynch, 2008).

As illustrated by the failed attempt in 1922 to produce a realistic 6 hour weather forecast, the generation of initial conditions or the *analysis*, based on observations is not straightforward. It requires a function that maps a set of sparse (sometimes indirect) observations to a physically consistent model state, which consists of all spatially discretized prognostic variables of the weather prediction model. It therefore justifies a separate research field known as *data assimilation (DA)*. First data assimilation attempts were based

on simple interpolation techniques, which were superseded by more sophisticated methods where *a priori* computed model states, referred to as the *background*, are used to supplement observations (Bergthorsson and Döös, 1955). A significant development for data assimilation was achieved by accounting for background and observation errors via “statistical interpolation” (Gandin, 1963), for which the foundation can be traced back to work by Kolmogorov (1941) and Wiener (1949). This method linearly combines current observations with its estimated error variance and a background state (which can be climatological or a model prediction computed from previous initial conditions) with its estimated error variance, such that the analysis is unbiased and its error variance is minimized, i.e. the analysis is the *Best Linear Unbiased Estimate (BLUE)* of the true state, obtained by optimizing the weights assigned to the observations and background respectively. It is important to note that statistical interpolation can be formulated as a variational least squares method (Lorenç, 1986). In other words, the problem of minimizing a function of weights can be reformulated to a problem of minimizing a quadratic cost function of a model state. This offers a more natural framework to incorporate any dynamical constraints and solve for the analysis iteratively, which is sometimes necessary due to computational restrictions.

## 1.1 Operational data assimilation algorithms

With the quality of weather prediction models and atmospheric observations available today, the background state is usually set as the most recent model forecast available at the time of assimilation, serving as a *first guess* of the true state of the atmosphere. This first guess is compared to available observations for quality control. Observations that are too far from the first guess are rejected to avoid physically inconsistent initial conditions as well as assimilation of poor data. The process of generating initial conditions is repeated in time to always incorporate newly available observations and is therefore referred to as the *data assimilation cycle*, which is schematically illustrated in Figure 1.1.1.

Data assimilation algorithms operational today are still based on least squares methods, although they differ in their representation of the error statistics as well as the implementation. Which algorithm to choose depends on the resolvable scale of the numerical weather prediction model, the climatology of the region of interest, the availability of observations and their nature, and computational capacity. Perhaps the most apparent operational algorithm is 3D-Var, which minimizes a cost function  $J_{3DVar}(\mathbf{x})$ , where  $\mathbf{x}$  is a model state, consisting of two terms: the squared distance between  $\mathbf{x}$  and the background  $\mathbf{x}^f$ , weighted by the inverse of the error covariance matrix of the background  $\mathbf{B}^{-1}$ , and the squared distance between  $\mathbf{x}$  in observation space and the observations  $\mathbf{y}$ , weighted by the inverse of the error covariance matrix of the observations  $\mathbf{R}^{-1}$

$$J_{3DVar}(\mathbf{x}) = \frac{1}{2} \|\mathbf{x} - \mathbf{x}^f\|_{\mathbf{B}^{-1}}^2 + \frac{1}{2} \|\mathbf{y} - \mathbf{H}\mathbf{x}\|_{\mathbf{R}^{-1}}^2, \quad (1.1.1)$$

where  $\mathbf{H}$  is the linearized observation operator that maps a model state to observation space. The matrix  $\mathbf{B}$  is stationary and based on many simplifying assumptions of clima-

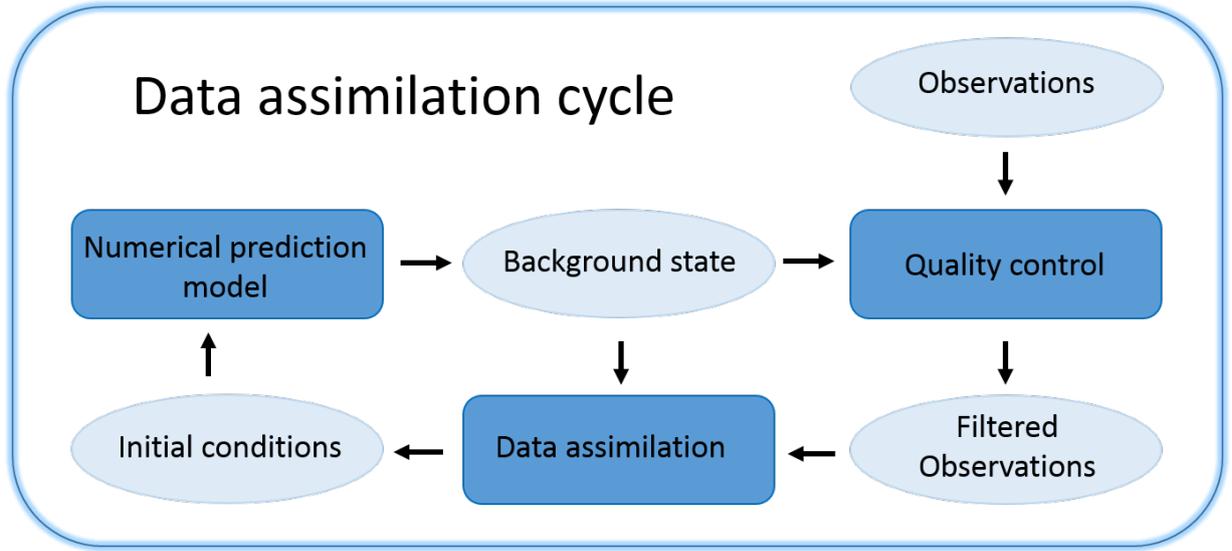


Figure 1.1.1: Schematic illustration of the data assimilation cycle.

tological information and chosen such that the analysis is well balanced. Though 3D-Var performs adequately, it has the disadvantage that it cannot handle assimilation of observations at the correct time. Typically, initial conditions are computed at fixed hours, which does not necessarily match the time that observations are made. 4D-Var offers a (costly) solution by using a forward linearized model operator  $\mathbf{M}_t$  that propagates a model state from initial time to time  $t$ , to compare the analysis with the observations at the correct time  $t = 0, 1, \dots, T$ . As a consequence, the minimization is done over a trajectory  $\{\mathbf{M}_t \mathbf{x} : t = 0, 1, \dots, T\}$  instead of over a single state

$$J_{4D_{strong}}(\mathbf{x}) = \frac{1}{2} \|\mathbf{x} - \mathbf{x}^f\|_{\mathbf{B}^{-1}}^2 + \frac{1}{2} \sum_{t=0}^T \|\mathbf{y}_t - \mathbf{H}\mathbf{M}_t \mathbf{x}\|_{\mathbf{R}^{-1}}^2. \quad (1.1.2)$$

This formulation was referred to as strong constraint 4D-Var by Sasaki (1970), because the linearized model serves as a strong constraint for the considered trajectories. If computational capacity allows it, the cost function should be relaxed to account for *model error*. Model error is defined as the error arising from deficiencies of the model to represent the true (projected) atmospheric state. Sources of model error are for example simplifications of model equations and their discretization in time and space. Therefore, instead of comparing observations with a forwarded model state  $\mathbf{M}_t \mathbf{x}$ , they should be compared to a state  $\mathbf{x}_t$  which appears in an additional term of the cost function that measures the squared distance between  $\mathbf{x}_t$  and the corresponding model state  $\mathbf{M}_t \mathbf{x}$ , weighted by the inverse of the covariance matrix of the error of the linearized model  $\mathbf{Q}$

$$J_{4D_{weak}}(\mathbf{x}) = \frac{1}{2} \|\mathbf{x} - \mathbf{x}^f\|_{\mathbf{B}^{-1}}^2 + \frac{1}{2} \sum_{t=0}^T \|\mathbf{y}_t - \mathbf{H}\mathbf{x}_t\|_{\mathbf{R}^{-1}}^2 + \frac{1}{2} \sum_{t=0}^T \|\mathbf{x}_t - \mathbf{M}_t \mathbf{x}\|_{\mathbf{Q}^{-1}}^2. \quad (1.1.3)$$

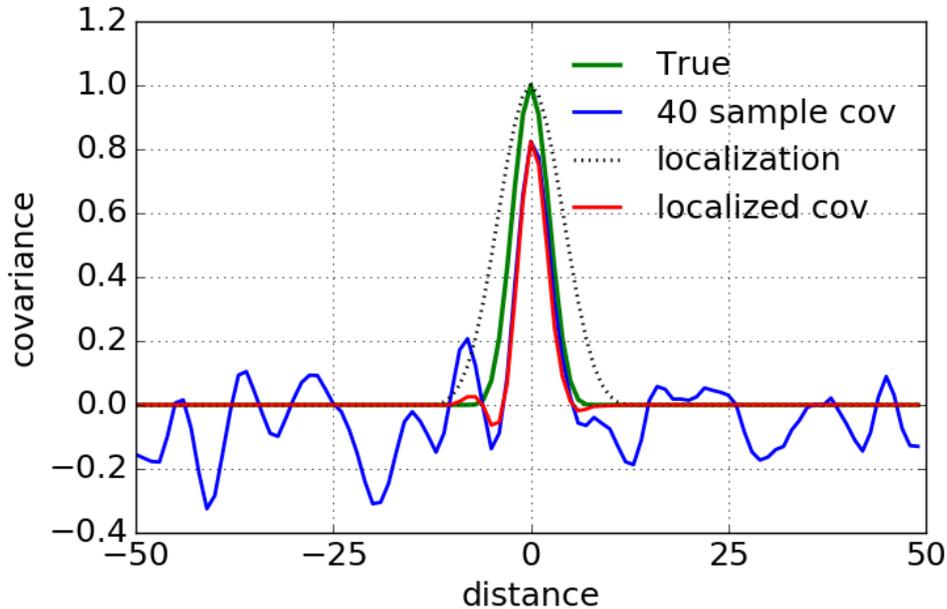


Figure 1.1.2: Illustrative example of sampling error and localization. The green solid line depicts the true covariance and the blue solid line the sample covariance with 40 samples. Signal to noise ratio is large when covariances are small. When covariance localization is applied, i.e. the sample covariance is multiplied by the localization function (dotted black line), the spurious correlations are damped and the result is the solid red line.

Though these variational methods are very successful, they lack non-linear flow dependent information of the background error statistics. For this reason ensemble methods like the Ensemble Kalman Filter (EnKF) (Evensen, 1994, 2003) have become important for atmospheric data assimilation. Instead of using a fixed climatological matrix  $\mathbf{B}$ , ensemble methods aim to compute flow dependent error statistics in the form of error covariance matrix  $\mathbf{P}_t^f$  from an ensemble which consists of several parallel model simulations representing the probability distribution of the background field. The cost function corresponding to the EnKF is then

$$J_{EnKF}(\mathbf{x}) = \frac{1}{2} \|\mathbf{x} - \mathbf{x}^f\|_{\mathbf{P}_t^{f-1}}^2 + \frac{1}{2} \|\mathbf{y} - \mathbf{H}\mathbf{x}\|_{\mathbf{R}^{-1}}^2. \quad (1.1.4)$$

A problem that arises when applying ensemble methods, is that computational capacity only allows for a modest ensemble size, causing severe spurious correlations and rank deficiency in  $\mathbf{P}^f$ . The solution is to use known physical properties of the model to post process the sample error covariance matrix  $\mathbf{P}^f$ . This process is referred to as *localization*. A popular localization technique is multiplying each sample covariance  $\text{Cov}[x_j, x_l]$  with  $0 \leq c(d_{j,l}) \leq 1$ , where  $d_{j,l}$  is the Euclidean distance between  $x_j$  and  $x_l$ , thereby reducing spurious correlations caused by sampling errors. This approach is illustrated in Figure

## 1.1.2.

Many operational centers seek to combine ensemble methods with variational methods to benefit from the advantages of both, creating hybrids like 3DEnVar and 4DEnVar (Buehner, 2005; Liu et al., 2008). The German Weather Service uses a hybrid 3DEnVar for its global model ICOSahedral Nonhydrostatic (ICON, Zängl et al., 2015). Its regional model is called Consortium for Small Scale Modeling (COSMO, Baldauf et al., 2011) for which data assimilation system Kilometer-Scale Ensemble-based Data Assimilation (KENDA, Schraff et al., 2016) is utilized. KENDA is based on an EnKF-type method called Localized Ensemble Transform Kalman Filter (LETKF) (Hunt et al., 2007), which will be introduced in Chapter 2. In Chapter 4 we employ the COSMO-KENDA weather forecasting system and describe it there in more detail.

## 1.2 General challenges for convective scale DA

As mentioned before the choice of data assimilation method depends on scale, regional climate, type and availability of observations, computational capacity, but also on the goal of the weather prediction system. It is natural that the type of challenges arising from data assimilation depends on the same aspects. In this thesis we are mostly concerned with the prediction of clouds and precipitation for the mid-latitudes. We aim for practically applicable methods and are therefore constrained by the computational capacity of the German Weather Service. For a thorough overview of operational convective scale data assimilation systems and the corresponding challenges, we refer to Gustafsson et al. (2018).

Most challenges for regional numerical weather prediction systems arise from the chaotic and highly nonlinear nature of convection. Unfortunately, high impact events such as thunderstorms are especially hard to forecast (Keil et al., 2014). Many studies (Lorenz, 1969; Rotunno and Snyder, 2008; Selz and Craig, 2015) indicate intrinsic limits to the predictability of both small and large scale atmospheric features. These studies also show that error growth is the fastest at small scales. Yet, with a dense observation network, a good representation of the evolution of error statistics and a method that can process all this in a statistically and dynamically consistent way, we should be able to make progress in predicting precipitation that is tied to high impact events such as storms and floods. Below we provide a brief overview of the challenges that occupy the data assimilation community.

### 1.2.1 Observation related challenges

Due to the low predictability of rapidly evolving convective events it is important to have a dense observational network to feed to the prediction model. However, the challenge is not only the availability of observations, but also how to use them. In fact, there are many relevant observations that are not assimilated at all, such as various data obtained from satellites, wind and solar power production systems, and even mobile phones. Apart from any possible political obstacles, the assimilation of a certain data type requires a proper,

possibly nonlinear, forward operator, which maps a model state to a field that allows direct comparison to the data. The development of such operators and estimating the corresponding error variances, is not always straightforward (Zeng, 2014; Scheck et al., 2016). Moreover, the existence of correlated observation errors remains a significant challenge. Though addressed in many important studies (Weston et al., 2014; Bormann et al., 2016; Campbell et al., 2017), they are still ignored in operational weather prediction systems. In addition, the currently implemented data assimilation algorithm at the German Weather Service is constrained in the number of observations it can process. This is due to the rank deficiency of covariance matrix  $\mathbf{P}_t^f$ , inherited from a limited ensemble size. Suggestions to alleviate this issue exist (Sommer and Janjić, 2018), but have yet to be implemented and thoroughly tested.

As motivated above, a dense observational network is needed to track the evolution of convection. On the other hand, it has been hypothesized that predictability of convection can benefit from identification of synoptical forcings as well. It is however not evident how to project observational error statistics to different scales and assimilate them as such (Janjić et al., 2018). Finally, it is not yet clear which type of observations dominate the potential quality of the model prediction. Studies have aimed to identify the relative importance of observations assimilated operationally (Sommer and Weissmann, 2016; Necker et al., 2018), but little can be said about the potential of observation types that have not yet been assimilated. Fabry and Sun (2010) did a study that indicates high sensitivity of the model state to vertical moisture, wind and temperature profiles, suggesting the need for dense and accurate measurements of these quantities. However, if assimilating these additional observations would actually improve a forecast cannot (yet) be verified. Further discussion of these important observation related challenges is beyond the scope of this thesis. Instead we focus on challenges discussed next.

### 1.2.2 Model related challenges

Naturally, the accuracy of the numerical prediction model is a key factor for the quality of the weather forecast. However, since we know that our models will never be perfect, it is important to deal with the resulting model error properly to ensure that the uncertainty of a model prediction is well interpretable. Since model errors contribute to the background error, they should be represented in the background error statistics. This is especially important for the computation of initial conditions, as overestimation of the accuracy of the background and underestimation of its covariances causes suboptimal use of observations which can lead to filter divergence (Dee, 1995).

Uncertainties can be addressed by summarizing model error at time of analysis in a covariance matrix  $\mathbf{Q}$ , as in equation (1.1.3). One could sample from matrix  $\mathbf{Q}$  to properly represent model error in an ensemble. However, constructing matrix  $\mathbf{Q}$  is an extensive challenge on its own, as there are many sources of model error, all interacting with each other. Moreover, covariances are not necessarily enough to describe the full model error statistics. Model equations are often simplified and discretized in time and space. As a result, many relevant physical processes are not resolved and their mean impact on the

model equations needs to be modeled separately in a parametrization scheme. Several successful stochastic physical parametrization schemes have been developed with the aim to account for the mean as well as the variability of the impact of unresolved processes on the mean flow (see Berner et al. (2017) for a review). However, in operational regimes these schemes are not sufficient to fully avoid under dispersion of the error covariance matrix  $\mathbf{P}^f$ . It remains therefore necessary to explicitly address model error in the data assimilation framework as well.

To complement any use of stochastic physical parametrization schemes, model error is typically accounted for by necessary *inflation* techniques embedded in the data assimilation algorithm. Perhaps the simplest form is multiplicative inflation (Anderson and Anderson, 1999), where error covariance matrix  $\mathbf{P}^f$  is multiplied with a factor  $\rho > 1$ . Such covariance inflation techniques are helpful, but they are not designed to target specific model errors, leading to suboptimal results. As an alternative, several successful additive inflation approaches have been proposed, where model error samples are added to the initial conditions in a stochastic manner. These samples can be obtained from climatological systems with the aim to account for large scale errors (Rhodin et al., 2013; Zängl et al., 2015), or from a higher resolution system to account for sub-grid scale errors (Zeng et al., 2019). Sommer and Janjić (2018) proposed augmenting the prior ensemble with similar samples instead of adding them to the posterior ensemble, thereby creating a large synthetic ensemble at affordable cost to improve the quality of  $\mathbf{P}^f$ . They show in an idealized setup that this method reduces the need of multiplicative inflation and localization.

Additive inflation techniques, though effective, usually do not directly consider observational data, which is a rich source of flow dependent information. In this thesis we therefore use observations to estimate perturbed parameters with data assimilation to represent model error. This model error can be the uncertainty of the parameters themselves, or quantities that are affected by them. By projecting model error onto parameters that are estimated along with the state using data assimilation (Jazwinski, 1970; Evensen, 2009), we hope to represent and even reduce the model uncertainty. Many studies show that in low resolution models, under ideal conditions, i.e. conditions in which the only model error source is the estimated parameter, this approach has the potential to significantly improve forecasts (e.g., Aksoy et al., 2006; Koyama and Watanabe, 2010; Schirber et al., 2013; Ruiz et al., 2013a). However, it is not clear how the joint state and parameter estimation algorithm reacts to the highly non-linear relations and non-Gaussian error statistics inherited from convection. Nor is it clear how the parameter estimation is affected by model errors that are not directly related to the parameter. Ruiz and Pulido (2015) took a step towards operational complexity in a simple atmospheric circulation model by introducing model error in the form of perturbed parameters that are not estimated. Though the estimated parameters did not converge to their corresponding perfect model value, the analysis error and the forecast skill did improve by compensating for the perturbed parameters that were not assimilated. It is difficult to predict how parameter estimation impacts state predictions in the presence of unknown and correlated model errors. Only few studies investigated this in quasi-operational setup (Annan et al., 2005; Kondrashov et al., 2008; Schirber et al., 2013; Doron et al., 2013; Simon et al., 2015). All of these studies were conducted on large

scale atmospheric or ocean models. The overarching conclusion of these studies is that forecasts generally benefit from parameter estimation on short time scales, but in some cases biases are introduced on climatological scales. In addition to experiments with the assimilation of real observations, some of these studies conducted corresponding idealized experiments where different forms of model error were introduced to investigate the accuracy of the parameter estimation under varying conditions. They all found the same as Ruiz and Pulido (2015): parameters do not converge to their true value in the presence of model error, even if they do in a perfect model setting. The effectiveness of parameter estimation is therefore sensitive to the application. The only explored meso-scale application that we know of was demonstrated by Hu et al. (2010), who successfully estimated two global parameters in the planetary boundary layer with the assimilation of wind profiler observations with the goal to improve the simulation of transport and dispersion of pollutants. We want to explore if we can improve the simulation of clouds and precipitation in operational convective-scale model COSMO by estimating one spatially varying parameter in the planetary boundary layer with the operational data assimilation system KENDA. The results are presented in Chapter 4. To the best of our knowledge there are no studies that investigate the feasibility of augmented state parameter estimation using EnKF-type algorithms in a convection permitting model with quasi-operational settings.

### 1.2.3 Method related challenges

The least squares approach on which all operational data assimilation methods in weather forecasting are based, only considers first and second moment statistics. Using Bayes's theorem, it can be shown that if all errors are unbiased and Gaussian distributed, the analysis is the most likely state, provided covariance matrices  $\mathbf{P}^f$  and  $\mathbf{R}$  are correct. However, at convective scales this assumption is often violated. Precipitation is an obvious example of a non-Gaussian variable. Indeed, zero precipitation often occurs, whereas negative precipitation is absurd. As a result, the corresponding probability distribution is heavily skewed, and/or multi-modal. Ignoring this can result in poor and in some cases even unphysical initial conditions.

In contrast to the EnKF, particle filters (van Leeuwen, 2009) try to estimate full error statistics, making them suitable for non-Gaussian systems. Although the benefits of particle filters have been established, the computational demands are too severe for high dimensional systems (Snyder et al., 2008). Promising efforts like implicit sampling (Chorin et al., 2010; Weir et al., 2013) and localized particle filters (Poterjoy, 2016) are made to alleviate this problem, but performance on applications such as weather forecasting has yet to be demonstrated. For this reason, EnKF-based algorithms are still a popular branch of research. Indeed, despite their sub-optimality in non-Gaussian frameworks, they have proven capable of dealing with high dimensional systems, which is vital for many practical applications. Therefore, instead of modifying particle filters to make them suitable for high dimensional systems, we are interested in strategies to modify the EnKF to deal with non-Gaussian distributions.

Ensemble based smoothers (ES) do not assimilate data sequentially in time like the

EnKF. Instead they compute a global update by simultaneously assimilating all data available, thereby emphasizing the optimization of the full trajectory of the state rather than current forecasts. ES have demonstrated their worth in dealing with nonlinearity. The OSA-EnKF (Gharanti et al., 2015) and the IEnKS (Bocquet and Sakov, 2014, 2013) have shown great potential, especially for combined state and parameter estimation. However, smoothers require additional model evaluations, which is the dominating computational expense for convective scale applications. Furthermore, localization, which is necessary to deal with spurious correlations arising from undersampling, is not straight forward due to the temporal dimension included in the algorithm's cost function (Bocquet, 2016; Desroziers, 2016).

Many modifications of the EnKF have been suggested to relax the Gaussian assumption. Applying transformations on non-Gaussian variables to Gaussianize the distributions (Gaussian anamorphosis) has been attempted Bertino et al. (2003), but this approach can lead to instability (Bocquet and Sakov, 2013). Static anamorphosis is more efficient on distributions that are not too dynamical (Bocquet and Sakov, 2013) which is not the case for convective scale applications. Further, state properties such as mass conservation can be lost (Janjić et al., 2014) and transformed observation errors can lead to biased estimates (Bishop, 2016). The GIGG-EnKF (Bishop, 2016) on the other hand, directly deals with non-Gaussianity by allowing for each observed variable and corresponding observation the assumption of either a Gamma or Inverse Gamma prior distribution and/or likelihood. Results are promising for the estimation of observed state variables. However, similar to the EnKF, mapping the analysis from observation space to model space is done via simple correlations between the observed and unobserved quantities. Therefore this algorithm cannot be exploited for parameter estimation applications, since parameters are typically not represented in observation space.

In Chapter 3 we evaluate two recently developed ensemble algorithms that either explicitly incorporate constraints such as mass conservation and positivity of precipitation (QPEns, Janjić et al., 2014) or introduce higher order moments such as skewness (QF, Hodyss, 2011, 2012) to deal with non-Gaussian distributions corresponding to both the state and parameters. Although the importance of respecting physical conservation principles has long been recognized in numerical weather prediction modeling, it has only recently been shown that imposing them during data assimilation might have a positive impact as well (Janjić et al., 2014; Zeng et al., 2017). However, whether or not the impact would be positive for convective scale data assimilation is not obvious. It is clear though, that non-Gaussianity in convective scale data assimilation needs to be taken into account. Both QPEns and QF are able to handle non-Gaussian error statistics by using different approaches: QPEns through imposing constraints and QF with explicitly incorporating skewness.

## 1.3 Scientific Goal

We aim to investigate the feasibility of addressing model error by projecting uncertainties onto model parameters that are perturbed and estimated along with the state in a data assimilation framework.

### 1.3.1 Approach

We first investigate the feasibility of combined state and parameter estimation on an idealized test case, where we evaluate the performance of two data assimilation algorithms that were respectively designed to ensure certain dynamical properties of the initial conditions (QPEns) and account for skewed probability distributions of both the background and observations (QF). The model used for this study is the modified shallow water model (Würsch and Craig, 2014), which was designed for the purpose of testing data assimilation algorithms for convective-scale applications. In particular the model considers key properties of convection that form a challenge for data assimilation: conditional instability in the form of a regime switch, and a time lag between the onset of convection and its observation. This study is meant to quantify the relative importance of conserving physical quantities and considering higher order statistical moments in the calculation of the initial conditions for convective-scale applications.

In the second part of this work, the feasibility of augmented state parameter estimation is explored in a quasi-operational setup of weather prediction system COSMO-KENDA. Specifically, we estimate the roughness length parameter  $z_0$ , which is a measure of the surface roughness that appears in the planetary boundary layer parametrization. Both the parameter itself and the parametrization of surface fluxes which are directly influenced by  $z_0$  contain uncertainties (e.g., Rabin et al., 1990). Our goal is to project all model error related to surface fluxes onto  $z_0$  and estimate it along with the state variables to improve the forecast of clouds and precipitation.

The focus of the first part of this work lies on algorithmic advances (Chapter 3), whereas the second part focuses on the application (Chapter 4). Progress is made to enable integration of the algorithmic related findings to the operational setup (Chapter 5), but full applicability in operational systems remains a challenge for further research.

### 1.3.2 Challenges of parameter estimation

Many studies have shown the feasibility of joint state and parameter estimation ranging from high resolution (Tong and Xue, 2008) to climate models (Annan et al., 2005) in an idealized setup. The same studies identify important challenges that arise from the augmented state problem. A good overview of these challenges is given in Ruiz et al. (2013a).

### 1.3.2.1 Parameter uncertainty representation

A crucial matter is the representation of parameter uncertainty. If neither dynamics nor some form of inflation is applied to the parameters, the prediction system ebbs away the parameter perturbations, thereby prohibiting any further changes to the estimated parameter value. It is therefore necessary to explicitly model the uncertainty of the parameters in some way. Usually solutions are integrated in the data assimilation algorithm in the form of additive noise (Gelb, 1974) or multiplicative inflation (Aksoy et al., 2006). However, it must be stressed that this choice could significantly impact the parameter evolution, since by design the feedback between static model parameters and the state is one sided, thereby gravely restricting adjustments in the structure of the error covariance matrix of the parameters. Modeling the uncertainty of static model parameters is therefore somewhat artificial in the sense that the user (partially) prescribes the error variance and covariances corresponding to the parameter. This topic is discussed further in the Chapters 3 and 4.

### 1.3.2.2 Localization for parameters

A complication that arises for parameter estimation is the need for localization. Localization for dynamical variables is usually based on a correlation length scale, i.e. the mean influence radius of a dynamical state element. This approach could also make sense for spatially varying parameters (such as the roughness length  $z_0$ ), although the parameter's influence radius might be hard to determine and be too small if the parameter field is non-smooth. Indeed, a spatially correlated variable has a far larger influence radius than an uncorrelated one. Determining a localization strategy for parameters is therefore not straightforward, even if they are spatially varying. For global parameters, these classic localization techniques make no sense at all, since the parameter's influence is global by design. Aksoy et al. (2006) introduced artificial spatial structure for these parameters to circumvent the problem, but this could create biases. We therefore propose a new localization method for global parameters in Chapter 3. In Chapter 4 we discuss localization for the spatially varying roughness length parameter  $z_0$ .

## 1.4 Outline

In Chapter 2 we provide a mathematical background of the data assimilation algorithms used and how parameters are estimated. In Chapter 3 we evaluate the QPEns and the QF for the joint state and parameter problem on an idealized test case. A method for localization of global parameters is proposed and we discuss possibilities for representing the parameter uncertainty. In Chapter 4 we estimate the roughness length in near operation setup in COSMO-KENDA and investigate its effect on the representation of clouds by comparing model equivalents to reflectance observations obtained from the Spinning Enhanced Visible and InfraRed Imager (SEVIRI) instrument aboard the Meteosat Second Generation (MSG) satellite. Several dynamical models for the roughness length are tested and compared to a control run with fixed roughness length. The experiments are run for a

high impact summer period in 2016 which is divided into a strong and a weak synoptically forced week, to examine the sensitivity to large scale driving mechanisms. As the QPEs explored in Chapter 3 is computationally expensive, it is not applicable in the near operational setup at this stage of research. We therefore propose a new minimization algorithm in Chapter 5 that exploits certain properties of the problem, resulting in a reduction of the computation time for the QPEs. In addition we explore the possibility of training a neural network to reproduce the initial conditions obtained from the QPEs. In Chapter 6 we provide a conclusion and lay out suggestions for further research.

# Chapter 2

## Mathematical background of ensemble methods

Weather forecasts are based on numerical models that are designed to simulate the time evolution of an atmospheric state. To obtain a relevant weather forecast, the numerical model requires *initial conditions* that represent the current weather situation. The process of using measurements of relevant atmospheric quantities to produce these initial conditions is called *data assimilation*. In this thesis the focus lies on ensemble methods inspired by the Kalman Filter. In section 2.1 we introduce the Kalman Filter, the Ensemble Kalman Filter (EnKF) and the Ensemble Transform Kalman Filter (ETKF) in detail. We cover localization as way to deal with the consequences of a limited ensemble size in section 2.2 and explain how parameter are estimated with the augmented state approach in section 2.3.

### 2.1 Ensemble Kalman filters

The main strength of a Kalman Filter is its ability to take flow dependent uncertainties into account. Typically, a weather prediction model is imperfect due to approximations and processes that cannot be resolved on the chosen numerical grid. The measurements that are used to initialize the model contain uncertainties as well, such as instrument or representation errors. In this section we first introduce the Kalman Filter, which assumes a linear prediction model. Then we relax this assumption and describe the stochastic EnKF and the Ensemble Transform Kalman Filter (ETKF).

#### 2.1.1 Kalman Filter

Let us assume we have an imperfect linear model that describes the time evolution of state vector  $\mathbf{x} \in \mathbb{R}^n$ . Let us also assume we have independent, imperfect and indirect

measurements  $\mathbf{y} \in \mathbb{R}^m$  of  $\mathbf{x}$ . Usually  $m \ll n$ .

$$\mathbf{x}_t = \mathbf{M}_t \mathbf{x}_{t-1} + \mathbf{w}_t, \quad \mathbf{w}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_t) \quad (2.1.1a)$$

$$\mathbf{y}_t = \mathbf{H} \mathbf{x}_t + \mathbf{v}_t, \quad \mathbf{v}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_t) \quad (2.1.1b)$$

where  $M_t \in \mathbb{R}^{n \times n}$  and matrix  $\mathbf{H} \in \mathbb{R}^{m \times n}$  is a linear *observation operator* that maps a state vector from model space to observation space. The subscript  $t = 1, 2, \dots$  indicates the discrete times at which observations are available. The goal of the Kalman Filter is to estimate the time evolution of unknown state vector  $\mathbf{x}$ , using the model and observations described by (2.1.1).

Suppose we have an estimate  $\mathbf{x}_{t-1}^a$  of  $\mathbf{x}_{t-1}$  with error covariance matrix  $\mathbf{P}_{t-1}^a$  at time  $t-1$ . We use model (2.1.1a) to propagate this estimate and corresponding error covariance matrix to time  $t$

$$\mathbf{x}_t^f = \mathbf{M}_t \mathbf{x}_{t-1}^a \quad (2.1.2)$$

$$\mathbf{P}_t^f = \mathbf{M}_t \mathbf{P}_{t-1}^a \mathbf{M}_t^T + \mathbf{Q}_t. \quad (2.1.3)$$

Model prediction  $\mathbf{x}_t^f$  is called the *background* and serves as a *first guess* of the true state. Note that  $\mathbf{x}_t^f$  and observations  $\mathbf{y}_t$  serve as two independent sources that describe the true state vector  $\mathbf{x}_t$ , each with known error covariance matrix. It is therefore reasonable to take a weighted average of  $\mathbf{x}_t^f$  and  $\mathbf{y}_t$  to produce an updated estimate

$$\begin{aligned} \mathbf{x}_t^a &= \mathbf{K}_t \mathbf{y}_t + (\mathbf{I} - \mathbf{K}_t \mathbf{H}) \mathbf{x}_t^f \\ &= \mathbf{x}_t^f + \mathbf{K}_t (\mathbf{y}_t - \mathbf{H} \mathbf{x}_t^f). \end{aligned} \quad (2.1.4)$$

Note that the corresponding error covariance matrix is

$$\mathbf{P}_t^a = (\mathbf{I} - \mathbf{K}_t \mathbf{H}) \mathbf{P}_t^f (\mathbf{I} - \mathbf{K}_t \mathbf{H})^T + \mathbf{K}_t \mathbf{R}_t \mathbf{K}_t^T. \quad (2.1.5)$$

The best linear unbiased estimate is found by minimizing the mean square error, or, equivalently, the trace of  $\mathbf{P}_t^a$ , yielding

$$\mathbf{K}_t = \mathbf{P}_t^f \mathbf{H}^T (\mathbf{H} \mathbf{P}_t^f \mathbf{H}^T + \mathbf{R}_t)^{-1}. \quad (2.1.6)$$

This ‘‘optimal’’  $\mathbf{K}_t$  is called the *Kalman Gain*. Substituting the Kalman gain in (2.1.5) results in a simplified expression for the analysis error covariance matrix:

$$\mathbf{P}_t^a = (\mathbf{I} - \mathbf{K}_t \mathbf{H}) \mathbf{P}_t^f. \quad (2.1.7)$$

The updated estimate  $\mathbf{x}_t^a$  of  $\mathbf{x}_t$  is referred to as the *analysis* or simply the *initial conditions*.

Perhaps a more general derivation of the Kalman Filter can be developed using Bayes’ theorem. The analysis is the most likely state, given all previous and current observations.

Applying Bayes's theorem to our problem yields

$$p(\mathbf{x}_t = \mathbf{x} | \mathbf{y}_t, \dots, \mathbf{y}_0) = \frac{p(\mathbf{y}_t | \mathbf{x}_t = \mathbf{x}) p(\mathbf{x}_t = \mathbf{x} | \mathbf{y}_{t-1}, \dots, \mathbf{y}_0)}{p(\mathbf{y}_t | \mathbf{y}_{t-1}, \dots, \mathbf{y}_0)} \\ \propto \exp -\frac{1}{2} \left\{ (\mathbf{y}_t - \mathbf{H}\mathbf{x})^T \mathbf{R}_t^{-1} (\mathbf{y}_t - \mathbf{H}\mathbf{x}) + (\mathbf{x} - \mathbf{x}_t^f)^T \mathbf{P}_t^{f-1} (\mathbf{x} - \mathbf{x}_t^f) \right\}, \quad (2.1.8)$$

where  $p(\mathbf{x}_t | \mathbf{y}_t, \dots, \mathbf{y}_0)$  is the *posterior* distribution,  $p(\mathbf{y}_t | \mathbf{x}_t)$  the *observation likelihood* and  $p(\mathbf{x}_t | \mathbf{y}_{t-1}, \dots, \mathbf{y}_0)$  the *prior* distribution. For simplicity we assume for now that  $\mathbf{P}_t^f$  and  $\mathbf{R}_t$  are invertible. Maximizing posterior distribution (2.1.8) is equivalent to minimizing cost function

$$J(\mathbf{x}) = \frac{1}{2} \left\{ (\mathbf{y}_t - \mathbf{H}\mathbf{x})^T \mathbf{R}_t^{-1} (\mathbf{y}_t - \mathbf{H}\mathbf{x}) + (\mathbf{x} - \mathbf{x}_t^f)^T \mathbf{P}_t^{f-1} (\mathbf{x} - \mathbf{x}_t^f) \right\} \quad (2.1.9)$$

resulting in equations (2.1.4) and (2.1.6). The algorithm is summarized in Algorithm 2.1.1.

---

#### Algorithm 2.1.1 Kalman Filter

---

**Require:**

Initial state  $(\mathbf{x}_0^a, \mathbf{P}_0^a)$

Model error statistics  $\{\mathbf{Q}_t : t = 1, 2, \dots\}$

Observations and corresponding error statistics  $\{(\mathbf{y}_t, \mathbf{R}_t) : t = 1, 2, \dots\}$

**for**  $t = 1, 2, \dots$  **do**

Prediction:

$$\mathbf{x}_t^f = \mathbf{M}_t \mathbf{x}_{t-1}^a$$

$$\mathbf{P}_t^f = \mathbf{M}_t \mathbf{P}_{t-1}^a \mathbf{M}_t^T + \mathbf{Q}_t$$

Update:

$$\mathbf{K} = \mathbf{P}_t^f \mathbf{H}^T (\mathbf{H} \mathbf{P}_t^f \mathbf{H}^T + \mathbf{R}_t)^{-1}$$

$$\mathbf{x}_t^a = \mathbf{x}_t^f + \mathbf{K} (\mathbf{y}_t - \mathbf{H} \mathbf{x}_t^f)$$

$$\mathbf{P}_t^a = (\mathbf{I} - \mathbf{K} \mathbf{H}) \mathbf{P}_t^f$$

**end for**

---

### 2.1.2 Stochastic EnKF

The Kalman Filter was designed for relatively small linear problems. Unfortunately, weather prediction models are rarely linear. We therefore relax the linear assumption and assume

$$\mathbf{x}_t = \mathbf{f}(\mathbf{x}_{t-1}) + \mathbf{w}_t, \quad \mathbf{w}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_t) \quad (2.1.10a)$$

$$\mathbf{y}_t = \mathbf{h}(\mathbf{x}_t) + \mathbf{v}_t, \quad \mathbf{v}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_t) \quad (2.1.10b)$$

where  $\mathbf{f}$  and  $\mathbf{h}$  are nonlinear functions. A straightforward way to modify the Kalman Filter for (2.3.1) is using tangent linear models  $\mathbf{M}_t = \frac{\partial \mathbf{f}}{\partial \mathbf{x}}(\mathbf{x}_t)$  and  $\mathbf{H}_t = \frac{\partial \mathbf{h}}{\partial \mathbf{x}}(\mathbf{x}_t)$  in (2.1.3), (2.1.4) and (2.1.6). The resulting algorithm is called the extended Kalman Filter (EKF). However, as the size of the state increases, computing the error covariance matrix  $\mathbf{P}_t^f$  using (2.1.3) becomes infeasible. Also, linearization of  $\mathbf{f}$  and  $\mathbf{h}$  might be a poor approximation as nonlinearities can be severe. To avoid direct computation of  $\mathbf{P}^f$  and linearization of  $\mathbf{f}$  and  $\mathbf{h}$ , the EnKF uses an ensemble to estimate the the background error statistics.

Let ensemble  $\{\mathbf{x}_{t-1,i}^a : i = 1, 2, \dots, N_{ens}\}$  with corresponding ensemble mean  $\bar{\mathbf{x}}_{t-1}^a = \frac{1}{N_{ens}} \sum_{i=1}^{N_{ens}} \mathbf{x}_{t-1,i}^a$  consist of  $N_{ens}$  independent samples of the posterior distribution at time  $t-1$ . Then, at time  $t$ ,  $\{\mathbf{x}_{t,i}^f = \mathbf{f}(\mathbf{x}_{t-1,i}^a) + \mathbf{w}_{t,i} : i = 1, 2, \dots, N_{ens}\}$  with ensemble mean  $\bar{\mathbf{x}}_t^f = \frac{1}{N_{ens}} \sum_{i=1}^{N_{ens}} \mathbf{x}_{t,i}^f$  and  $\{\mathbf{y}_{t,i} = \mathbf{y}_t + \mathbf{v}_{t,i} : i = 1, 2, \dots, N_{ens}\}$  are independent samples of the prior distribution and the observation likelihood respectively. From the prior ensemble, the prior sample error covariance matrix can be computed, i.e.

$$\mathbf{P}_t^f := \frac{1}{N_{ens} - 1} \mathbf{X}_t^f \mathbf{X}_t^{fT},$$

where  $\mathbf{X}_t^f \in \mathbb{R}^{n \times N_{ens}}$  is the matrix whose  $i^{th}$  column is  $\mathbf{x}_{t,i}^f - \bar{\mathbf{x}}_t^f$ . However, as seen in Algorithm 2.1.1, the relevant terms are  $\mathbf{P}_t^f \mathbf{H}_t^T$  and  $\mathbf{H}_t \mathbf{P}_t^f \mathbf{H}_t^T$ . With nonlinear operator  $\mathbf{h}$ , these terms become  $\frac{1}{N_{ens}-1} \mathbf{X}_t^f \mathbf{Y}_t^{fT}$  and  $\frac{1}{N_{ens}-1} \mathbf{Y}_t^f \mathbf{Y}_t^{fT}$  respectively, where  $\mathbf{Y}_t^f \in \mathbb{R}^{n \times N_{ens}}$  is the matrix whose  $i^{th}$  column is  $\mathbf{h}(\mathbf{x}_{t,i}^f) - \mathbf{h}(\bar{\mathbf{x}}_t^f)$  with  $\mathbf{h}(\bar{\mathbf{x}}_t^f) = \frac{1}{N_{ens}} \sum_{i=1}^{N_{ens}} \mathbf{h}(\mathbf{x}_{t,i}^f)$ . Ensemble  $\{\mathbf{x}_{t,i}^a = \mathbf{x}_{t,i}^f + \mathbf{K}_t (\mathbf{y}_{t,i} - \mathbf{h}(\mathbf{x}_{t,i}^f)) : i = 1, 2, \dots, N_{ens}\}$ , where  $\mathbf{K}_t = \mathbf{X}_t^f \mathbf{Y}_t^{fT} (\mathbf{Y}_t^f \mathbf{Y}_t^{fT} + (N_{ens} - 1) \mathbf{R}_t)^{-1}$  then represents the posterior distribution at time  $t$ . It should be stressed that a single ensemble member is not necessarily a meaningful estimate of the true state. Rather, the ensemble as a whole represents the statistical properties of the true state. For a meaningful estimate one could compute the ensemble mean, or perform a *deterministic run* parallel to the ensemble. This deterministic run is not perturbed during the prediction step and is updated using unperturbed observations. See Algorithm 2.1.2 for a concise description of the EnKF.

### 2.1.3 ETKF

A downside of the stochastic EnKF is that by representing the observation likelihood with an ensemble of perturbed observations, an additional source of sampling error is introduced. As a result, equation (2.1.7) is not satisfied exactly. The Extended Transform Kalman Filter (ETKF) avoids this, by explicitly imposing (2.1.7) on the analysis ensemble:

$$\mathbf{X}_t^a \mathbf{X}_t^{aT} = (\mathbf{I} - \mathbf{K}_t \mathbf{H}_t) \mathbf{X}_t^f \mathbf{X}_t^{fT} \quad (2.1.11)$$

where  $\mathbf{X}_t^a \in \mathbb{R}^{n \times k}$  is the matrix whose  $i^{th}$  column is  $\mathbf{x}_{t,i}^a - \bar{\mathbf{x}}_t^a$ . The posterior ensemble mean  $\bar{\mathbf{x}}_t^a$  is computed analogous to the EnKF:

$$\bar{\mathbf{x}}_t^a = \bar{\mathbf{x}}_t^f + \mathbf{K}_t \left( \mathbf{y}_t - \overline{\mathbf{h}(\mathbf{x}_t^f)} \right) \quad (2.1.12)$$

---

**Algorithm 2.1.2** EnKF

---

**Require:**Initial ensemble  $\{\mathbf{x}_{0,i}^a : i = 1, 2, \dots, k\}$ Initial deterministic state  $\mathbf{x}_0^a$ Model error statistics  $\{\mathbf{Q}_t : t = 1, 2, \dots\}$ Observations and corresponding error statistics  $\{(\mathbf{y}_t, \mathbf{R}_t) : t = 1, 2, \dots\}$ **for**  $t = 1, 2, \dots$  **do**

Prediction:

$$\mathbf{x}_{t,i}^f = \mathbf{f}(\mathbf{x}_{t-1,i}^a) + \mathbf{w}_{t,i}, \quad \mathbf{w}_{t,i} \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_t)$$

$$\mathbf{x}_t^f = \mathbf{f}(\mathbf{x}_{t-1}^a)$$

$$\mathbf{X}_t^f = \begin{pmatrix} \mathbf{x}_{t,1}^f & \mathbf{x}_{t,2}^f & \cdots & \mathbf{x}_{t,N_{ens}}^f \end{pmatrix}$$

$$\mathbf{Y}_t^f = \begin{pmatrix} \mathbf{h}(\mathbf{x}_{t,1}^f) & \mathbf{h}(\mathbf{x}_{t,2}^f) & \cdots & \mathbf{h}(\mathbf{x}_{t,N_{ens}}^f) \end{pmatrix}$$

Update:

$$\mathbf{y}_{t,i} = \mathbf{y}_t + \epsilon_{t,i}, \quad \epsilon_{t,i} \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_t)$$

$$\mathbf{K} = \mathbf{X}_t^f \mathbf{Y}_t^{fT} \left( \mathbf{Y}_t^f \mathbf{Y}_t^{fT} + (N_{ens} - 1) \mathbf{R}_t \right)^{-1}$$

$$\mathbf{x}_{t,i}^a = \mathbf{x}_{t,i}^f + \mathbf{K} \left( \mathbf{y}_{t,i} - \mathbf{h}(\mathbf{x}_{t,i}^f) \right)$$

$$\mathbf{x}_t^a = \mathbf{x}_t^f + \mathbf{K} \left( \mathbf{y}_t - \mathbf{h}(\mathbf{x}_t^f) \right)$$

**end for**

---

To determine the posterior ensemble perturbations we substitute ansatz  $\mathbf{X}_t^a = \mathbf{X}_t^f \mathbf{W}_t^a$  with  $\mathbf{W}_t^a \in \mathbb{R}^{k \times n}$  into (2.1.11):

$$\mathbf{X}_t^f \mathbf{W}_t^a \mathbf{W}_t^{aT} \mathbf{X}_t^{fT} = (\mathbf{I} - \mathbf{K}_t \mathbf{H}_t) \mathbf{X}_t^f \mathbf{X}_t^{fT} \quad (2.1.13a)$$

$$= \mathbf{X}_t^f \left( \mathbf{I} - (k-1)^{-1} \mathbf{Y}_t^{fT} \left( (k-1)^{-1} \mathbf{Y}_t^f \mathbf{Y}_t^{fT} + \mathbf{R}_t \right)^{-1} \mathbf{Y}_t^f \right) \mathbf{X}_t^{fT} \quad (2.1.13b)$$

$$= \mathbf{X}_t^f (k-1) \left( (k-1) \mathbf{I} + \mathbf{Y}_t^{fT} \mathbf{R}_t^{-1} \mathbf{Y}_t^f \right)^{-1} \mathbf{X}_t^{fT}. \quad (2.1.13c)$$

leading to the following posterior ensemble perturbations

$$\mathbf{X}_t^a = (k-1)^{\frac{1}{2}} \mathbf{X}_t^f \left( (k-1) \mathbf{I} + \mathbf{Y}_t^{fT} \mathbf{R}_t^{-1} \mathbf{Y}_t^f \right)^{-\frac{1}{2}}. \quad (2.1.14)$$

It can be shown that the choice of square root formulation (2.1.14) provides an unbiased posterior ensemble, in the sense that the mean of the ensemble perturbations is zero, i.e.  $\mathbf{X}_t^a \mathbf{e} = 0$  where  $\mathbf{e}$  is the vector with elements equal to 1 of appropriate size. The ETKF is outlined in Algorithm 2.1.3.

## 2.2 Localization

When the Gaussian assumption is valid and  $N_{ens}$  is large enough, the EnKF and ETKF perform very well. However, in practice we can often not afford an ensemble size large enough to ensure filter convergence. The solution is to use known physical properties of the model to post process the sample error covariances computed by the undersized ensemble. For example, we know that the covariance of variables at distant grid points is very small. In fact, on average, the true covariance resembles a Gaussian function of the variables' spatial distance. This knowledge can be used to suppress severe spurious sample correlations. A natural way is to Schur multiply the sample error covariance matrix  $\mathbf{P}_t^f$  with a stationary *a priori* chosen localization matrix  $\mathbf{C}$ . This technique is called *covariance localization* (see Figure 1.1.2). However, as can be seen in Algorithm 2.1.2 and Algorithm 2.1.3, it is not necessary to compute the full error covariance matrix  $\mathbf{P}_t^f$ . In addition, if the observation operator is nonlinear, applying covariance localization on  $\frac{1}{N_{ens}-1} \mathbf{X}_t^f \mathbf{Y}_t^{fT}$  and  $\frac{1}{N_{ens}-1} \mathbf{Y}_t^f \mathbf{Y}_t^{fT}$  is not straightforward. A popular alternative is therefore *domain localization*, where one computes the analysis for each grid point separately, using only a subset of observations. The advantages of this technique are mostly computational, as the large problem is reduced to many smaller ones, which can be solved in parallel.

## 2.3 Augmented state parameter estimation

In weather prediction applications, observations are spatially sparse with respect to the model grid. Yet, via the covariances given in  $\mathbf{P}_t^f$  they can provide useful information about

---

**Algorithm 2.1.3** ETKF

---

**Require:**Initial ensemble  $\{\mathbf{x}_{0,i}^a : i = 1, 2, \dots, k\}$ Initial deterministic state  $\mathbf{x}_0^a$ Model error statistics  $\{\mathbf{Q}_t : t = 1, 2, \dots\}$ Observations and corresponding error statistics  $\{(\mathbf{y}_t, \mathbf{R}_t) : t = 1, 2, \dots\}$ **for**  $t = 1, 2, \dots$  **do**

Prediction:

$$\mathbf{x}_{t,i}^f = \mathbf{f}(\mathbf{x}_{t-1,i}^a) + \mathbf{w}_{t,i}, \quad \mathbf{w}_{t,i} \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_t)$$

$$\mathbf{x}_t^f = \mathbf{f}(\mathbf{x}_{t-1}^a)$$

$$\mathbf{X}_t^f = \left( \mathbf{x}_{t,1}^f \mathbf{x}_{t,2}^f \cdots \mathbf{x}_{t,N_{ens}}^f \right)$$

$$\mathbf{Y}_t^f = \left( \mathbf{h}(\mathbf{x}_{t,1}^f) \mathbf{h}(\mathbf{x}_{t,2}^f) \cdots \mathbf{h}(\mathbf{x}_{t,N_{ens}}^f) \right)$$

Update:

$$\mathbf{K} = \mathbf{X}_t^f \mathbf{Y}_t^{fT} \left( \mathbf{Y}_t^f \mathbf{Y}_t^{fT} + (N_{ens} - 1) \mathbf{R}_t \right)^{-1}$$

$$\bar{\mathbf{x}}_t^a = \bar{\mathbf{x}}_t^f + \mathbf{K} \left( \mathbf{y}_t - \overline{\mathbf{h}(\mathbf{x}_t^f)} \right)$$

$$\mathbf{X}_t^a = (k - 1)^{\frac{1}{2}} \mathbf{X}_t^f \left( (k - 1) \mathbf{I} + \mathbf{Y}_t^{fT} \mathbf{R}_t^{-1} \mathbf{Y}_t^f \right)^{-\frac{1}{2}}$$

$$\mathbf{x}_{t,i}^a = \bar{\mathbf{x}}_t^a + \mathbf{X}_t^a(i)$$

$$\mathbf{x}_t^a = \bar{\mathbf{x}}_t^a + \mathbf{K} \left( \mathbf{y}_t - \mathbf{h}(\mathbf{x}_t^f) \right)$$

**end for**

---

many state elements that are not directly observed. They can do the same for uncertain model parameters.

Let  $\theta \in \mathbb{R}^p$  be a model parameter that we wish to estimate, which is not represented in observation space. Then

$$\tilde{\mathbf{x}}_t := \begin{pmatrix} \mathbf{x}_t \\ \theta_t \end{pmatrix} = \begin{pmatrix} \mathbf{f}(\mathbf{x}_{t-1}, \theta_{t-1}) \\ \theta_{t-1} \end{pmatrix} + \tilde{\mathbf{w}}_t := \tilde{\mathbf{f}}(\tilde{\mathbf{x}}_{t-1}) + \tilde{\mathbf{w}}_t, \quad \mathbf{w}_t \sim \mathcal{N}(\mathbf{0}, \tilde{\mathbf{Q}}_t) \quad (2.3.1a)$$

$$\mathbf{y}_t = \mathbf{h}(\mathbf{x}_t) + \mathbf{v}_t, \quad \mathbf{v}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_t) \quad (2.3.1b)$$

where  $\tilde{\mathbf{x}} \in R^{n+p}$  is the augmented state and  $\tilde{\mathbf{Q}}_t \in R^{(n+p) \times (n+p)}$  the corresponding model error covariance. When provided with an initial ensemble of the augmented state and  $\{\tilde{\mathbf{Q}}_t : t = 1, 2, \dots\}$ , Algorithm 2.1.2 or Algorithm 2.1.3 can be applied to the augmented state, thereby yielding an online estimate of the model parameter  $\theta$  as well as the model state  $\mathbf{x}$ . The augmented error covariance matrix  $\tilde{\mathbf{P}}_t^f$  is of the form

$$\tilde{\mathbf{P}}_t^f := \begin{pmatrix} \mathbf{P}_t^f & \mathbf{P}_{[\mathbf{x}, \theta]t}^f \\ \mathbf{P}_{[\mathbf{x}, \theta]t}^f & \mathbf{P}_{\theta t} \end{pmatrix} \quad (2.3.2)$$

where  $\mathbf{P}_{[\mathbf{x}, \theta]t}^f = \text{Cov}[\mathbf{x}_t^f, \theta_t^f]$  and  $\mathbf{P}_{\theta t}$  is the error covariance matrix of the parameters. Note that  $\mathbf{P}_{\theta t}$  does not have a subscript because  $\mathbf{P}_{\theta(t-1)}^a = \mathbf{P}_{\theta t}^f$ . In this thesis we do not explicitly consider correlations between parameter errors and the remaining model error, i.e.

$$\tilde{\mathbf{Q}}_t = \begin{pmatrix} \mathbf{Q}_t & \mathbf{0} \\ \mathbf{0} & \mathbf{Q}_{\theta t} \end{pmatrix} \quad (2.3.3)$$

where  $\mathbf{Q}_{\theta t} \in \mathbb{R}^{p \times p}$  will be discussed in detail in later chapters.

In the remaining of this thesis, we refrain from writing subscript  $t$  to indicate time, except when helpful to avoid confusion. The stochastic EnKF shall further be referred to simply as the EnKF.

# Chapter 3

## Evaluating data assimilation algorithms in an idealized setup

This Chapter appears in Quarterly Journal of the Royal Meteorological Society as Ruckstuhl and Janjić (2018). We reprint it here with some minor modifications.

We evaluate two recently developed EnKF based algorithms that either explicitly incorporate constraints such as mass conservation and positivity of precipitation (QPEnS), or introduce higher order moments (QF) on the joint state and parameter estimation problem. We compare their results to the localized EnKF on a common idealized test case. The test case uses perfect model experiments with the one dimensional modified shallow water model that was designed to mimic important properties of convection. These aspects include an acute regime switch when convection is triggered (conditional instability) and a significant time lag between the onset of convection and its observation. The comparison between the QF, the QPEnS and the EnKF should provide insight in the relative impact of ignoring conservation laws and physical bounds, and the skewness of relevant distributions for convective scale applications. This will aid in establishing a course for further research, i.e. either combining the algorithms or putting more effort in one direction instead. The algorithms are introduced in more detail in section 3.1

As pointed out in Aksoy et al. (2006), localization in both state and parameter space is needed when the posterior distribution is undersampled. To deal with the typically lower dimension of the parameters, the spatial updating technique was introduced. In section 3.2 we elaborate on this technique, and propose an alternative formulation to reduce computational costs. In the second part of this section we describe how we deal with severe underdispersion in parameter space, which is a common phenomenon for estimation of static parameters (see for example Simon et al., 2015).

In section 3.3 we present the experimental setup, where the modified shallow water model is explained and the settings for both the model and the data assimilation algorithms are specified. This is followed by a discussion of the results in section 3.4, in which we first focus on the sensitivity of the algorithms to the localization radius, the observation coverage and frequency without parameter uncertainty. Then, model error due to parameter uncertainty is introduced and it is investigated how parameter estimation affects the

estimate of the dynamical state for the different algorithms. Finally, a summary is provided and some conclusions and further research questions are stated in section 3.5.

### 3.1 Data assimilation algorithms

The data assimilation algorithms that are discussed in this chapter are modified versions of the EnKF described in Algorithm 2.1.2 (Evensen, 1994; Houtekamer and Mitchell, 1998; Evensen, 2003). With the goal to clearly indicate the respective modifications, we introduce the cost function for each ensemble member  $i$ ,  $i = 1, 2, \dots, N_{ens}$

$$J(\delta \mathbf{x}_i) := \delta \mathbf{x}_i^T \mathbf{P}^f \delta \mathbf{x}_i + [\mathbf{v}_i - \mathbf{H} \delta \mathbf{x}_i]^T \mathbf{R}^{-1} [\mathbf{v}_i - \mathbf{H} \delta \mathbf{x}_i] \quad (3.1.1)$$

where  $N_{ens}$  is the ensemble size. At the minimum of cost function  $J$ ,  $\delta \mathbf{x}_i = \mathbf{x}_i^a - \mathbf{x}_i^f$  is the analysis increment of size  $n \times 1$  that is used to calculate analysis ensemble member  $\mathbf{x}_i^a$  based on background ensemble member  $\mathbf{x}_i^f$  (all of size  $n \times 1$ ). The matrix

$$\mathbf{P}^f = \frac{1}{N_{ens} - 1} \sum_{i=1}^{N_{ens}} [\mathbf{x}_i^f - \bar{\mathbf{x}}^f] [\mathbf{x}_i^f - \bar{\mathbf{x}}^f]^T := \frac{1}{N_{ens} - 1} \sum_{i=1}^{N_{ens}} \epsilon_i^f (\epsilon_i^f)^T$$

is the background error covariance matrix with  $\bar{\mathbf{x}}^f$  representing the ensemble mean,  $\mathbf{R}$  the observation error covariance matrix and  $\mathbf{H}$  an observation operator, which for simplicity we assume is linear, of size  $p \times n$ . The *innovation vector*  $\mathbf{v}_i = \mathbf{v} - \epsilon_i$ , where  $\mathbf{v} = \mathbf{y} - \bar{\epsilon}^o - \mathbf{H} \bar{\mathbf{x}}^f$  and  $\epsilon_i = \epsilon_i^o - \bar{\epsilon}^o + \mathbf{H} \epsilon_i^f$  is subject to perturbed observations  $\epsilon_i^o$ , whose sample has bias  $\bar{\epsilon}^o$ . For the EnKF, the analysis increments  $\delta \mathbf{x}_i$  are found by solving

$$\min_{\delta \mathbf{x}_i} J(\delta \mathbf{x}_i) \quad \text{for } i = 1, 2, \dots, N_{ens} \quad (3.1.2)$$

or, equivalently,

$$\delta \mathbf{x}_i = \mathbf{P}^f \mathbf{H}^T (\mathbf{H} \mathbf{P}^f \mathbf{H}^T + \mathbf{R})^{-1} \mathbf{v}_i. \quad (3.1.3)$$

Formulations (3.1.2) and (3.1.3) of the EnKF allow us to clearly identify the differences among the algorithms discussed in this chapter. Note however that since  $\mathbf{P}^f$  is not generally invertible, our abuse of notation in (3.1.1) strictly speaking means that a projection into the lower dimensional space would need to be done first before minimization.

#### 3.1.1 Quadratic Programming Ensemble

A side effect of using data assimilation with the EnKF is the generation of a possibly unphysical analysis state (Kivman, 2003; Pan and Wood, 2006; Simon and Bertino, 2009; Janjić et al., 2014; Zeng and Janjić, 2016). In such cases, the simple solution is to set the unphysical values to the nearest physical one. For example, in convective scale applications negative precipitation values could be set to zero. As a consequence, total mass is altered and bias is introduced (see Janjić et al., 2014, for a simple example). The QPEns algorithm,

suggested in Janjić et al. (2014), addresses this obstacle by applying physical constraints to minimization problem (3.1.2):

$$\begin{aligned} \min_{\delta \mathbf{x}_i} \quad & J(\delta \mathbf{x}_i) \\ \text{subject to} \quad & c_j(\delta \mathbf{x}_i) \leq 0, \quad j \in \{1, 2, \dots, m_1\} \\ & g_k(\delta \mathbf{x}_i) = 0, \quad k \in \{1, 2, \dots, m_2\} \end{aligned} \quad (3.1.4)$$

for  $i = 1, 2, \dots, N_{ens}$ , where  $m_1$  and  $m_2$  indicate the number of inequality and equality constraints respectively. Here,  $c_j$  and  $g_k$  will depend on the physical constraints that one wishes to impose. In Janjić et al. (2014), the effect of imposing positivity and mass constraints was investigated on simple test cases.

Since a full minimization problem has to be solved for each ensemble member, the computational cost of the QPEns heavily depends on the number and the nature of the constraints. For example, inequality constraints are algorithmically more challenging than equality constraints. Sophisticated minimization algorithms, such as interior point methods and Sequential Quadratic Programming (SQP), handle inequality constraints either by incorporating them in the cost function with a penalty term, or by solving a sequence of equality constrained subproblems instead. The non-linearity level of the constraints is an important factor as well. The higher the non-linearity, the harder the problem.

In this study we consider linear constraints only, such that problem (3.1.4) can be solved using quadratic programming (QP). For further details on QP we refer to Gill (1981), where various constrained minimization algorithms are thoroughly discussed. For the extension on QPEns to nonlinear equality constraints we refer to Zeng et al. (2017).

### 3.1.2 Quadratic Filter

The most restricting assumption that is needed to derive the EnKF is that all relevant distributions are Gaussian. At convective scales this assumption is typically not valid. In fact, the very nature of convection is highly non-linear and spatially irregular, which leads to skewed (and even multi-modal) distributions. The QF alleviates this issue by calculating third and fourth moments from the ensemble, in addition to the first two. The algorithm is based on an expansion which represents the analysis increment in powers of the innovation. Truncation of this expansion at the quadratic term, along with ignoring higher order terms between 3 or more variables, yields the following formulation of the QF:

$$\delta \mathbf{x}_i = \left( \tilde{\mathbf{P}} \tilde{\mathbf{H}}^T \right) \left( \tilde{\mathbf{H}} \tilde{\mathbf{P}} \tilde{\mathbf{H}}^T + \tilde{\mathbf{R}} \right)^{-1} \tilde{\mathbf{v}}_i \Big|_n \quad (3.1.5)$$

where  $\tilde{\mathbf{P}} = \begin{bmatrix} \mathbf{P}^f & \mathbf{P}_3 \\ \mathbf{P}_3^T & \mathbf{P}_4 \end{bmatrix}$ ,  $\tilde{\mathbf{R}} = \begin{bmatrix} \mathbf{R} & \mathbf{R}_3 \\ \mathbf{R}_3^T & \mathbf{R}_4 \end{bmatrix}$ ,  $\tilde{\mathbf{H}} = \begin{bmatrix} \mathbf{H} & 0 \\ 0 & \mathbf{H} \end{bmatrix}$  and  $\tilde{\mathbf{v}}_i = [\mathbf{v}_i^T, (\mathbf{v} \odot \mathbf{v})^T - (\epsilon_i \odot \epsilon_i)^T]^T$ , with  $\odot$  representing element-wise (Schur) multiplication. The subscript  $n$  in (3.1.5) indicates that only the first  $n$  rows of the concerning vector are needed, where  $n$  is the

dimension of  $\delta\mathbf{x}_i$ . The matrices  $\mathbf{P}_3$  ( $\mathbf{R}_3$ ) and  $\mathbf{P}_4$  ( $\mathbf{R}_4$ ) hold information about third and fourth moments respectively and are obtained by  $\tilde{\mathbf{P}} = \tilde{\mathbf{Z}}\tilde{\mathbf{Z}}^T$ , where

$$\tilde{\mathbf{Z}} = \frac{1}{\sqrt{N-1}} \begin{bmatrix} \epsilon_1^f & & & \epsilon_N^f \\ \epsilon_1^f \odot \epsilon_1^f - \text{diag}(\mathbf{P}^f) & \dots & & \epsilon_N^f \odot \epsilon_N^f - \text{diag}(\mathbf{P}^f) \end{bmatrix}$$

and

$$\tilde{\mathbf{R}} = \begin{bmatrix} \mathbf{R} & \mathbf{T} \\ \mathbf{T}^T & \mathbf{F} - \mathbf{r}_d \mathbf{r}_d^T + 4\mathbf{R} \odot (\mathbf{H}\mathbf{P}^f\mathbf{H}^T) \end{bmatrix}.$$

Here  $\mathbf{r}_d = \text{diag}(\mathbf{R})$ ,  $\mathbf{T}$  and  $\mathbf{F}$  are the third and fourth moment matrices of the observation likelihood respectively. Note that for computational reasons, only higher order moments between at most two variables are taken into account. This ensures that the dimension of  $\mathbf{P}_3$  ( $\mathbf{R}_3$ ) and  $\mathbf{P}_4$  ( $\mathbf{R}_4$ ) equals that of  $\mathbf{P}^f$  ( $\mathbf{R}$ ). In particular,  $\mathbf{P}_{3_{ij}}$  ( $\mathbf{R}_{3_{ij}}$ ) and  $\mathbf{P}_{4_{ij}}$  ( $\mathbf{R}_{4_{ij}}$ ) correspond to the same variables as  $\mathbf{P}_{ij}^f$  ( $\mathbf{R}_{ij}$ ), thereby avoiding complications with respect to localization.

Additional computational costs with respect to the EnKF arise mainly from an augmented observation space. In cases where running the model forecast dominates the computation time, the additional costs of using the QF may be very reasonable. Also, in contrast to the QPEs, the QF is suitable for square root formulations such as the ETKF (Algorithm 2.1.3). However, like the EnKF, physical bounds are not necessarily respected (Posselt et al., 2014), as is the case for the QPEs.

A full derivation of the algorithm along with a discussion of the effect of skewed distributions in EnKF context can be acquired in Hodyss (2011) and Hodyss (2012).

## 3.2 Augmented state parameter estimation

There are two difficulties related to augmented state parameter estimation in ensemble based algorithms. Localization in parameter space  $\theta$  is not straightforward since it is not obvious how to best relate global model parameter values to the analysis grid points. This complication can be circumvented using the spatial updating technique introduced by Aksoy et al. (2006). Here, we suggest an alternative to it. Secondly, augmented state data assimilation algorithms suffer from underdispersion in parameter space. Our approach to address this problem is discussed in section 3.2.2.

### 3.2.1 Localization in parameter space

To illustrate our approach, we consider the case in which one state field and one global scalar parameter are estimated using the EnKF with covariance localization, i.e.  $\mathbf{P}_{Loc} = \mathbf{C} \odot \mathbf{P}^f$ , where  $\mathbf{C}$  is a localization matrix and  $\mathbf{P}_{Loc}$  is the localized error covariance matrix. Global parameters become an issue when localization is required. Localizing only the state and not the parameters yields indefinite matrices, which can cause any algorithm to fail.

Let

$$\mathbf{C} = \begin{bmatrix} \mathbf{C}_{(\mathbf{x},\mathbf{x})} & \mathbf{C}_{(\theta,\mathbf{x})}^T \\ \mathbf{C}_{(\theta,\mathbf{x})} & \mathbf{C}_{(\theta,\theta)} \end{bmatrix}$$

be the matrix we localize the augmented state with, where  $\mathbf{C}_{(\mathbf{x},\mathbf{x})}$  is a correlation matrix of size  $n \times n$ ,  $\mathbf{C}_{(\theta,\mathbf{x})}$  is of size  $1 \times n$  and  $\mathbf{C}_{(\theta,\theta)} = 1$  is a scalar. By Schur complement, the matrix  $\mathbf{C}$  is positive semidefinite if and only if  $\mathbf{C}_{(\mathbf{x},\mathbf{x})} - \mathbf{C}_{(\theta,\mathbf{x})}^T \mathbf{C}_{(\theta,\mathbf{x})}$  is. If no localization is applied to the parameters, then  $\mathbf{C}_{(\theta,\mathbf{x})} = \mathbf{e}^T$ , where  $\mathbf{e}$  is the vector with elements equal to 1 of appropriate size. However,  $\mathbf{e}^T \mathbf{C}_{(\mathbf{x},\mathbf{x})} \mathbf{e} - \mathbf{e}^T \mathbf{e} \mathbf{e}^T \mathbf{e} < n^2 - n^2 = 0$ , and therefore the localization matrix  $\mathbf{C}$  is not positive semidefinite when only the state is localized. In addition, localization is typically applied to suppress spurious correlations caused by undersampling, which is an issue that should also be dealt with in parameter space.

To localize global parameters in the same way as the state, Aksoy et al. (2006) proposed to map global parameters to homogeneous fields of appropriate dimension  $l$  (depending on the dimension one wishes to apply parameter localization to). These “dummy” fields serve to create an artificial spatial dependency of parameters, thereby allowing similar localization techniques in both state and parameter space. After the analysis increment  $[\delta \mathbf{x}_i^T, \delta \theta_i^T]^T$  of size  $(n + l) \times 1$  is computed, the now spatially inhomogeneous parameter field  $\delta \theta_i$  is mapped back to a scalar by performing the spatial average for each ensemble member  $i$ . In mathematical terms, the parameter for each ensemble member is updated according to

$$\delta \theta_i = \frac{1}{l} \mathbf{e}^T \mathbf{C}_{(\theta\mathbf{e},\mathbf{x})} \odot \mathbf{P}_{(\theta\mathbf{e},\mathbf{x})}^f \mathbf{H}^T \mathbf{S}^{-1} \mathbf{v}_i, \quad (3.2.1)$$

where  $\mathbf{P}_{(*,**)}^f$  contains the covariances between  $*$  and  $**$ , and

$$\mathbf{S} = \mathbf{H} \mathbf{C}_{(\mathbf{x},\mathbf{x})} \odot \mathbf{P}_{(\mathbf{x},\mathbf{x})}^f \mathbf{H}^T + \mathbf{R}.$$

Note that  $\mathbf{P}_{(\theta\mathbf{e},\mathbf{x})}^f$  of size  $l \times n$  consists of  $l$  identical rows equal to  $\mathbf{P}_{(\theta,\mathbf{x})}^f$  of size  $1 \times n$  and if  $l = n$  then  $\mathbf{C}_{(\theta\mathbf{e},\mathbf{x})}$  can be chosen equal to  $\mathbf{C}_{(\mathbf{x},\mathbf{x})}$  of size  $n \times n$ .

One could argue that creating an artificial spatial dependency leads to a bias in the use of observations. Indeed, the weight that an observation is given for the parameter update is modified by a spatial dependency that does not exist. One can remedy this by using the identity matrix as correlation matrix for the parameters,  $\mathbf{C}_{(\theta\mathbf{e},\mathbf{x})} = \mathbf{I}$ . This way, each observation is given a relative weight that is solely determined by its error statistics and its correlation to the parameter, so that no biases of observation weights should occur. For  $\mathbf{C}_{(\theta\mathbf{e},\mathbf{x})} = \mathbf{I}$  one can rewrite (3.2.1) as:

$$\delta \theta_i = \frac{1}{n} \mathbf{P}_{(\theta,\mathbf{x})}^f \mathbf{H}^T \mathbf{S}^{-1} \mathbf{v}_i, \quad (3.2.2)$$

where for  $l = n$  we substituted  $\frac{1}{l} \mathbf{e}^T \mathbf{C}_{(\theta\mathbf{e},\mathbf{x})} \odot \mathbf{P}_{(\theta\mathbf{e},\mathbf{x})}^f = \frac{1}{n} \text{diag}\{\mathbf{P}_{(\theta\mathbf{e},\mathbf{x})}^f\} = \frac{1}{n} \mathbf{P}_{(\theta,\mathbf{x})}^f$ . Note that augmenting the parameter space to a higher dimension is not necessary here, which confirms the absence of an artificial spatial dependency of global parameters. We therefore refer to this method as *global updating*. The formula for updating the parameter (3.2.2) can

be obtained by a simple scalar parameter augmentation of the state that ensures positive definiteness of the localized error covariance matrix (see Appendix B). Global updating, where  $l = 1$ , has the additional advantage of being computationally cheaper than spatial updating, where  $l > 1$ . For the EnKF this saves  $\mathcal{O}(n(l-1))$  operations. For the QF the reduction in computational costs is even larger. However, it is the QPEns that benefits most from the new approach, as the optimization problems are solved in model space instead of observation space.

### 3.2.2 Dynamical model for parameters

For any simple augmented state algorithm, the parameter spread eventually collapses due to the static nature of the parameters. For  $N_{ens} \rightarrow \infty$  and under perfect (Gaussian) conditions this is justified, since the only uncertainty related to parameters is the initial conditions, which should be recovered by the algorithm. However, for smaller ensemble sizes this spread collapse leads to underdispersion in parameter space.

An overview of possible approaches to circumvent this phenomenon is provided by Ruiz et al. (2013b). We choose to impose stochastic dynamics on the parameters to represent the parameter uncertainty, as for example proposed in Gelb (1974). This approach allows the user to partially *choose* the parameter distribution (and hence its uncertainty), leaving the mean to be determined by the estimation algorithm. Specifically, we set for each ensemble member  $i, i = 1, 2, \dots, N_{ens}$

$$\theta_{t,i}^f = \theta_{t-1,i}^a + f(\mathbf{X}_{t,i}) \quad (3.2.3)$$

where  $t$  is the time index,  $f : [0, 1] \rightarrow [l_\theta - \theta_{t-1,i}^a, u_\theta - \theta_{t-1,i}^a]$  is a linear, bijective function and  $\mathbf{X}_{t,i} \sim \text{Beta}(\alpha_{t,i}, \beta_{t,i})$ . The relation between the parameters  $\alpha_{t,i}$  and  $\beta_{t,i}$  is chosen such that  $\mathbf{E}[f(\mathbf{X}_{t,i})] = 0$ . The degree of freedom that is left can for instance be used to set the variance or shape of the distribution. Note that since the Beta distribution is bounded, the parameters cannot exceed the prescribed range  $[l_\theta, u_\theta]$  as long as the posterior ensemble members do not exceed it. This is an advantage over using a Gaussian distribution or inflation type methods, where one would need to truncate the parameter values of the ensemble members that exceed the bounds, which affects the mean and the spread of the ensemble. An alternative would be to use a transformation of variables to guarantee that the parameters stay within their bounds, which may introduce additional nonlinearities in the parameter estimation problem.

## 3.3 Experiment setup

The purpose of this study is to compare the performance of the QPEns, the QF and the EnKF and evaluate the relative importance of explicitly taking higher order moments into account and respecting physical conservation laws and bounds for convective scale applications. To that end, we conducted a set of numerical 'twin' experiments with the modified shallow water model, where a model run (further referred to as the *nature run*) is considered the true state of the atmosphere.

The modified shallow water model (Würsch and Craig, 2014) consists of the following equations for the velocity  $u$ , rain  $r$  and water height level of the fluid  $h$  respectively:

$$\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} + \frac{\partial(\phi + \gamma^2 r)}{\partial x} = \beta_u + D_u \frac{\partial^2 u}{\partial x^2}, \quad (3.3.1)$$

with

$$\phi = \begin{cases} \phi_c & \text{if } h > h_c \\ gh & \text{else,} \end{cases} \quad (3.3.2)$$

$$\frac{\partial r}{\partial t} + u \frac{\partial r}{\partial x} = D_r \frac{\partial^2 r}{\partial x^2} - \alpha r - \begin{cases} \delta \frac{\partial u}{\partial x}, & h > h_r \quad \text{and} \quad \frac{\partial u}{\partial x} < 0 \\ 0, & \text{else,} \end{cases} \quad (3.3.3)$$

$$\frac{\partial h}{\partial t} + \frac{\partial(uh)}{\partial x} = D_h \frac{\partial^2 h}{\partial x^2}. \quad (3.3.4)$$

Here,  $h_c$  represents the level of free convection. When this threshold is reached the geopotential  $\phi$  takes on a lower, constant value  $\phi_c$ . The parameters  $D_u$ ,  $D_r$ ,  $D_h$  are the corresponding diffusion constants,  $\gamma := \sqrt{gh_0}$  is the gravity wave speed for absolute fluid layer  $h_0$  ( $h_0 < h_c$ ). The small stochastic Gaussian forcing  $\beta_u$  is added at random locations to the velocity at every model time step, in order to trigger perturbations and hence convection. Note that this implies that the location of convection is mostly random. The parameter  $\delta$  is the production rate and  $\alpha$  the removal rate of rain. When  $h$  reaches the rain threshold  $h_r$  (higher than  $h_c$ ), rain is 'produced' by adding rain water mass to the potential, leading to a decrease of the water level and of buoyancy. The model conserves mass, so the spatial integral over  $h$  is constant in time.

For the numerical implementation of the model, the one dimensional domain, representing 125 km is discretized with 250 points, yielding the state vector  $\mathbf{x} = [\mathbf{u}^T \mathbf{h}^T \mathbf{r}^T]^T \in \mathbb{R}^{750}$ . The time variable is discretized into time steps of 5 seconds. The Gaussian stochastic forcing  $\beta_u$  has a half width of 4 grid points and an amplitude of 0.002 m/s. The parameters that are estimated along with the dynamical state are  $\alpha \in [l_\alpha, u_\alpha]$ ,  $\phi_c \in [l_{\phi_c}, u_{\phi_c}]$  and  $h_r \in [l_{h_r}, u_{h_r}]$  (see Table 5.1.1 for values of upper and lower bounds). As previously stated, the multiplicative parameter  $\alpha$  regulates how fast rain decays. Parameter  $h_r$  is responsible for a possible switch between the regimes, since it determines when it starts to rain. Parameter  $\phi_c$  is a value of geopotential once the level of free convection is reached and influences the total time required for a cloud to develop to full height. Each ensemble

Parameter	Lowerbound	Upperbound
	$l$	$u$
$\alpha$	0.0003	0.001
$\phi_c$	899.7	899.9
$h_r$	90.15	90.25

Table 3.3.1: Lower and upper bounds for the parameters to be estimated.

member as well as the nature run is initialized as a random model run, with parameter

values drawn randomly from the uniform distributions  $\mathcal{U}(l_\alpha, u_\alpha)$ ,  $\mathcal{U}(l_{\phi_c}, u_{\phi_c})$  and  $\mathcal{U}(l_{h_r}, u_{h_r})$  respectively. After an appropriate spin up time, 250 assimilation cycles are performed. All experiments discussed below need about 50 cycles to converge to a constant mean RMSE value. Therefore, the average of the RMSE over the last 100 cycles is computed to assign a scalar RMSE value to each experiment. For statistical significance, the final score used in the results is the average RMSE over 250 experiments.

Unless stated otherwise, observations are taken from the nature run every 60 model time steps (equivalent to 5 minutes in real time). A Gaussian observation error is added to the wind  $u$  and height  $h$  fields with a standard deviation of  $\sigma_u = 0.001$  m/s and  $\sigma_h = 0.02$  m, and a lognormal error is added to the rain  $r$  field with parameters  $\mu = -8$  and  $\sigma = 1.8$ , yielding a very small observation bias of 0.000825 and standard deviation of 0.00185. For all variables the observation error is roughly 10% of the maximum deviation from the variable mean. To mimic radar data, observations for all variables are available only on grid points where rain above a threshold of 0.005 dBZ is measured. Therefore, the rain data has a known observation bias of 0.000825 dBZ where  $r > 0.005$  dBZ in the nature run. Note that with a probability of 6.7%, observations corresponding to a dry grid point in the nature run are also assimilated. For rain, these observations have a bias larger than 0.005 dBZ. A random selection, amounting to 25% of the remaining grid points, of additional wind observations are assimilated, which represents additional data available (for example obtained from aircraft).

To deal with undersampling, covariance localization according to Gaspari and Cohn (1999) is applied with a fixed localization radius of 6 grid points for all three algorithms. This corresponds to the localization radius for which the EnKF yields minimum analysis RMSE values of the rain variable for small ensemble sizes. For the spatial updating technique described by (3.2.1), the localization matrix for the parameters is the same as for the state. For global updating we use  $\mathbf{C}_{\text{COV}(\theta, \mathbf{x})} = \frac{1}{n} \mathbf{e} \mathbf{e}^T$  as discussed in section 3.2.1. For the dynamical model of the parameters according to (3.2.3) we choose to use the second degree of freedom to set a lower limit on the parameter spread. This limit is set to 25% of the initial spread. An interior point method is used to solve the quadratic minimization problems of the QPEns. The constraints that are applied are mass conservation, i.e.  $\mathbf{e}^T (\mathbf{h}^a - \mathbf{h}^f) = \mathbf{e}^T \delta \mathbf{h} = 0$ , and positivity of precipitation, i.e.  $\mathbf{r}^a = \delta \mathbf{r} + \mathbf{r}^f \geq 0$ . For the QF and the EnKF negative values for rain are set to zero if they occur.

## 3.4 Results

Results are generally presented in the form of RMSE as a function of ensemble size. For the analysis RMSE we use solid lines, for the background RMSE we use dashed lines. Colors are used when necessary to indicate the algorithm: blue for the EnKF, green for the QF and red for the QPEns. Markers correspond to different experiment settings. For example, round markers correspond to experiments with default settings, where only the state is estimated, i.e. the parameters are correct and fixed throughout the assimilation. Diamond markers are used for the joint state and parameter estimation experiments.

### 3.4.1 State estimation

To evaluate the behavior of the different algorithms without parameter uncertainty, we first discuss Figure 3.4.1, where the average RMSE of the analysis and the background is plotted against the ensemble size. Both the QF and QPEns are more sensitive to the ensemble size than the EnKF. They therefore need an ensemble size larger than a certain threshold ( $N_{ens} > N_{QF}$  and  $N_{ens} > N_{QPEns}$ ) to outperform the EnKF, with usually  $N_{QF} > N_{QPEns}$ . To accurately calculate higher order moments, a larger sample size is needed, so it is clear why the QF becomes superior to the EnKF only for sufficiently large ensemble sizes. With regards to the QPEns we hypothesize that the error covariance matrix needs to be sufficiently accurate to use the mass constraint beneficially. When the error covariance matrix is poor, the mass constraint might cause spurious convection instead of preventing it. The background shows roughly the same behaviour as the analysis, although the distinctions between the algorithms are somewhat damped, as the analysis increments of the QF appear slightly smaller and those of the QPEns slightly larger than those of the EnKF. In particular, the threshold  $N_{QF}$  is smaller for the background than for the analysis. The top of the right side of Figure 3.4.1 displays the analysis state RMSE of the rain field split up into rainy and dry regions. The QF gains its overall advantage relative to the EnKF in the rainy regions only. However, the RMSE in the dry regions decreases for the QF and increases for the EnKF as the ensemble size grows. This indicates that a threshold above which the QF outperforms the EnKF exists in dry regions as well, but is higher than in the rainy regions. The QPEns does especially well in dry regions when the ensemble size is large enough. The lower two plots on the right show the absolute error in mass of  $\mathbf{h}$ , and  $\mathbf{r}$  respectively. Recall that both the model and the QPEns conserve the mass of  $\mathbf{h}$ , which means that the mass of  $\mathbf{h}$  is kept constant throughout the forecast. Absolute mass errors of  $\mathbf{h}$  do not decrease for the EnKF and the QF when the number of ensemble members is increased and cause mass errors in  $\mathbf{r}$  as well. The QF and EnKF even add errors in the analysis step for rain mass instead of reducing them. The slight increase in absolute mass of rain for QPEns with the number of ensemble members is the result of the bias in observations from dry regions. To confirm this, we performed experiments using the nature run to distinguish rainy from dry regions (instead of the observations). In this case, the increase in absolute mass of rain for QPEns does not exist (not shown). However, using the nature run to distinguish rainy from dry regions results in less data being assimilated. Recall that we assimilate information on all three variables only where it rains to mimic radar data. This therefore leads to higher RMSE's and absolute mass errors for the QF and the EnKF, but the same relative behavior as in Figure 3.4.1.

To distinguish between the respective effects of mass conservation  $\mathbf{e}^T(\mathbf{h}^a - \mathbf{h}^f) = 0$  and positivity constraints  $\mathbf{r}^a \geq 0$ , we in addition conducted experiments in which only mass conservation was applied, and ones where only positivity constraints were applied. The results with only mass conservation are very close to the QPEns results, where both type of constraints are imposed (not shown). Experiments where only the positivity of rain is constrained, yield slightly higher RMSEs than those corresponding to the EnKF (not shown). This is consistent with the findings of Janjić et al. (2014), where it was demon-

strated that mass conservation is needed to benefit from the positivity constraints. We conclude that the effect of mass conservation is both dominant compared to the positivity constraints and necessary for improved accuracy. We therefore assume that for our test case the distinctions between EnKF and QPEns arise mainly the result of mass conservation.

The relative behavior of the algorithms depends on the experiment setup. To investigate the sensitivity of the setup further, we vary the localization radius, the observation coverage of  $u$  and the observation frequency. In our experiments, the observation frequency affects the total number of observations assimilated. In particular, by doubling the observation frequency, the total number of observations assimilated is also doubled. The results are shown in Figures 3.4.2, 3.4.3 and 3.4.4.

Comparing the performance of the algorithms for different localization radii in Figure 3.4.2, we conclude that  $N_{QF}$  and  $N_{QPEns}$  are positively correlated to the localization radius. In particular we see a strong superiority of the QPEns for the narrow localization radius. When the localization radius is too small, balances are destroyed (Kepert, 2009; Greybush et al., 2011), conservation laws are violated (Janjić et al., 2014; Zeng and Janjić, 2016) and the natural spatial smoothness of the state variables is lost. This introduces noise, i.e. undesirable small scale fluctuations in all fields. It is for this reason that the absolute mass error of  $h$  for the EnKF and the QF is much larger for the small localization radius. The absolute mass error of  $h$ , which is caused by noise that can lead to spurious clouds, extends to the rain field, also creating a strong absolute error in precipitation for the EnKF and the QF. This is avoided for the QPEns, owing to the mass conservation constraint. The top right plot of Figure 3.4.2 supports the hypothesis that mass conservation contributes to preventing the introduction of noise and spurious clouds caused by narrow localization, as the smaller RMSE of rain for the QPEns is inherited from dry regions only. For large localization radii, mass is better conserved within the EnKF resulting in smaller RMSE for the state and smaller absolute mass errors.

As the time between assimilation cycles increases, the prior becomes less Gaussian, and therefore more is gained from taking higher order moments and constraints into account. This is confirmed by Figure 3.4.3. For temporal sparse observations, the non-Gaussianity caused by rapidly developing and intermittent clouds has time to manifest itself in the prior distribution. Since the QF partially accounts for non-Gaussianity the absolute mass errors are clearly smaller for the QF than for the EnKF. As reasoned above, the absolute mass errors are related to noise and spurious clouds. By constraining the mass, noise and therefore spurious clouds are suppressed, as is again confirmed in the top right plot of Figure 3.4.3, which shows that for temporal sparse observations, the distinctions between the performance of the methods comes mainly from dry regions, i.e. from the presence of noise and spurious convection.

Figure 3.4.4 displays information about the behavior of the algorithms with respect to the observation coverage of  $u$ . There is no significant relative difference in performance between the EnKF and the QF when the observation coverage of  $u$  is increased. The gradients of the graphs as well as  $N_{QF}$  are similar for all observation coverages. It is for the QPEns that large relative diversity is detected. The  $h$  and  $r$  variables are driven by the  $u$  variable, since the stochastic perturbations in the wind field determine the location

of the clouds. Due to the time lag between both the development of a cloud in  $\mathbf{h}$  and the production of rain in  $\mathbf{r}$ , the model forecast benefits most from wind observations. By observing the entire wind field, the background has good skill with respect to detecting arising clouds, which is not the case when no additional wind observations are assimilated. When the background is accurate in the location of the clouds, there is less room for spurious clouds. Also, the more accurate the background, the less weight is put on the observations and therefore less noise is introduced. As a result, mass errors are smaller. So as the observation coverage of  $\mathbf{u}$  increases, the benefit of imposing mass conservation decreases. However, when the observation coverage of  $\mathbf{u}$  decreases, the same reasoning as for the previous figures applies, i.e. there is more room for spurious clouds due to lack of forecast skill, which the QPEns can suppress by imposing mass conservation. Again, the top right plot of Figure 3.4.4 clearly supports this.

In general the QF is superior in the rainy regions and the QPEns in the dry regions. Likewise, separating the  $h$  variables according to  $h > h_c$  and  $h \leq h_c$ , highlights the superiority of the QF in cloudy regions and that of the QPEns in the non-cloudy regions (not shown). Due to the mass restriction, the QPEns can only reach the full height of the cloud if the non-cloudy regions are estimated close to perfection, i.e. no spurious mass at all, which is practically unattainable. It is therefore not surprising that the QPEns excels in dry and non-cloudy regions, but lacks mass in cloudy and rainy regions. Since the QF takes higher order moments into account, we expect a better performance than the EnKF as long as the sample size is larger than a critical threshold. This threshold is larger for non-cloudy and dry regions, because the prior distribution is more skewed in those regions and needs more samples to accurately estimate its third moment. The excellence of the QF and the weakness of the QPEns in the wet regions are generally slightly enhanced in the background (not shown), probably due to nonlinearity.

The QPEns is by far the most sensitive to the experiment setup. When observations are temporally and/or spatially sparse, the QPEns excels. However, when they become more dense in the wind field, the ensemble size threshold  $N_{QPEns}$  above which the QPEns beats the EnKF is increased. In addition, the QPEns can handle narrow localization radii which the EnKF and the QF cannot. To further illustrate the effect that mass conservation has on the state estimate, we refer to Figure 3.4.5, where a snapshot of a random experiment with 50 ensemble members and similar setup as the square plot in Figure 3.4.4 is presented for the EnKF and the QPEns. For all state variables, the difference in behavior between the respective algorithms is primarily visible in non-cloudy regions (compare truth (red) and ensemble mean (blue)). Here, undesirable small scale fluctuations (noise), are significantly smaller for the QPEns.

### 3.4.2 Parameter estimation

To illustrate the effect of imposing dynamics on the parameters and to compare the two approaches for parameter localization as discussed in section 3.2, we refer to Figure 3.4.6, where the results for the different experiments are shown for the EnKF. For both spatial and global updating, the RMSE of the state and the parameters is clearly reduced when

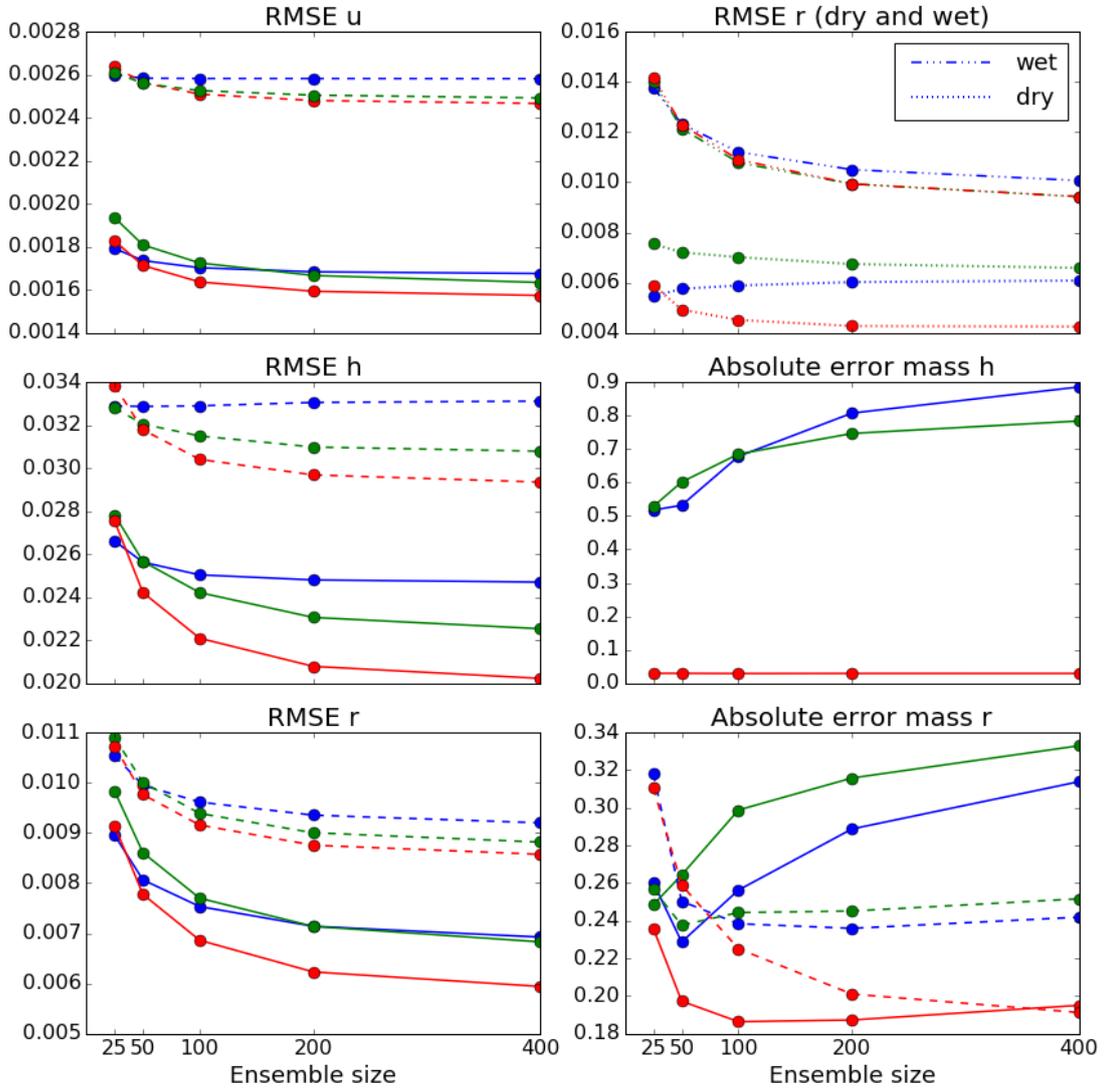


Figure 3.4.1: Results of state estimation with localization radius 6, 5 minutes between assimilation cycles, and 25% additional wind observations. For the background (dashed) and analysis (solid) corresponding to EnKF (blue), QF (green) and QPEns (red), the plots show: Left, the RMSE of the state as a function of ensemble size. Top right, the analysis RMSE of  $r$  split up into rainy (dotted) and dry (dash-dotted) regions. Lower right, the absolute mass error of  $h$  and  $r$  respectively.

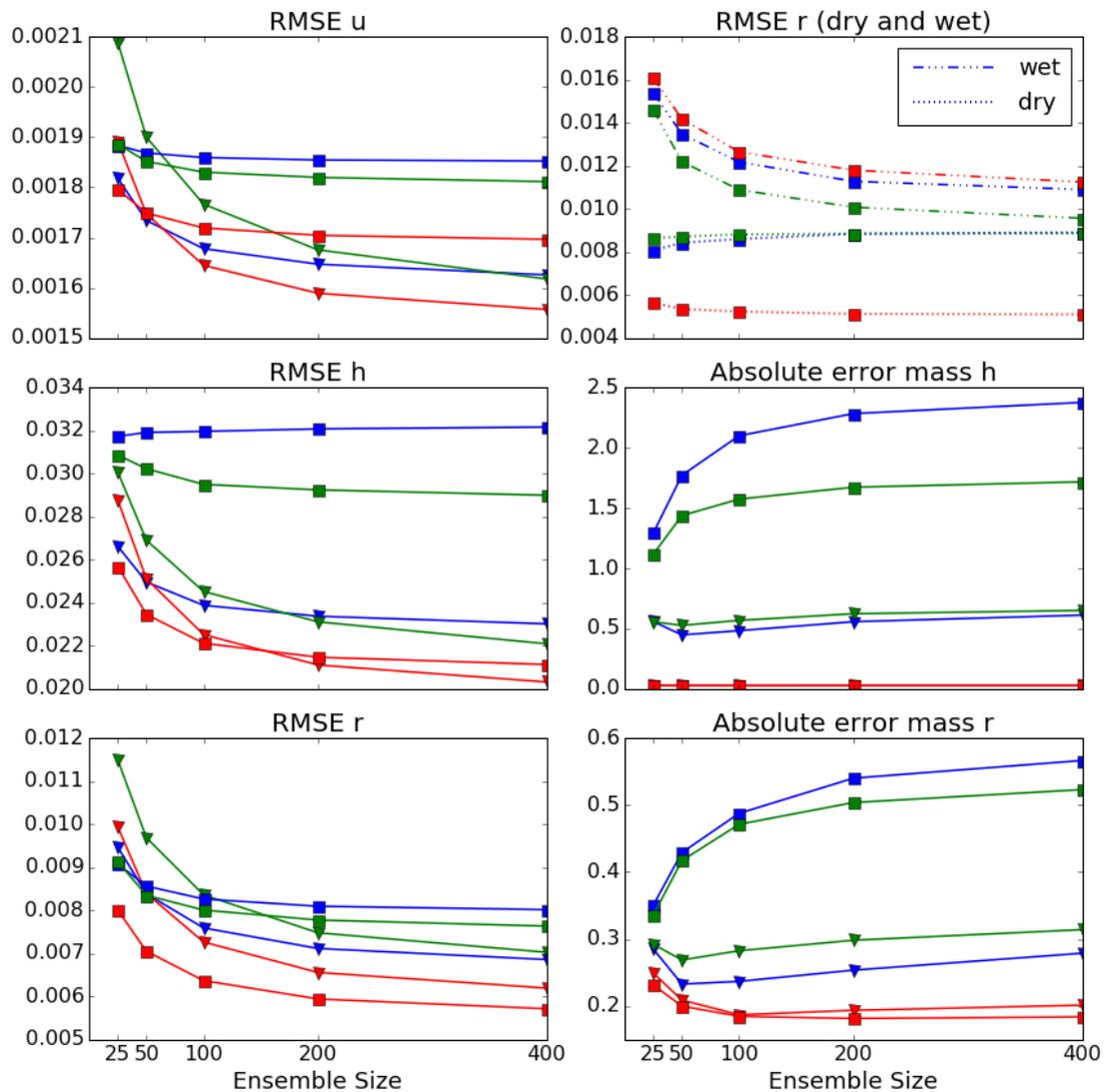


Figure 3.4.2: Results of state estimation with 5 minutes between assimilation cycles, and 25% additional wind observations. For the analysis state of EnKF (blue), QF (green) and QPEns (red) and for localization radius 2 (square) and 10 (triangle), the plotted quantities are as in Figure 3.4.1.

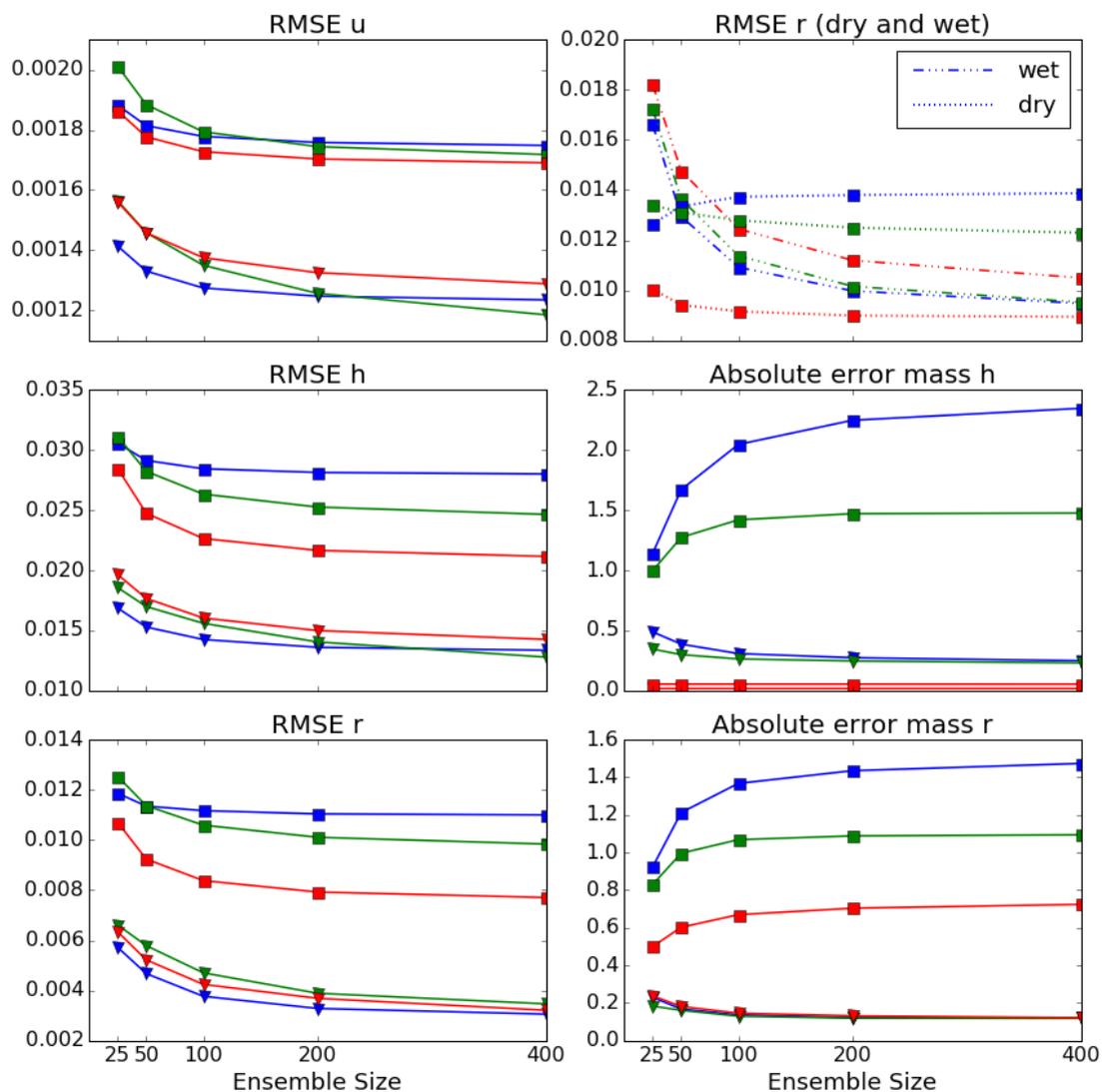


Figure 3.4.3: Results of state estimation with localization radius 6 and 25% additional wind observations. For the analysis state of EnKF (blue), QF (green) and QPEns (red) and for an assimilation cycle period of 1 min (triangle) and 30 min (square), the plotted quantities are as in Figure 3.4.1.

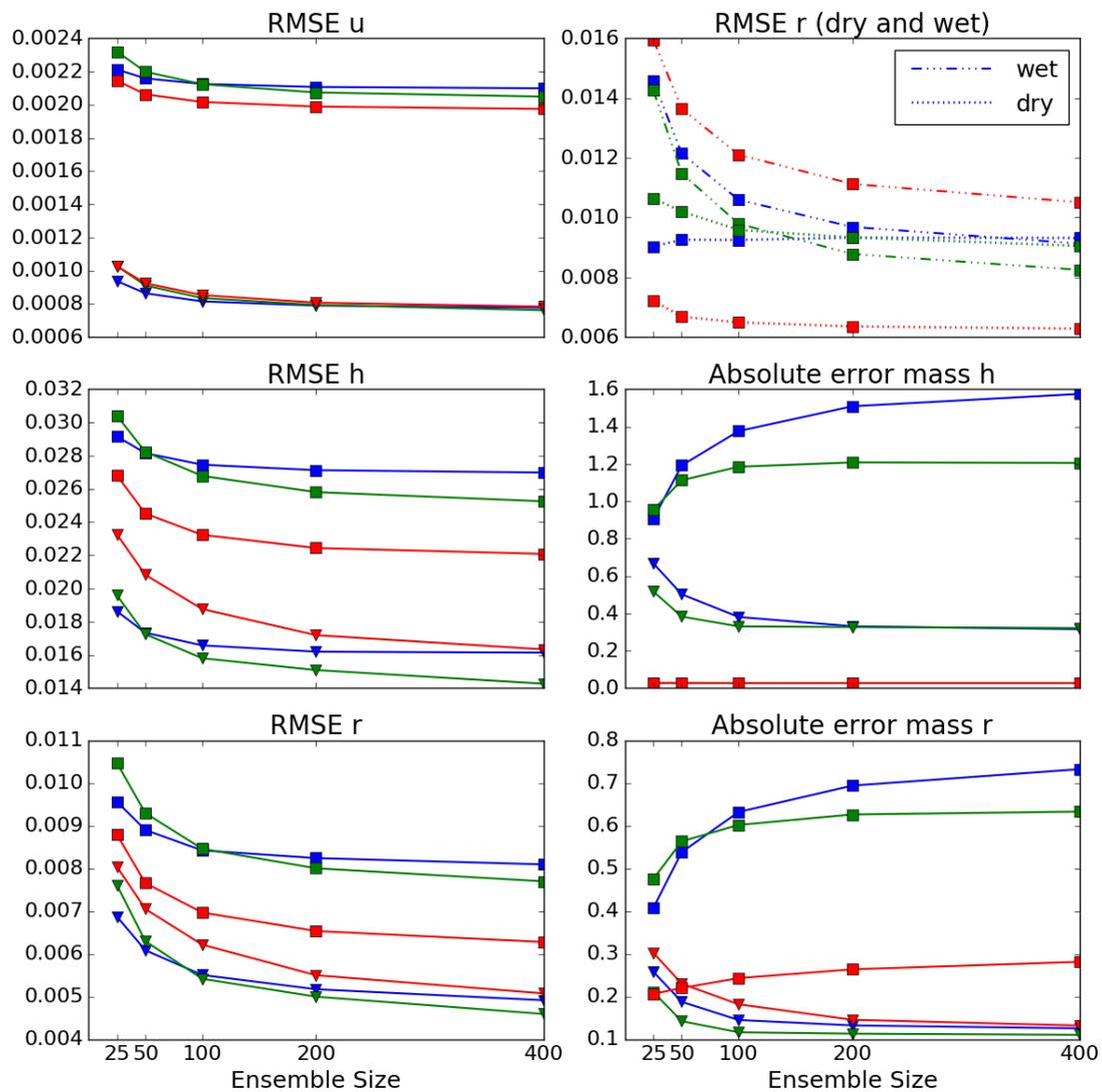


Figure 3.4.4: Results of state estimation with localization radius 6 and 5 minutes between assimilation cycles. For the analysis state of EnKF (blue), QF (green) and QPEns (red) and for no additional wind observations (square) and 100% wind observation coverage (triangle), the plotted quantities are as in Figure 3.4.1.

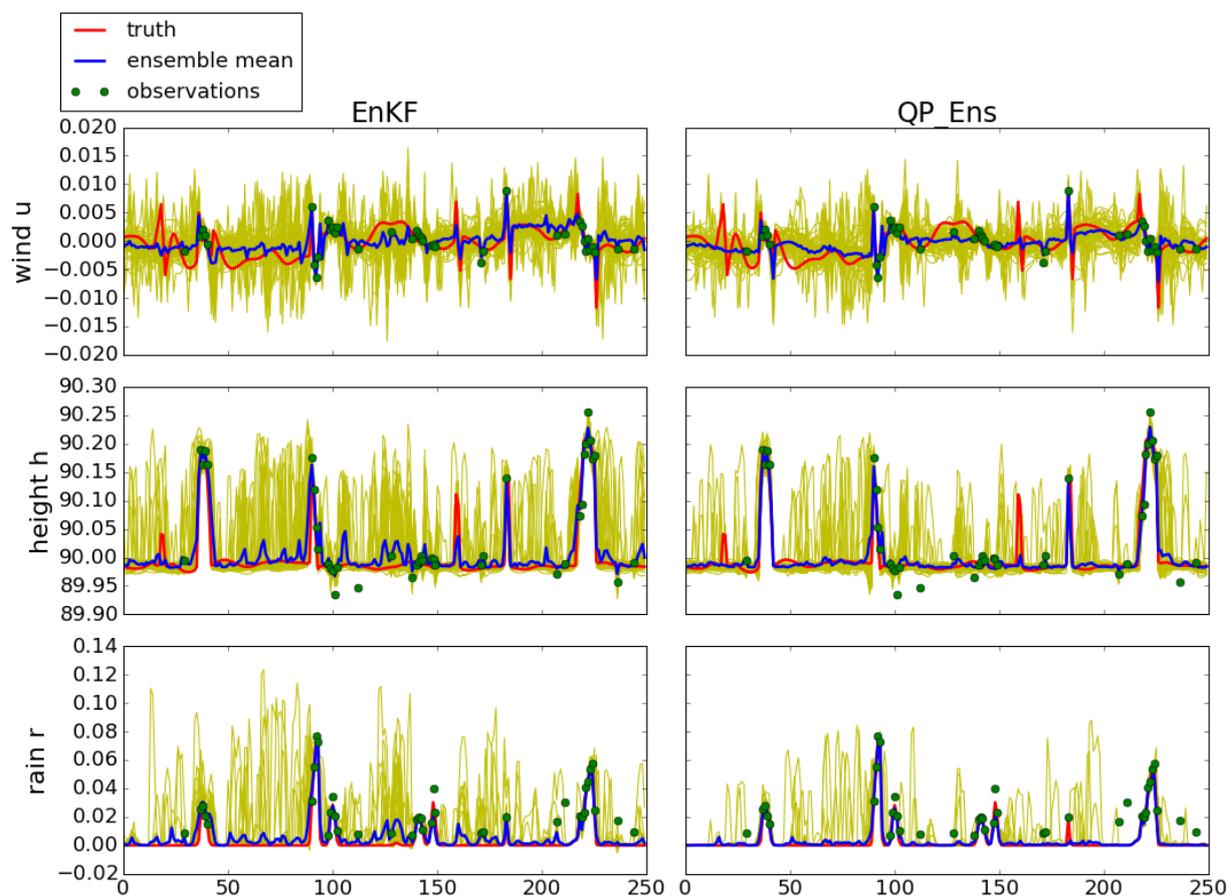


Figure 3.4.5: Snapshot after 50 assimilation cycles resulting from the EnKF (left) and the QPEns (right) for 50 ensemble members. The truth (red line), the ensemble analysis mean (blue line), the observations (green dots) and the ensemble members (yellow lines) are shown for the velocity (top panel), the height (middle panel) and the rain (bottom panel) for all 250 grid points of the model. Cloudy regions are represented by peaks in the height and rain field.

the dynamical model is used to maintain a certain parameter spread. The difference is on average larger for small ensemble sizes, because spread collapse is a slower process for large ensemble sizes. Although it is not clear why the absolute error for  $\alpha$  increases with ensemble size for spatial updating, its effect on  $r$  does make sense. Indeed, since  $\alpha$  influences the rain decay rate, the error in  $\alpha$  is strongly correlated to the RMSE of  $r$ . Overall, global updating appears to perform as good, if not better, than spatial updating. With the additional advantage of significantly reduced computational costs, we consider global updating superior for this application. The remainder of this paper is therefore dedicated to the discussion of experiments with global updating and with dynamical model for parameters.

From Figure 3.4.7 we can conclude that applying parameter estimation affects the relative behavior of the algorithms with respect to the state. There are three important differences. First, the thresholds  $N_{QF}$  and  $N_{QPEns}$  are smaller than for state estimation without model error due to parameter uncertainty. In particular, QPEns outperforms EnKF now for all ensemble sizes. Second, the contribution of dry regions to the RMSE is relatively larger with parameter estimation. This is also reflected by the increased absolute mass errors. Third, estimating parameters along with the state yields a stronger dependence of the state on the ensemble size for the QF. Already for small ensemble sizes the parameter estimates are better for the QF than for the EnKF. For the QPEns the parameters remain within their respective bounds anyway (not shown) and therefore no constraints are imposed on the parameters. The only influence the QPEns has on parameter estimation with respect to the EnKF is through the constraints imposed on the state. The non-Gaussianity related to the parameters is not dealt with. Therefore, for parameter estimation the gradient of the graph corresponding to the QPEns becomes closer to that of the EnKF. Taking higher order moments into account, however, is directly beneficial for parameter estimation (at least for  $\alpha$  and  $h_r$ ), as is confirmed by the light blue line in Figure 3.4.7, where the state is updated with the EnKF and the parameters with the QF. Given a sufficiently large ensemble size, the QF can represent the distributions related to the parameters more accurately than the EnKF and the QPEns, and since the model space is essentially increased when parameter estimation is applied, the QF becomes more dependent on ensemble size. It therefore has the potential to beat both the EnKF and QPEns as the ensemble size increases for parameter estimation. The analysis increments are comparable to those obtained from the experiments without parameter uncertainty, meaning that the QF has the smallest increments and the QPEns the largest.

Since no constraints are applied to the parameters, comparing the EnKF and the QPEns gives us information about the influence of state errors on parameter estimation. The smaller error of  $\phi_c$  for the QPEns indicates that state errors can certainly influence the estimates of the parameters. The parameter errors, in turn, also influence the state estimates. Indeed, the QPEns yields smaller state errors than the EnKF already for small ensemble sizes, which is not the case for state estimation only. Comparing the dark blue and light blue lines in Figure 3.4.7 also supports this statement. Taking skewness into account for the parameter updates, yields better estimates of the parameters. The fact that the state errors are also reduced (even though no skewness is taken into account for

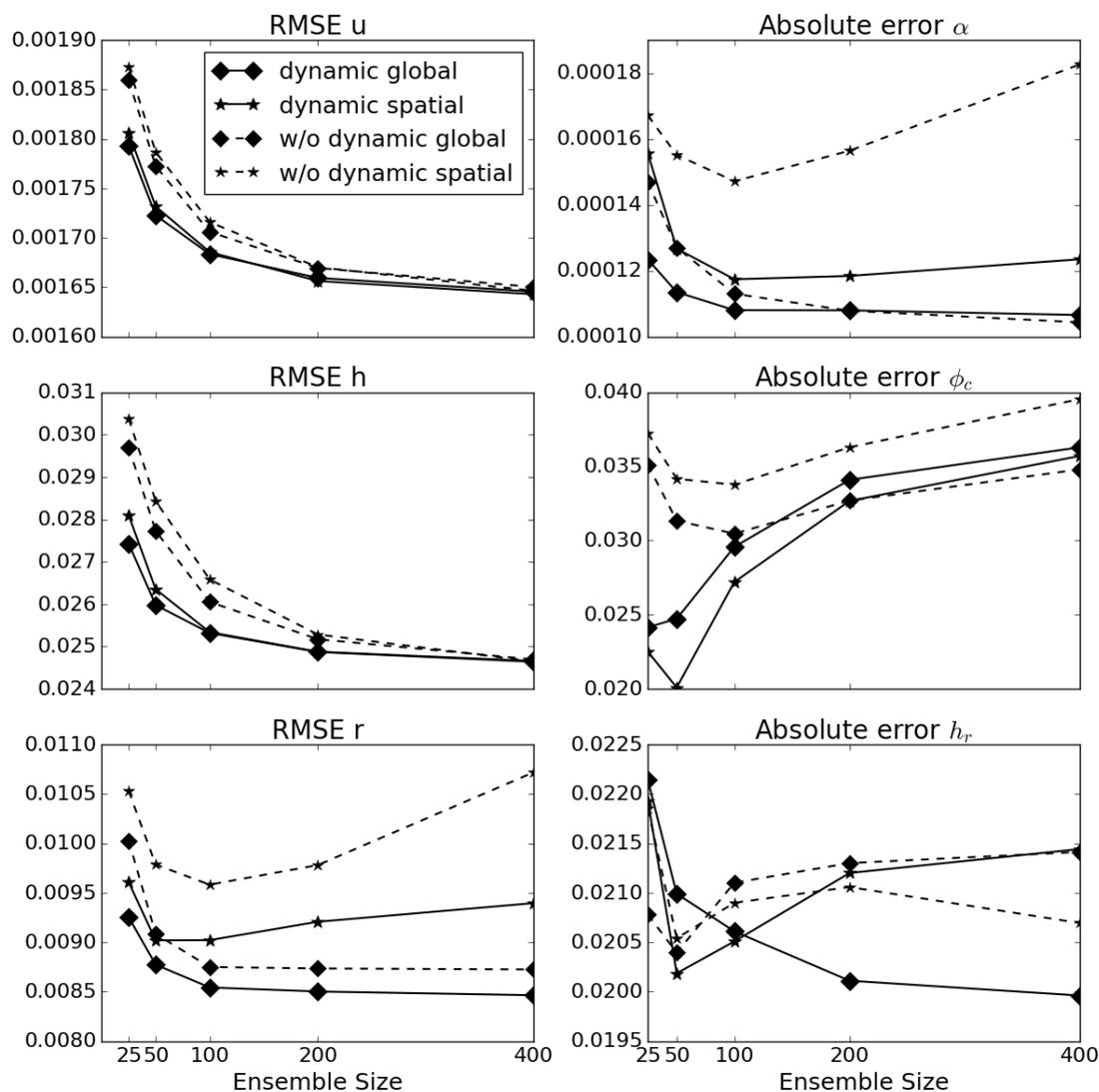


Figure 3.4.6: Average RMSE of the state and the parameters versus ensemble size for global (diamond markers) and spatial (star markers) updating, with (solid) and without (dashed) dynamical model for parameters for the EnKF.

the state updates), confirms the influence of parameter errors on the state estimates.

Interestingly, the bias of  $\phi_c$  is very close to its absolute error, which indicates that the algorithm systematically underestimates the true parameter value. The optimization problem appears to have a local minimum for small  $\phi_c$ . This explains why the quality of the estimate for  $\phi_c$  deteriorates as the ensemble size increases. Besides, a smaller parameter error does not always yield smaller RMSE of the state: compare for example the QF and the EnKF for small ensemble sizes. This suggests that the additional degrees of freedom introduced from treating  $\phi_c$  as uncertain, compensate for errors that are not parameter related. Figure 3.4.8 demonstrates that this is indeed the case, i.e. the RMSE for state and parameter estimation is sometimes lower than for state estimation without parameter error. In experiments where only  $\phi_c$  is estimated, this effect is stronger (not shown). The graph with triangular markers in Figure 3.4.8 shows the RMSE of the analysis state resulting from state estimation with fixed, but wrong parameter values. These experiments are meant to represent the effect that model error due to wrong parameter values have on the state estimates. The parameter values for these experiments were chosen such that they are equal to the initial parameter errors of the ensemble mean corresponding to the experiments with parameter estimation. Figure 3.4.8 also shows the benefit of parameter estimation on the RMSE/spread ratio. Note however that the spread remains too small even for large ensemble sizes, which is apparently not only caused by undersampling and model error (otherwise the RMSE/spread ratio for the round marker experiments would converge to 1 with ensemble size). We hypothesize that instead, non-Gaussianity contributes to the lack of spread, which cannot be resolved with typical inflation techniques. Indeed, we found that in our experiments multiplicative inflation increases the RMSE as well as the spread, thereby failing to be effective.

Figure 3.4.9 shows the time evolution of the parameter absolute errors for 50 ensemble members. The error of  $\alpha$  was reduced between 30% and 45%, that of  $\phi_c$  between 45% and 70% and that of  $h_r$  between 20% and 30%. As prescribed by the dynamical model for the parameters, the spread converges to 25% of its original value. Although the RMSE of the state converges after 50 cycles, the parameter errors still go down after 250 cycles, with the exception of  $h_r$  for the EnKF and the QPEnS. This indicates that augmented state parameter estimation with the EnKF approach is not very flexible. It would need many assimilation cycles to adjust to any changes in the true parameter values. We therefore argue that this method is suited for static or slowly varying (for instance on seasonal timescales) parameter values, but probably less suited for fast varying parameters, such as those that vary with weather. In the latter case one could apply Sampling Error Correction (SEC) introduced by Anderson (2012, 2016). This method is based on Monte Carlo simulations that serve to calculate a statistically based correction term for a given sample correlation, ensemble size and prior assumption of the correlation distribution. These corrections are computed off-line, resulting in a simple look up table for the sampling error correction.

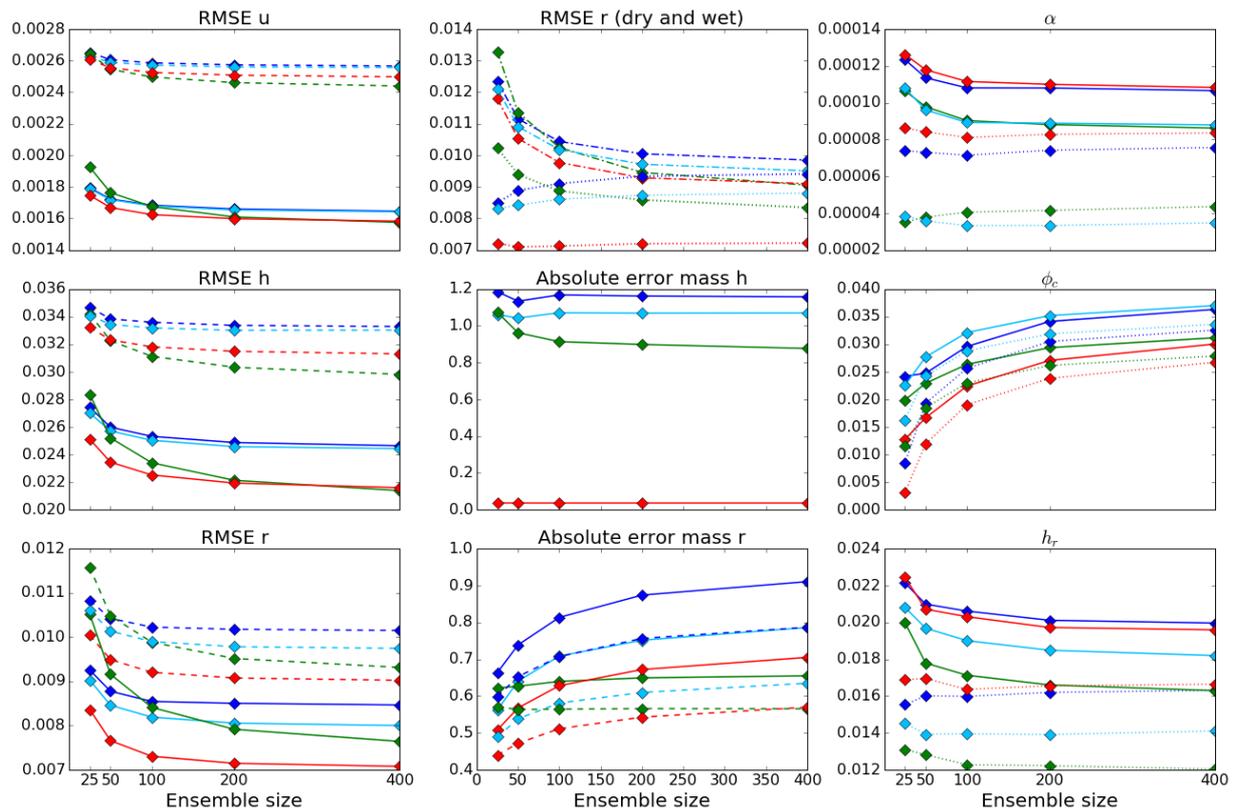


Figure 3.4.7: Average analysis (solid) and background (dashed) RMSE of the state and absolute error (solid) and bias (dotted) of the parameters versus ensemble size for the EnKF (blue), QF (green) and QPEns (red). The light blue line corresponds to experiments in which the EnKF is applied to the state and the QF is applied to the parameter estimation, i.e. higher order moments are taken into account for parameter updates only.

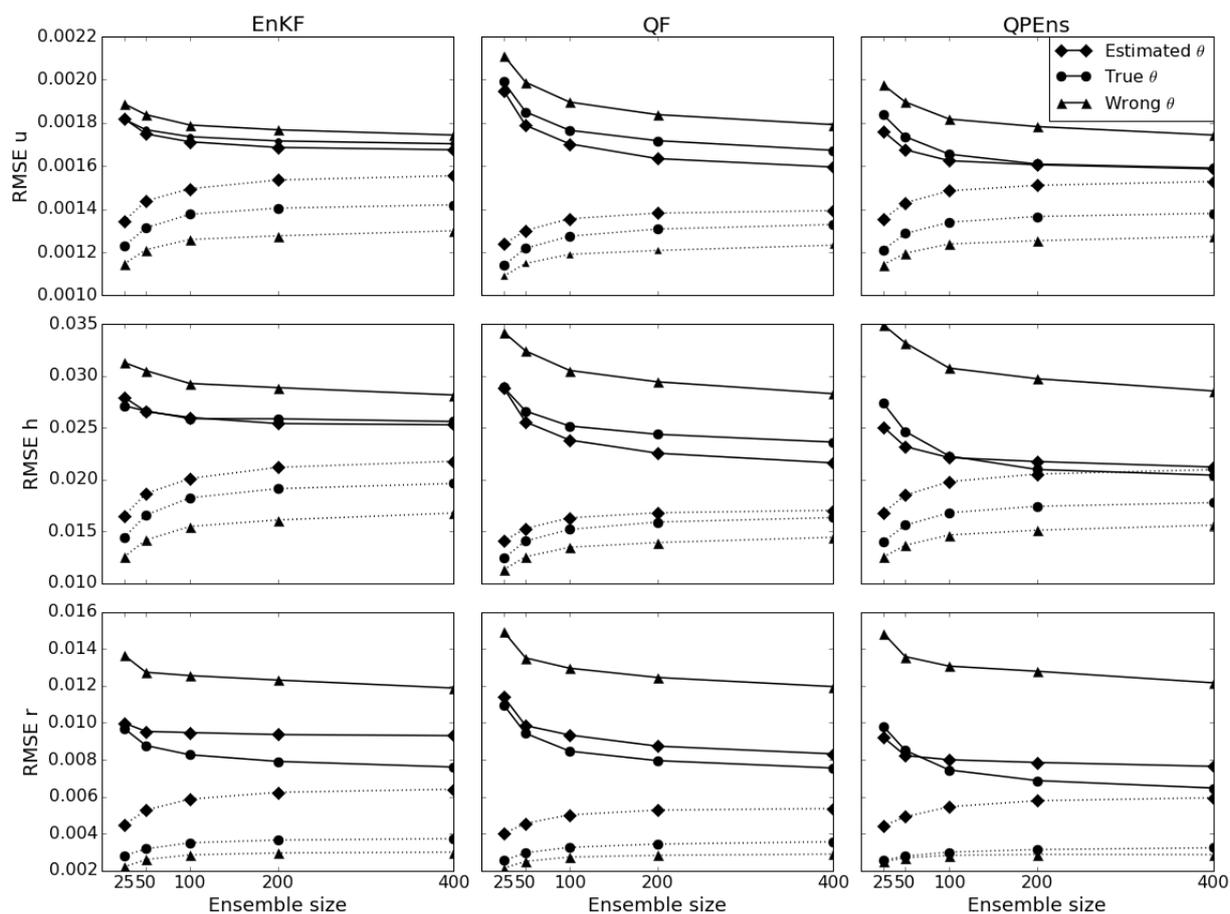


Figure 3.4.8: Average analysis RMSE (solid) and analysis spread (dotted) of the state versus ensemble size for the EnKF (left), QF (middle) and QPEns (right). The different markers correspond to experiments in which all parameters are: estimated (diamond), fixed and correct (circle), and fixed and incorrect (triangle). Note that the solid line graphs corresponding to the diamond markers are the same as in Figure 3.4.7, and the solid line graphs corresponding to the circle markers are the same as in Figure 3.4.1.

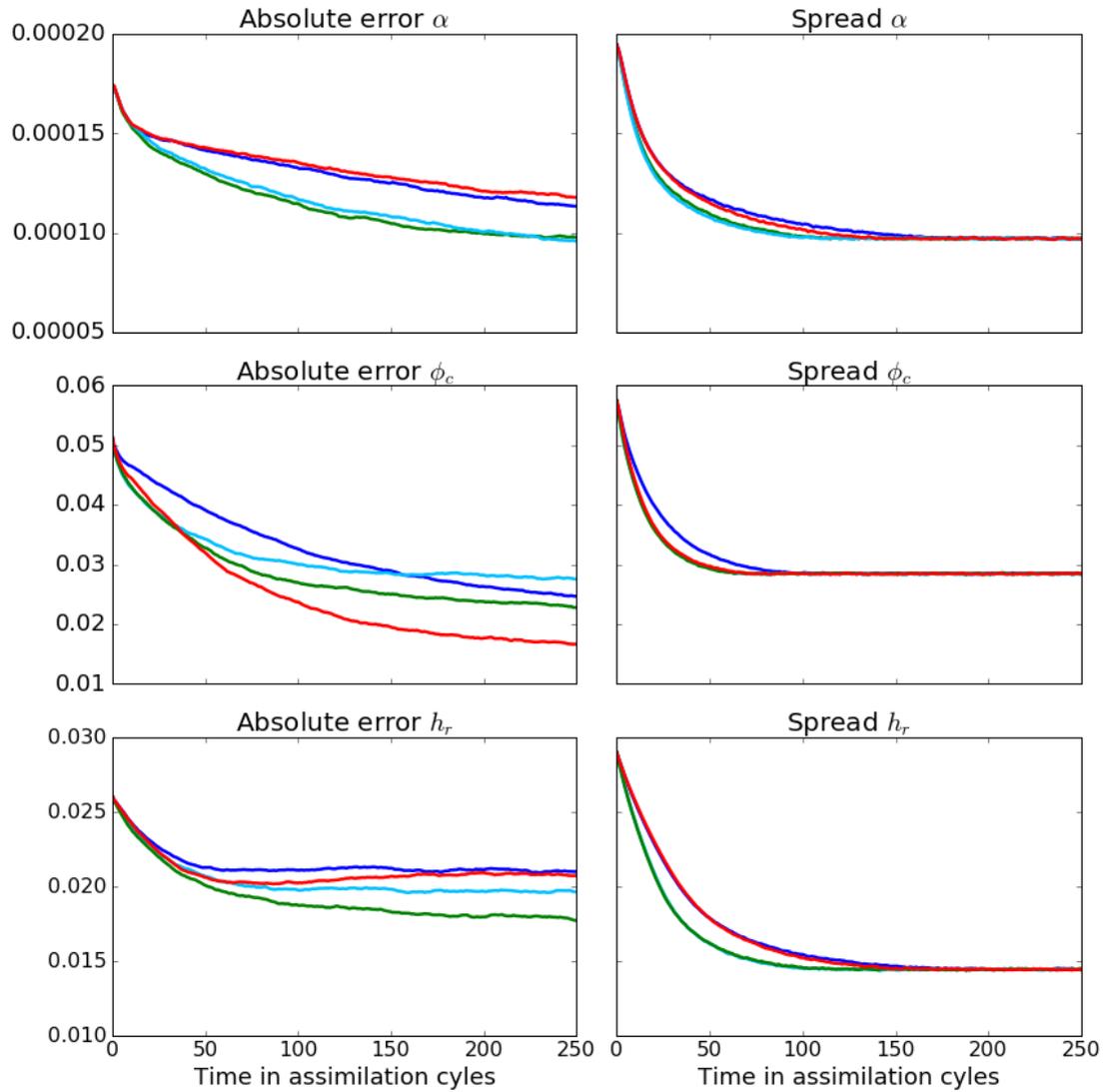


Figure 3.4.9: Average absolute error (left) and spread (right) of the parameters versus time in assimilation cycles for 50 ensemble members.

## 3.5 Conclusion

In this paper we compared the model state analysis RMSE for three different EnKF-based algorithms applied to the modified shallow water model (Würsch and Craig, 2014). In addition, feasibility of parameter estimation using the augmented state approach (Jazwinski, 1970; Evensen, 2009) was investigated. We used a stochastic dynamical model based on a Beta distribution for parameters to prevent underdispersion in parameter space and keep parameters within their bounds. To deal with global parameter localization, we introduced a computationally cheap and successful modification of spatial updating, which we call global updating.

Decreasing the localization radius introduces noise that leads to spurious convection, which leads to a mass error of  $h$  and therefore  $r$ . We have shown that in this case, preserving the mass of  $h$  strongly contributes to suppressing this noise and, by extension, the spurious convection that arises from it. It was shown in Lange and Craig (2014) that, given a realistic ensemble size and observation coverage and frequency, the localization radius that yields the smallest RMSE of the analysis state does not necessarily yield the best forecasts, presumably due to the introduction of noise. They argue that the localization radius should be chosen sufficiently large to avoid too much spurious convection, at the expense of local forecast accuracy. The QPEns might offer the possibility of suppressing noise, while still achieving local accuracy by decreasing the localization radius. It was also shown that for spatial and temporal sparse observations, which is currently the case in practice, mass conservation plays a significant beneficial role as well. This serves as a motivation to further develop the QPEns to make it computationally affordable.

The QF, which could be computationally suited for practical applications, shows benefits (especially in rainy regions) for ensemble sizes larger than some critical value, which depends on the localization radius and the observation frequency. Whether incorporating higher order moments is beneficial or not will therefore depend on the application. However, when applied to parameter estimation only, the QF supersedes the EnKF for all ensemble sizes and all state variables that were tested in this study (compare light and dark blue lines in Figure 3.4.7). We therefore conclude that explicitly accounting for higher order moments can be effective and affordable for parameter estimation problems.

We detected a strong positive feedback loop of errors between the state and parameters. As a result, the advantages of taking higher order moments into account and imposing constraints are enhanced when parameter estimation is applied. An interesting, though not surprising side effect of augmented state parameter estimation, is the algorithm's unpredictable use of the additional degrees of freedom. While our goal might be to estimate the parameters to the best possible accuracy, the algorithm might compensate for other errors instead, yielding poorer parameter estimates, but better state estimates. This behavior was detected with respect to  $\phi_c$ . Still, the parameters were all estimated fairly well, in the sense that the error was in all cases reduced and in some cases even up to 70% for a modest ensemble size of 50.

In practical applications, where other forms of model error influence state estimation as well, the consequences of parameter estimation are not predictable. Error compensations

might be beneficial for some, but not all errors. This was clearly highlighted in the study of Simon et al. (2015), where combined state and parameter estimation was conducted on an ecosystem model for the North Atlantic and Arctic Oceans. Therefore, the next research step is to apply parameter estimation to models of operational complexity, and investigate the forecast quality in realistic scenarios, which is done in Chapter 4.

# Chapter 4

## Estimation of roughness lengths in COSMO-KENDA

For the transport of heat and moisture, for the build up of convective available potential energy (CAPE), and the initiation of convection, accurate surface fluxes as well as a proper parametrization of the boundary layer are required. Numerical weather prediction models usually have limited skill in the boundary layer due to errors in the land surface boundary conditions, surface fluxes and parameterization of vertical mixing. The calculation of surface fluxes and vertical mixing depends on very crude estimated parameters such as the roughness lengths and the turbulent length scale. Modest changes to these parameters can significantly affect the initialization and development of clouds in numerical models. Here, our focus is on representing and reducing the uncertainty in the representation of the surface fluxes by estimating the roughness length with the augmented state approach. The goal is to thereby improve the simulation of convective clouds in the COSMO model.

In section 4.1 we introduce the COSMO model and explain the role of the roughness length in context of the model equations. Then we discuss the observations that are assimilated and used for verification in section 4.2, followed by a description of the data assimilation system KENDA, which is operational at the German Weather Service, in section 4.3. As discussed in section 1.3.2 and again in section 3.2.2, a dynamical model for the roughness length is needed to successfully estimate it with the augmented state approach. Our choice for this model, along with a description of the different weather regimes of our test period and the experiment setup are described in section 4.4. The model equivalents (i.e. estimates of the true state of the atmosphere in observation space) of our experiments are verified against SEVIRI observations and radar reflectivity to measure the quality of the representation of clouds and precipitation respectively. The results are analyzed in section 4.5, followed by a conclusion in section 4.6. This chapter is completed with a critical discussion of possible improvements and suggestions for future work in section 4.7.

## 4.1 COSMO

The COSMO-Model is a non-hydrostatic limited-area numerical atmospheric prediction model which is developed and maintained by the national weather services of the COSMO consortium for operational weather forecasting as well as research. Its dynamical core  $\mathcal{D}(\mathbf{x})$  is based on the primitive thermo-hydrodynamical equations describing compressible flow in a moist atmosphere, derived from the basic conservation laws for mass, heat, momentum and energy. Its practical formulation consists of a set of prognostic equations describing wind  $u, v, w$ , pressure perturbation  $p'$ , temperature  $T$ , total air density  $\rho$  and mass fraction of water in different phases  $q^v, q^l, q^f$ . At the German Weather Service the convective scale configuration COSMO-DE (Baldauf et al. (2011)) is employed, which covers Germany and parts of neighboring countries. The horizontal grid spacing is 2.8 km, allowing deep convection to be resolved partially. The 50 hybrid vertical layers extend from the surface to a height of 22 km ( $\sim 40$  hPa) and gradually shift from terrain-following at the bottom to horizontally flat at the top. Lateral boundary conditions are provided twice a day by the prediction system of the operational global model ICON, which runs globally with a resolution of 40 km for ensemble members and 20 km for the deterministic run, but over Europe the resolution is increased to 13 km and 6.5 km respectively. The boundary conditions at the ground are provided by the multilayer soil model TERRA (Doms et al., 2011).

### 4.1.1 Parameterizations

While the dynamical core equations are capable of representing key aspects of atmospheric flow, including deep convection in part, the numerical grid is too coarse to resolve all relevant processes explicitly. These unresolved physical processes, illustrated in Figure 4.1.1, drive heat and momentum budgets at the grid scale in the form of radiation, convection and diffusion. Representing them is crucial for achieving predictive skill, motivating the need for parameterization schemes. By design, these schemes are expressed in terms of grid-scale variables and appear as additive terms in the dynamical core equations.

The main approach for developing parameterizations is relying on the assumption that the mean flow varies considerably slower in time and space than its deviations, allowing for scale separation  $\psi = \bar{\psi} + \psi'$ . Here  $\psi$  is any variable of the flow,  $\bar{\psi}$  its mean over the corresponding grid-box and  $\psi'$  its subgrid-scale perturbation. The contribution of the mean values  $\bar{\psi}$  on the flow is resolved by the dynamical core  $\mathcal{D}(\mathbf{x})$ , but the significant contributions of the subgrid-scale perturbations  $\psi'$  are predicted by a parametrization scheme  $\mathcal{P}(\mathbf{x}, \theta)$ , which depends on many unknowns  $\theta$  in the form of external forcing or tuning parameters. Combining the dynamical core and the parameterization, the full numerical weather prediction model can be summarized as

$$\mathbf{x}_{t+1} = \mathcal{D}(\mathbf{x}_t + \Delta\mathbf{x}_p) \quad (4.1.1)$$

where

$$\Delta\mathbf{x}_p = \mathcal{P}(\mathbf{x}_t, \theta) \Delta t \quad (4.1.2)$$

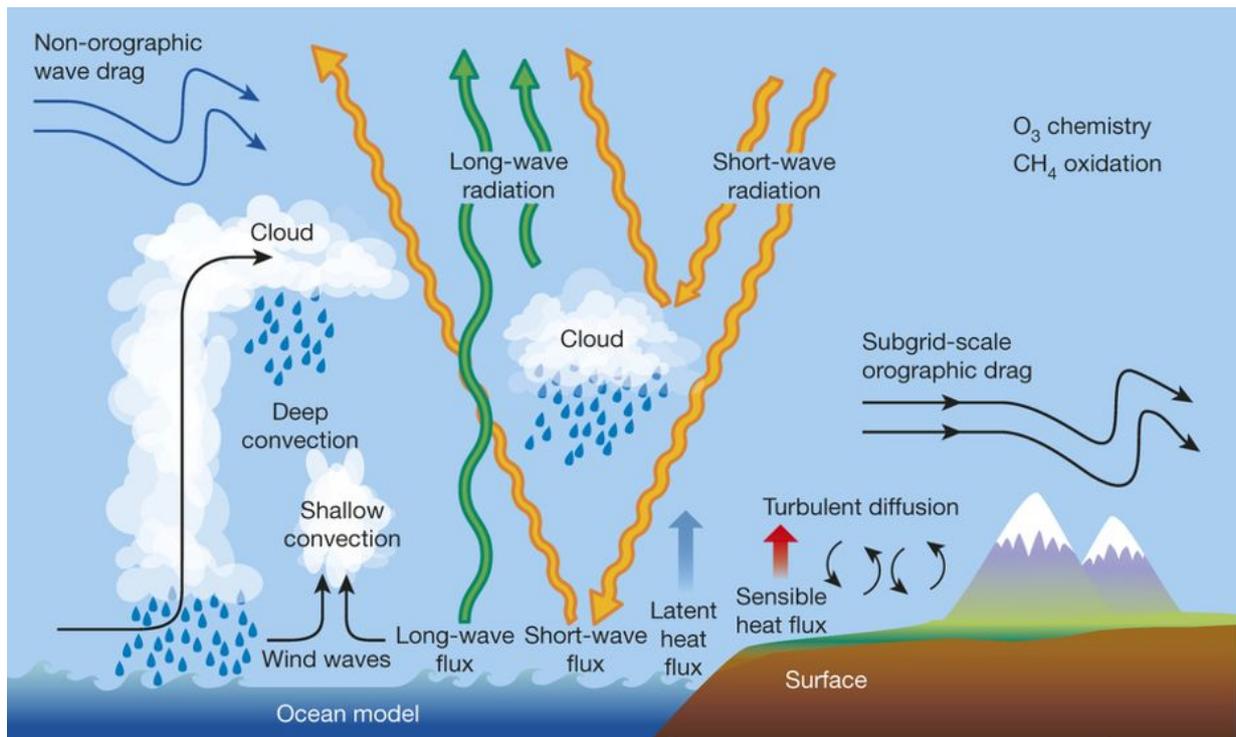


Figure 4.1.1: Illustration of parametrized physical processes. Figure adopted from Bauer Peter et al. (2015).

is the estimated subgrid-scale contribution to the grid-scale flow. A key difficulty for model developers is that the subgrid processes, including their mutual interaction, are not fully understood. In addition, complexifying the physics of a parametrization typically results in replacing or even adding unknown parameters to the scheme. In practice these free parameters are tuned manually to obtain the desired result, usually the best averaged (climatological) fit to observations or high resolution model runs.

Many physical processes are parametrized in COSMO. In particular, *shallow convection* is based on the Tiedtke scheme (Tiedtke, 1989), *clouds* are represented using a convection-allowing Lin-Farley-Orville-type one moment bulk microphysical scheme, including cloud droplets, cloud ice, rain, snow and graupel (Lin et al., 1983; Reinhardt and Seifert, 2006), the heating rate caused by *radiation* is computed according to Ritter and Geleyn (1992), subgrid-scale *turbulence* is parametrized based on the turbulent kinetic energy (TKE) equation described in Raschendorfer (2001) using parametrized *surface fluxes* as lower bound. The roughness length influences the dynamical flow via the surface fluxes and is further described in the next sections.

### 4.1.2 Subgrid-scale turbulence parametrization

Turbulent fluxes provide an exchange of momentum, heat and humidity between the earth's surface and the free atmosphere via the planetary boundary layer. Representing their effect well is vital for a successful numerical simulation of atmospheric flows. The mean contribution of the turbulent flux of momentum  $T$ , sensible heat  $H$ , and moisture  $F^v$  to the atmospheric flow appears in the prognostic equation for wind, temperature and mass fraction of water vapor as source terms  $M_u, M_v, M_w, M_T, M_{q^v}$  respectively. These source terms are functions of the prognostic variables and describe the amount of mixing that occurs due to subgrid-scale turbulence. They are most relevant in the boundary layer and are negligible at higher altitudes where geostrophic balance dominates. Since the vertical scales of motion are much smaller than the horizontal in the boundary layer, only the effect of vertical mixing on horizontal flows is considered. All horizontal fluxes are assumed negligible.

The parametrization of turbulent fluxes is based on the K-theory, which relates the subgrid-scale flux to the gradient of the corresponding variable and a diffusion coefficient for transport. The divergence of these fluxes yields their mean contribution to the flow:

$$M_u = \frac{\partial P_u}{\partial z} \quad (4.1.3a)$$

$$M_v = \frac{\partial P_v}{\partial z} \quad (4.1.3b)$$

$$M_w = 0 \quad (4.1.3c)$$

$$M_T = \frac{\partial H_T}{\partial z} \quad (4.1.3d)$$

$$M_{q^v} = \frac{\partial F_{q^v}}{\partial z} \quad (4.1.3e)$$

where  $P_u, P_v, H_T$  and  $F_{q^v}$  are the kinematic fluxes of momentum, heat and moisture respectively. A lower bound for the turbulent fluxes is provided by surface fluxes.

### 4.1.3 Surface Fluxes

Surface fluxes provide a coupling between the soil and the atmosphere. For convenience, the surface layer is subdivided into two layers. The lowest layer, extending from the ground to the aerodynamical roughness length  $z_0$ , is where roughness elements like trees and buildings disturb the logarithmic vertical wind profile. This layer is used to parametrize feedback between the soil and the surface. The layer above, called the Prandtl-layer, extends from  $z_0$  to the

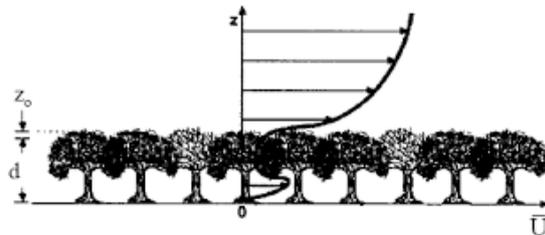


Figure 4.1.2: Illustration of flow over forest canopy showing wind speed as a function of height. Figure adopted from Stull (1988).

height of the lowest grid level above the ground  $h$ . Within this layer it is assumed that the mean wind speed increases logarithmically with height (see Figure 4.1.2). The fluxes in the Prandtl-layer vary by less than 10% of their magnitude with height, allowing for a constant flux approximation. These constant surface fluxes serve as the lower bound for  $P_u$ ,  $P_v$  (momentum),  $H_T$  (heat) and  $F_{q^v}$  (water vapor) in equation (4.1.3), and are parameterized by a drag-law formulation

$$P_u^s = C_m \|\mathbf{u}(h)\|_2 (u(h) - u_{z_0}) \quad (4.1.4a)$$

$$P_v^s = C_m \|\mathbf{u}(h)\|_2 (v(h) - v_{z_0}) \quad (4.1.4b)$$

$$H_T^s = C_h \|\mathbf{u}(h)\|_2 (T(h) - T_{z_0}) \quad (4.1.4c)$$

$$F_{q^v}^s = C_q \|\mathbf{u}(h)\|_2 (q^v - q_{z_0}^v), \quad (4.1.4d)$$

where  $C_m$ ,  $C_h$  and  $C_q$  are the transfer coefficients for momentum, turbulent heat and moisture in the Prandtl-layer and  $\|\mathbf{u}(z)\|_2 = \sqrt{(\mathbf{u}(z)^2 + \mathbf{v}(z)^2)}$  is the wind speed. The values  $u_{z_0}$ ,  $v_{z_0}$ ,  $T_{z_0}$  and  $q_{z_0}^v$  are the horizontal wind, temperature and specific humidity at height  $z_0$  and are provided by the soil-surface coupling parametrization. Note that by definition of roughness length the wind is zero at height  $z_0$ , i.e.  $u_{z_0} = v_{z_0} = 0$ . It is assumed that  $C_h = C_q$ .

As suggested by Figure 4.1.2, the wind speed relates logarithmically to height. A similar empirical relation exists for heat. Using similarity theory, under statically neutral conditions, the following dimensionless relations hold:

$$\frac{\|\mathbf{u}(h)\|_2}{u_*} = \frac{1}{\kappa} \ln \left( \frac{h}{z_0} \right) \quad (4.1.5a)$$

$$\frac{T(h) - T_{z_0}}{\theta_*} = \frac{1}{\kappa} \ln \left( \frac{h}{z_h} \right), \quad (4.1.5b)$$

where the friction velocity  $u_*$  and the surface-layer temperature scale  $\theta_*$  are defined as

$$u_*^2 = \sqrt{(P_u^s)^2 + (P_v^s)^2} \quad (4.1.6a)$$

$$\theta_* = \frac{H_T^s}{u_*}. \quad (4.1.6b)$$

The value  $z_h = \min(z_0, z_{h_{max}})$  is introduced to avoid excessive heat exchange over rough terrain. Using (4.1.4) and (4.1.5) the transfer coefficients for neutral conditions are

$$C_{m,n} = \left( \frac{u_*}{\|\mathbf{u}\|_2} \right)^2 = \kappa^2 \ln \left( \frac{h}{z_0} \right)^{-2} \quad (4.1.7a)$$

$$C_{h,n} = \frac{\theta_*}{T(h) - T_{z_0}} \frac{u_*}{\|\mathbf{u}(h)\|_2} = \kappa^2 \ln \left( \frac{h}{z_0} \right)^{-1} \ln \left( \frac{h}{z_h} \right)^{-1}. \quad (4.1.7b)$$

To generalize the computation of the transfer coefficients to non-neutral conditions, (4.1.7) is multiplied by stability functions  $f_m(Ri_B, h/z_0)$  and  $f_h(Ri_B, h/z_0, h/z_h)$  respectively,

where  $Ri_B$  is the bulk Richardson number indicating the static stability condition, yielding

$$C_m = C_{m,n} f_m(Ri_B, h/z_0) \quad (4.1.8a)$$

$$C_h = C_{h,n} f_h(Ri_B, h/z_0, h/z_h) \quad (4.1.8b)$$

Equations (4.1.8) and (4.1.5) confirm our intuition that a lower roughness length implies less exchange between the surface and the atmosphere, but also stronger wind near the ground.

#### 4.1.4 Representation of roughness length

The parametrization of the roughness length  $z_0$  depends on subgrid-scale orography  $z_{0,or}$  and land use  $z_{0,lu}$ . In COSMO only the mean and variance of sub-grid scale orography is taken into account,  $z_{0,or} = a_0 \sigma^2 \arctan(\Delta x/b_0)$ , where  $\sigma^2$  is the variance of the subgrid-scale orography scaled by its mean,  $a_0 = 10^{-5}m$  and  $b_0 = 2.5m$ . The roughness lengths  $z_{0,i}$  corresponding to the different land uses (grass, crops, forest, urban, etc.) within one grid-box of area  $A$ , are logarithmically averaged weighted by their respective areas  $A_i$ :

$$z_{0,lu} = h \exp \left( - \frac{A}{\sum_{i=1}^{i=I} A_i / \ln \left( \frac{h}{z_{0,i}} \right)} \right). \quad (4.1.9)$$

The final roughness length value used in COSMO is the sum of the two contributions:

$$z_0 = z_{0,or} + z_{0,lu}. \quad (4.1.10)$$

The resulting two dimensional constant roughness length field that is used in COSMO is visualized in Figure 4.1.3.

## 4.2 Observations

There are many relevant observations for weather forecasting. However, due to reasons mentioned in section 1.2.1, only the observations discussed in 4.2.1 and 4.2.2 are assimilated at the German Weather Service for the regional weather forecast.

### 4.2.1 Conventional observations

Conventional observations measure prognostic variables of COSMO, such as temperature, wind, pressure and relative humidity. Synoptic surface data (SYNOP) provides hourly measurements of surface pressure, horizontal wind at 10m above the surface, and temperature and humidity at 2m above the surface. Due to subgrid-scale orography, only surface pressure and 10m wind in the north of Germany is consistently assimilated. Other surface measurements are usually rejected because they differ too much from the interpolated

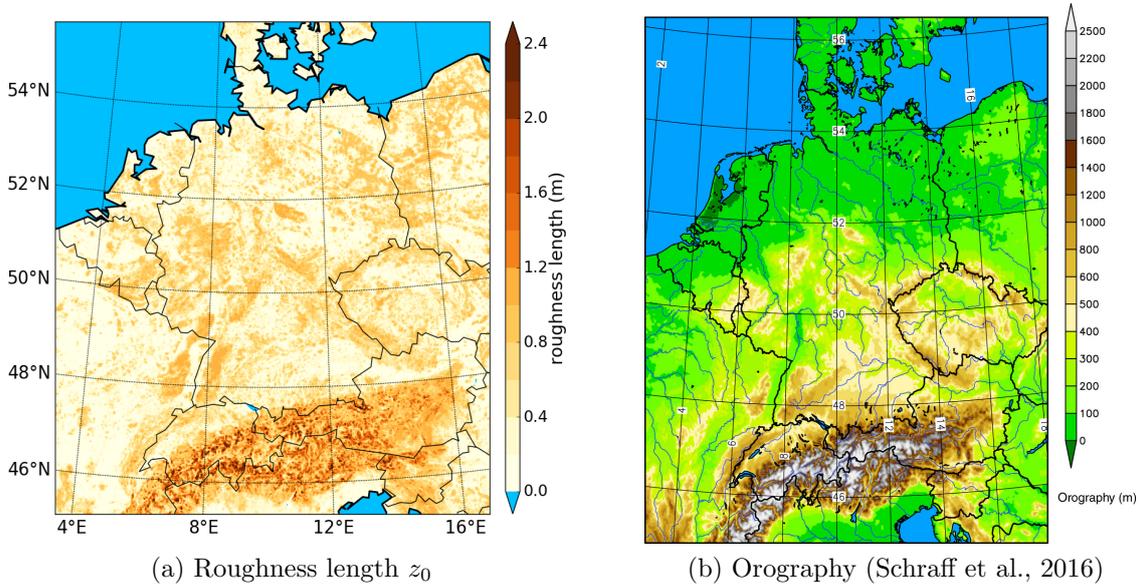


Figure 4.1.3: The roughness length (left) and the orography (right) of the operational COSMO-DE domain with 2.8 km horizontal resolution. The domain size is about 1170 km  $\times$  1280 km.

model output, see Figure 4.2.1. Upper air data such as radiosondes (TEMP), wind profilers (PROF) and aircraft reports (AIREP) do not share this problem and are therefore assimilated when available. Radiosondes are typically launched at 00:00 and 12:00 UTC with some in addition launching at 06:00 and 18:00 UTC. PROF measurements of horizontal wind are available approximately half-hourly during the whole day and the majority of AIREP horizontal wind and temperature data are collected during the daytime. Since 2017 wind and temperature observations obtained from aircrafts within the range of airport radars in the surveillance Mode-S are also assimilated (Lange and Janjić, 2016).

## 4.2.2 Radar data

Aside from conventional observations, the German Weather Service operates a network of 17 C-band Doppler radars over the COSMO-DE domain, which provides reflectivity and radial wind measurements every 5 minutes. For now, this data is used to derive precipitation rates, which are assimilated according to the latent heat nudging technique (Stephan et al., 2008), where for each ensemble member the temperature increment is increased based on the scaling between observed and predicted precipitation to create up-draft motion (and eventually precipitation) where necessary. In the future, however, it is planned to use the LETKF to assimilate radar reflectivity, as is done with conventional observations. An efficient radar forward operator EMVORADO (Zeng et al., 2014, 2016) that converts a model state to its reflectivity equivalent is available, but some obstacles remain as indicated in Chapter 3. For example, the LETKF is not designed to handle

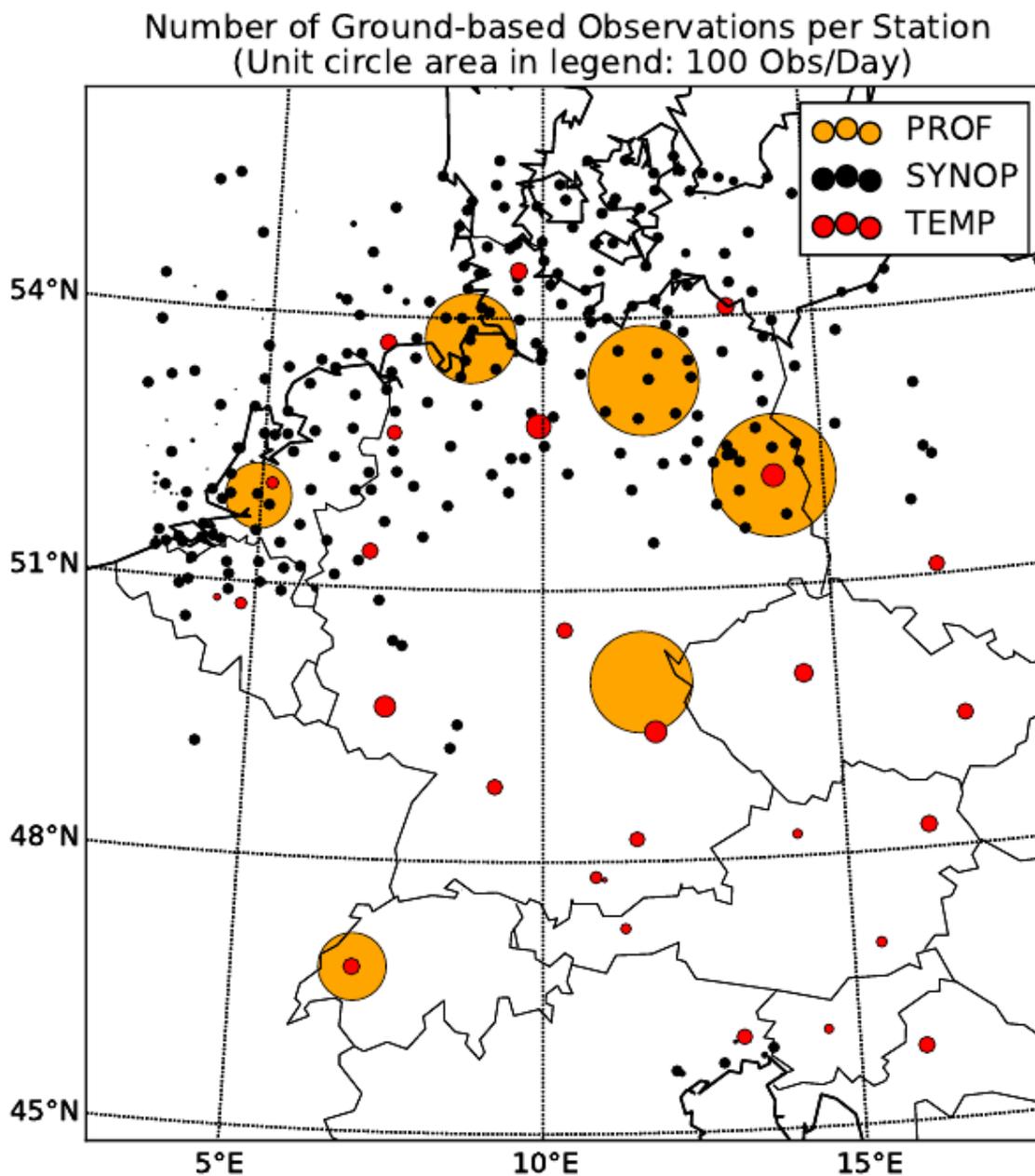


Figure 4.2.1: Domain extent of COSMO-DE with conventional observations. The circles indicate positions of the surface-based stations. The areas of the circles correspond to the average daily number of single observations of the wind variable per station. The average number of observations per day are 11 851 PROF, 5813 SYNOP, and 1571 TEMP. Figure adopted from Lange and Janjić (2016).

the non-Gaussianity and non-linearity involved with the assimilation of radar reflectivity. Problems in particular arise when observations indicate a precipitative event, which no ensemble member predicted. This situation happens frequently due to the chaotic and nonlinear nature of convection as well as undersampling. Nevertheless, experiments conducted by Bick et al. (2016) indicate high potential for assimilation of radar reflectivity with the LETKF. It was found that temporal thinning is necessary to reduce temporally correlated observation errors. Therefore only the latest available batch of reflectivity measurements are assimilated. In addition, spatial superobbing is applied to reduce spatially correlated errors, which essentially combines several observations into one “super observation”. More details and an illustrative example for superobbing of radar reflectivity can be found in Bick et al. (2016). Also, all reflectivities smaller than 5 dBZ are assimilated as no-reflectivity information.

### 4.2.3 Satellite data

Assimilation of satellite data has played an important role in the steady increase of forecast skill over the past decades. Nonetheless, this rich source of observational data is far from being fully exploited, especially for convective scale NWP, where cloud related information is crucial. So far, the use of satellite data has mostly been limited to clear sky radiances, as it is not straightforward how to handle radiances that have been effected by clouds. Yet, it is cloud affected radiances that are most relevant to convective scale data assimilation. In particular, visible satellite images obtained from the METOSAT SEVIRI instrument contain high-resolution information about clouds, which can be used for data assimilation purposes as well as model output verification. For this application, Scheck et al. (2016) recently proposed the forward operator based on the Method for Fast Satellite Image Synthesis (MFASIS) to generate synthetic images from the COSMO-DE output that is fast and compact enough for operational use. Visible satellite images are available every 5 minutes over Europe and every 15 minutes elsewhere. Since applying this forward operator to assimilate visible satellite images is currently in research mode, we here use it for verification purposes only.

## 4.3 KENDA

The data assimilation system operational at the German Weather Service is based on the LETKF (Hunt et al., 2007), which is a localized version of the ETKF introduced in Algorithm 2.1.3. To deal with undersampling, model and observation errors and computational limits, different techniques are applied in addition to the standard LETKF equations, as discussed in this section.

### 4.3.1 Localization

Localization is done by applying for each grid-point  $i$  a Gaspari-Cohn weight function  $L(d_i)$  to the observations  $y \in \mathbb{R}^m$ , where  $d_i \in \mathbb{R}^m$  is the Euclidean distance between the observations and grid-point  $i$  (Gaspari and Cohn, 1999). As a result, the state vector is analyzed at each grid-point independently and each state element is influenced only by observations within a certain radius. The size of this radius is a tunable parameter that should depend on physical knowledge and mathematically substantiated limitations. For example, the degrees of freedom for a state element update is limited by the ensemble size. Consequently, a maximum amount of information in the form of observations that can be processed effectively by the LETKF exists, rendering any additional observations obsolete. It is important to assimilate the data on the correct scale, i.e. the localization radius should be sufficiently large to recover more than just small scale fluctuations from the data, yet small enough to successfully avoid spurious correlations that arise from undersampling. Taking these trade-offs into account, for 40 ensemble members the localization radius is set per grid-point such that 100 observations are assimilated, subject to a lower and upper limit of 50 km and 100 km respectively. For a graphical view of the resulting localization radii we refer to Lange and Janjić (2016). Due to the high and constant density of radar data, a fixed localization radius of 16km is set for this particular observation type.

### 4.3.2 Spread inflation techniques

Representation of correlated model error is crucial to avoid filter divergence. The background error covariance matrix  $\mathbf{P}^f$  determines how the observation weights effect the analysis increment. On average these error covariances are underestimated due to lack of model error representation. To deal with this, an adaptive multiplicative covariance inflation scheme was implemented in KENDA (Houtekamer et al., 2005). This scheme relies on the statistical property that the error covariance of the innovation should equal the sum of  $\mathbf{R}$  and  $\mathbf{P}^f$  in observation space. This property is compressed to the following natural multiplicative inflation factor:

$$\rho = \frac{(\mathbf{y} - \mathbf{h}(\mathbf{x}^f))^T (\mathbf{y} - \mathbf{h}(\mathbf{x}^f)) - \text{trace}(\mathbf{R})}{\text{trace}(\mathbf{Y}^f \mathbf{Y}^{fT})}. \quad (4.3.1)$$

Note that adaptive multiplicative covariance inflation also takes sampling error into account.

Additional inflation in the form a relaxation to prior perturbations (RTPP, Zhang et al., 2004) is used operationally in the KENDA system. Here, a weighted average of the observation weights  $\mathbf{W}^a$  and the identity matrix is used to update the first guess ensemble. The effect is that the analysis ensemble perturbations are relaxed towards those of the first guess ensemble, which has more spread. A fixed weight of only 25% is given to  $\mathbf{W}^a$ . Note that in contrast to the adaptive multiplicative covariance inflation scheme, RTPP does not effect the ensemble mean, only the ensemble perturbations.

A different approach to represent model error that is available in KENDA is by additive inflation. Here model error samples obtained from climatological systems are used to represent large scale model errors. These samples are added *a posteriori* to the analysis ensemble (Rhodin et al. (2013)). The intended effect is the increase of spread in a directed manner.

### 4.3.3 Observation errors

Observations errors arise from the instrument, the forward operator and unresolved scales, i.e. point measurements are subject to small scale perturbations that models cannot resolve. The total error is therefore difficult to define. In addition, for practical reasons observations errors are assumed temporally and spatially uncorrelated, although this does not necessarily reflect reality, as for example for radar data. To compensate for any correlations, the error variance needs to be increased. Altogether, the diagonal values of the observation error covariance matrix  $\mathbf{R}$  are tuned rather than derived from physical knowledge of each error source separately. This tuning is done based on Desroziers statistics (Desroziers et al., 2005a,b), where the theoretical relations between  $\mathbf{R}$  and the statistics of the background and analysis innovations are exploited. The resulting average observation errors for surface pressure are about 0.5 hPa and those assumed for upper air measurements are given in Table 4.3.1. For radar reflectivity an error of 10 dBz is assumed.

Level (hPa)	Wind ( $\text{m s}^{-1}$ )	Temperature (K)	Rel. humidity (%)
300	2.38	0.56	14
400	2.08	0.54	13
500	1.92	0.55	13
700	1.89	0.68	12
850	2.00	0.85	13
1000	1.95	1.07	9

Table 4.3.1: Upper air observational error standard deviations derived from Desroziers statistics for wind, temperature and relative humidity per pressure level. Adopted from Schraff et al. (2016).

### 4.3.4 Implementation notes

The KENDA system calculates the analysis weights in observation space on a coarser grid to reduce computational costs. The weights are then interpolated to the model grid before converting the solution to full model space (Yang et al., 2009). Without significant loss of accuracy, a horizontal coarsening factor of 3 is applied and the number of vertical layers is reduced from 50 to 30, leaving only 6.67% of the grid points to be analyzed in observation space. Furthermore, techniques like saturation adjustment and hydrostatic balancing are applied to the computed analysis, to ensure well balanced initial conditions (Rhodin et al.,

2013). Finally, the prognostic variables of TKE and the mass fraction of water in its different phases are left unchanged, because the LETKF is not yet consistently producing well balanced, physical initial conditions for these variables.

## 4.4 Experiment setup

The experiments are conducted in a quasi-operational setting using a basic cycling environment (BACY) developed by the German Weather Service. Our setting differs from the operational setting in the following ways. We assimilate radar reflectivity directly with the LETKF, instead of the nudging technique. We apply only additive inflation to regulate the spread, motivated by the results presented in Zeng et al. (2019). Finally, the prognostic variables of TKE and the mass fraction of water in its different phases are updated along with the standard variables by the LETKF.

We focus on the period from the 28<sup>th</sup> of May to the 9<sup>th</sup> of June in 2016, which we split in two periods of 6 days, motivated by the weather situation (see section 4.4.2). We use a 40 member ensemble for the hourly cycling. Every hour from 6 UTC to 15 UTC we start a 6 hour forecast using 20 ensemble members, so that for each of the two weeks we have a total of 60 forecasts of 20 members available for verification. Note that our choice of forecast start times stems from the availability of visible satellite data.

For estimation of the roughness length ( $z_0$ ) we use the augmented state approach (Jazwinski, 1970; Evensen, 2009). Since  $z_0$  is two dimensional, horizontal localization is applied in exactly the same way as for the state variables.

### 4.4.1 Dynamical model for roughness length

Due to the static design of the roughness length parameter  $z_0$  in COSMO, a model propagation cannot provide any direct information about the uncertainty of  $z_0$ . Instead we have to explicitly model the uncertainty as a heuristic exercise. We choose the same approach as for the idealized setup experiments discussed in Chapter 3. For each ensemble member  $i$  at time  $t$ :

$$z_{0_{t,i}}^f = z_{0_{t-1,i}}^a + \mathbf{D}_{t-1} \mathbf{Q}^{\frac{1}{2}} \eta \quad (4.4.1)$$

where  $\mathbf{D}_{t-1}$  is a diagonal matrix that controls the ensemble spread of the roughness length,  $\mathbf{Q}^{\frac{1}{2}}$  is the error correlation matrix that specifies the correlations within the roughness length field and  $\eta \sim \mathcal{N}(0, \mathbf{I})$  determines the random realization of the stochastic model. Ideally this stochastic uncertainty model should take into account all targeted unrepresented model errors, in this case the roughness length itself and the surface fluxes. As there is no evident way to estimate this accumulated uncertainty, we follow what was done in Chapter 3 and choose  $\mathbf{D}_{t-1}$  such that the spread of  $z_{0_{t,i}}^f$  equals 25% of the original roughness length value implemented in COSMO-DE. It should be stressed that  $\mathbf{D}_{t-1}$  is a tunable matrix and may effect the experiments significantly, as it essentially determines the amplitude of the analysis increments of the roughness length. We set  $\mathbf{Q}^{\frac{1}{2}}$  such that the correlations are an approximate Gaussian function of the spatial distance between roughness length

elements. To test the effect of the assumed correlation length scale  $dx$  of the model error, we respectively do experiments with  $dx = 0$ ,  $dx = 5$  and  $dx = 25$  grid points.

The choice of our setup then implies that we assume that the sum of all targeted model errors projected onto the roughness length is Gaussian with correlation length scale of either 0, 5, or 25 grid points, where 5 grid points is the effective resolution of the model. Furthermore, it is assumed that the absolute errors are flow dependent with a constant standard deviation of 25% of the climatological mean, which we assume is the original roughness length parameter value. We clip the roughness length analysis values to the lower bound of 0.0002 before the dynamical model is applied. After the dynamical model is applied we clip again to a value of 0.0001, thereby ensuring a spread larger than zero.

We run a set of 5 experiments: a reference run referred to as *ref*, where  $z_0$  is not estimated,  $dx = 0$ ,  $dx = 5$  and  $dx = 25$  referring to experiments where  $z_0$  is estimated using the dynamical model for the respective correlation length scales, and, to determine whether any change with respect to *ref* is the result of parameter estimation or simply the adding of noise, a final experiment is conducted referred to as *perturb*, where  $z_0$  is perturbed according to the dynamical model with correlation length scale  $dx = 5$ , but the mean value is left unchanged, i.e.  $z_0$  is not estimated.

#### 4.4.2 Weather situation

Our experiments are conducted for subsets of the two week period from 27th of May to the 9th of June, when a large part of Europe was affected by a sequence of severe convective storms. These storms were the result of the combination of high moisture content, low thermal stability, weak wind speed, and large scale lifting caused by surface lows, i.e. pressure minima at the Earth's surface. These conditions persisted for two weeks due to an atmospheric blocking in the form of a large-scale ridge spreading over the North Atlantic and northern Europe, hampering mass exchange. Western and southern Germany remained under the influence of low pressure with moist and warm air, while drier air gradually prevailed in the northeast. During the first week of this blocking event, moist and warm air was advected ahead of a deep trough northeastwards towards central Europe, causing synoptic lifting. During the second week moisture was maintained mainly by evapotranspiration from local sources and advection from nearby countries. Due to the weak pressure gradient, horizontal flow was limited, causing the convective storms to be almost stationary (Piper et al., 2016), especially in the second week. The majority of the convective events during the whole period followed a typical diurnal cycle with peaks in the late afternoon (see Figure 4.4.1).

We distinguish between synoptic regimes that occurred throughout this convectively active period of two weeks by applying the convective adjustment time-scale measure ( $\tau_c$ ) introduced by Done et al. (2006) and Keil et al. (2014). The convective adjustment time-scale  $\tau_c$  is defined as the ratio of convective available potential energy over the rate of change of the convective available potential energy. It describes how fast the conditional instability (CAPE) is discharged by the release of moist convection and can be used to distinguish between synoptically driven convection and convection triggered by local processes in the

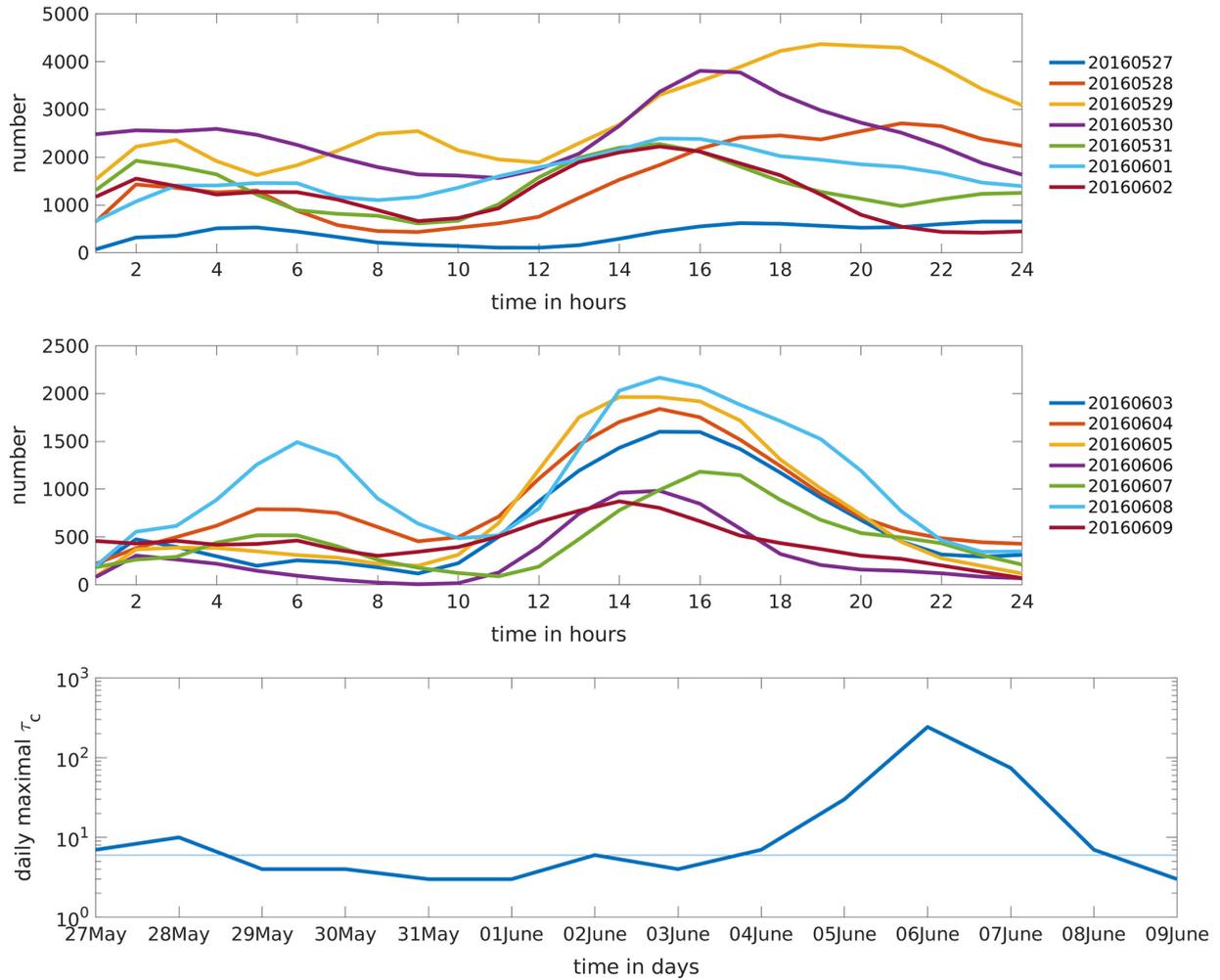


Figure 4.4.1: Hourly variations of numbers of model grid points with precipitation rate  $\geq 5.0$  mm/hour for different days in the first half (upper panel) and the second half (middle panel) of the test period. The lower panel shows the variations of daily maximal  $\tau_c$ . The  $y$ -axis is in log scale, and the horizontal line indicates the threshold value of 6 hours. All data is derived from the mean of individual ensemble forecasts produced using COSMO-DE. Picture adopted from Zeng et al. (2019).

boundary layer. In the first case, convection occurs as long as there is CAPE. As convection is a mechanism to stabilize the atmosphere, the rate of CAPE decay is of the same order as CAPE, leading to maximum  $\tau_c$  values of order 1. In the latter case, the onset of convection needs a trigger to overcome the energy barrier of convective inhibition (CIN), i.e. CAPE alone is not enough to initiate convection. As a result, CAPE is accumulated over a long time scale before convection occurs to stabilize the atmosphere, causing high values of  $\tau_c$  just before the outbreak of convection. Analogous to Keil et al. (2014), we define a day to be weak synoptically forced if  $\tau_c$  exceeds the threshold of 6 hours at least once, and strong synoptically forced otherwise. Figure 4.4.1 justifies our choice to split our experiments in two 6 days periods. From 28.05.2016 to 02.06.2016 the synoptic forcing is strong, and from 04.06.2016 to 09.06.2016 the forcing is weak, with the exception of 09.06.2016.

## 4.5 Results

As stronger synoptic forcing suggests weaker influence from subgrid-scale processes, we hypothesize that our approach is beneficial for the second week, and neutral for the first week of our test period. We therefore focus on the experiments conducted in the weakly forced weather regime, from 04.06.2016 to 10.06.2016. As we expect surface wind measurements to be the most important observations for the estimation of  $z_0$  (see Figure 4.5.1), we divide the domain in North (above 50 degrees latitude), where surface wind measurements are mostly assimilated, and South (below 50 degrees latitude), where surface wind measurements are mostly discarded. We start by analyzing the changes in the roughness length parameter and then continue to verification of atmospheric fields.

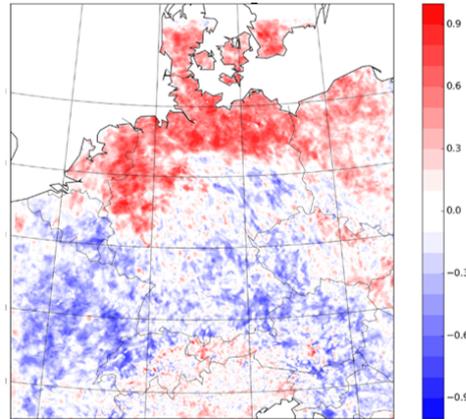


Figure 4.5.1: Snapshot of grid-point wise correlation between  $z_0$  and the U-component of the wind 10 meter above the surface for  $dx = 5$ .

### 4.5.1 Roughness length evolution

Figure 4.5.2 shows a snapshot of  $z_0$  after 6 days of hourly cycling. The spatial correlation inherited from the different dynamical models for  $z_0$  is clearly visible. For all experiments the orography, especially the Alps, remains featured and is even emphasized. In Figure 4.5.3 the spatial mean of the analysis increments for the different experiments is shown. The evolution of the mean increments clearly highlights a diurnal cycle. A natural assumption is that this is related to the diurnal cycle of the surface fluxes. Indeed, the spatial mean of the hourly differences of the momentum surface flux (dashed black line) evolve in sync with the  $z_0$  increments. The surface heat flux generally peaks earlier in the day (not

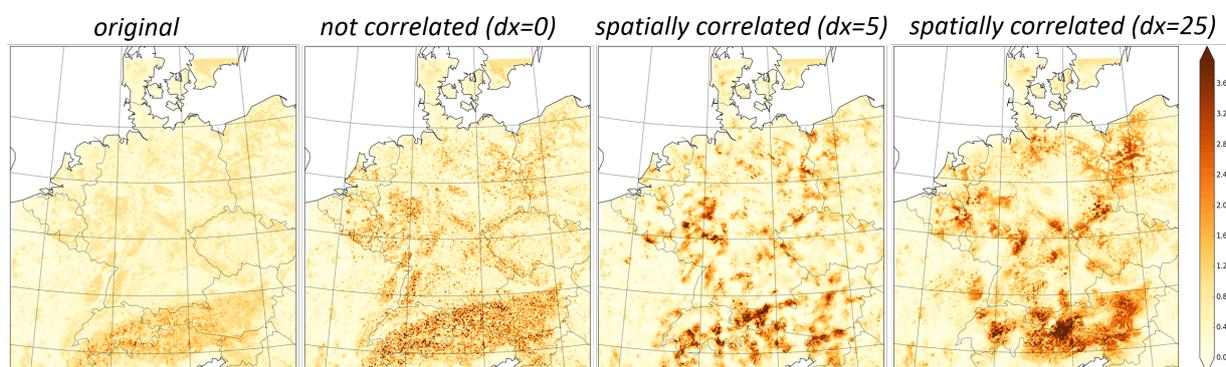


Figure 4.5.2: The deterministic roughness length parameter after 6 days of hourly cycling for *ref*,  $dx = 0$ ,  $dx = 5$  and  $dx = 25$  from left to right.

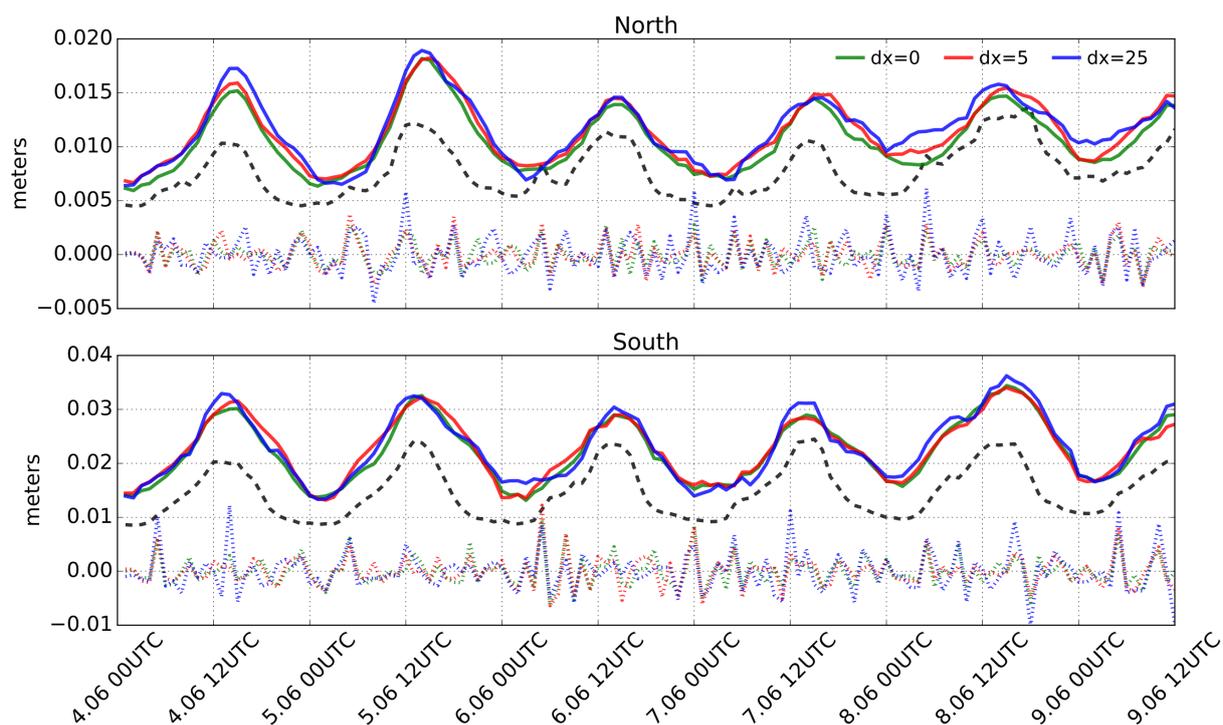


Figure 4.5.3: Absolute analysis increments averaged over space for the North (top) and the South (bottom). The increments are separated into a smoothed part (solid lines) and the remaining noise (dotted lines) using the Savitzky-Golay filter. The dashed black line is the average hourly absolute surface momentum flux differences corresponding to the reference run, normalized with some linear function for visualization purposes.

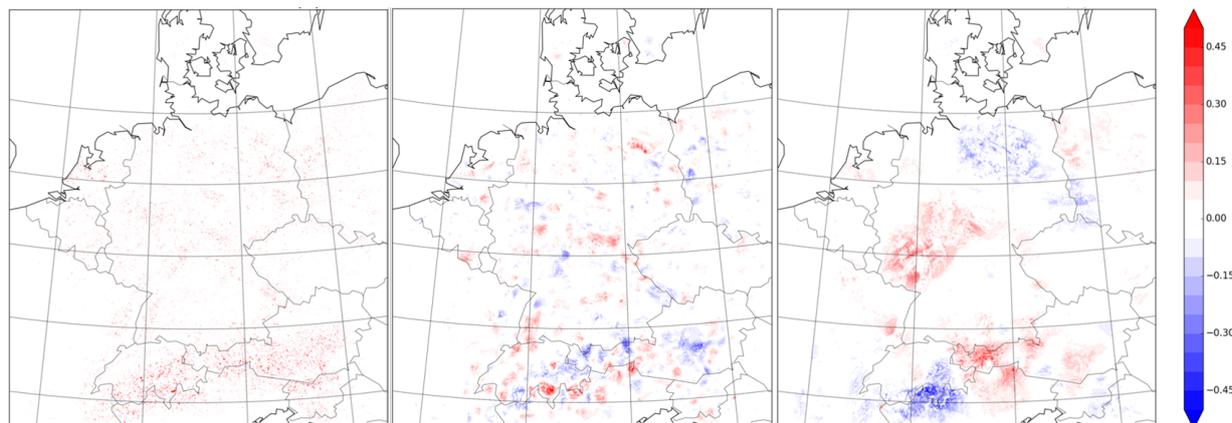


Figure 4.5.4: Snapshot (9th of June, 10 UTC) of analysis increments for  $dx = 0$  (left),  $dx = 5$  (middle),  $dx = 25$  (right).

shown). It is encouraging that the increments appear to be driven by physical processes rather than spurious correlations, at least during the day. That said, spurious correlations do influence the evolution of  $z_0$ , as can be interpreted from the noisy nature of the increments (dotted lines). Also, the increments in the South are roughly twice as large as in the North, most likely due to the influence of orography. Numerical weather prediction systems have difficulties providing accurate forecasts over mountainous terrain, because the effect of subgrid-scale orography is large and difficult to model. The model error is therefore expected to be large over the Alps, which the LETKF tries to compensate for by modifying the roughness length, leading to large increments. In addition, surface wind measurements, which are the most direct observations available for the roughness length, are not assimilated in the South, causing the roughness length to have more freedom.

From Figure 4.5.3 we can not establish a clear difference in behavior for the three dynamical models. Even histograms of the absolute increments (not shown), do not provide information on any distinction. However, when looking at the spatial distribution of the analysis increments, huge differences are found, see Figure 4.5.4. As expected, the smaller the correlation length scale of the dynamical model, the “spottier” the increments. This effect, though heavily reduced, is passed on to the surface momentum flux (Figures 4.5.5-4.5.6). Where  $dx = 25$  mainly accentuates the existing features of the reference run,  $dx = 5$  introduces lighter and darker spots of similar size as seen in the middle plot of Figure 4.5.4, and  $dx = 0$  clearly introduces higher resolution features. The statistical properties shown in Figure 4.5.7 highlights steady behavior in the North and more noisy behavior in the South, especially for  $dx = 25$ . Again, this is probably due to lack of assimilated surface wind measurements and the model error due to orography. In general  $dx = 0$  and  $dx = 5$  exhibit similar behavior. In both regions the standard deviation is steadily increasing, while the median appears to drop somewhat. The mean increases slightly, which is partially due to the imposed lower bound of 0.0002 meter. These features seem more pronounced in the South. For  $dx = 25$  however, the South exhibits different behavior than in the North. In

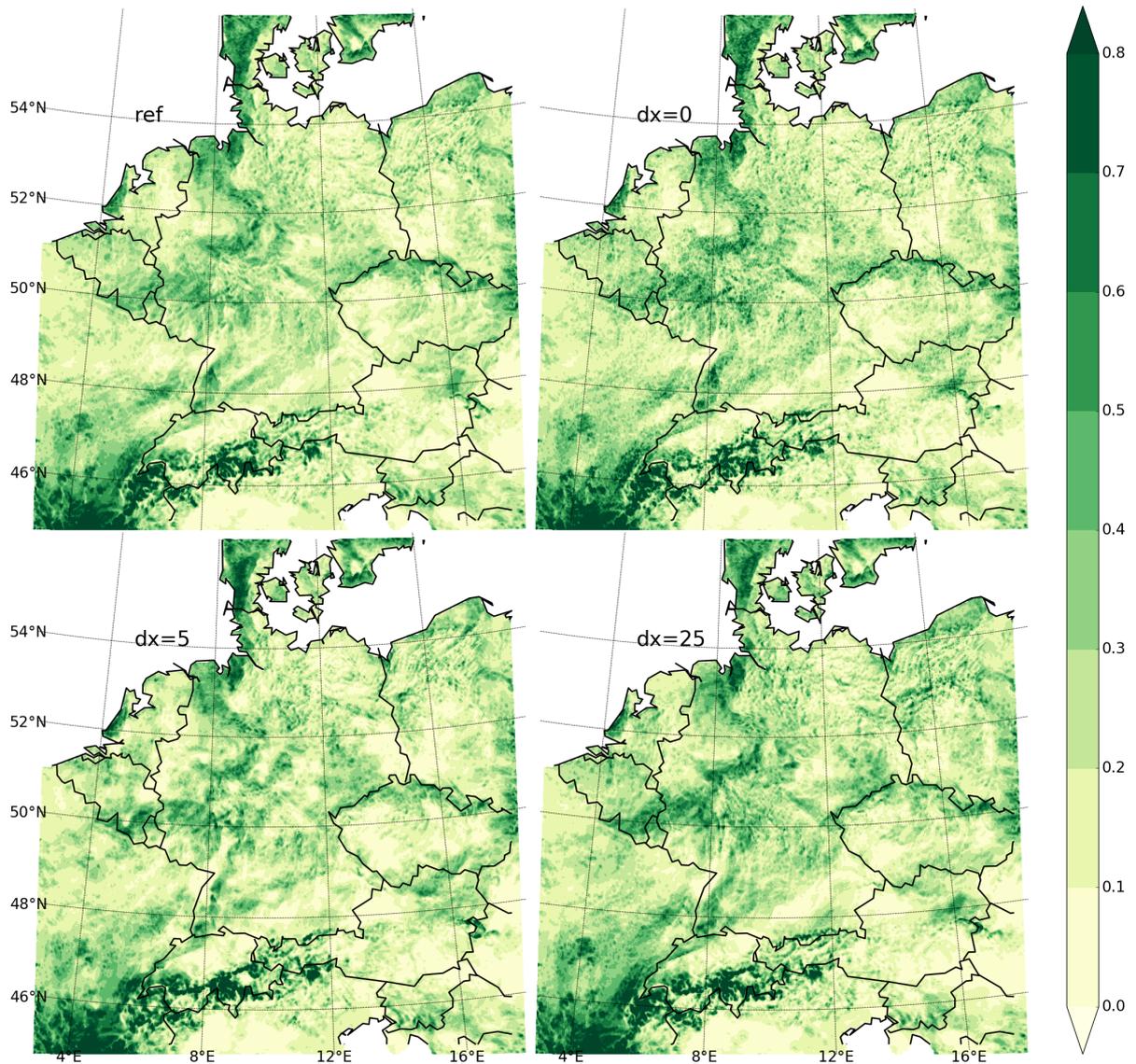


Figure 4.5.5: Snapshot (9th of June, 10 UTC) of surface momentum flux ( $\text{N}/\text{m}^2$ ) generated from the deterministic background for the different experiments. Clockwise from top left: *ref*,  $dx = 0$ ,  $dx = 5$  and  $dx = 25$ .

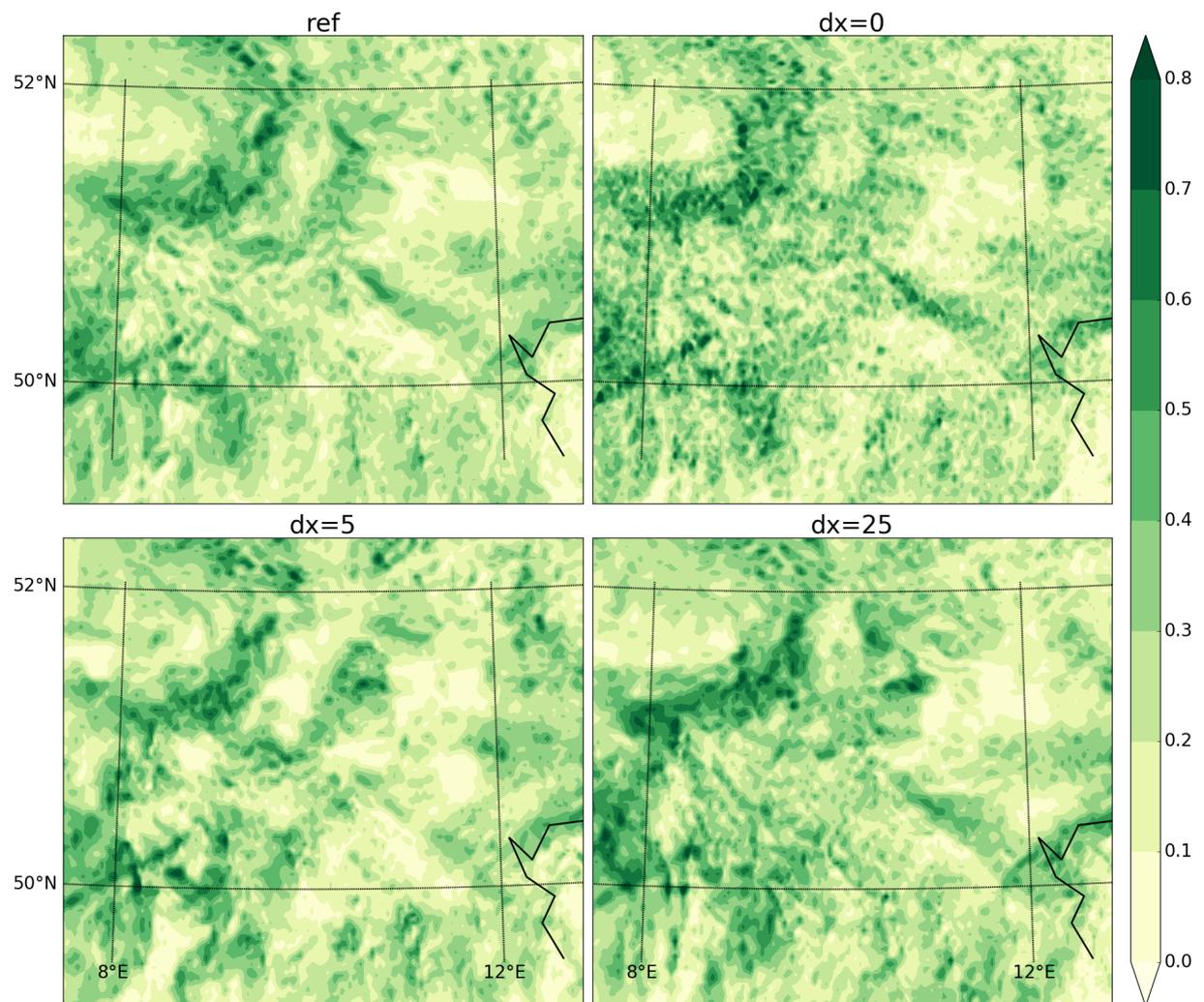


Figure 4.5.6: Figure 4.5.5 zoomed in at arbitrary location to highlight the scale of the differences in the momentum surface flux among the dynamical models.

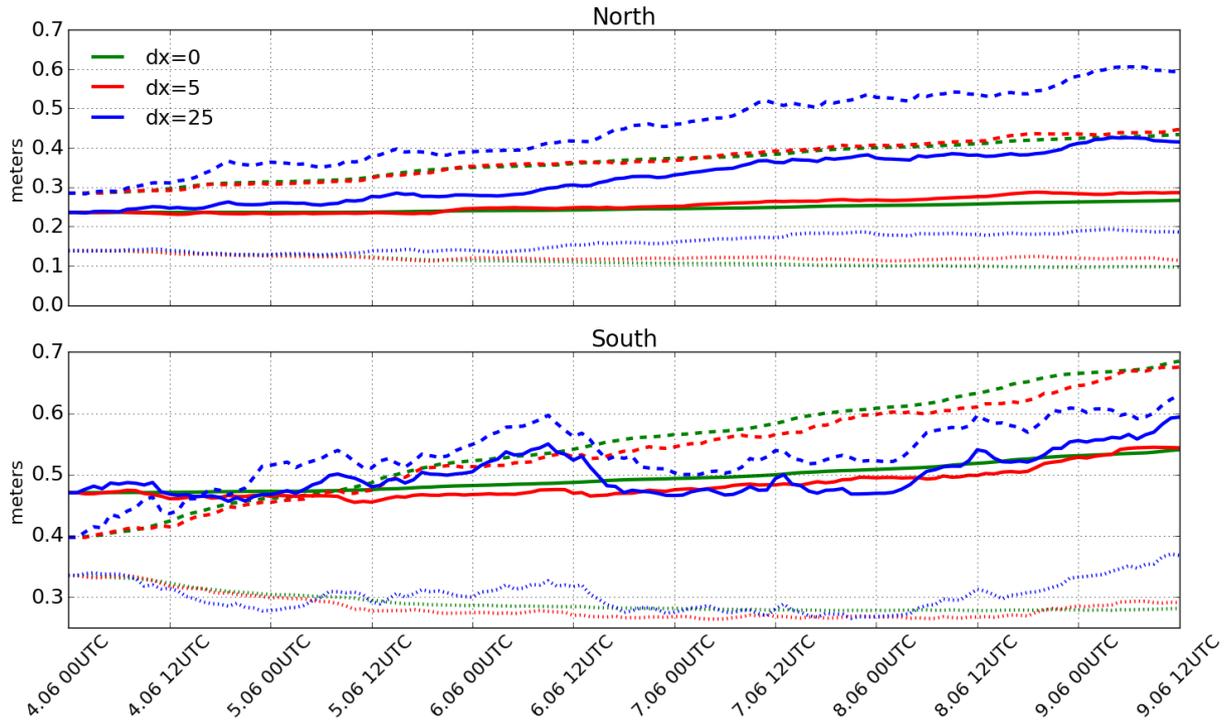


Figure 4.5.7: Evolution of statistical properties of the spatial distribution of  $z_0$ . The mean (solid), standard deviation (dashed) and median (dotted) are plotted for the North (upper plot) and the South (lower plot). Colors indicate the different experiments.

the North all statistical properties shown are higher than for the other experiments, which is not true in the South. An important note is that  $dx = 25$  introduces a large bias with respect to the original  $z_0$  in the North. Whether this is the result of an existing model error bias that should be corrected, or simply an imposed effect of the chosen dynamical model for  $z_0$ , is to be debated using the forecast scores discussed in a later section.

In general it seems the three dynamical models show very different behavior in terms of the spatial distribution of  $z_0$ , yet show very similar behavior in terms average change per cycle. Figure 4.5.7 suggests the sign of the increments differs among the experiments: a large correlation length scale appears to result in larger positive than negative increments, introducing a bias with respect to the original parameter.

## 4.5.2 Verification of atmospheric fields against observations

We first look at the average RMSE of the experiments with respect to the observations assimilated for the first guess and the analysis, see Figures 4.5.8 and 4.5.9. What is most notable, especially in the North, is that *perturb* has by far the largest overall RMSE of all experiments, including the control. We can also conclude from Figure 4.5.10 that randomly adding perturbations does not necessarily result in a larger spread. However, when the perturbations of  $z_0$  are used to allow change in the mean of  $z_0$ , the spread increases and the

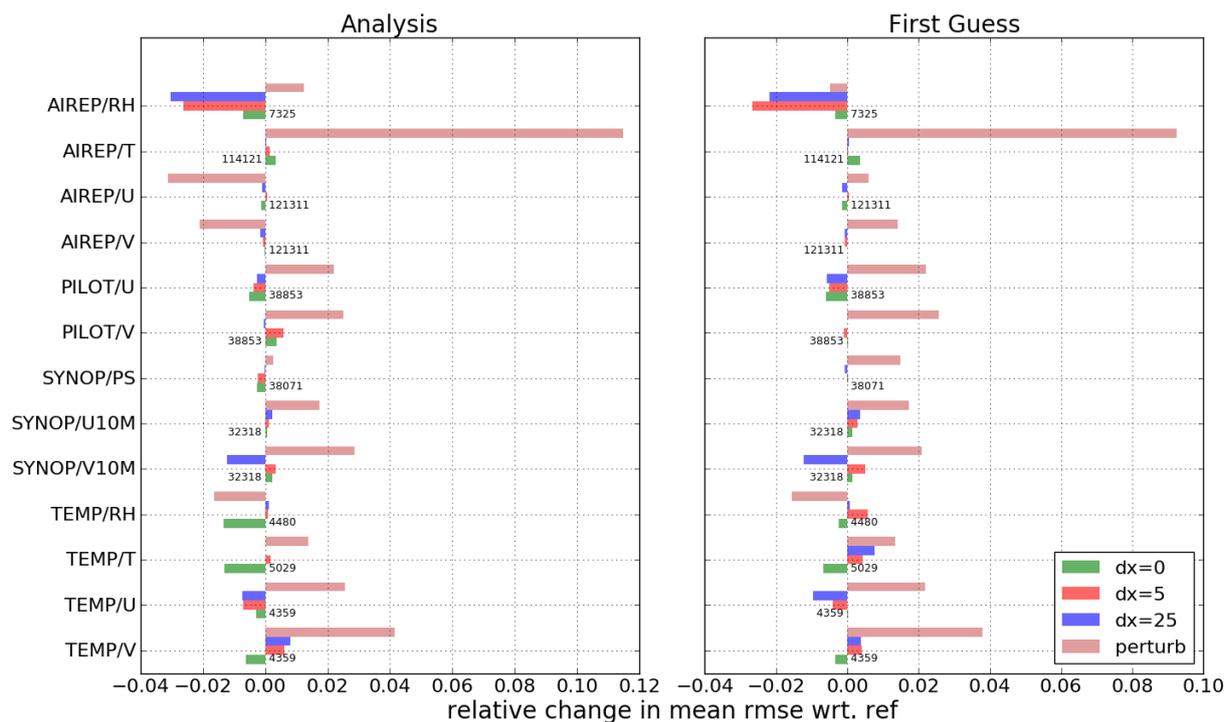


Figure 4.5.8: Relative change in mean RMSE compared to assimilated conventional observations for the Northern part of domain for the analysis (left) and 1 hour forecast (right). Numbers indicate the total number of assimilated observations.

RMSE decreases. This is not surprising since the LETKF seeks to minimize the distance between the first guess and the observations assimilated, using the additional degrees of freedom offered by the variable parameter for that purpose. However, initial conditions that are closer to the observations, do not guarantee they remain closer throughout a forecast and vice versa. We are interested in longer forecast lead times than 1 hour. Also, the quantities we are most interested in are clouds and precipitation, as they describe convective events.

#### 4.5.2.1 Reflectance

Since SEVIRI observations are not reliable over the Alps due to the possibility of snow, the domain is cut off at a latitude of approximately 47.5 degrees. Figure 4.5.12 offers a snapshot of the reflectance, which illustrates the influence that estimating  $z_0$  can have on the representation of clouds in COSMO-DE. The differences are mostly subtle, but there are areas where a clear difference can be spotted, see for example the circled area. After estimating the roughness length, both the deterministic run and the ensemble predict clouds with a reflectance larger than 0.3, where the control run does in neither. More quantitative results are presented in Figures 4.5.13-4.5.16 in the form of bias, RMSE, Fractions Skill Score (FSS), False Alarm Ratio (FAR), Equitable Threat Score (ETS) and

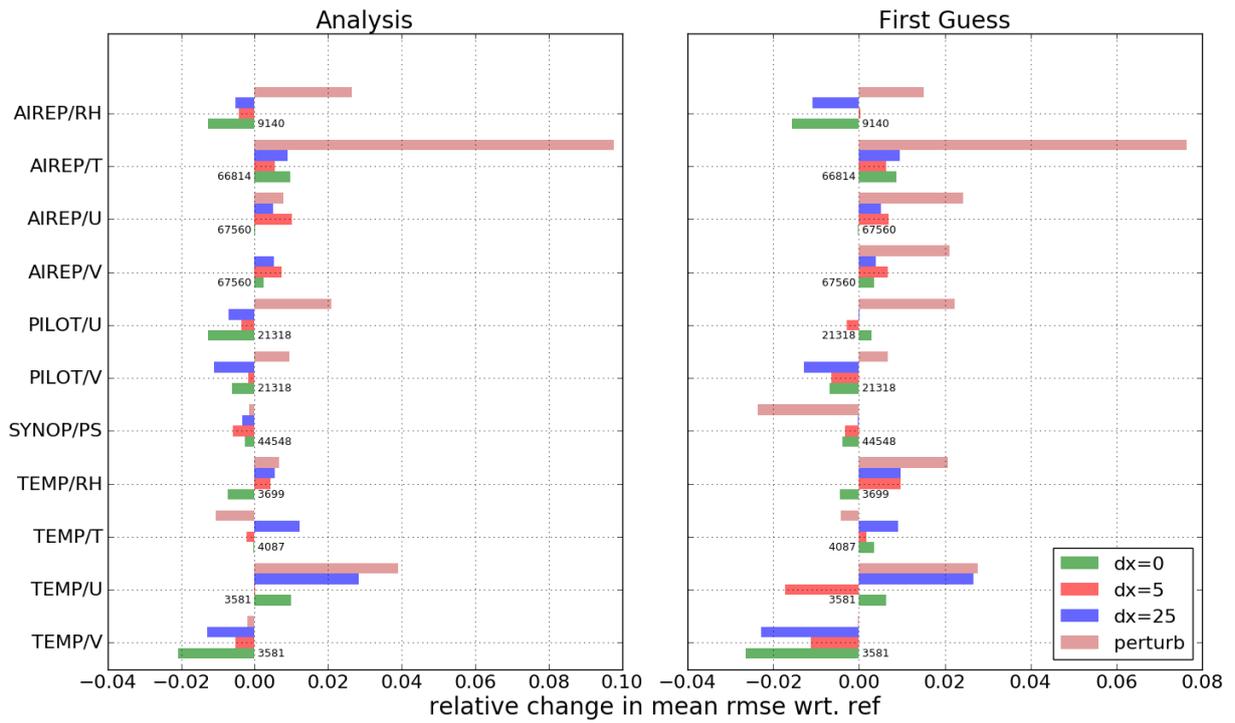


Figure 4.5.9: As Figure 4.5.8, but four the Southern part of the domain.

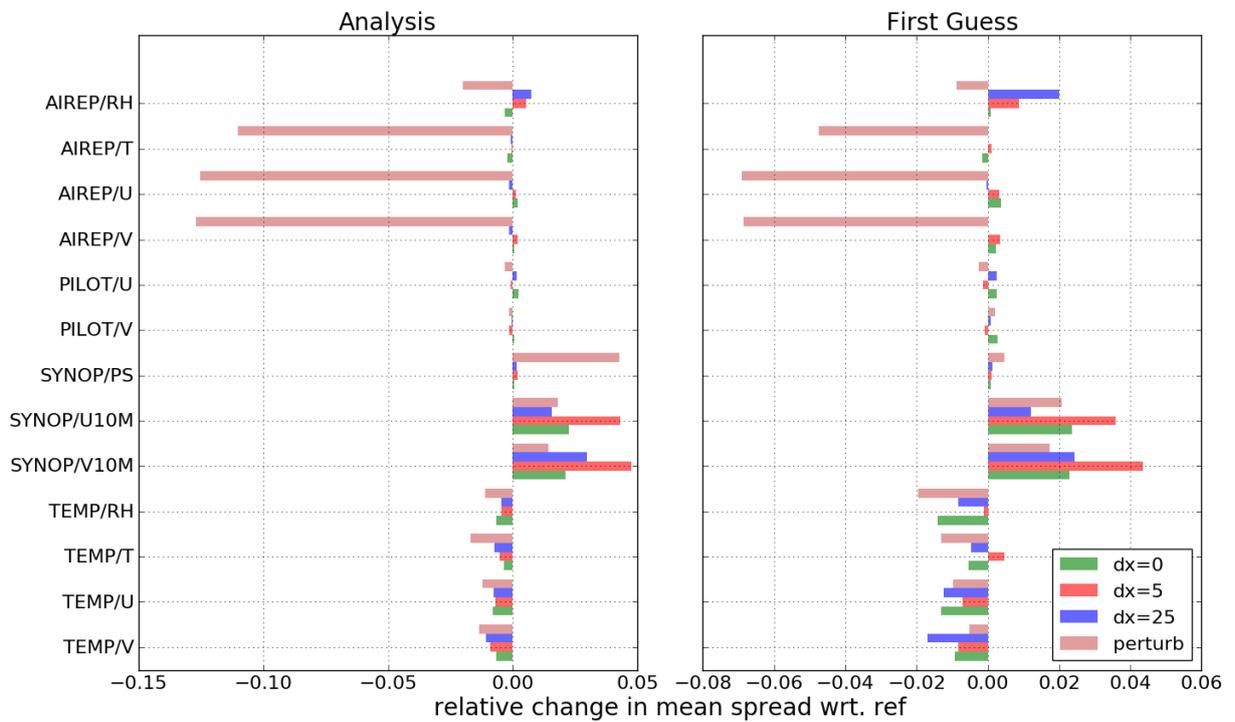


Figure 4.5.10: As Figure 4.5.8, but for the spread in observation space.

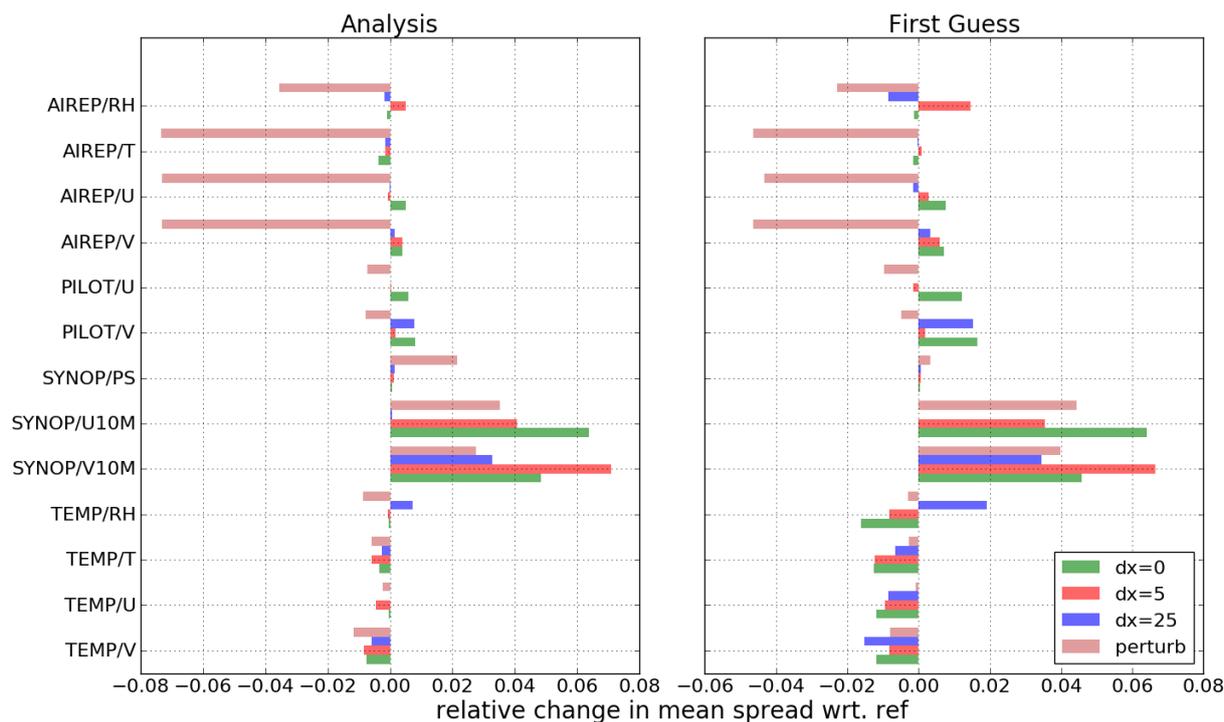


Figure 4.5.11: As Figure 4.5.10, but for the Southern part of the domain.

spread (see Appendix A) averaged over all 60 forecasts. As before, simply perturbing  $z_0$  without estimation is yielding poorer results than  $dx = 5$ , especially in the South. We can therefore conclude that any gain is due to updating  $z_0$  based on correlations with observed variables, rather than being the result of adding noise. Differences between North and South are most pronounced for  $dx = 0$ . In the North this experiment significantly outperforms all other experiments, whereas in the South it has the worse scores. In contrast,  $dx = 5$  is steadily performing better than both  $dx = 25$  and the control run over the entire domain, except for the bias, where  $dx = 25$  is superior. It is worth noting that the FAR and the ETS compliment each other in the sense that the FAR is meant to provide a measure for the amount of spurious convection, whereas the ETS measures the amount correctly predicted convection. Being superior in both scores indicates that better scores are not just the result of over- or under-forecasting. The FSS is popular to measure the skill for different scales. In Figures 4.5.15 and 4.5.16 we see that the FSS relative to the control run is more sensitive to the threshold than the scale. Again we see that  $dx = 0$  excels in the North, especially for high thresholds and longer forecast lead times, achieving an improvement of more than 5%.

#### 4.5.2.2 Precipitation

The verification product for precipitation is a combination of radar derived precipitation measurements and rain gauges. As this product is only available to us over Germany, the

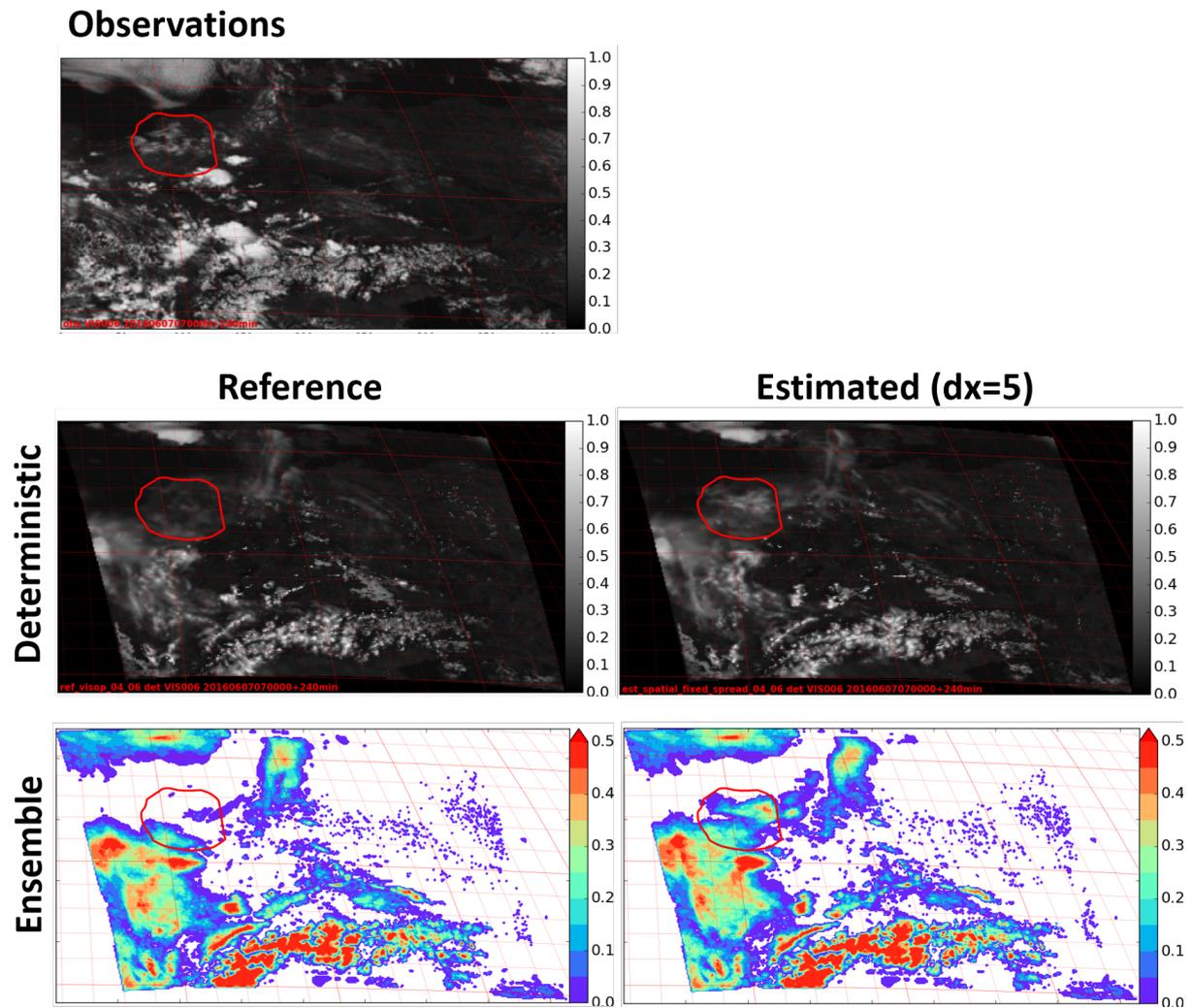


Figure 4.5.12: Snapshot of SEVIRI observations (top left) and model equivalents of *ref* (left) and  $dx = 5$  (right) for a 4 hour forecast started from 07.06.2016 at 7 UTC. The second row shows a snapshot of the deterministic run and the lowest row shows a probability plot generated from the corresponding ensemble.

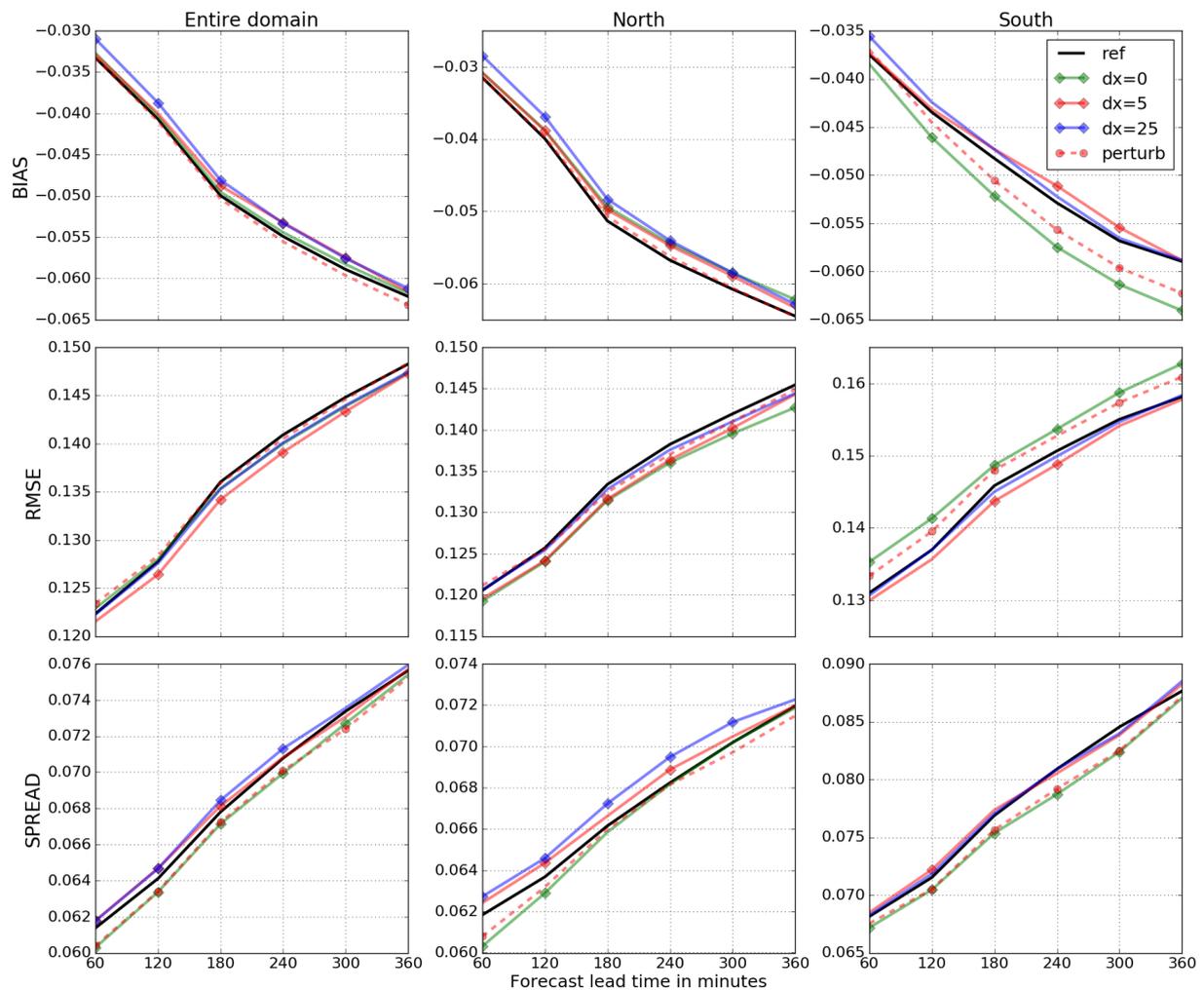


Figure 4.5.13: Reflectance bias (top), RMSE (middle) and spread (bottom) of the respective experiments for the entire domain (left), the North (middle) and the South (right) for the weak synoptically forced week. Markers indicate statistical significance according to the bootstrap method (Efron and Tibshirani, 1993), where the absolute differences compared to *ref* are calculated, followed by 10.000 bootstrap resamplings to determine the statistical significance for a 95 % confidence interval.

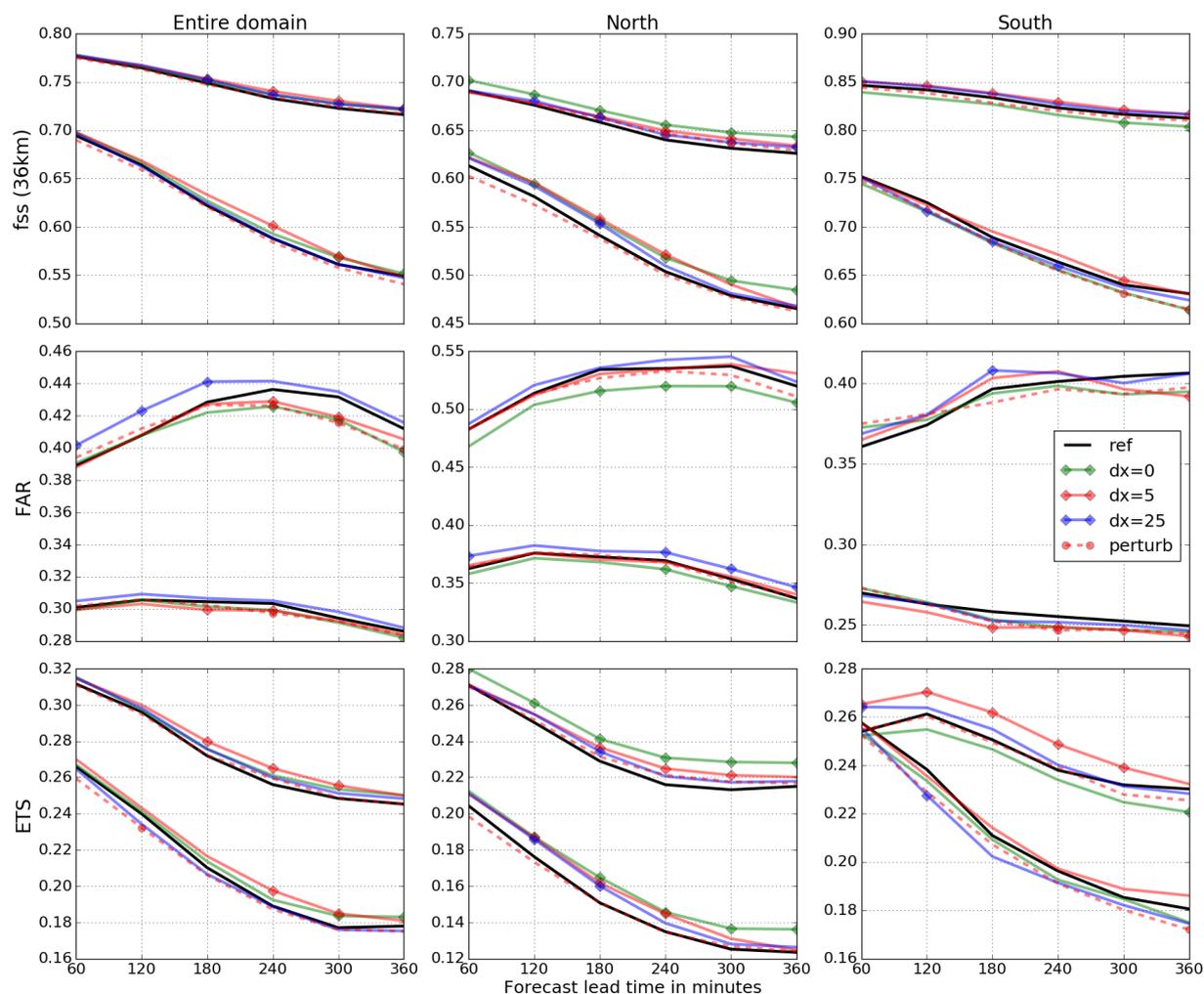


Figure 4.5.14: Reflectance FSS (top), FAR (middle) and ETS (bottom) for thresholds 0.3 and 0.5 of the respective experiments for the entire domain (left), the North (middle) and the South (right) for the weak synoptically forced week. Markers indicate the same as in Figure 4.5.13.

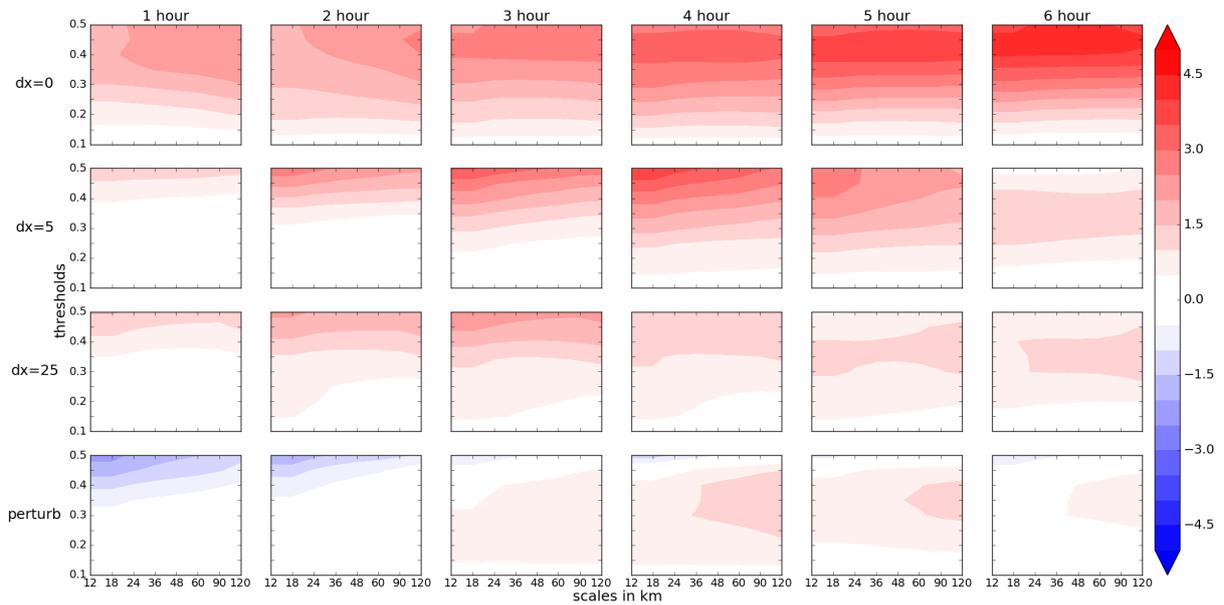


Figure 4.5.15: Relative reflectance FSS score in percentage (color) with respect to  $ref$  in the North for the weak synoptically forced week for the experiments (rows) for different lead times (columns).

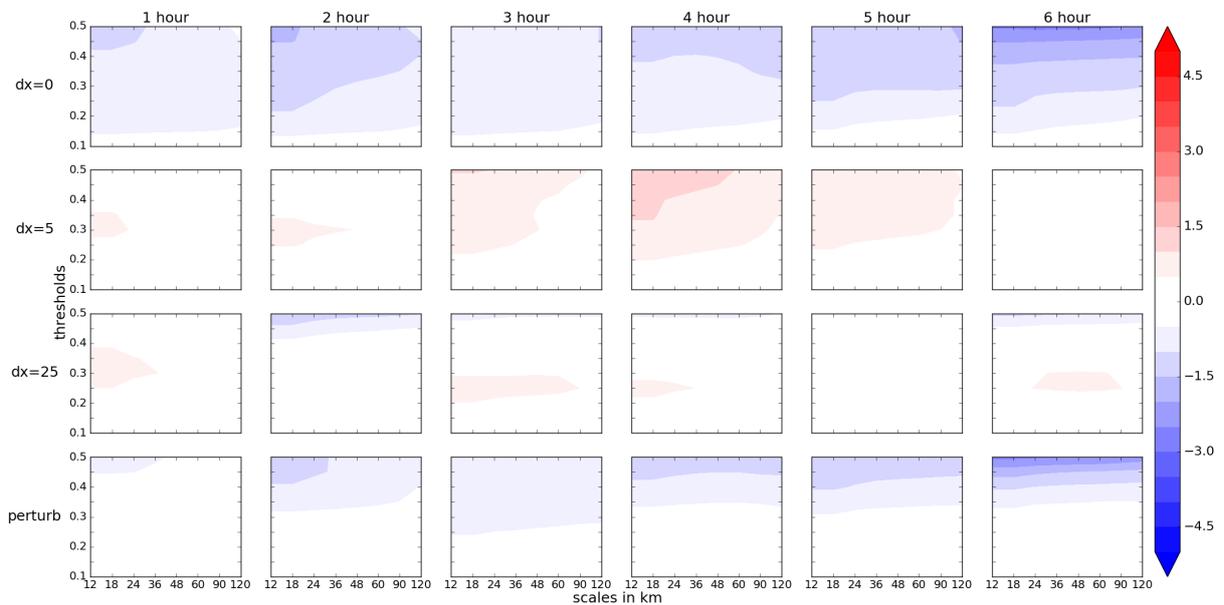


Figure 4.5.16: As Figure 4.5.15, but for the South.

verification region is masked to its borders. Due to the existence of outliers in radar data, the bias and RMSE are deemed inappropriate scores for reflectivity. Instead, scores like the FSS, FAR and ETS are used, where the concept of thresholds eliminates the sensitivity to outliers.

Differences among the experiments are smaller for reflectivity scores than for reflectance scores. The small difference might be due to that reflectivity is assimilated, thereby nudging the analyses towards the radar derived observations and suppressing variability among experiments. The only significant differences are in the North. Here it seems that  $dx = 5$  is predicting precipitation more accurately than the control run. Both in terms of correctly predicted convective events (ETS) and false alarms (FAR). Figure 4.5.18 also shows that the gain especially holds on smaller scales and higher thresholds. The advantage peaks at 2 to 3 hour forecast lead time where the gain is over 20 %.

#### 4.5.2.3 Strong synoptically forced week

Figures 4.5.19 to 4.5.21 show the verification scores for reflectance and precipitation for the strong synoptically forced week. As we already concluded that updating the mean of  $z_0$  is crucial to outperform the control run, we did not compute the *perturb* experiment for this period. Neither do we show the  $dx = 25$  experiment, as we found that a correlation length scale of 25 grid points is too large for the dynamical model of the roughness length.

The differences among the experiments are smaller than for the weak synoptically forced week (below 1% for the FSS). This supports our hypothesis that the effect of estimating the roughness length is smaller when the synoptic forcing is larger. In the North the scores indicate a slightly improved forecast with respect to the control run, except for the bias. In contrast to the weak synoptically forced week,  $dx = 0$  is doing better than the control run in the South. An exception is again the bias, though it should be noted that the bias is very small in the first few forecast hours in comparison to the North and the weak synoptically forced week. As the roughness length is less bounded in the South than in North due to lack of assimilated surface wind observations, it is less predictable how the LETKF utilizes the degrees of freedom inherited from the parameter estimation.

## 4.6 Conclusion

We conclude that estimating the roughness length with the augmented state approach can lead to better predictions of clouds and precipitation. The clear diurnal cycle of the analysis increments of the roughness length indicates the influence of physically based correlations. The diurnal cycle of surface heat fluxes peaks earlier in the day, suggesting the evolution of the roughness length is dominated by wind measurements.

In the North the scores were consistently better than the control run for a dynamical model with no ( $dx = 0$ ) or a modest ( $dx = 5$ ) correlation length scale. For a larger correlation length scale ( $dx = 25$ ) and ceasing to update the parameter mean (*perturb*) no

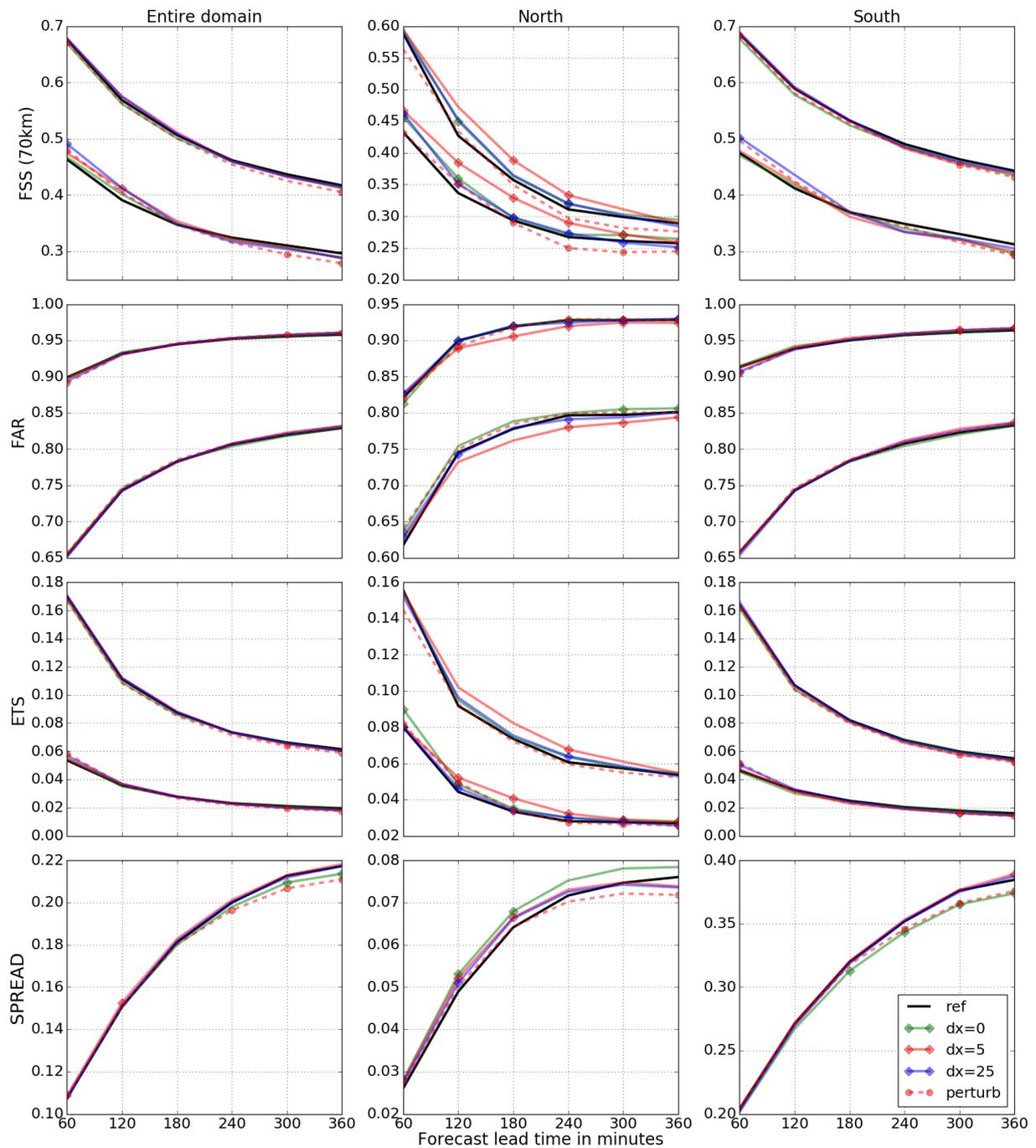


Figure 4.5.17: Reflectivity FSS (1<sup>st</sup> row), FAR (2<sup>nd</sup> row) and ETS (3<sup>rd</sup> row) for thresholds 1 mm per hour and 5 mm per hour and the spread (4<sup>th</sup> row) of the respective experiments for the entire domain (left), the North (middle) and the South (right) for the weak synoptically forced week. Markers indicate the same as in Figure 4.5.13.

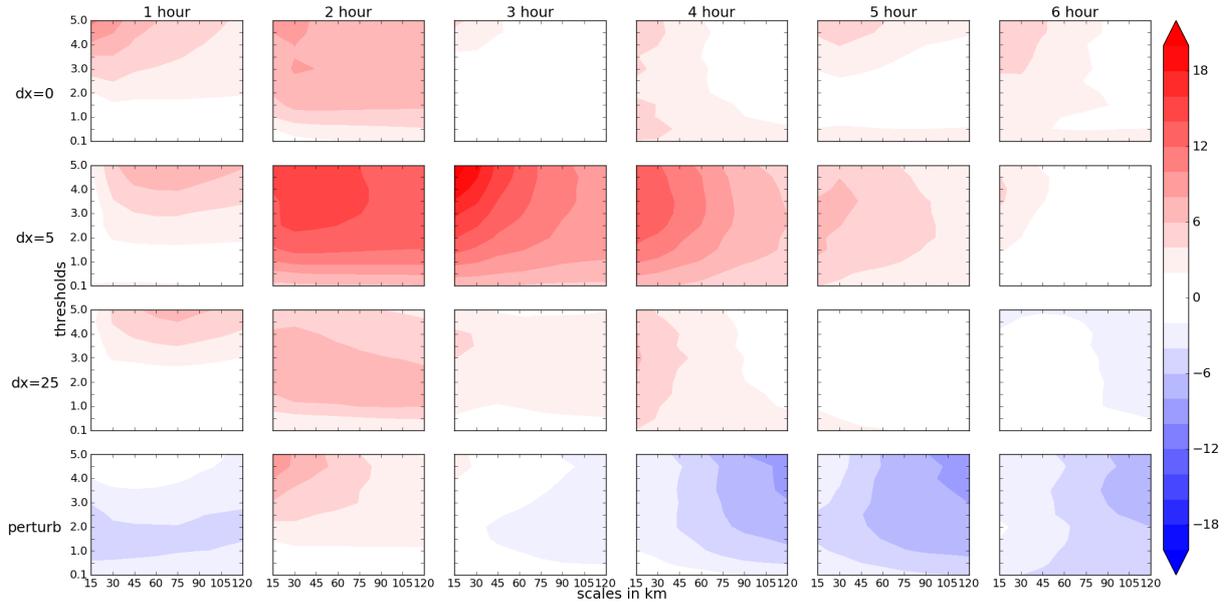


Figure 4.5.18: Relative reflectivity FSS score in percentage (color) with respect to *ref* for the weak synoptically forces week in the North for the remaining experiments (rows) for different forecast lead times (columns).

improvement over the control run was found, highlighting the importance of small scale analysis increments.

In the Northern part of the domain, experiment  $dx = 0$  was superior for the prediction of clouds, and  $dx = 5$  was superior for the prediction of precipitation. Yet, both experiments were consistently outperforming the control run, which was especially profound in the weak synoptically forced week, confirming the higher sensitivity of convection to the roughness length.

In the Southern part of the domain  $dx = 0$  portrayed unpredictable behavior. In this region, estimation of the roughness length with a correlation free dynamical model was clearly detrimental for the weak synoptically forced week, yet slightly favorable in the strong synoptically forced week. This unpredictable behavior can be attributed to the large model error inherited from orography and the lack of assimilated surface wind measurements. For  $dx = 5$  the spatial correlations imposed on the roughness length increase the influence radius of the observations assimilated, thereby reducing the freedom of the parameter. For the weak synoptically forced week, when convection is sensitive to the roughness length, this proved favorable in the South, where model error is large and surface wind observations sparse. The choice of correlation length scale for the dynamical model of the roughness length should therefore depend on the model error and observations assimilated.

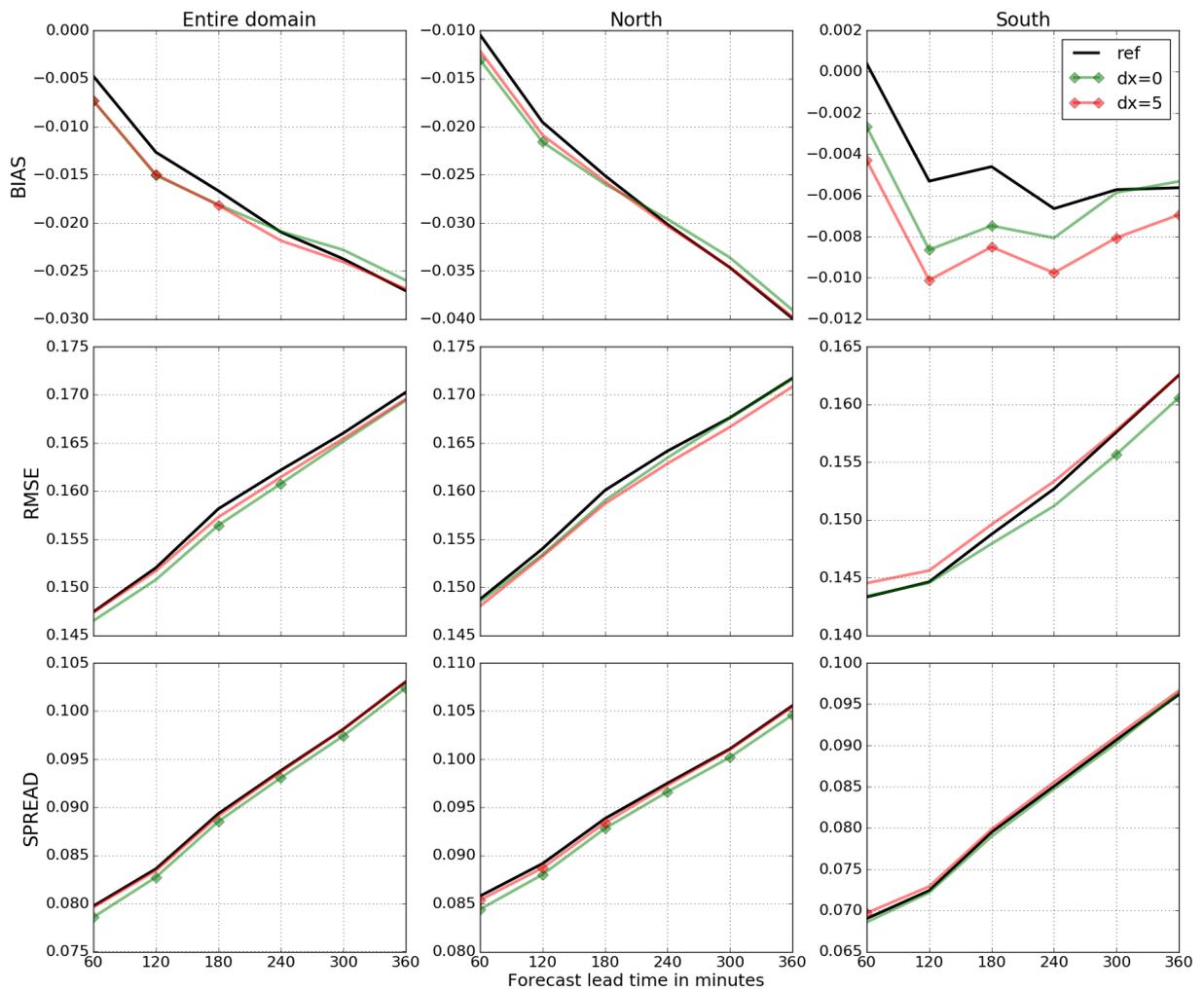


Figure 4.5.19: Similar as 4.5.13, but for the strong synoptically forced week.

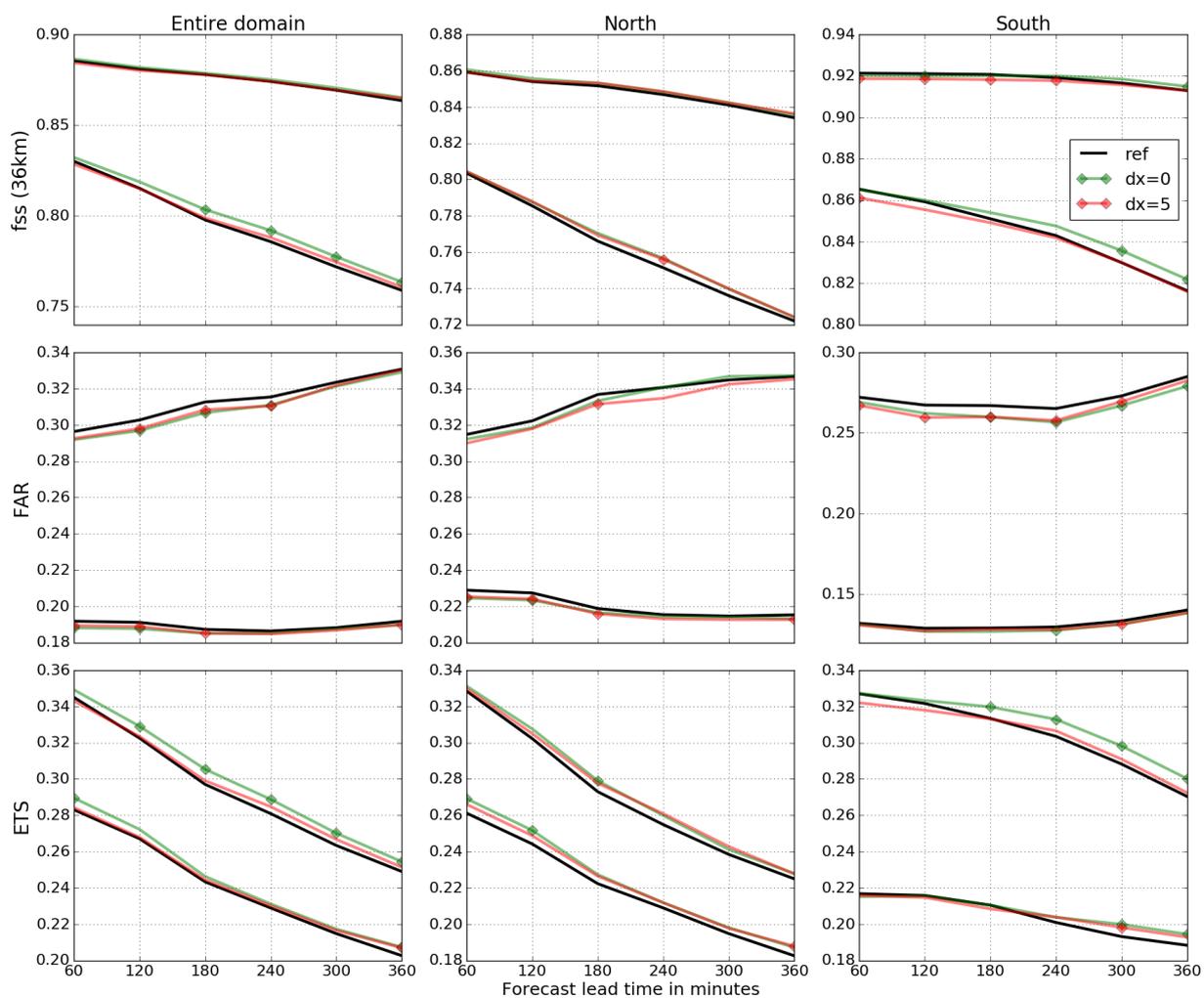


Figure 4.5.20: Similar as 4.5.14, but for the strong synoptically forced week.

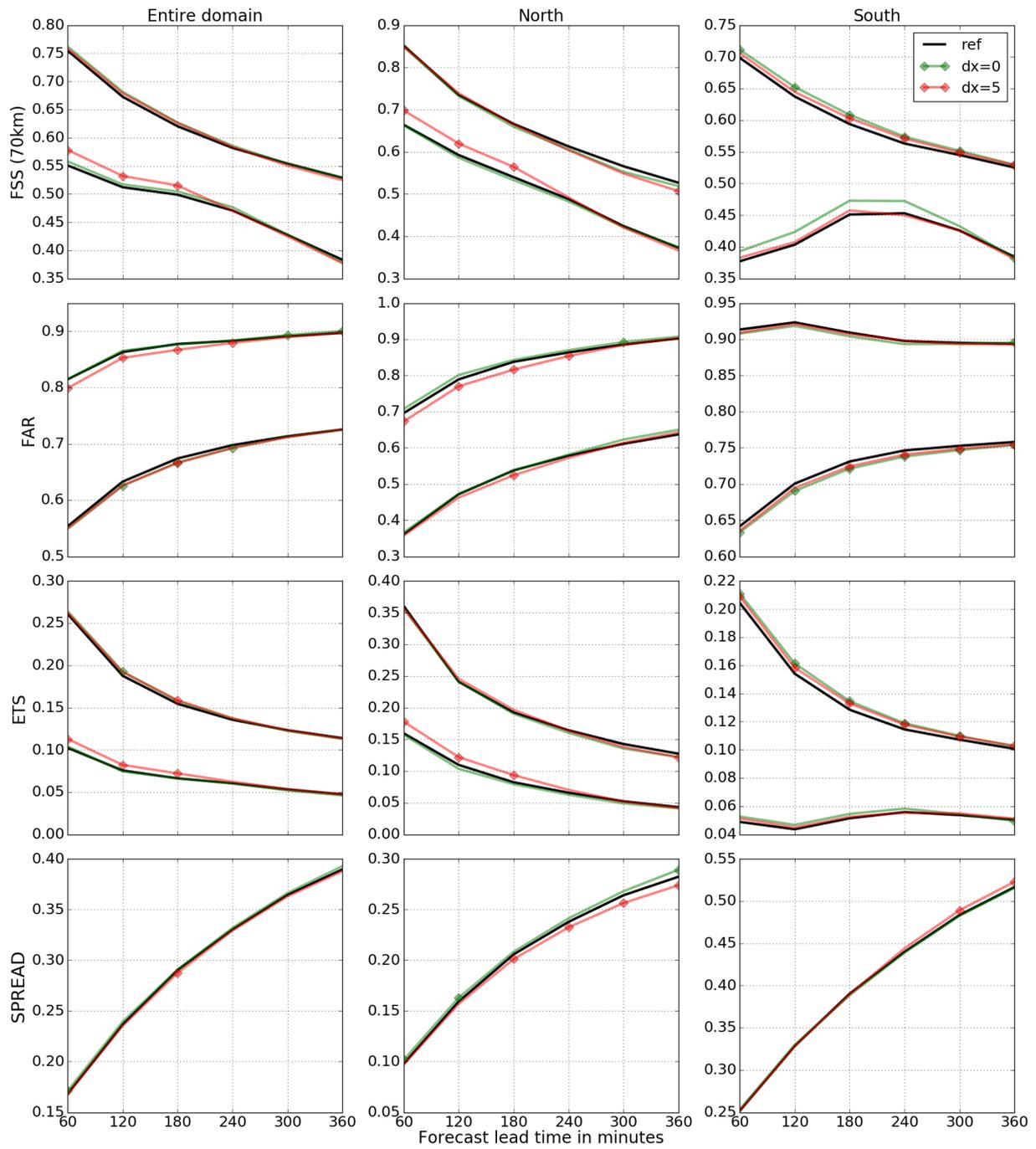


Figure 4.5.21: Similar as 4.5.17, but for the strong synoptically forced week.

## 4.7 Critical look and future work

The Conclusions presented in section 4.6 are promising, but they have the potential to be much better. In this section we propose some ideas on how to improve the parameter estimation.

### 4.7.1 Heat versus momentum flux

The roughness length for heat exchange ( $z_h$ ), though heavily correlated via the landscape type, is not the same as for momentum ( $z_0$ ). In fact, since momentum transfer is mainly caused by turbulent drag on roughness elements and heat exchange occurs via molecular thermal diffusion, it is necessary to add an additional resistance term to account for differences in transport mechanisms between momentum and heat (Owen and Thomson, 1963; Yang and Friedl, 2003). As explained in section 4.1.3, in the COSMO model the roughness length for heat  $z_h$  is set as  $z_h = \min(z_0, z_{h_{max}})$ , which does not allow decoupled estimation of  $z_0$  and  $z_h$ . Consequently, the analysis increments of  $z_0$  are the result of a compromise between reducing the errors in the momentum flux and the surface heat flux. In addition, the imposed upper bound of  $z_h$  causes complications for data assimilation as discussed in section 1.2.3 and Chapter 3. Furthermore, it has been shown that landscape variability plays an important role in the initiation of cumulus clouds via surface heat fluxes, see for example Rabin et al. (1990). However, from our results we deduce that the momentum flux dominated the parameter increments. This is possibly an effect of the upper bound  $z_h$ , which can cause severe spread reduction in  $z_h$ . It is therefore natural to estimate the parameters  $z_0$  and  $z_h$  separately, or at least estimate a coupling term between the two.

### 4.7.2 Spurious correlations

Spurious correlations are a serious problem for parameter estimation. As argued before, localization as done for the state is not necessarily appropriate for parameters, even for a spatially varying one, because the influence radius is smaller. This is especially the case for non-smooth parameters such as the roughness length. We partially addressed this issue by introducing a correlation length scale in the model, which essentially reduced the degrees of freedom. This strategy was indeed effective in the South where surface wind observations are extremely sparse. However, naturally, introducing artificial correlations is not ideal, as it may also introduce wrong correlations. This was probably the case for  $dx = 25$ . We believe a more natural way to reduce spurious correlations for parameter estimation is Sampling Error Correction (SEC, Anderson, 2012, 2016). As already mentioned in Chapter 3, this method is based on Monte Carlo simulations that serve to calculate a statistically based correction term for a given sample correlation, ensemble size and prior assumption of the correlation distribution. These corrections are computed off-line, resulting in a simple look up table for the sampling error correction.

### 4.7.3 Tuning

Although we have explored three different correlation length scales, the dynamical model would most certainly benefit from some tuning. We saw for example large differences among experiments between the forecast quality in the North and in the South. This suggests the need for a location depended correlation length scale. Another very important tuning exercise is the spread specification. The spread basically regulates the amplitude of the increments, which obviously influences the results strongly. Ideally the spread should be a function of time and space, based on the estimated strength of correlations with observed variables.

Throughout this work we have assumed a Gaussian dynamical model for  $z_0$ . However, as  $z_0$  relates lognormally to the surface fluxes, it might make sense to assume a skewed distribution as in Chapter 3. This would also alleviate the lower bound problem.

### 4.7.4 Resampling

To analyze the severity of spurious correlations and to aid the tuning, we propose to perform bootstrapping on the prior ensemble perturbations of  $z_0$  (personal communication with Martin Verlaan). Note that the analysis increment of the ensemble mean of  $z_0$  can be written as

$$\bar{z}_0^a - \bar{z}_0^f = \mathbf{X}_{z_0}^f \bar{\mathbf{w}}^a, \quad (4.7.1)$$

where  $\mathbf{X}_{z_0}^f$  are the ensemble perturbations of  $z_0$  on which  $\bar{\mathbf{w}}^a \in \mathbb{R}^{N_{ens} \times N_{ens}}$  does not depend since  $z_0$  is not represented in observation space (Hunt et al., 2007). Generating resampled analyses of  $\bar{z}_0^a - \bar{z}_0^f$  is therefore relatively cheap. From the result we can learn about when and where spurious correlations are severe, which could provide hints on how to regulate the spread. Indeed, the detrimental effect of spurious correlations can be suppressed by lowering the spread.

### 4.7.5 Scale separation

Although the results are promising, we should critically consider Figure 4.5.7. It is not clear if and when the standard deviation and the maximum of  $z_0$  converges. To verify that the parameter does not diverge, a longer test period is needed. Maintaining a flow dependent spread as suggested above should aid in preventing parameter divergence, but it might not be enough. In that case it might be favorable to split the parameter into its climatological part, which is slowly varying, and its remaining flow dependent part, which is rapidly varying (personal communication with Martin Verlaan). The climatological part could be estimated with time averaged correlations, to smooth out spurious and flow dependent correlations. The flow dependent deviations should be kept within a certain range of the climatological part either by resetting regularly, or imposing bound constraints. To investigate what works best, the parameter estimation should run over a long test period.

### 4.7.6 QPEns and QF

As discussed earlier, the roughness length has a non-linear relation to the state variables, making it natural to assume a skewed prior distribution. In contrast to the EnKF, the QF considers skewness in the calculation of the analysis increments. The results of Chapter 3 highlight the benefit of applying the QF for the parameters, even for small ensemble sizes.

Finally, in contrast to the modified shallow water experiments discussed in Chapter 3, the parameter  $z_0$  does hit the lower bound. Applying QPEns for bound constraints using the gradient projected method (section 5.1.2.2) is a far more consistent way of dealing with bound violation than the ad hoc clipping method we applied.

# Chapter 5

## Reducing computational costs for the QPEns

The results discussed in the previous chapter clearly highlight the advantage of the QPEns algorithm. Unfortunately, this algorithm is computationally very expensive, as one needs to solve a constrained minimization problem for each ensemble member. In this chapter, we focus on reducing the computational costs for the QPEns. We first propose an efficient QPEns algorithm where we exploit that the bound constraints (precipitation) and equality constraints (mass conservation) are disjoint in section 5.1. Then we illustrate the potential of simple neural networks to produce constrained initial conditions from unconstrained ones in section 5.2. These two very different approaches are tested on the modified shallow water model introduced in section 3.3. As the focus is on computational costs, only state estimation is considered.

### 5.1 QP for disjoint constraints

State of the art algorithms for a broad range of optimization problems are widely available. Even though these algorithms are already specialized for specific problem classes, practical problems often have additional characteristics that could be exploited. In this case we can combine algorithms designed for bound constraints and for equality constraints because the variables effecting the respective constraint types are disjoint. The contents of this section are based on Janjić et al. (2019).

#### 5.1.1 Problem

We consider the following problem

$$\begin{aligned} \min_{\mathbf{x}, \mathbf{y}} \quad & \mathcal{J}(\mathbf{x}, \mathbf{y}) \stackrel{\text{def}}{=} (\mathbf{g}_x^T, \mathbf{g}_y^T) \begin{pmatrix} \mathbf{x} \\ \mathbf{y} \end{pmatrix} + \frac{1}{2} (\mathbf{x}^T \mathbf{y}^T) \begin{pmatrix} \mathbf{G}_{xx} & \mathbf{G}_{xy} \\ \mathbf{G}_{xy}^T & \mathbf{G}_{yy} \end{pmatrix} \begin{pmatrix} \mathbf{x} \\ \mathbf{y} \end{pmatrix} \\ \text{subject to} \quad & \mathbf{A}\mathbf{x} = \mathbf{b} \quad \text{and} \quad \mathbf{1} \leq \mathbf{y} \end{aligned} \quad (5.1.1)$$

where  $\mathbf{x}, \mathbf{g}_x \in \mathbb{R}^n$ ,  $\mathbf{y}, \mathbf{g}_y, \mathbf{l} \in \mathbb{R}^p$ ,  $\mathbf{G}_{xx} \in \mathbb{R}^{n \times n}$  and  $\mathbf{G}_{yy} \in \mathbb{R}^{p \times p}$  are symmetric and real,  $\mathbf{G}_{xy} \in \mathbb{R}^{n \times p}$  is real,  $\mathbf{A} \in \mathbb{R}^{m \times n}$  is real with  $m \leq n$  of rank  $m$ ,  $\mathbf{b} \in \mathbb{R}^m$  and the inequality is understood component-wise. We refer to a problem of the type (5.1.1) as having *disjoint constraints* in the sense that the two sets of constraints of (5.1.1) involve disjoint sets of variables. We focus on the convex case, and assume that

$$\mathbf{G} \stackrel{\text{def}}{=} \begin{pmatrix} \mathbf{G}_{xx} & \mathbf{G}_{xy} \\ \mathbf{G}_{xy}^T & \mathbf{G}_{yy} \end{pmatrix}$$

is positive definite.

### 5.1.2 Background

In this section we cover some background for convex optimization. For more details, see for example Conn et al. (2000); Nocedal and Wright (2006).

Inequality constraints are generally more difficult to handle than equality constraints. It is therefore useful to distinguish between *active constraints* and *inactive constraints*. Constraints that meet their bounds are considered active and define the *working set* ( $\mathcal{W}$ ), all other constraints are considered inactive. The working set corresponding to the solution is called the *active set* ( $\mathcal{A}$ ). In what follows we refer to the subspace spanned by the inactive constraints as the *working face*. Note that if the active set of an inequality constrained problem is known, the problem is reduced to an equality constrained problem. Indeed, the constraints active at the solution can be seen as equality constraints and all other constraints do not affect the solution and can therefore be ignored. We introduce the following notation. If  $\mathbf{v}$  is a vector in  $\mathbb{R}^p$  and  $\mathcal{W} \subseteq \{1, \dots, p\}$ , we denote by  $\mathbf{v}^{\mathcal{W}}$  the vector  $\mathbf{v}$  reduced to its active component, that is

$$\mathbf{v}^{\mathcal{W}} = \begin{cases} [\mathbf{v}]_i & \text{if } i \in \mathcal{W} \\ \mathbf{1} & \text{otherwise,} \end{cases}$$

where  $[\mathbf{v}]_i$  denotes the  $i$ -th component of  $\mathbf{v}$ . Similarly,  $\mathbf{M}^{\mathcal{W}}$  is the matrix  $\mathbf{M}$  reduced to its active columns (and rows, if it is symmetric).

*Active set algorithms* aim to find the active set, by iteratively adding and removing constraints from the working set and solving corresponding equality constrained problems. In section 5.1.3 we show how we can use this theory to develop an efficient algorithm for problem (5.1.1).

#### 5.1.2.1 Equality constrained case

For  $l = -\infty$ , (5.1.1) reduces to an equality constrained problem.

$$\begin{aligned} \min_{\mathbf{x}, \mathbf{y}} \quad & \mathcal{J}(\mathbf{x}, \mathbf{y}) \stackrel{\text{def}}{=} (\mathbf{g}_x^T, \mathbf{g}_y^T) \begin{pmatrix} \mathbf{x} \\ \mathbf{y} \end{pmatrix} + \frac{1}{2} (\mathbf{x}^T \mathbf{y}^T) \begin{pmatrix} \mathbf{G}_{xx} & \mathbf{G}_{xy} \\ \mathbf{G}_{xy}^T & \mathbf{G}_{yy} \end{pmatrix} \begin{pmatrix} \mathbf{x} \\ \mathbf{y} \end{pmatrix} \\ \text{subject to} \quad & \mathbf{A}\mathbf{x} = \mathbf{b} \end{aligned} \quad (5.1.2)$$

Since the problem is convex, its solution  $(\mathbf{x}^*, \mathbf{y}^*)$  is found by solving the following KKT conditions

$$\nabla_x \mathcal{J}(\mathbf{x}^*, \mathbf{y}^*) = -\mathbf{A}^T \lambda^* \quad (5.1.3a)$$

$$\nabla_y \mathcal{J}(\mathbf{x}^*, \mathbf{y}^*) = \mathbf{0} \quad (5.1.3b)$$

$$\mathbf{A}\mathbf{x}^* - \mathbf{b} = \mathbf{0} \quad (5.1.3c)$$

for some  $\lambda^* \in \mathbb{R}^m$  (see appendix C). Let  $(\mathbf{x}_k, \mathbf{y}_k)$  be a feasible estimate of  $(\mathbf{x}^*, \mathbf{y}^*)$ , i.e.  $\mathbf{A}\mathbf{x}_k = \mathbf{b}$ . We substitute  $\mathbf{x} = \mathbf{s} + \mathbf{x}_k$  and  $\mathbf{y} = \mathbf{v} + \mathbf{y}_k$  into problem (5.1.2) to obtain the equivalent system of equations

$$\begin{pmatrix} \mathbf{G}_{xx} & \mathbf{G}_{xy} & \mathbf{A}^T \\ \mathbf{G}_{xy}^T & \mathbf{G}_{yy} & \mathbf{0} \\ \mathbf{A}_k & \mathbf{0} & \mathbf{0} \end{pmatrix} \begin{pmatrix} \mathbf{s} \\ \mathbf{v} \\ \lambda \end{pmatrix} = - \begin{pmatrix} \nabla_x \mathcal{J}(\mathbf{x}_k, \mathbf{y}_k) \\ \nabla_y \mathcal{J}(\mathbf{x}_k, \mathbf{y}_k) \\ \mathbf{0} \end{pmatrix} \quad (5.1.4)$$

Solving system (5.1.4) for  $(\mathbf{s}, \mathbf{v})$  yields the exact solution  $(\mathbf{x}^*, \mathbf{y}^*) = (\mathbf{x}_k + \mathbf{s}, \mathbf{y}_k + \mathbf{v})$  of problem (5.1.2).

### 5.1.2.2 Bound constrained case

For  $n = 0$ , (5.1.1) reduces to a bound constrained problem, which can be solved with a gradient projected method

$$\begin{aligned} \min_{\mathbf{y}} \quad & \mathcal{J}(\mathbf{y}) \stackrel{\text{def}}{=} \mathbf{g}^T \mathbf{y} + \frac{1}{2} \mathbf{y}^T \mathbf{G} \mathbf{y} \\ \text{subject to} \quad & \mathbf{l} \leq \mathbf{y}. \end{aligned} \quad (5.1.5)$$

Let  $\mathbf{y}_k$  be a feasible estimate of  $\mathbf{y}^*$  with corresponding working set  $\mathcal{W}_k$ , i.e.  $[\mathbf{l}]_i = [\mathbf{y}_k]_i, i \in \mathcal{W}_k$  and  $[\mathbf{l}]_i < [\mathbf{y}_k]_i, i \in \mathcal{W}_k^c$ . The next iterate  $\mathbf{y}_{k+1}$  is found in two steps.

First, we search for a minimum along the steepest descent direction  $\nabla \mathcal{J}(\mathbf{y}_k)$ . When a bound is encountered before a minimum is found, the search direction is “bent”, so that it stays feasible. More precisely, we search for a minimum along

$$h(\alpha) \stackrel{\text{def}}{=} \mathcal{J} \left( \max \left[ \mathbf{y}_k - \alpha \nabla \mathcal{J}(\mathbf{y}_k), \mathbf{l} \right] \right). \quad (5.1.6)$$

The resulting step size  $\alpha_k$  gives us the *Cauchy point*  $\hat{\mathbf{y}}_k = \mathbf{y}_k + \alpha_k \nabla \mathcal{J}(\mathbf{y}_k)$ , with corresponding working set  $\hat{\mathcal{W}}_k$ .

In the second step we partially solve

$$\begin{aligned} \min_{\mathbf{y}} \quad & \mathcal{J}(\mathbf{y}) \stackrel{\text{def}}{=} \mathbf{g}^T \mathbf{y} + \frac{1}{2} \mathbf{y}^T \mathbf{G} \mathbf{y} \\ \text{subject to} \quad & [\mathbf{l}]_i \leq [\mathbf{y}]_i, i \in \hat{\mathcal{W}}_k^c \quad \text{and} \quad [\mathbf{l}]_i = [\mathbf{y}]_i, i \in \hat{\mathcal{W}}_k \end{aligned} \quad (5.1.7)$$

Note that the components  $i \in \hat{\mathcal{W}}_k$  of  $\mathbf{y}$  are constant in (5.1.7) so we may minimize the problem in working space:

$$\begin{aligned} \min_{\mathbf{y}^{\hat{\mathcal{W}}_k}} \quad & \mathcal{J}(\mathbf{y}^{\hat{\mathcal{W}}_k}) \\ \text{subject to} \quad & \mathbf{l}^{\hat{\mathcal{W}}_k} \leq \mathbf{y}^{\hat{\mathcal{W}}_k} \end{aligned} \quad (5.1.8)$$

One could for example use the conjugate gradient method (Hestenes and Stiefel, 1952) to minimize cost function  $\mathcal{J}(\mathbf{y}^{\hat{\mathcal{W}}_k})$ . One would update the working set when a new bound is encountered and restart the procedure in the new (lower dimensional) working face. The resulting feasible estimate solution of problem (5.1.8) defines the next iterate  $\mathbf{y}_{k+1}$ .

These steps are repeated until no further gain is made by moving in steepest descent direction whilst staying in the feasible region. In other words, the solution is found when the problem is minimized in working space,  $\nabla^{\mathcal{W}_k} \mathcal{J}(\mathbf{y}_k) = \mathbf{0}$ , and the gradient of the quadratic with respect to the active variables is non negative  $\nabla^{\mathcal{W}_k} \mathcal{J}(\mathbf{y}_k) \geq \mathbf{0}$ .

If the problem is well conditioned, step two might be unnecessary, because the steepest descent method performs well enough, i.e.  $\mathbf{y}_{k+1} \leftarrow \hat{\mathbf{y}}_k$ . For poorly conditioned problems however, the Cauchy point serves primarily to update the estimate of the active set. (Partially) solving (5.1.8) is required to provide a significantly improved estimate of the solution. There is no general rule to what degree (5.1.8) should be solved for optimal efficiency. Instead, heuristic approaches are usually key to optimize the gradient projected method.

### 5.1.3 The new algorithms

We can apply an active set algorithm that maintains feasibility with respect to the equality constraints, while using projection techniques as described above to enforce feasibility of  $\mathbf{y}$ . At each iteration  $k$  a working set is defined as

$$\mathcal{W}_k \stackrel{\text{def}}{=} \{i \in \{1, \dots, p\} \mid [\mathbf{y}_k]_i = [\mathbf{l}]_i \text{ and } \nabla_{\mathbf{y}} \mathcal{J}(\mathbf{x}_k, \mathbf{y}_k) > 0\},$$

which represents all  $\mathbf{y}$ -components that are active at iteration  $k$  and would leave the feasible region if moved along the steepest descent direction. Therefore, all  $\mathbf{y}$ -components corresponding to the working set are temporarily assumed constant and are excluded from the search direction. The search direction is found by solving system (5.1.4) in working face. Note that this is equivalent to solving subproblem

$$\begin{aligned} \min_{\mathbf{x}, \mathbf{y}} \quad & \mathcal{J}(\mathbf{x}, \mathbf{y}) \stackrel{\text{def}}{=} (\mathbf{g}_x^T, \mathbf{g}_y^T) \begin{pmatrix} \mathbf{x} \\ \mathbf{y} \end{pmatrix} + \frac{1}{2} (\mathbf{x}^T \mathbf{y}^T) \begin{pmatrix} \mathbf{G}_{xx} & \mathbf{G}_{xy} \\ \mathbf{G}_{xy}^T & \mathbf{G}_{yy} \end{pmatrix} \begin{pmatrix} \mathbf{x} \\ \mathbf{y} \end{pmatrix} \\ \text{subject to} \quad & \mathbf{A}\mathbf{x} = \mathbf{b} \text{ and } [\mathbf{y}]_i = [\mathbf{l}]_i, i \in \mathcal{W}_k \end{aligned} \quad (5.1.9)$$

The step size computation is analogous to the gradient projection method, thereby ensuring feasibility of  $\mathbf{y}$ , while maintaining feasibility with respect to the equality constraint. The latter only holds because the constraints are disjoint! Once all  $\mathbf{y}$ -components outside of the

working set are optimized, i.e.  $[\nabla_y \mathcal{J}(\mathbf{x}_k, \mathbf{y}_k)]^{\mathcal{W}_k^c} = \mathbf{0}$ , the solution is found. The algorithm is described in Algorithm 5.1.1.

**Algorithm 5.1.1: QP algorithm for disjoint constraints**

**Step 0: Initialization.** A feasible starting point  $(\mathbf{x}_0, \mathbf{y}_0)$  is given (i.e.,  $\mathbf{A}\mathbf{x}_0 = \mathbf{b}$ ,  $\mathbf{y}_0 \geq \mathbf{l}$ ), as well as an accuracy threshold  $\epsilon > 0$ . Set  $k = 0$ .

**Step 1: Working set update.**

$$\mathcal{W}_k \stackrel{\text{def}}{=} \{i \in \{1, \dots, p\} \mid [\mathbf{y}_k]_i = [\mathbf{l}]_i \text{ and } \nabla_y \mathcal{J}(\mathbf{x}_k, \mathbf{y}_k) > \mathbf{0}\} \quad (5.1.10)$$

$$\mathcal{W}_k^c \stackrel{\text{def}}{=} \{i \in \{1, \dots, p\} \mid i \notin \mathcal{W}_k\} \quad (5.1.11)$$

**Step 2: Termination test.** Terminate if  $\|[\nabla_y \mathcal{J}(\mathbf{x}_k, \mathbf{y}_k)]^{\mathcal{W}_k^c}\| \leq \epsilon$ .

**Step 3: Search direction computation.** Solve

$$\begin{pmatrix} \mathbf{G}_{xx} & \mathbf{G}_{xy}^{\mathcal{W}_k^c} & \mathbf{A}^T \\ (\mathbf{G}_{xy}^{\mathcal{W}_k^c})^T & \mathbf{G}_{yy}^{\mathcal{W}_k^c} & \mathbf{0} \\ \mathbf{A} & \mathbf{0} & \mathbf{0} \end{pmatrix} \begin{pmatrix} \mathbf{s}_k \\ \mathbf{v}_k^{\mathcal{W}_k^c} \\ \mathbf{w}_k \end{pmatrix} = - \begin{pmatrix} \nabla_x^{\mathcal{W}_k^c} \mathcal{J}(\mathbf{x}_k, \mathbf{y}_k) \\ \nabla_y^{\mathcal{W}_k^c} \mathcal{J}(\mathbf{x}_k, \mathbf{y}_k) \\ \mathbf{0} \end{pmatrix} \quad (5.1.12)$$

**Step 4: Projected search.** Determine  $\alpha > 0$  such that  $(\mathbf{x}_{k+1}, \mathbf{y}_{k+1})$  is the first minimizer of  $\mathcal{J}(\mathbf{x}_k + \alpha \mathbf{s}_k, \max[\mathbf{y}_k + \alpha \mathbf{v}_k, \mathbf{l}])$ , where  $\mathbf{v}_k$  is obtained from  $\mathbf{v}_k^{\mathcal{W}_k^c}$  by setting  $[\mathbf{v}_k]_i = 0$  for  $i \in \mathcal{W}_k$ .

System (5.1.12) can be solved using a Krylov solver like MINRES or GMRES (see Saad (1996) for a description of these methods), or by a “constrained preconditioned” conjugate gradient method (see Gould et al. (2001); Gould and Toint (2002)). If this is the case, any preconditioner must also be reduced (in its  $\mathbf{y}$  part) to the subset of currently active variables  $\mathcal{W}_k$ . If dimension and sparsity of  $\mathbf{G}$  allows (which is typically not the case in weather forecasting), a stable factorization can also be used to solve (5.1.12) accurately.

To illustrate Algorithm 5.1.1 we perform a similar twin experiment as in the Chapter 3, where we consider a modified shallow model run to be the true state  $\mathbf{z} = (\mathbf{u}^T, \mathbf{h}^T, \mathbf{r}^T)^T$ , which we call the nature run. Given an estimate  $\tilde{\mathbf{z}} = (\tilde{\mathbf{u}}^T, \tilde{\mathbf{h}}^T, \tilde{\mathbf{r}}^T)^T$  and observations  $\mathbf{z}^{obs} = \mathbf{H}\mathbf{z} + \epsilon$ ,  $\epsilon \sim \mathcal{N}(0, \mathbf{R})$  of the nature run, we minimize a quadratic cost function based on the error covariance matrix of the state estimate  $\mathbf{P}$  and the observations  $\mathbf{R}$  respectively, in order to find an improved estimate  $\mathbf{z}^* = (\mathbf{u}^{*T}, \mathbf{h}^{*T}, \mathbf{r}^{*T})^T$  of the true state. We constrain

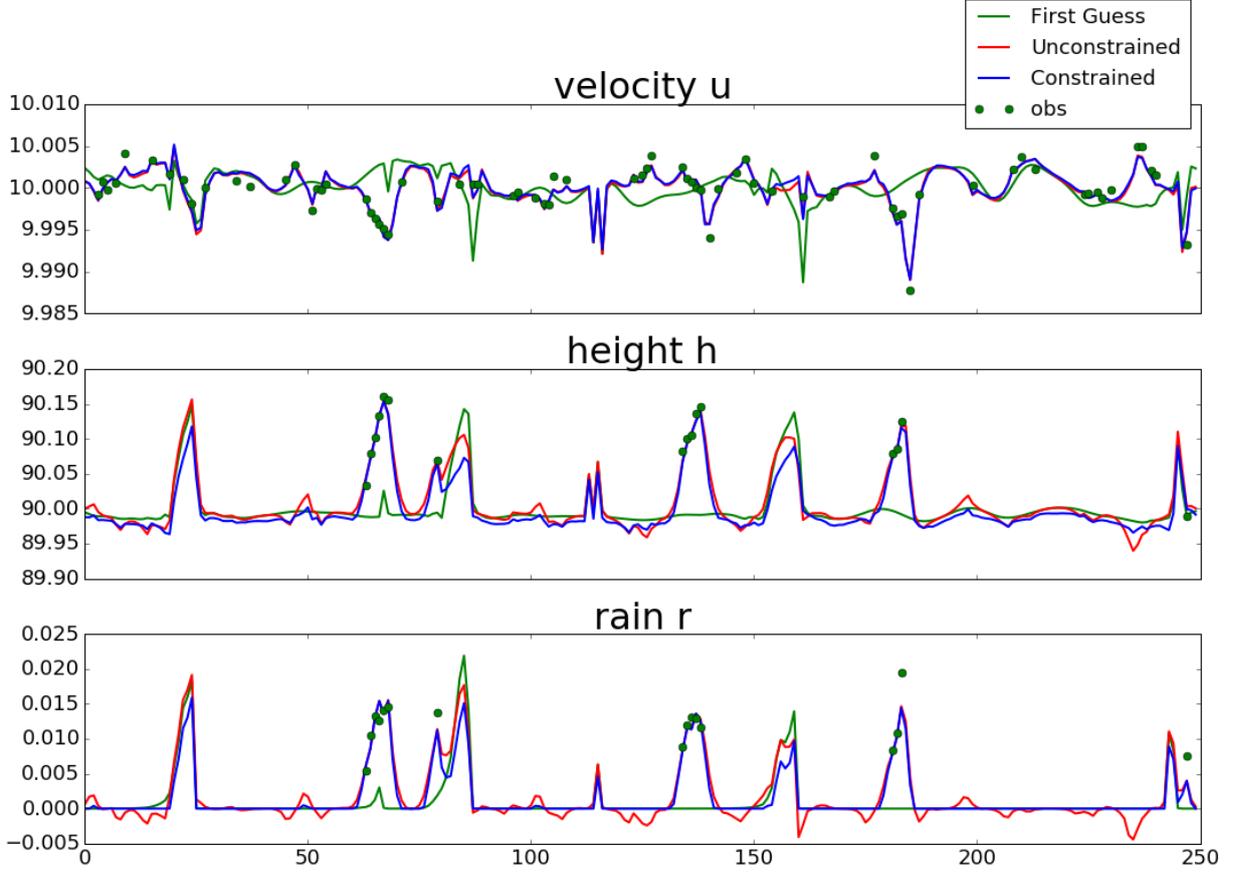


Figure 5.1.1: Results of minimization for unconstrained (red) and constrained (blue) problems based on prior estimate (green line) and observations (green circles).

the mass of  $\mathbf{h}^*$  such that  $\mathbf{e}^T \mathbf{h}^* = \mathbf{e}^T \tilde{\mathbf{h}}$  and  $\mathbf{r}^* \geq \mathbf{0}$ . The minimization problem to be solved is:

$$\min_{\mathbf{z}} \mathcal{J}(\mathbf{z}) \stackrel{\text{def}}{=} \frac{1}{2}(\mathbf{z} - \tilde{\mathbf{z}})^T \mathbf{P}^{-1}(\mathbf{z} - \tilde{\mathbf{z}}) + \frac{1}{2}(\mathbf{H}\mathbf{z} - \mathbf{z}^{obs})^T \mathbf{R}^{-1}(\mathbf{H}\mathbf{z} - \mathbf{z}^{obs}) \quad (5.1.13)$$

subject to

$$\mathbf{e}^T \mathbf{h} = \mathbf{e}^T \tilde{\mathbf{h}} \quad \text{and} \quad \mathbf{r} \geq \mathbf{0}. \quad (5.1.14)$$

Therefore, in our setup  $n = 750$ ,  $p = 250$ ,  $\mathbf{x} = (\mathbf{u}^T, \mathbf{h}^T)^T$  and  $\mathbf{y} = \mathbf{r}$ ,  $\mathbf{G} = \mathbf{P}^{-1} + \mathbf{H}^T \mathbf{R}^{-1} \mathbf{H}$  and  $\mathbf{g} = -\mathbf{P}^{-1} \tilde{\mathbf{z}} - \mathbf{H}^T \mathbf{R}^{-1} \mathbf{z}^{obs}$ . A natural feasible starting point is  $(\mathbf{x}_0, \mathbf{y}_0) = ((\tilde{\mathbf{u}}^T, \tilde{\mathbf{h}}^T)^T, \tilde{\mathbf{r}})$ . We use an LU decomposition with pivoting to solve (5.1.12) accurately. Table 5.1.1 illustrates the performance of the algorithm.

As illustrated in Table 5.1.1, Algorithm 5.1.1 converges in only five iterations on this example. If a more general interior point method like the CVXOPT package Andersen et al. (2010) is applied for minimization of this problem, the number of (more expensive) iterations required is typically between ten and twenty.

	$J(\mathbf{z})$	$ \mathcal{A}_k^c $	$\ [\nabla_y \mathcal{J}(\mathbf{x}_k, \mathbf{y}_k)]^{\mathcal{A}_k^c}\ $	$\alpha_k$
1	-1.799206e+03	122	3.840914e+03	1.
2	-1.803981e+03	156	4.038438e+02	1.
3	-1.805237e+03	160	3.795432e+01	1.
4	-1.805271e+03	162	5.449233E-01	0.9991
5	-1.805271e+03	162	1.875281E-11	1.

Table 5.1.1: Illustration of performance of the Algorithm 5.1.1.

Note that in our case, matrix  $\mathbf{A}$  has a very simple form ( $\mathbf{A} = (\mathbf{0}_u^T, \mathbf{e}_h^T, \mathbf{0}_r^T)$ ) and has rank one. We can therefore easily project the problem into the nullspace of  $\mathbf{A}$  by defining  $\mathbf{Z} = \mathbf{I} - \mathbf{A}^T \mathbf{A} / \mathbf{h}^2$ , the projection onto this nullspace, and applying the change of variable  $\mathbf{x} = \mathbf{Z}\tilde{\mathbf{x}}$  for  $\tilde{\mathbf{x}} \in \mathbb{R}^{n-1}$ , which leads to the problem

$$\min_{\tilde{\mathbf{x}}, \mathbf{y}} \tilde{\mathcal{J}}(\tilde{\mathbf{x}}, \mathbf{y}) \stackrel{\text{def}}{=} ((\mathbf{Z}^T \mathbf{g}_x)^T, \mathbf{g}_y^T) \begin{pmatrix} \tilde{\mathbf{x}} \\ \mathbf{y} \end{pmatrix} + \frac{1}{2} (\tilde{\mathbf{x}}^T \mathbf{y}^T) \begin{pmatrix} \mathbf{Z}^T \mathbf{G}_{\tilde{x}\tilde{x}} \mathbf{Z} & \mathbf{Z}^T \mathbf{G}_{xy} \\ \mathbf{G}_{xy}^T \mathbf{Z} & \mathbf{G}_{yy} \end{pmatrix} \begin{pmatrix} \tilde{\mathbf{x}} \\ \mathbf{y} \end{pmatrix} \quad (5.1.15)$$

subject to

$$\mathbf{y} \geq \mathbf{l}. \quad (5.1.16)$$

Problem (5.1.15)-(5.1.16) is now a bound constrained quadratic problem, to which standard techniques based on gradient projection method can be applied, including for large scale instances (see Conn et al. (1992), for example). A simple version of the resulting algorithm (based on Conn et al. (1992)) is now stated as Algorithm 5.1.2 on the following page.

	$J(\mathbf{z})$	$ \mathcal{A}_k^c $	$\ [\nabla_y \mathcal{J}(\mathbf{x}_k, \mathbf{y}_k)]^{\mathcal{A}_k^c}\ $	$\alpha_k$	CG its	faces	$\ \mathbf{z}_k - \mathbf{z}^*\ $
1	-1.794983e+03	191	9.445814E-07	6.921e-07	1216	284	8.521e-02
2	-1.804782e+03	167	2.441864E-05	7.204e-06	862	11	2.193e-02
3	-1.805271e+03	162	2.576893E-06	1.012e-05	847	7	1.392e-12

Table 5.1.2: Illustration of performance of the Algorithm 5.1.2. In this table, ‘‘CG its’’ stands for the total number of CG iterations at major iteration  $k$  and ‘‘faces’’ is the number of explored faces at iteration  $k$ . To illustrate the accuracy, the difference is calculated between result of each major iteration  $\mathbf{z}_k$  to  $\mathbf{z}^*$  an end solution of Algorithm 5.1.1 on page 85.

As mentioned earlier, Algorithm 5.1.2 could be implemented without Step 4 if mere convergence is wanted, but performing conjugate gradient iterations as suggested in Conn et al. (1992), very often significantly reduces the number of outer iterations. This was also observed for our test problem. If subproblems in Step 4 are solved accurately, Algorithm 5.1.2 requires three outer iterations, as illustrated in Table 5.1.2. If the number of conjugate gradient iterations per outer iteration is fixed *a priori* (a standard practice in weather forecasting), the number of outer iterations increases, and could reach twenty, but

**Algorithm 5.1.2: Projected QP algorithm for disjoint constraints**

**Step 0: Initialization.** A feasible starting point  $(\tilde{\mathbf{x}}_0, \mathbf{y}_0)$  is given (i.e.  $\mathbf{y}_0 \geq \mathbf{1}$ ), as well as an accuracy threshold  $\epsilon > 0$ . Compute the projection  $\mathbf{Z}$  onto the null space of  $\mathbf{A}$  and set  $k = 0$ .

**Step 1: Active set update.** Define

$$\mathcal{W}_k \stackrel{\text{def}}{=} \{i \in \{1, \dots, p\} \mid [\mathbf{y}_k]_i = l_i\} \quad \text{and} \quad \mathcal{W}_k^c \stackrel{\text{def}}{=} \{1, \dots, p\} \setminus \mathcal{W}_k. \quad (5.1.17)$$

**Step 2: Termination test.** Terminate if the following conditions hold:

- $\|[\nabla_{\mathbf{y}} \tilde{\mathcal{J}}(\tilde{\mathbf{x}}_k, \mathbf{y}_k)]^{\mathcal{W}_k^c}\| \leq \epsilon$
- $\|\nabla_{\tilde{\mathbf{x}}} \tilde{\mathcal{J}}(\tilde{\mathbf{x}}_k, \mathbf{y}_k)\| \leq \epsilon$
- $\nabla_{y_i} \tilde{\mathcal{J}}(\tilde{\mathbf{x}}_k, \mathbf{y}_k) \geq 0$  for  $i \in \mathcal{W}_k$ .

**Step 3: Find the Cauchy point and determine its active set.** Determine  $\alpha > 0$  such that  $(\tilde{\mathbf{x}}_k^c, \mathbf{y}_k^c)$  is the first minimizer of

$$\tilde{\mathcal{J}}\left(\mathbf{x}_k - \alpha \nabla_{\tilde{\mathbf{x}}} \tilde{\mathcal{J}}(\tilde{\mathbf{x}}_k, \mathbf{y}_k), \max\left[\mathbf{y}_k - \alpha \nabla_{\mathbf{y}} \tilde{\mathcal{J}}(\tilde{\mathbf{x}}_k, \mathbf{y}_k), \mathbf{1}\right]\right).$$

Set

$$\mathcal{W}_{k,C} \stackrel{\text{def}}{=} \{i \in \{1, \dots, p\} \mid [\mathbf{y}_k^c]_i = l_i\}, \quad \text{and} \quad \mathcal{C}_k \stackrel{\text{def}}{=} \{1, \dots, p\} \setminus \mathcal{W}_{k,C}. \quad (5.1.18)$$

**Step 4: Minimization beyond the Cauchy point.** Apply the CG algorithm to find an approximate minimizer  $(\tilde{\mathbf{x}}_{k+1}, \mathbf{y}_{k+1}^{\mathcal{C}_k})$  of

$$\left((\mathbf{Z}^T \mathbf{g}_x)^T, \mathbf{g}_{y^{\mathcal{C}_k}}^T\right) \begin{pmatrix} \tilde{\mathbf{x}} \\ \mathbf{y}^{\mathcal{C}_k} \end{pmatrix} + \frac{1}{2} (\tilde{\mathbf{x}}^T \mathbf{y}^{\mathcal{C}_k, T}) \begin{pmatrix} \mathbf{Z}^T \mathbf{G}_{xx} \mathbf{Z} & \mathbf{Z}^T \mathbf{G}_{xy^{\mathcal{C}_k}} \\ \mathbf{G}_{xy^{\mathcal{C}_k}}^T \mathbf{Z} & \mathbf{G}_{y^{\mathcal{C}_k} y^{\mathcal{C}_k}} \end{pmatrix} \begin{pmatrix} \tilde{\mathbf{x}} \\ \mathbf{y}^{\mathcal{C}_k} \end{pmatrix} \quad (5.1.19)$$

subject to

$$\mathbf{y}^{\mathcal{C}_k} \geq \mathbf{1}^{\mathcal{C}_k}. \quad (5.1.20)$$

Terminate the CG once one (or more) bound(s) of indices  $j_1, \dots, j_s$  are encountered, after a maximum number of iterations or once it has converged. If CG was terminated because bounds were encountered, restart it after redefining  $\mathcal{C}_k = \mathcal{C}_k \setminus \{j_1, \dots, j_s\}$ . Repeat this process until the size of  $\mathcal{C}_k$  does not decrease anymore.

a)	$J(\mathbf{z})$	$ \mathcal{A}_k^c $	$\ [\nabla_y \mathcal{J}(\mathbf{x}_k, \mathbf{y}_k)]^{\mathcal{A}_k^c}\ $	$\alpha_k$	CG its	faces	$\ \mathbf{z}_k - \mathbf{z}^*\ $
1	-7.263261e+02	22	2.675923e+05	6.921e-07	25	25	4.425e-01
2	-1.006441e+03	87	1.020320e+05	6.717e-07	25	25	4.347e-01
3	-1.232950e+03	82	4.254892e+04	5.089e-06	25	25	4.087e-01
4	-1.499552e+03	86	5.519945e+04	2.545e-05	25	25	3.388e-01
5	-1.546278e+03	82	2.679016e+04	4.146e-06	25	25	3.275e-01
6	-1.612699e+03	111	7.176333e+04	2.601e-05	25	25	2.896e-01
7	-1.662727e+03	127	1.739912e+04	2.296e-06	25	25	2.837e-01
8	-1.747531e+03	146	1.382792e+04	5.212e-05	25	25	2.345e-01
9	-1.764352e+03	159	1.176661e+04	1.912e-05	25	19	2.180e-01
10	-1.776980e+03	166	6.041509e+03	4.779e-07	25	15	1.991e-01
11	-1.790153e+03	168	8.311407e+03	3.071e-06	25	9	1.654e-01
12	-1.797375e+03	167	5.098743e+03	7.904e-07	25	9	1.322e-01
13	-1.800946e+03	165	6.175714e+03	5.926e-07	25	7	1.03e-01
14	-1.802543e+03	162	2.556256e+03	6.887e-07	25	3	8.614e-02
15	-1.803445e+03	163	2.419666e+03	7.022e-07	25	3	7.179e-02
16	-1.803929e+03	164	2.268422e+03	7.029e-07	25	4	6.274e-02
17	-1.804628e+03	166	2.016213e+03	5.976e-07	25	5	4.417e-02
18	-1.805009e+03	164	7.389449e+02	6.18e-07	25	3	2.942e-02
19	-1.805173e+03	163	5.190906e+02	8.036e-07	25	2	1.823e-02
20	-1.805230e+03	162	7.279170e+02	7.597e-07	25	2	1.012e-02
21	-1.805250e+03	162	3.956734e+02	5.145e-07	25	2	8.605e-03
22	-1.805258e+03	162	3.498790e+02	6.562e-07	25	1	6.014e-03
b)	$J(\mathbf{z})$	$ \mathcal{A}_k^c $	$\ [\nabla_y \mathcal{J}(\mathbf{x}_k, \mathbf{y}_k)]^{\mathcal{A}_k^c}\ $	$\alpha_k$	CG its	faces	$\ \mathbf{z}_k - \mathbf{z}^*\ $
1	-7.263261e+02	45	2.663518e+05	6.921e-07	50	50	4.425e-01
2	-1.167649e+03	104	5.380895e+04	6.717e-07	50	50	4.196e-01
3	-1.474787e+03	103	2.454138e+04	1.569e-05	50	50	3.490e-01
4	-1.704972e+03	144	4.199822e+04	7.083e-05	50	50	2.468e-01
5	-1.762680e+03	161	8.488606e+03	3.372e-06	50	39	2.221e-01
6	-1.790840e+03	171	1.082978e+04	1.369e-06	50	22	1.615e-01
7	-1.801730e+03	168	1.130804e+04	7.58e-07	50	7	8.449e-02
8	-1.805271e+03	167	1.423371e+03	7.466e-07	50	7	4.866e-02
9	-1.805064e+03	164	1.374224e+03	7.58e-07	50	6	2.496e-02
10	-1.805221e+03	164	9.809614e+02	6.055e-07	50	2	1.112e-02
11	-1.805263e+03	163	4.830760e+02	9.67e-07	50	2	3.722e-03
12	-1.805267e+03	162	8.241262e+01	6.635e-07	50	3	2.662e-03
13	-1.805267e+03	162	4.826981e+01	7.39e-07	50	1	2.414e-03

c)	$J(\mathbf{z})$	$ \mathcal{A}_k^c $	$\ [\nabla_y \mathcal{J}(\mathbf{x}_k, \mathbf{y}_k)]^{\mathcal{A}_k^c}\ $	$\alpha_k$	CG its	faces	$\ \mathbf{z}_k - \mathbf{z}^*\ $
1	-1.794806e+03	189	1.243605e+03	6.921e-07	400	282	9.172e-02
2	-1.804771e+03	169	8.766907e+01	7.032e-06	400	12	2.224e-02
3	-1.805271e+03	162	5.298667e+00	1.001e-05	400	3	9.024e-05
4	-1.805271e+03	162	1.756899e-02	7.86e-07	400	1	3.036e-07
d)	$J(\mathbf{z})$	$ \mathcal{A}_k^c $	$\ [\nabla_y \mathcal{J}(\mathbf{x}_k, \mathbf{y}_k)]^{\mathcal{A}_k^c}\ $	$\alpha_k$	CG its	faces	$\ \mathbf{z}_k - \mathbf{z}^*\ $
1	-1.794983e+03	191	3.675456e+00	6.921e-07	800	284	8.521e-02
2	-1.804782e+03	167	1.621029e-04	7.204e-06	800	11	2.193e-02
3	-1.805271e+03	162	8.003332e-06	1.012e-05	800	7	1.262e-10

Table 5.1.3: Illustration of performance of the Algorithm 5.1.2 when the maximum number of CG iterations per major iteration is fixed to a) 25, b) 50, c) 400 and d) 800 respectively. The notation follows that of Table 5.1.2.

the cost of each outer iteration decreases. The behavior of the algorithm with the number of CG iterations fixed *a priori* to 25, 50, 400 and 800 is illustrated in Table 5.1.3 (the algorithm is also stopped if converged). Fixing the total number of CG iterations per outer iteration limits number of CG restarts during one major iteration and reduces accuracy as well as cost. For example, for a fixed number of 25 CG iterations, the solutions obtained by both algorithms only coincide to two significant digits, while if 800 CG iterations are allowed, they share ten significant digits. When allowing the number of CG iterations to increase from 25 to 800, the total number of CG iterations performed increases from 550 to 2400 and reaches 3172 in case where no limit is set, while the cost increases by 75%.

#### 5.1.4 Performance summary

The first of our methods, Algorithm 5.1.1, is more efficient than an interior point approach on this representative example, but still requires solving the KKT system (5.1.12), which is impractical in weather forecasting applications due to problem size and frequency of solution. By contrast, Algorithm 5.1.2 exploits the low rank of the linear equality constraints and uses a well-known iterative approach to compute an approximate solution while ensuring feasibility. If the size of the problem is such that the conjugate gradient algorithm is allowed to converge, the number of outer iterations required by Algorithm 5.1.2 is smaller or comparable to that required by Algorithm 5.1.1. If the number of conjugate gradient iterations is limited from the start (as is often the case in weather forecasting applications), the number of outer iterations typically increases and finding the optimal equilibrium between accuracy and cost then depends on the problem at hand. A further advantage of Algorithm 5.1.2 is that it applies the conjugate gradient to a subproblem whose size is significantly smaller than that of the KKT system (5.1.12) (remember that  $p \approx n/3$ ).

## 5.2 Neural Network to replace QPEns

Artificial neural networks, further referred to as neural networks (NN), are powerful tools to approximate arbitrary non-linear functions (Nielsen, 2015). A NN learns to recognize patterns based on example, rather than being explicitly programmed. An important advantage is that no direct knowledge of the function is needed. Instead, a data set consisting of input-output pairs is used to train the NN to predict the output corresponding to a given input. Especially in the fields of image recognition and natural language processing, NN's are state-of-the-art and have become a standard tool (LeCun Yann et al., 2015). In numerical weather prediction NN's are not yet fully integrated, though interest is rising quickly. Explored applications include (but are not limited to) post processing of raw model output based on observations (Rasp and Lerch, 2018), representing subgrid processes in weather and climate models using high resolution model simulations (Krasnopolsky et al., 2013; Rasp et al., 2018), combining NN with a knowledge based model as a hybrid forecasting approach (Pathak et al., 2018b) and replacing the numerical weather prediction model all together (Dueben and Bauer, 2018; Pathak et al., 2018a).

Fully replacing data assimilation by a NN has been attempted by Cintra and de Campos Velho (2014). They trained on a cycling data set produced by the LETKF and show that the trained NN performs nearly as good as the LETKF with significantly reduced computational effort. We aim to produce better results than standard data assimilation algorithms at minimal additional computational costs, by training on data produced by the QPEns. The work presented in this section is joint work with Stephan Rasp.

### 5.2.1 What is a neural network?

A NN is a function that maps a given input  $\mathbf{x} \in \mathbb{R}^n$  to a desired output  $\mathbf{y} \in \mathbb{R}^m$ . This function has a network infrastructure consisting of a sequence of layers, including one *input layer*, a number, say  $L$ , of *hidden layers*, and one *output layer*. The nodes in the input layer  $\mathcal{N}_{0,j}, j \in \{1, 2, \dots, n\}$  serve to receive the user's input  $\mathbf{x}$  and pass it on to the first hidden layer  $\hat{y}_{0j} \leftarrow x_j, j \in \{1, 2, \dots, n\}$ . Nodes of a hidden layer  $\mathcal{N}_{i,j}, i \in \{1, \dots, L\}, j \in \{1, 2, \dots, J_i\}$  receive an affine transformation of the output of the previous layer  $\hat{x}_{i,j} = \sum_{l=1}^{J_{i-1}} \hat{y}_{i-1,l} w_{i-1,l,j} + b_{i-1,j}$  with coefficients  $w_{i-1,l,j}$  and  $b_{i-1,j}$  denoted as *weights* and *bias* respectively, and pass it through an *activation function*  $g_{ij} : \mathbb{R} \rightarrow \mathbb{R}$  to produce output  $\hat{y}_{ij}$ . The output layer is similar as a hidden layer, but produces the output  $\mathbf{y}$ . A graphical description of a NN is presented in Figure 5.2.1.

### 5.2.2 Convolutional neural network

In our application it is unnecessary to fully connect all elements of input  $\mathbf{x}$  to all elements of output  $\mathbf{y}$ , as both vectors have a correlation length scale significantly smaller than their full size. One could train a NN for each grid point separately, but this becomes cumbersome as the number of grid points increases. Since for our application all grid points are statistically homogeneous, training one grid point would be enough, but as soon a local factors such as

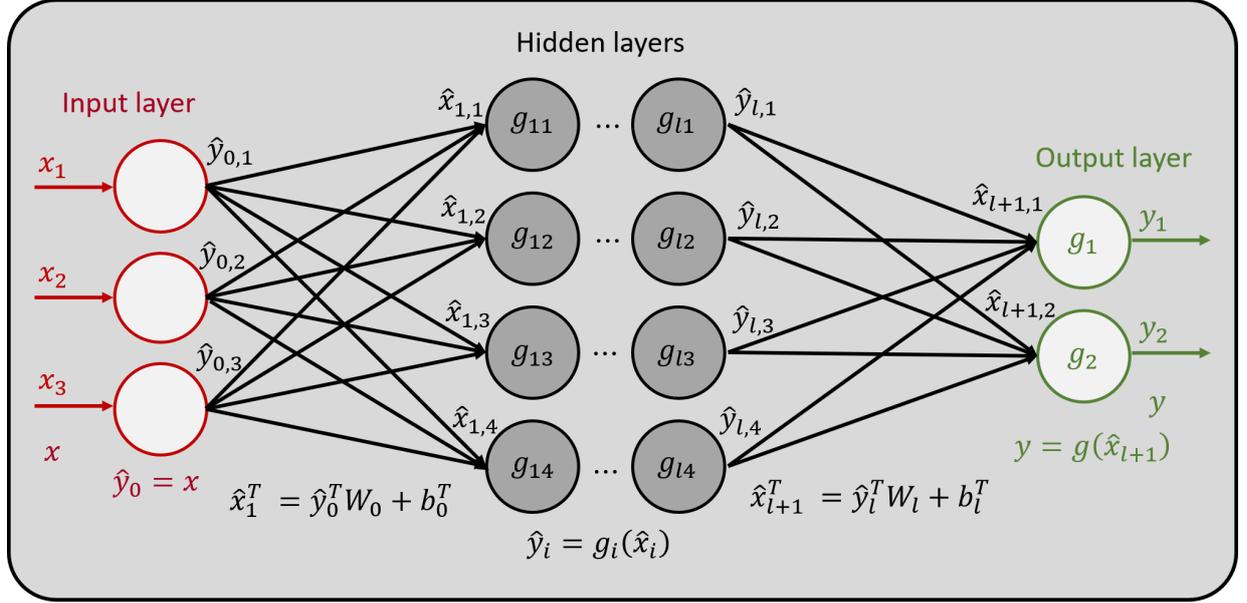


Figure 5.2.1: Schematic illustration of a neural network.

ography play a role, the advantage is lost. Convolutional neural networks (CNN) offer a natural framework for localization and detecting spatial correlations, thereby not wasting computational resources on training redundant node connections.

CNN's include *convolutional layers* consisting of a number of *filters*  $\mathcal{F}$ , each characterized by a *kernel*

$$\mathcal{K}(r, l, l^{\mathcal{F}}) = \begin{cases} W_{l+r, l^{\mathcal{F}}}, & \text{if } -r \leq l \leq r \\ 0, & \text{otherwise} \end{cases} \quad (5.2.1)$$

where  $\mathbf{W} \in \mathbb{R}^{R \times m}$  is a weight matrix,  $R = 2r + 1 \in \mathbb{N}$  the number of elements in the support of the kernel, which we refer to as the kernel size, and  $m$  the number of input vectors  $\hat{\mathbf{y}} \in \mathbb{R}^n$ , which is determined by the number of filters in the previous layer. A filter computes the convolution of its input  $\hat{\mathbf{Y}} \in \mathbb{R}^{n \times m}$  and its kernels and adds a bias  $\mathbf{b}$ :

$$\hat{x}_j = (\hat{\mathbf{Y}} * \mathcal{K})[j] \stackrel{\text{def}}{=} \sum_{l^{\mathcal{F}}=1}^m \sum_{l=-r}^r \hat{Y}_{j-l, l^{\mathcal{F}}} \mathcal{K}(r, l, l^{\mathcal{F}}) + b_j, \quad (5.2.2)$$

see Figure 5.2.2. So a filter processes localized input data, where the localization radius is  $r$  and the localization function is  $\mathcal{K}(r, l)$  with  $l$  being the distance  $|i - j|$  between the input element  $\hat{y}_i$  and output element  $\hat{x}_j$ . Let us assume that all filters  $\mathcal{F}_{i, i^{\mathcal{F}}}, i^{\mathcal{F}} \in \{1, 2, \dots, N_i^{\mathcal{F}}\}$  in convolutional layer  $i$  have the same size  $R_i = 2r_i + 1$  and that the output of the previous layer  $\hat{\mathbf{Y}}_{i-1}$  has shape  $n_{i-1} \times m_{i-1}$ . Each filter convolves  $\hat{\mathbf{Y}}_{i-1}$  with its corresponding kernel  $\mathcal{K}_{i, i^{\mathcal{F}}}$  according to (5.2.2), yielding matrix  $\hat{\mathbf{X}}_i \in \mathbb{R}^{n-iR_i \times N_i^{\mathcal{F}}}$ . An activation function is applied to each of the elements to obtain  $\hat{\mathbf{Y}}_i$ .

The number of layers, the size of the kernels, the number of filters per layer and the activation functions are predetermined by the user. The corresponding weight and bias matrices are learned by the CNN in the training process. Note that we have described a one dimensional CNN, as our model is one dimensional. In practice CNN's are mostly applied to two or even three dimensional objects, in which case the spacial dimensions of the kernels is augmented accordingly.

Unfortunately, mathematically based guidelines about the effect of different CNN architectures are still lacking. Intuitively, the complexity of the function and the amount of data available are important factors to choose the total number of filters. How to organize the CNN given a total number of filters on the other hand, is more complex. Studies indicate that deep CNNs (many hidden layers) are more efficient than shallow CNNs (few hidden layers), but finding a good balance remains a heuristic exercise. Guidelines on the choice of activation functions also remain vague. It is known that activation functions corresponding to hidden layers should be nonlinear on at least part of the domain  $(-\infty, \infty)$  to allow nonlinear relations between the input  $\mathbf{x}$  and output  $\mathbf{y}$ . For choosing the activation functions of the output layer one could think about the desired range of the output  $\mathbf{y}$ . For example, we could use the so called “relu” function, which is the identity ( $x \mapsto x$ ) on the positive axis and zero ( $x \mapsto 0$ ) on the negative axis, to yield a non-negative output  $\mathbf{y}$ .

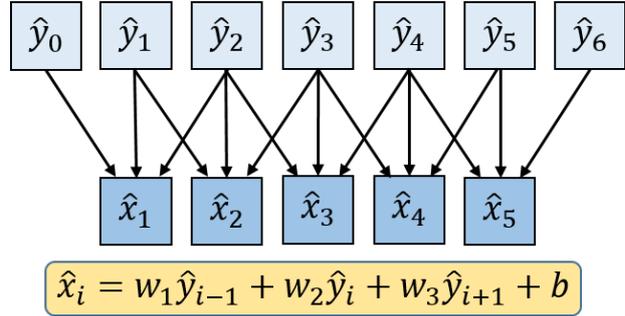


Figure 5.2.2: Example of a filter operation with  $R = 3$ ,  $m = 1$ ,  $n = 7$ .

### 5.2.3 Training a neural network

Training a NN is an iterative process that minimizes a loss function  $f$

$$\min_{\mathbf{W}, \mathbf{b}} f(\mathcal{M}(\mathbf{X}^{tr}, \mathbf{W}, \mathbf{b}), \mathbf{Y}^{tr}) \quad (5.2.3)$$

where  $\mathcal{M}$  is the function defined by the NN with weights  $\mathbf{W}$  and biases  $\mathbf{b}$  and  $(\mathbf{X}^{tr}, \mathbf{Y}^{tr})$  is the training data set composed of  $N^{tr}$  input-output pairs  $(\mathbf{x}^{tr}, \mathbf{y}^{tr})$ . The most commonly used algorithm to solve (5.2.3) is stochastic gradient descent, where one iteratively moves the decision variables (in this case  $\mathbf{W}$  and  $\mathbf{b}$ ) in direction of the negative gradient  $-\nabla_{\mathbf{W}, \mathbf{b}} f(\mathcal{M}(\mathbf{X}^{tr}, \mathbf{W}, \mathbf{b}), \mathbf{Y}^{tr})$ . To avoid any ill conditioning it is recommended to train the NN on normalized data. In NN language one refers to computing  $\mathcal{M}(\mathbf{X}^{tr}, \mathbf{W}, \mathbf{b})$  as *forward propagation* and computing the gradient as *backward propagation*. Since the training data set is usually large, it is split up into *batches*. Backward propagation is done subsequently for all batches, completing an *epoch*. A straightforward loss function would be the mean

squared error (MSE) defined as

$$f(\mathcal{M}(\mathbf{X}^{tr}, \mathbf{W}, \mathbf{b}), \mathbf{Y}^{tr}) \stackrel{\text{def}}{=} \frac{1}{N^{tr}n} \sum_{j=1}^{N^{tr}} \sum_{i=1}^n (\mathcal{M}(X_{ij}^{tr}, \mathbf{W}, \mathbf{b}) - Y_{ij}^{tr})^2. \quad (5.2.4)$$

However, any loss function fitting the problem at hand can be used. For example, even though physical properties like mass conservation cannot be imposed as a hard constraint, one could add a term to the loss function to penalize any constraint violation in a weak sense.

### 5.2.4 Experiment setup

The experiments in this section are a first exploration of the possibility of combining neural networks with data assimilation. For this reason, we did not spend a lot of effort on maximizing the performance of our CNNs. The presented results are, therefore, to be seen only as a first step in this research direction. We test two different data assimilation settings to investigate the sensitivity to the performance gap between the EnKF and QPEns. The training data is produced with the modified shallow water model.

#### 5.2.4.1 Data assimilation settings

The data assimilation settings should be such that the QPEns performs better than the EnKF. We fixed the number of ensemble members to 50, the localization radius to 6 grid points and the number of additional wind measurements to 10%. We distinguish between two cases, one where the EnKF diverges and the QPEns converges and one where both algorithm converge. To accomplish this we set the time between assimilation cycles to 30 minutes and 5 minutes respectively.

#### 5.2.4.2 Training data

We aim to produce initial conditions of the same quality as the ones produced by the QPEns by upgrading the initial conditions produced by the EnKF. To that end, we generated data sets which includes the EnKF and QPEns cycling data  $\{(\mathbf{E}_t^f, \mathbf{E}_t^a) : t = 1, 2, \dots, T\}$  and  $\{(\mathbf{Q}_t^f, \mathbf{Q}_t^a) : t = 1, 2, \dots, T\}$ , where  $\mathbf{E}$  stands for EnKF and  $\mathbf{Q}$  for QPEns, the superscript  $f$  denotes the background and  $a$  the analysis. In parallel we created the data set  $\{\mathbf{X}_t^a : t = 1, 2, \dots, T\}$ , where  $\mathbf{X}_t^a$  is the unconstrained solution calculated from  $\mathbf{Q}_t^f$ . All three data sets contain the entire ensemble of  $N_{ens} = 50$  members, such that  $(*)^{(*)}_t \in \mathbb{R}^{N_{ens} \times n \times 3}$ , where the last dimension represents the 3 variables ( $u, h, r$ ) and  $n$  is the number of grid points. The observations  $\{O_t : t = 1, 2, \dots, T\}$  are also available to us.

The output of our training set  $\mathbf{Y}^{tr} \in \mathbb{R}^{N_{ens}T \times n \times 3}$  is simply a reshaped and normalized version of the data set  $\{\mathbf{Q}_t^a : t = 1, 2, \dots, T\}$ . For the input of our training set  $\mathbf{X}^{tr}$  we chose to use both the observations  $\{\mathbf{O}_t : t = 1, 2, \dots, T\}$  and the unconstrained solutions  $\{\mathbf{X}_t^a : t = 1, 2, \dots, T\}$ , yielding  $\mathbf{X}^{tr} \in \mathbb{R}^{N_{ens}T \times n \times 6}$ , where an observation vector  $\mathbf{O}_t$  is copied

$N_{ens}$  times to obtain  $\mathbf{O}_t^* \in \mathbb{R}^{N_{ens} \times n \times 3}$  and unobserved elements are given a value of -1 to match the required shape. For  $u$  and  $h$  the input and output data set is normalized by subtracting the climatological mean before dividing by the climatological standard deviation. For  $r$ , we do not subtract the climatological mean to maintain positivity.

As our initial background state is generated from an ensemble of random initial conditions, it takes several data assimilation cycles for the filter to converge. Consequently, the difference between the unconstrained solution (EnKF) and the constrained solution (QPEns) is “time dependent during the first few cycles. To allow the CNN to learn this time dependency we need to feed it data that includes information on the stage of the corresponding cycling sequence. This could for example be a combination of the background and observations or the unconstrained initial conditions  $X^a$  so that the CNN can detect how accurate the background is. Another way is to explicitly include the time index  $t$  to each input, which is what we do here. Since this time dependency is only an issue for the first few cycles, we train a separate CNN for  $t \leq t_0$ , where  $t_0 = 10$ , which we refer to as CNN-0. To that end we generated 1000 additional experiments running from  $t = 0$  to  $t = t_0$  with different random number seeds to create the training data set for CNN-0. The CNN for  $t > t_0$ , which will simply be referred to as CNN does not require any special treatment.

A validation data set exactly as the training data set but with a different random seed number is created to monitor the training process. For both the training data set and the validation data set the size is set to  $T = 10,000$ .

### 5.2.4.3 Fully convolutional neural network (CNN)

We chose to use a CNN with 3 convolutional hidden layers, consisting of 32 filters each with kernels of size 5 and the “relu” activation function

$$g(x) = \begin{cases} x, & \text{for } x \geq 0 \\ 0, & \text{for } x < 0 \end{cases}.$$

The output layer is a convolutional layer as well, where the number of filters is determined by the desired shape of the output of the CNN, which is a model state  $(u, h, r) \in \mathbb{R}^{n \times 3}$ . The output layer has therefore 3 filters and the kernel size is again 5. Note that the “localization radius”, that is, the maximum influence radius of a variable as assumed by the CNN is  $(5 - 1)/2 * 4 = 8$ , where 4 is number of layers and 5 the kernel size. We use a linear activation function for  $u$  and  $h$  and the “relu” activation function for  $r$  to ensure non-negativity of rain. We set the batch size to 4096 and do 25 epochs. These choices are more or less arbitrary as our focus is not on optimizing the CNN architecture and training process. The training is done with python library Keras (Chollet et al., 2015).

## 5.2.5 Results

We assign the name  $E30$  to the data set corresponding to a cycling period setting of 30 minutes, and  $E5$  to the data set corresponding to a cycling period setting of 5 minutes.

Table 5.2.1 summarizes what the CNN has learned. For  $E30$  the CNN was able to reduce the gap between the constrained and unconstrained solution for all variables. This is not the case for  $E5$ , where the CNN actually increases the gap for variable  $u$ . This can be explained by the observation that the RMSE between the input and output is smallest for  $u$ . Note that in terms of MSE (which is the CNN's loss function) this is even more pronounced, making  $u$  a low priority for the CNN. In both data sets, the CNN is able to remove the bias in  $h$  almost entirely. For  $u$  the bias is increased, but the bias in  $u$  and  $r$  is an order of magnitude smaller than for  $h$ . The total improvement in terms of relative RMSE is 40% for  $E30$  and 27% for  $E5$ . The CNN is better at reducing the gap between the constrained and unconstrained solution for  $E30$ , most likely because the increments are significantly larger. (Again, this is emphasized even more when the increments are measured in MSE.) For CNN-0 the total improvement is 52% for  $E30$  and 37% for  $E5$ . In what follows we focus on the CNNs for  $t > t_0$ .

	<b>u</b>	<b>h</b>	<b>r</b>	bias <b>u</b>	bias <b>h</b>	bias <b>r</b>
Input	7.5e-02	1.7e-01	1.4e-01	1.8e-02	1.0e-01	1.8e-02
Prediction	7.1e-02	9.8e-02	6.9e-02	2.6e-02	9.0e-03	2.3e-03
Improvement (%)	5	44	50	-43	91	88
Input	2.6e-02	4.0e-02	6.0e-02	1.9e-03	1.3e-02	1.8e-03
Prediction	3.1e-02	3.8e-02	2.7e-02	3.1e-03	5.8e-05	1.9e-03
Improvement (%)	-17	5.2	55	-65	100	-4.2

Table 5.2.1: Mean RMSE and bias of the variables (columns) of input  $\mathbf{X}^{tr}$  (top row) and the CNN prediction (middle row) with respect to the output  $\mathbf{Y}^{tr}$  for a random validation data set that was not used for training. The last row shows the improvement of the prediction towards the output compared to the input. The top table corresponds to  $E30$ , the bottom table to  $E5$ .

Figure 5.2.3 shows the value of the loss function for the training and validation data set as function of epochs for  $E30$  and  $E5$  respectively. It should be noted that the training loss is the average over the loss of the batches of the training data set computed during the forward propagation. The validation loss is the loss over the entire validation data set computed after the weights have been updated. Consequently, the validation loss is likely to be smaller than the training loss when the learning curve of the CNN is steep. We see that this is indeed the case for both data sets. The red curve does not reduce beyond the blue curve which means the CNN is under fitting. This indicates that the CNNs used to compute the entries of Table 5.2.1 is not optimal for the given training data. Possible approaches to further optimize the CNNs are simply training them further (for example by performing more epochs or optimizing the backpropagation algorithm), or introduce more complexity in their architecture (for example more layers and/or nodes).

What we really care about is how the CNNs perform when applied within the data assimilation cycling. We compare performance of the EnKF, QPEs and the CNN applied to the initial conditions computed by the EnKF in Figures 5.2.4-5.2.6. The CNN is able to

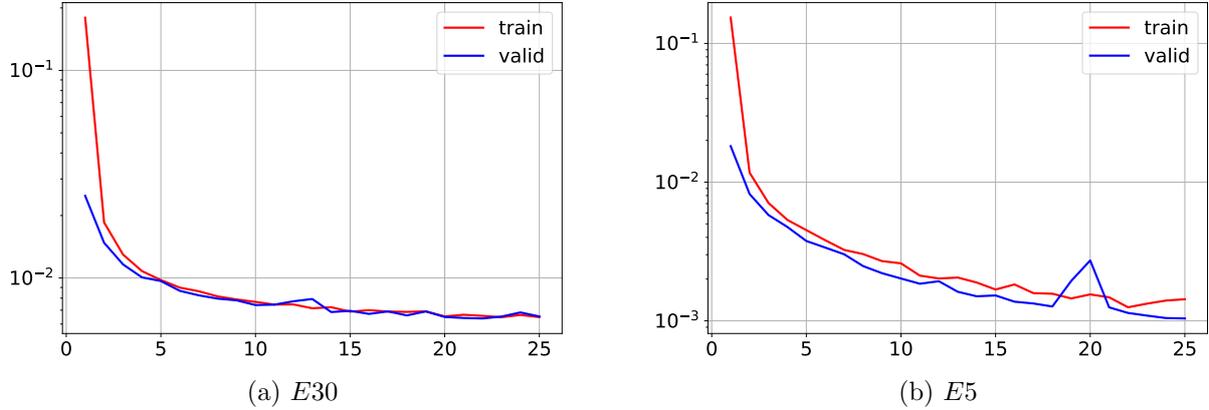


Figure 5.2.3: Value of the loss function for the training (red) and validation (blue) data set as function of epochs for  $E30$  (left) and  $E5$  (right).

reduce the gap between the EnKF and the QPEnS completely for the RMSE, the spread and the mass of  $r$  for both data sets, despite the imperfection of the CNN as illustrated in Table 5.2.1. The mass constraint of  $h$  is also almost satisfied when the CNN is applied. So how can this be? We found that applying the CNN only to  $h$  yields comparable results as shown in Figures 5.2.4-5.2.6. When applying the CNN only to  $u$  and  $r$ , the CNN became redundant. We therefore hypothesize that the bias correction done by the CNN in  $h$  severely reduces the accumulation of spurious convection that follows from using the EnKF, which feeds back to all variables.

## 5.3 Conclusion

We have presented two approaches to deal with the computational expense of the QPEnS. First, we suggested two projection algorithms which exploit the disjoint nature of constraints typically occurring in weather forecasting applications. While projection methods may be inefficient when the combinatorial aspect of selecting the correct active bounds dominate and many faces need to be explored at each major iteration, they do perform well compared to interior point algorithms when the gradient quickly provides a good identification of the active constraints. Indeed, interior points methods handle inequality constraints as penalty terms in the costfunction, thereby ceasing the need to find the active set. Projection methods on the other hand, solve a smaller problem per iteration due to the dismissal of inactive constraints. In our representative application, the active set is found efficiently, making the proposed algorithms superior to interior points methods. The results are encouraging (and have already spurred some interest from the weather forecasting operational centers), but we are aware that adapting the proposed method(s) to a real production environment remains a significant task, as preconditioning and the details of the face changing mechanism will need thought and fine tuning.

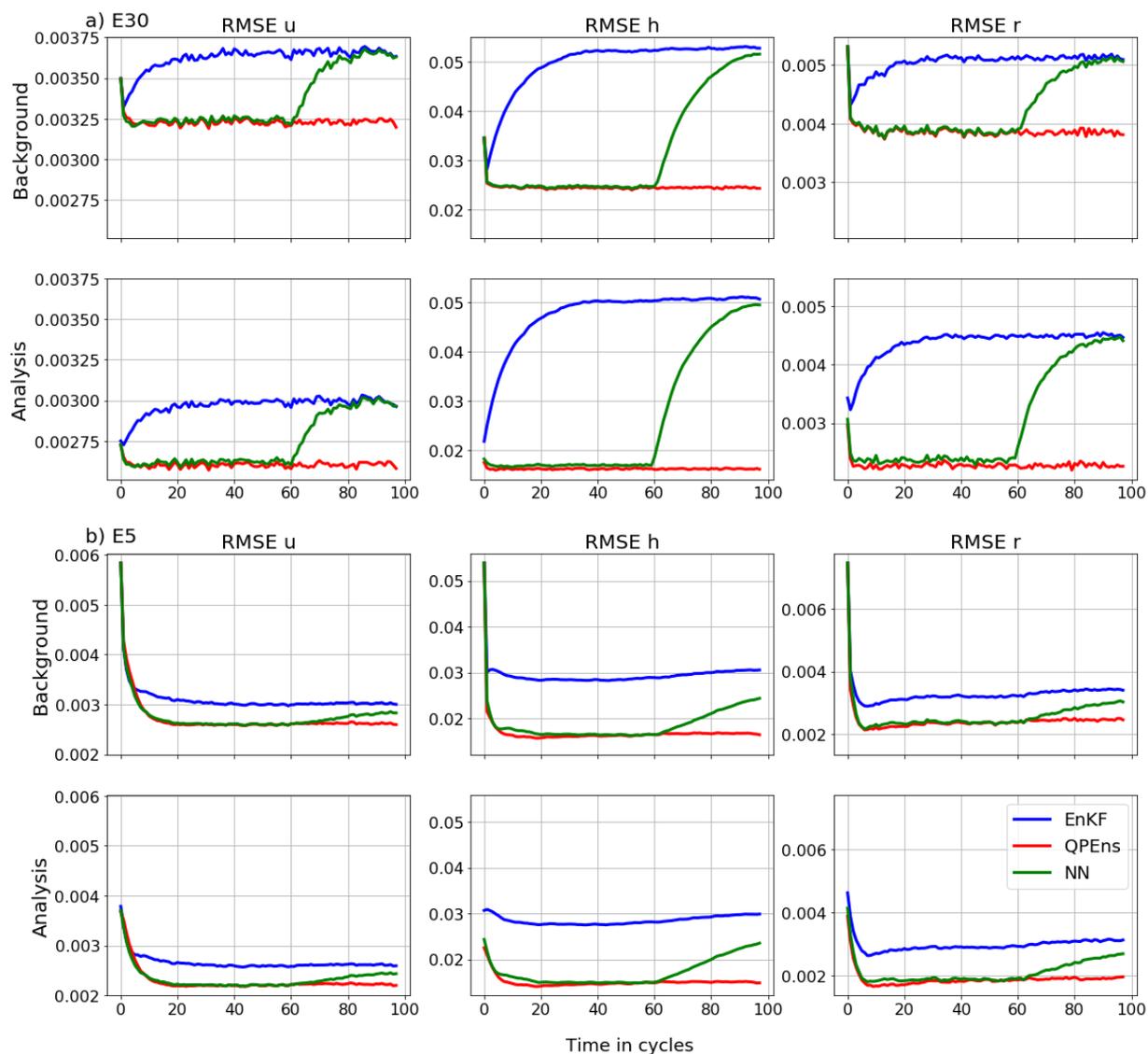


Figure 5.2.4: RMSE of ensemble mean of the variables (columns) for the first guess (top row) and analysis (bottom row) as a function of cycles for the EnKF (blue), the QPEns (red) and the EnKF with CNN (green). After 60 cycles the CNN is no longer applied to highlight its importance. The top plot corresponds to  $E30$  and the bottom plot to  $E5$ .

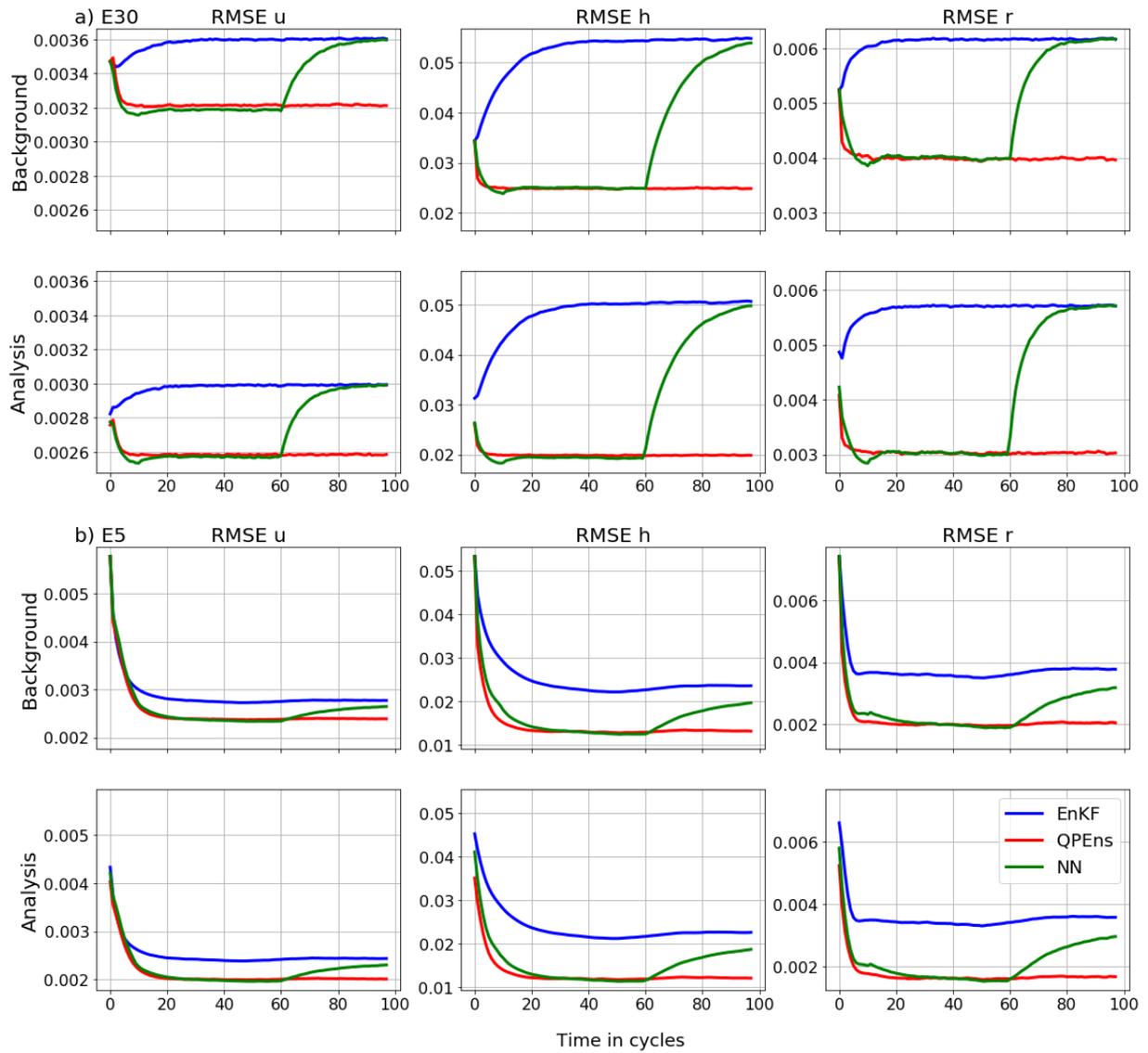


Figure 5.2.5: As Figure 5.2.4, but for the spread.

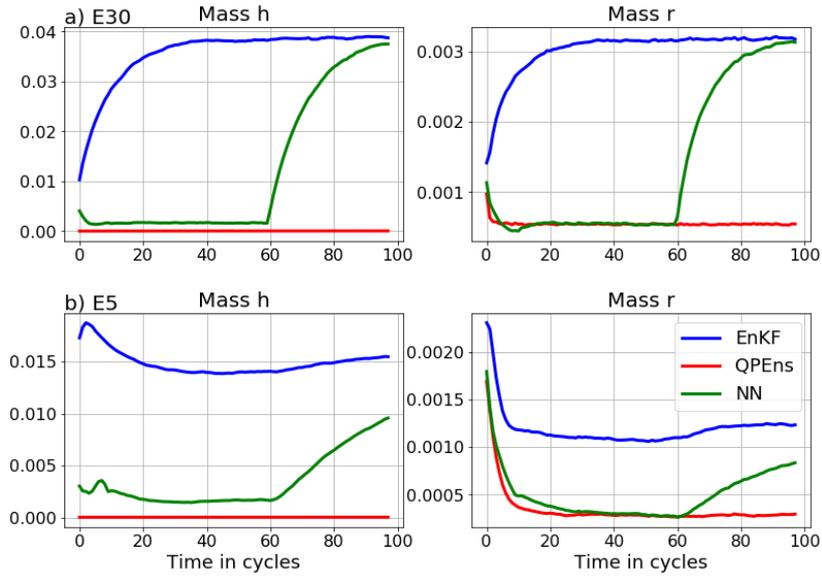


Figure 5.2.6: As Figure 5.2.4, but for the mass. Note that since the model is conserving mass, the first guess and analysis of  $h$  are identical, so only the analysis is shown.

Second, we proposed the use of a CNN to reproduce the initial conditions generated by the QPEs from the corresponding unconstrained initial conditions. We showed that reducing the bias in the  $h$  variable was enough to yield analyses and backgrounds with a similar RMSE and spread as the QPEs. We found that training the CNN to reduce the relative gap between the input and output given a certain training data size, was easier for  $E30$ , where the increments are larger. For  $E5$  the CNN was only able to reduce the bias in  $h$  and the RMSE of  $r$ . However, with neural network architectures better customized to this application, we believe the training process can be optimized with respect to required time and data size. In addition, our results suggest that a bias correction in  $h$  was the key to good cycling performance. Since in general biases tend to accumulate over time, we could expect similar benefits from bias corrections in operational applications. Fortunately, bias should be an easy quantity to learn for a NN, thus making this approach very appealing.

# Chapter 6

## Conclusion

In this thesis we discussed the potential of reducing model error by estimating static model parameters with the augmented state approach. We explored two different experiment setups. In Chapter 3 an idealized setup with the modified shallow water model (Würsch and Craig, 2014) was used to investigate the benefits of two recently developed data assimilation algorithms that alleviate the Gaussian assumption of the EnKF. These algorithms were the computationally expensive QPEs, which was designed to satisfy imposed constraints such as mass conservation and non-negativity of precipitation, and the QF, which includes higher order moments in the calculation of the initial conditions. The same setup was used in Chapter 5, where the focus lied on reducing the computational costs of the QPEs. The purpose of the second experiment setup was to investigate convective scale parameter estimation in near operational setup. In Chapter 4 the regional COSMO-KENDA weather prediction model employed at the German Weather Service was used to estimate the two dimensional roughness length parameter appearing in the boundary layer scheme.

The key challenges of convective scale parameter estimation that were widely discussed in this thesis are managing the ensemble spread of the parameters, handling sampling errors and dealing with non-Gaussian error statistics. Even though we have not fully resolved these challenges, we did take a solid step forward and presented clear beneficial impact of our approaches. In addition we provided various suggestions for potential further improvements.

To manage the ensemble spread of the parameters, we chose to design a dynamical model in the form of stochastic perturbations and apply it to the parameters after each analysis update. This dynamical model is meant to represent the uncertainty of a targeted model error. For bounded global parameters, we found the Beta distribution convenient, as by choosing the two degrees of freedom appropriately, one can regulate the spread, while ensuring feasible parameter values. For local parameters we argued that multivariate distributions should be considered, making the Gaussian distribution an accessible choice. For the roughness length, which is per definition not a spatially smooth parameter, we introduced spatial correlations anyway to represent the model error related to surface fluxes. A correlation length scale of 25 grid points was concluded to be too large. A correlation length scale of 5 grid points, which is considered to be the smallest resolvable scale by the

numerical weather prediction model, was superior for the prediction of precipitation. Yet, for the prediction of clouds no correlation at all yielded even better results. The optimal correlation length scale might depend on the density of relevant observations assimilated such as surface wind measurements. Another tunable aspect of the dynamical model is the spread regulation. In both the idealized and the near operational setup we chose the distribution parameters such that the spread is 25 % of the mean value of the model parameters. This is arbitrary and should be investigated further. Indeed, where and when the model error is larger, the spread should also be larger.

Sampling errors are an issue for both the state variables and the parameters. For the roughness length we chose to apply the same localization technique as for the state variables. Though our results were positive, state variables typically have a larger correlation length scale than parameters. Therefore, using the same localization radius for the parameters does not sufficiently reduce the sampling errors. On the other hand, with a sparse observation network using a narrow localization radius will leave the roughness length in part of the domain invariant. The dynamical model can address parts of this problem, for example by introducing a potentially artificial correlation length scale. We showed that this is especially helpful when the observation density is low, which is the case in the Southern part of the COSMO-DE domain. Also, when and where the model variables are less sensitive to the parameters, the spread could be reduced to avoid spurious parameter updates. A concrete suggestion is to regulate the spread of the roughness length based on the diurnal cycle. We also suggested using statistical resampling methods to identify spurious correlations and adjusting the spread accordingly. Though potentially effective, these approaches alleviate the negative effect of spurious correlations, rather than target the core of the problem. For global parameters we proposed dividing the correlation between the parameters and the state variables by the number of grid points, which worked well in the idealized setup. A downside is the loss of flexibility. Indeed, the evolution of the parameters is severely slowed down, thereby failing to address model errors on small time scales. As an alternative we suggested sampling error correction (Anderson, 2012) for rapidly varying model error.

To deal with non-Gaussian error statistics we evaluated two recently developed modifications of the EnKF in an idealized setup for the joint state and parameter problem. The QPEns (Janjić et al., 2014), which solves constrained minimization problems to conserve physical properties of the model in the computation of the initial conditions, and the QF (Hodyss, 2011), which considers the skewness in addition to the covariance of the error statistics. When applied to both the state and the parameters, we found that the QF requires a large ensemble size to beat the EnKF. We suggested to try to use a smaller localization radius for the skewness and kurtosis than for the covariance, to weaken the ensemble size requirement. On the other hand, when the QF is applied to the parameters only, already for a small ensemble size the parameters are estimated more accurately, which feeds back positively to the state. In contrast, the QPEns, was of no direct use to the parameters as they did not violate any constraints, but a clear benefit was found for the state, which then fed back positively to the parameters. Though superior to all tested algorithms for realistic ensemble sizes, the QPEns lacks computational efficiency.

We proposed exploiting the disjointness of the bound and equality constraints, which reduced the computational costs. We also explored the possibility of training a convolutional neural network to reproduce the initial conditions obtained from the QPEs and tested it in the idealized setup. Although the neural network did not fully learn to transform the unconstrained to the constrained solution, it was enough to obtain similar results as the QPEs when applied in the data assimilation cycle.

Overall we saw a clear positive effect of parameter estimation in both the idealized and the near operational setup. We found that the roughness length increments follow the diurnal cycle of the momentum surface flux, suggesting that the evolution of the roughness length is dominated by wind measurements. In particular, we hypothesize that the better scores in the North indicate the importance of proper use of surface wind measurements. We believe that further improvements can be achieved by applying the QF for updating the roughness length, synchronizing the spread of roughness length to the diurnal cycle and reducing spurious correlations via sampling error correction. The QPEs showed great potential for state estimation and reducing the computational costs further should definitely be on the research agenda.

The idea of estimating parameters with data assimilation has been around for decades, but in operational applications model errors were too large and the observing system too sparse to constrain the parameters reasonably (based on a conversation with Jeffery Anderson). In addition, ensemble based data assimilation algorithms that allow flow dependent error statistics have not been operational for very long in numerical weather prediction systems. With an increasing model resolution and an increasingly dense observation network we believe parameter estimation for improved weather prediction should be explored further. We demonstrated its benefits on a specific example. However, this technique can be applied to any parameter in any numerical prediction model as long as it includes well observed model variables that are sensitive to the chosen parameters.



# Appendix A

## Verification scores

In this appendix we define and briefly discuss verification scores that are used in this work. In what follows, let  $x_j$  indicate the  $j^{\text{th}}$  member and  $\bar{x} = \frac{1}{N_{\text{ens}}} \sum_{j=1}^{N_{\text{ens}}} x_j$  the mean of a model state ensemble,  $y^m = h(x) \in \mathbb{R}^{N_o}$  a single model state,  $y_j^m = h(x_j) \in \mathbb{R}^{N_o}$  the  $j^{\text{th}}$  ensemble member and  $\bar{y}^m = \frac{1}{N_{\text{ens}}} \sum_{j=1}^{N_{\text{ens}}} y_j^m \in \mathbb{R}^{N_o}$  the ensemble mean in observation space and  $y^o \in \mathbb{R}^{N_o}$  the corresponding observations.

### A.1 Bias

The bias of a model state is calculated with respect to the observation:

$$\text{BIAS}(y^m) = \frac{1}{N_o} \sum_{i=1}^{N_o} (y^m - y^o)_i \quad (\text{A.1.1})$$

A positive bias indicates a systematic overestimation of the observations, whereas a negative bias indicates a systematic underestimation of the observations. The amplitude is a measure of the severity of the over-or underestimation.

### A.2 Root Mean Squared Error (RMSE)

The root mean squared error (RMSE) is a frequently used measure of the errors between an estimator and the values observed:

$$\text{RMSE}(y^m) = \sqrt{\frac{1}{N_o} \sum_{i=1}^{N_o} (y^m - y^o)_i^2} \quad (\text{A.2.1})$$

The lower the RMSE, the better the estimator fits the observational data. Note that errors are penalized disproportionately, so the RMSE is sensitive to large errors.

### A.3 Spread

The ensemble spread is the average distance between a single ensemble member and the corresponding ensemble mean:

$$\text{SPREAD}(\{x_j : j = 1, 2, \dots, N_{ens}\}) = \sqrt{\frac{1}{N_{ens} - 1} \sum_{j=1}^{N_{ens}} (x_j - \bar{x})^2} \quad (\text{A.3.1})$$

The spread is a measure for the predicted uncertainty of the estimate, and should therefore be related to the RMSE, i.e. a larger RMSE should correspond to a larger spread. The spread is therefore often used to predict the magnitude of the forecast error.

### A.4 Fractions Skill Score (FSS)

The Fraction Skill Score (FSS) (Roberts and Lean, 2008) is a measure that was introduced to assess the skill of a forecast over a certain spatial scale. For instance, when the prediction of a convective event is spatially displaced by a few grid-points with respect to the observations, we might still consider it a skillful forecast. Yet, the RMSE would see a large error both where the event was predicted and where it actually took place, because the error is calculated grid-point wise. This phenomenon is typically referred to as the double penalty. The FSS avoids this by measuring the skill over a regional average. For each grid-point  $i$  the fraction of grid-points  $p_{j,i}^m$  for which  $y_j^m$  exceeds a certain threshold  $\alpha$  within a certain region of size  $n \times n$  is calculated. This is done for all ensemble members and the observations, yielding  $\{p_{j,i}^m : j = 1, 2, \dots, N_{ens}, i = 1, 2, \dots, N\}$  and  $\{p_i^o : i = 1, 2, \dots, N\}$  respectively.

$$\text{FSS}_{\alpha,n}(\{y_j : j = 1, 2, \dots, N_{ens}\}) = 1 - \frac{\overline{\frac{1}{N} \sum_{i=1}^N (p_{j,i}^m - p_i^o)^2}}{\frac{1}{N} \sum_{i=1}^N p_{j,i}^{m2} + \frac{1}{N} \sum_{i=1}^N p_i^{o2}} \quad (\text{A.4.1})$$

where  $\overline{(\cdot)}$  indicates the mean over the ensemble dimension. So the FSS compares the number of grid-points exceeding threshold  $\alpha$  within a  $n \times n$  area in the predicted field to the observation field. It can therefore be used to view the forecast skill as a function of scale ( $n$ ) and threshold ( $\alpha$ ), which is highly informative for quantities which appear in the form of cells that can be advected, such as precipitation and clouds.

### A.5 False Alarm Ratio (FAR) and Equitable Threat Score (ETS)

Both the FAR and the ETS are based on hit or miss events, where an event is defined by an observation exceeding a certain threshold  $\alpha$  (Wilks, 2006). The FAR is defined as

		Event observed	
		YES	NO
Event forecasted	YES	Hit	False alarm
	NO	Miss	Dry

Table A.5.1

the ratio of False Alarms to observed events, which for example can be applied to detect spurious convection. The score ranges from 0 to 1, with 0 being a perfect score. The ETS is a measure for the fraction of correctly predicted events, given the current climate. This score ranges from  $-1/3$  to 1, with 1 being the perfect score and 0 is the no skill level.

$$\text{FAR}_\alpha (\{y_j : j = 1, 2, \dots, N_{ens}\}) = \frac{\#\text{False alarms}}{\#\text{False alarms} + \#\text{Hits}} \quad (\text{A.5.1})$$

$$\text{ETS}_\alpha (\{y_j : j = 1, 2, \dots, N_{ens}\}) = \frac{\#\text{Hits} - X_\alpha}{\#\text{Hits} + \#\text{Misses} + \#\text{False alarms} - X_\alpha}, \quad (\text{A.5.2})$$

where  $\#\text{Hits}$ ,  $\#\text{False alarms}$ ,  $\#\text{Misses}$  denotes the total number of Hits, False alarms and Misses of the whole ensemble as defined by Table A.5.1 respectively, and

$$X_\alpha = \frac{(\#\text{Hits} + \#\text{Misses})(\#\text{Hits} + \#\text{False alarms})}{\#\text{Hits} + \#\text{Misses} + \#\text{False alarms} + \#\text{Dry}}. \quad (\text{A.5.3})$$



# Appendix B

## Proof of positive definite localized error covariance matrix

Assume  $\mathbf{C}_{(\theta, \mathbf{x})} = a\mathbf{e}\mathbf{e}^T$  for some  $a \in (0, 1)$ . In our case  $\mathbf{C}_{(\mathbf{x}, \mathbf{x})}$  is obtained by using positive definite matrix  $\mathbf{A}$  of Gaspari and Cohn for each of the 3 fields  $u, h$  and  $r$  of the model, i.e.

$$\mathbf{C}_{(\mathbf{x}, \mathbf{x})} = \begin{bmatrix} \mathbf{A} & \mathbf{A} & \mathbf{A} \\ \mathbf{A} & \mathbf{A} & \mathbf{A} \\ \mathbf{A} & \mathbf{A} & \mathbf{A} \end{bmatrix} \quad (\text{B.0.1})$$

Such a matrix is positive semidefinite since for any vector  $\mathbf{z} = [\mathbf{z}_1 \ \mathbf{z}_2 \ \mathbf{z}_3]$ ,  $\mathbf{z}^T \mathbf{C}_{(\mathbf{x}, \mathbf{x})} \mathbf{z} = (\mathbf{z}_1 + \mathbf{z}_2 + \mathbf{z}_3)^T \mathbf{A} (\mathbf{z}_1 + \mathbf{z}_2 + \mathbf{z}_3)$ , which is 0 only if  $(\mathbf{z}_1 + \mathbf{z}_2 + \mathbf{z}_3) = \mathbf{0}$ , i.e. if  $\mathbf{e}^T \mathbf{z} = 0$ , and otherwise it is positive.

For any vector  $\tilde{\mathbf{z}} = [\mathbf{z} \ y]$ ,  $\tilde{\mathbf{z}}^T \mathbf{C} \tilde{\mathbf{z}} = \mathbf{z}^T \mathbf{C}_{(\mathbf{x}, \mathbf{x})} \mathbf{z} + 2ay\mathbf{e}^T \mathbf{z} + y^2$ . If  $\mathbf{z}$  is such that  $\mathbf{z}^T \mathbf{C}_{(\mathbf{x}, \mathbf{x})} \mathbf{z} = 0$  then  $\mathbf{e}^T \mathbf{z} = 0$  and the result is  $y^2 > 0$ . If  $\mathbf{z}$  is such that  $\mathbf{z}^T \mathbf{C}_{(\mathbf{x}, \mathbf{x})} \mathbf{z} > 0$  then there exist  $a$  small enough to insure positivity of  $\mathbf{C}$ . In our experiments  $a = 1/n$  was enough for matrix  $\mathbf{C}$  to be positive semidefinite.



# Appendix C

## KKT conditions

Here we define the necessary and sufficient conditions for equality constrained and inequality constrained minimization problem respectively. These conditions are referred to as the KKT (Karush–Kuhn–Tucker) conditions corresponding to a minimization problem. For more details we refer to (Nocedal and Wright, 2006).

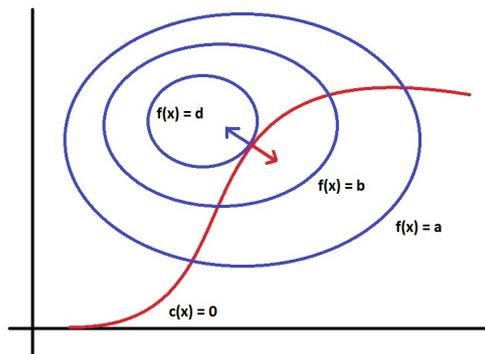
### C.0.1 Equality constrained optimization

Let us consider the following equality constrained problem

$$\begin{aligned} \min_x \quad & f(x) \\ \text{subject to} \quad & c_i(x) = 0, \quad i \in \{1, 2, \dots, m\}. \end{aligned} \tag{C.0.1}$$

Problems of type (C.0.1) are usually handled with the method of Lagrange multipliers. In this section the fundamentals of this method are explained.

Let us assume for now that  $m = 1$ . The goal is to find an  $x^*$  for which  $f(x)$  is minimized along  $c(x) = 0$ . Suppose  $x_0$  satisfies  $c(x_0) = 0$ . We want to establish whether the value of the objective function can be decreased by walking along the constraint in the neighborhood of  $x_0$ . If so, we continue to walk along the constraint until we find a point  $x^*$  for which  $f(x)$  can no longer be decreased along  $c(x) = 0$ . From the picture on the right it is clear that at  $x^*$  the respective gradients of the objective function and the constraint must be parallel. In other words, for the solution  $x^*$  holds



$$\nabla f(x^*) = -\lambda \nabla c(x^*) \tag{C.0.2a}$$

$$c(x^*) = 0 \tag{C.0.2b}$$

for some  $\lambda \in \mathbb{R}$ , which we call a *Lagrange multiplier*. The value of the Lagrange multiplier is not relevant in this context. For  $m > 1$  the right hand side of equation (C.0.2a) translates into a linear combination of the gradients of the constraints, yielding a vector of Lagrange multipliers  $\lambda \in \mathbb{R}^m$ . To incorporate these conditions into one equation we introduce the Lagrangian function, which is defined as follows:

$$\mathcal{L}(x, \lambda) = f(x) + \lambda^T c(x) \quad (\text{C.0.3})$$

where  $c(x) = [c_1(x), c_2(x), \dots, c_m(x)]^T$ . A necessary condition for a solution  $x^*$  of problem (C.0.1) is then

$$\nabla_{x,\lambda} \mathcal{L}(x, \lambda) = 0. \quad (\text{C.0.4})$$

Equation (C.0.4) defines the KKT conditions of problem (C.0.1). If the Hessian of the Lagrangian is positive definite on the null space of  $\nabla c(x^*)$ , i.e.  $p^T \nabla_x^2 \mathcal{L}(x^*, \lambda^*) p > 0$  for all  $p \neq 0$  such that  $\nabla c(x^*) p = 0$ , then all solutions of system (C.0.4) are *local* solutions of problem (C.0.1). In the special case that the constraints are convex and the objective function is (strictly) convex, a solution of system (C.0.4) is a (unique) *global* solution of problem (C.0.1).

## C.0.2 Inequality constrained optimization

Previously we discussed necessary conditions for the solution of a minimization problem subject to equality constraints. In this section we formulate these conditions for problem Let us consider the following equality constrained problem

$$\begin{aligned} \min_x \quad & f(x) \\ \text{subject to} \quad & c_i(x) \leq 0, \quad i \in \{1, 2, \dots, m\}. \end{aligned} \quad (\text{C.0.5})$$

It is helpful to introduce a term to distinguish between *active* and *inactive* constraints. A constraint is called active when  $c_i(x) = 0$  and inactive when  $c_i(x) < 0$ . The set of indices corresponding to the active constraints is called the *working set*,  $\mathcal{W}$ .

Let  $\mathcal{W}^*$  and  $\lambda^*$  denote the working set and Lagrange multipliers of the solution  $x^*$  respectively. From the previous section we know that for the constraints corresponding to the working set the following must hold

$$\nabla f(x^*) = - \sum_{i \in \mathcal{W}^*} \lambda_i^* \nabla c_i(x^*). \quad (\text{C.0.6})$$

In contrast to the equality constraint case, the values of the Lagrange multipliers are of some importance, in particular their sign. In order to intuitively understand the relevance of the signs of the Lagrange multipliers, we again consider the case when  $m = 1$ .

Suppose that the constraint is active at some  $x$  with corresponding Lagrange multiplier  $\lambda < 0$ , such that equation (C.0.6) holds. Then  $\nabla f(x)$  and  $\nabla c(x)$  are pointed in the

same direction, meaning that decreasing the constraint leads to decrease in the objective function. Ergo, even though (C.0.6) holds,  $x$  is not the solution because of the sign of the Lagrange multiplier. In general we conclude that the Lagrange multipliers  $\lambda^*$  corresponding to the active constraints at the solution  $x^*$  are positive. We therefore add another condition:

$$\lambda_i^* \geq 0, \quad i \in \mathcal{W}^* \quad (\text{C.0.7})$$

We can summarize these necessary conditions in the following manner;

$$\nabla_x \mathcal{L}(x^*, \lambda^*) = 0 \quad (\text{C.0.8a})$$

$$c_i(x^*) \leq 0, i \in \{1, 2, \dots, m\} \quad (\text{C.0.8b})$$

$$\lambda_i^* \geq 0, i \in \{1, 2, \dots, m\} \quad (\text{C.0.8c})$$

$$\lambda_i^* c_i(x^*) = 0, i \in \{1, 2, \dots, m\} \quad (\text{C.0.8d})$$

Note that these conditions imply that the Lagrange multipliers corresponding to the inactive constraints are zero. Equations (C.0.8) are called the KKT conditions of problem (C.0.5).

All (local) solutions of problem (C.0.5) satisfy these KKT conditions. Similar to the equality constrained case, if the constraints are convex and the objective function is (strictly) convex, a solution of system (C.0.8) is a (unique) *global* solution of problem (C.0.5).



# List of Abbreviations

<b>Abbreviation</b>	<b>Name</b>
1D	One dimensional
2D	Two dimensional
3D	Three dimensional
3D-Var	Three Dimensional Variational Data Assimilation
3DEnVar	Three Dimensional Ensemble Variational Data Assimilation
4D-Var	Four Dimensional Variational Data Assimilation
4DEnVar	Four Dimensional Ensemble Variational Data Assimilation
AIREP	Aircraft observations
BACY	BASic CYcling environment
BLUE	Best Linear Unbiased Estimate
CAPE	convective available potential energy
CG	Conjugate Gradient
CIN	Convective INhibition
CNN	Convolutional Neural Network
COSMO	Consortium for Small Scale Modeling
CVXOPT	python software for ConVeX OPTimization
DA	Data Assimilation
dBZ	decibel relative to Z
DWD	Deutscher Wetterdienst
ECMWF	European Centre for Medium-Range Weather Forecasts
EKF	Extended Kalman Filter
EMVORADO	Efficient Modular VOlume scanning RADar Operator
EnKF	Ensemble Kalman Filter
ES	Ensemble Smoothers
ETKF	Ensemble Transform Kalman Filter
FAR	False Alarm Rate
GIGG-EnKF	Gamma, InverseGamma and Gaussian Ensemble Kalman Filter
GMRES	Generalized Minimal RESidual method
HErZ	Hans-Ertel Centre for Weather Research
ICON	ICOsahedral Nonhydrostatic (atmospheric model)
IEnKS	Iterative Ensemble Kalman Smoother
IR	Infrared

---

KENDA	Kilometer-Scale Ensemble-based Data Assimilation
KKT	Karush-Kuhn-Tucker
LETKF	Local Ensemble Transform Kalman Filter
METEOSAT	Meteorological Satellite
MFASIS	Method for FAst Satellite Image Simulation
MINRES	MINimum RESidual method
ModeS	ModeS EnHanced Surveillance winds
MSG	Meteosat Second Generation
NIR	Near Infrared
NN	Neural Network
NWP	Numerical Weather Prediction
OSA-EnKF	One Step Ahead Ensemble Kalman Filter
OSSE	Observation System Simulation Experiment
PILOT	Balloon wind profile observations
PROF	Wind profiler observations
QF	Quadratic Filter
QPEns	Quadratic Programming Ensemble
RMSE	Root Mean Square Error
RTPP	RelaxationToPrior Perturbations
SEVIRI	Spinning Enhanced Visible and Infrared Imager
SQP	Sequential Quadratic Programming
SYNOP	Surface observations
TEMP	Radiosonde observations
TERRA	COSMO surface parametrization scheme
TKE	Turbulent Kinetic Energy
UTC	Coordinated Universal Time
VIS	Visible
VISOP	Visible Operator

# Bibliography

- Abbe, C., 1901: The physical basis of long-range weather forecasts. *Monthly Weather Review*, **29** (12), 551–561, doi:10.1175/1520-0493(1901)29[551c:TPBOLW]2.0.CO;2, [https://doi.org/10.1175/1520-0493\(1901\)29\[551c:TPBOLW\]2.0.CO;2](https://doi.org/10.1175/1520-0493(1901)29[551c:TPBOLW]2.0.CO;2).
- Ackinson, G. D., 1973: Impact of weather on military operations: past, present, future. Tech. rep., US Army War College.
- Aksoy, A., F. Zhang, and J. W. Nielsen-Gammon, 2006: Ensemble-based simultaneous state and parameter estimation with MM5. *Geophysical Research Letters*, **33** (12), doi:10.1029/2006GL026186, 112801.
- Andersen, M. S., J. Dahl, and L. Vandenberghe, 2010: Implementation of nonsymmetric interior-point methods for linear optimization over sparse matrix cones. *Mathematical Programming*, **2** (3-4), 167.
- Anderson, J., 2012: Localization and Sampling Error Correction in Ensemble Kalman Filter Data Assimilation. *Monthly Weather Review*, **140**, 2359–2371, doi:10.1175/MWR-D-11-00013.1.
- Anderson, J. L., 2016: Reducing Correlation Sampling Error in Ensemble Kalman Filter Data Assimilation. *Monthly Weather Review*, **144** (3), 913–925, doi:10.1175/MWR-D-15-0052.1, <https://doi.org/10.1175/MWR-D-15-0052.1>.
- 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.
- Annan, J., J. Hargreaves, N. Edwards, and R. Marsh, 2005: Parameter estimation in an intermediate complexity Earth system model using an ensemble Kalman filter. *Ocean Modelling*, **8**, 135–154.
- Baldauf, M., A. Seifert, J. Foerstner, D. Majewski, M. Raschendorfer, and T. Reinhardt, 2011: Operational Convective-Scale Numerical Weather Prediction with the COSMO Model: Description and Sensitivities. *Mon. Wea. Rev.*, **139**, 38873905, doi:10.1175/MWR-D-10-05013.1.

- Bauer Peter, Thorpe Alan, and Brunet Gilbert, 2015: The quiet revolution of numerical weather prediction. *Nature*, **525**, 47, URL <https://doi.org/10.1038/nature1495610.1038/nature14956>.
- Bergthorsson, P. and B. Döös, 1955: Numerical Weather Map Analysis. *Tellus*, **7**, 329.
- Berner, J., U. Achatz, L. Batté, L. Bengtsson, A. d. l. Cámara, H. M. Christensen, M. Colangeli, D. R. B. Coleman, D. Crommelin, S. I. Dolaptchiev, C. L. E. Franzke, P. Friederichs, P. Imkeller, H. Järvinen, S. Juricke, V. Kitsios, F. Lott, V. Lucarini, S. Mahajan, T. N. Palmer, C. Penland, M. Sakradzija, J.-S. von Storch, A. Weisheimer, M. Weniger, P. D. Williams, and J.-I. Yano, 2017: Stochastic parameterization: Toward a new view of weather and climate models. *Bulletin of the American Meteorological Society*, **98** (3), 565–588, doi:10.1175/BAMS-D-15-00268.1, <https://doi.org/10.1175/BAMS-D-15-00268.1>.
- Bertino, L., G. Evensen, and H. Wackernagel, 2003: Sequential data assimilation techniques in oceanography. *International Statistical Reviews*, **71**, 223–241.
- Bick, T., C. Simmer, S. Trömel, K. Wapler, H.-J. Hendricks Franssen, K. Stephan, U. Blahak, C. Schraff, H. Reich, Y. Zeng, and R. Potthast, 2016: Assimilation of 3D radar reflectivities with an ensemble Kalman filter on the convective scale. *Quarterly Journal of the Royal Meteorological Society*, **142** (696), 1490–1504, doi:10.1002/qj.2751, <https://rmets.onlinelibrary.wiley.com/doi/pdf/10.1002/qj.2751>.
- Bishop, C. H., 2016: The GIGG-EnKF: ensemble Kalman filtering for highly skewed non-negative uncertainty distributions. *Quart. J. Roy. Meteor. Soc.*, **142**, 1395–1412.
- Bjerknes, V., 1904: Das Problem der Wettervorhersage, betrachtet vom Standpunkte der Mechanik und der Physik. *Meteorol. Z.*
- Bocquet, M., 2016: Localization and the iterative ensemble Kalman smoother. *Quart. J. Roy. Meteor. Soc.*, **142** (695), 1075–1089.
- Bocquet, M. and P. Sakov, 2013: Joint state and parameter estimation with an iterative ensemble Kalman smoother. *Nonlin. Processes Geophys.*, **20**, 803–818.
- Bocquet, M. and P. Sakov, 2014: An iterative ensemble Kalman smoother. *Quart. J. Roy. Meteor. Soc.*, **140**, 1521–1535.
- Bormann, N., M. Bonavita, R. Dragani, R. Eresmaa, M. Matricardi, and A. McNally, 2016: Enhancing the impact of IASI observations through an updated observation-error covariance matrix. *Quarterly Journal of the Royal Meteorological Society*, **142** (697), 1767–1780, doi:10.1002/qj.2774, <https://rmets.onlinelibrary.wiley.com/doi/pdf/10.1002/qj.2774>.

- 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.
- Campbell, W. F., E. A. Satterfield, B. Ruston, and N. L. Baker, 2017: Accounting for Correlated Observation Error in a Dual-formulation 4D Variational Data Assimilation System. *Monthly Weather Review*, **145** (3), 1019–1032, doi:10.1175/MWR-D-16-0240.1, <https://doi.org/10.1175/MWR-D-16-0240.1>.
- Chollet, F. et al., 2015: Keras. <https://keras.io>.
- Chorin, A. J., M. Morzfeld, and X. Tu, 2010: Implicit Particle filters for data assimilation. *Commun. Appl. Math. Comput. Sci.*, **5**, 221–240.
- Cintra, R. S. C. and H. F. de Campos Velho, 2014: Data Assimilation by Artificial Neural Networks for an Atmospheric General Circulation Model: Conventional Observation. *CoRR*, abs/1407.4360, URL <http://arxiv.org/abs/1407.4360>, 1407.4360.
- Conn, A. R., N. I. M. Gould, and P. L. Toint, 1992: *LANCELOT: a Fortran package for large-scale nonlinear optimization (Release A)*. Number 17 in springer series in computational mathematics ed., Springer Verlag.
- Conn, A. R., N. I. M. Gould, and P. L. Toint, 2000: *Trust-Region Methods*. MPS-SIAM Series on Optimization.
- Dee, D. P., 1995: On-line estimation of error covariance parameters for atmospheric data assimilation. *Mon. Wea. Rev.*, **123**, 1128–1145.
- Desroziers, G., 2016: Improving spatial localization in 4D-EnVar. *Quart. J. Roy. Meteor. Soc.*, **142** (701), 3171–3185.
- Desroziers, G., L. Berre, B. Chapnik, and P. Poli, 2005b: Diagnosis of observation, background and analysis-error statistics in observation space. *Quarterly Journal of the Royal Meteorological Society*, **131** (613), 3385–3396, doi:10.1256/qj.05.108, <https://rmets.onlinelibrary.wiley.com/doi/pdf/10.1256/qj.05.108>.
- Desroziers, G., P. Brousseau, and B. Chapnik, 2005a: Use of randomization to diagnose the impact of observations on analyses and forecasts. *Quart. J. Roy. Meteorol. Soc.*, **131**, 2821–2837, doi:10.1256/qj.04.151.
- Doms, G., J. Förstner, E. Heise, H.-J. Herzog, D. Mironov, M. Raschendorfer, T. Reinhardt, B. Ritter, R. Schrodin, J.-P. Schulz, and G. Vogel, 2011: A Description of the Nonhydrostatic Regional COSMO Model. Part II: Physical Parameterization. Tech. rep., Consortium for Smallscale Modeling (COSMO).

- Done, J. M., G. C. Craig, S. L. Gray, P. A. Clark, and M. E. B. Gray, 2006: Mesoscale simulations of organized convection: Importance of convective equilibrium. *Quarterly Journal of the Royal Meteorological Society*, **132** (616), 737–756, doi:10.1256/qj.04.84, <https://rmets.onlinelibrary.wiley.com/doi/pdf/10.1256/qj.04.84>.
- Doron, M., P. Brasseur, J.-M. Brankart, S. N. Losa, and M. A., 2013: Stochastic estimation of biogeochemical parameters from Globcolour ocean colour satellite data in a North Atlantic 3D ocean coupled physicalbiogeochemical model. *Journal of Marine Systems*, **117**118, 81–95.
- Dueben, P. and P. Bauer, 2018: Challenges and design choices for global weather and climate models based on machine learning. *Geosci. Model Dev.*, **11**, 3999–4009, URL <https://doi.org/10.5194/gmd-11-3999-2018>.
- Efron, B. and R. J. Tibshirani, 1993: *An Introduction to the Bootstrap*. Abingdon, UK: Chapman & Hall/CRC.
- Evensen, G., 1994: Sequential data assimilation with a nonlinear quasi-geostrophic model using Monte Carlo methods to forecast error statistics. *Journal of Geophysical Research*, **99**, 10 143–10 162.
- Evensen, G., 2003: The ensemble Kalman filter: theoretical formulation and practical implementation. *Ocean Dynamics*, **53**, 343–367.
- Evensen, G., 2009: The Ensemble Kalman Filter for combined state and parameter estimation. *IEEE Control. Syst. Mag.*, **29**, 83–104.
- Fabry, F. and J. Sun, 2010: For How Long Should What Data Be Assimilated for the Mesoscale Forecasting of Convection and Why? Part II: On the Observation Signal from Different Sensors. *Monthly Weather Review*, **138** (1), 256–264, doi:10.1175/2009MWR2884.1, <https://doi.org/10.1175/2009MWR2884.1>.
- Frisinger, H., 1978: *History of Meteorology to 1800*. Science History Publications.
- Gandin, L. S., 1963: *Objective Analysis of Meteorological Fields*. Gidrometeor. Izdat., Leningrad, English translation: Israel Program for Scientific Translation, Jerusalem, 1965.
- Gaspari, G. and S. E. Cohn, 1999: Construction of correlation functions in two and three dimensions. *Quart. J. Roy. Meteor. Soc.*, **125**, 723–757.
- Gelb, A., 1974: *Applied Optimal Estimation*. The M.I.T. Press, 382 pp.
- Gharamti, M., B. Ait-El-Fquih, and I. Hoteit, 2015: An iterative ensemble Kalman filter with one-step-ahead smoothing for state-parameters estimation of contaminant transport models. *Journal of Hydrology*, **527**, 442–447.
- Gill, P. E., 1981: *Practical Optimization*. Academic Press, London.

- Gould, N. I. M., M. E. Hribar, and J. Nocedal, 2001: On the solution of equality constrained quadratic problems arising in optimization. *SIAM Journal on Scientific Computing*, **23** (4), 1376–1395.
- Gould, N. I. M. and P. L. Toint, 2002: An iterative working-set method for large-scale non-convex quadratic programming. *Applied Numerical Mathematics*.
- 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.
- Gustafsson, N., T. Janjić, C. Schraff, D. Leuenberger, M. Weissman, H. Reich, P. Brousseau, T. Montmerle, E. Wattrelot, A. Bucánek, M. Mile, R. Hamdi, M. Lindskog, J. Barkmeijer, M. Dahlbom, B. Macpherson, S. Ballard, G. Inverarity, J. Carley, C. Alexander, D. Dowell, S. Liu, Y. Ikuta, and T. Fujita, 2018: Survey of data assimilation methods for convective-scale numerical weather prediction at operational centres. *Quarterly Journal of the Royal Meteorological Society*, n/a–n/a, doi:10.1002/qj.3179, URL <http://dx.doi.org/10.1002/qj.3179>, qJ-17-0086.R2.
- Hestenes, M. R. and E. Stiefel, 1952: Methods of conjugate gradients for solving linear systems. *Journal of research of the National Bureau of Standards*, **49**, 409–436.
- Hodyss, D., 2011: Ensemble State Estimation for Nonlinear Systems Using Polynomial Expansions in the Innovation. *Monthly Weather Review*, **139**, 3571–3588.
- Hodyss, D., 2012: Accounting for Skewness in Ensemble Data Assimilation. *Monthly Weather Review*, **140**, 2346–2358.
- 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., 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. *Monthly Weather Review*, **133** (3), 604–620, doi:10.1175/MWR-2864.1, <https://doi.org/10.1175/MWR-2864.1>.
- Hu, X.-M., F. Zhang, and J. W. Nielsen-Gammon, 2010: Ensemble-based simultaneous state and parameter estimation for treatment of mesoscale model error: A real-data study. *Geophysical Research Letters*, **37** (8), doi:10.1029/2010GL043017, URL <https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2010GL043017>, <https://agupubs.onlinelibrary.wiley.com/doi/pdf/10.1029/2010GL043017>.
- Hunt, B. R., E. J. Kostelich, and I. Szunyogh, 2007: Efficient Data Assimilation for Spatiotemporal Chaos: A local Ensemble Transform Kalman filter. *Physica D*, **230**, 112–126.
- Janjić, T., D. McLaughlin, S. E. Cohn, and M. Verlaan, 2014: Conservation of mass and preservation of positivity with ensemble-type Kalman filter algorithms. *Mon. Wea. Rev.*, **142**, 755–773.

- Janjić, T., Y. Ruckstuhl, and P. L. Toint, 2019: An algorithm for optimization with disjoint linear constraints and its application for predicting rain. *to be submitted*.
- Janjić, T., N. Bormann, M. Bocquet, J. A. Carton, S. E. Cohn, S. L. Dance, S. N. Losa, N. K. Nichols, R. Potthast, J. A. Waller, and P. Weston, 2018: On the representation error in data assimilation. *Quarterly Journal of the Royal Meteorological Society*, **144** (713), 1257–1278, doi:10.1002/qj.3130, <https://rmets.onlinelibrary.wiley.com/doi/pdf/10.1002/qj.3130>.
- Jazwinski, A. H., 1970: *Stochastic Processes and Filtering Theory*. Academic Press, 376 pp.
- Keil, C., F. Heinlein, and G. C. Craig, 2014: The convective adjustment time-scale as indicator of predictability of convective precipitation. *Quarterly Journal of the Royal Meteorological Society*, **140** (679), 480–490, doi:10.1002/qj.2143, <https://rmets.onlinelibrary.wiley.com/doi/pdf/10.1002/qj.2143>.
- Kepert, J., 2009: Covariance localisation and balance in an ensemble Kalman filter. *Quart. J. Roy. Meteor. Soc.*, **135**, 1157–1176.
- Kivman, G. A., 2003: Sequential parameter estimation for stochastic systems. *Nonlin. Processes Geophys.*, **10**, 253–259, doi:10.5194/npg-10-253-2003.
- Kolmogorov, A., 1941: Interpolation and extrapolation of stationary random sequences. *Bulletin of the Academy of Sciences*.
- Kondrashov, D., C. Sun, and M. Ghil, 2008: Data Assimilation for a Coupled OceanAtmosphere Model. Part II: Parameter Estimation. *Monthly Weather Review*, **136** (12), 5062–5076, doi:10.1175/2008MWR2544.1, URL <https://doi.org/10.1175/2008MWR2544.1>, <https://doi.org/10.1175/2008MWR2544.1>.
- Koyama, H. and M. Watanabe, 2010: Reducing forecast errors due to model imperfections using ensemble kalman filtering. *Monthly Weather Review*, **138** (8), 3316–3332, doi:10.1175/2010MWR3067.1, URL <https://doi.org/10.1175/2010MWR3067.1>, <https://doi.org/10.1175/2010MWR3067.1>.
- Krasnopolsky, V. M., M. S. Fox-Rabinovitz, and A. A. Belochitski, 2013: Using ensemble of neural networks to learn stochastic convection parameterizations for climate and numerical weather prediction models from data simulated by a cloud resolving model. *Adv Artif Neural Syst*, **2013**, URL <http://dx.doi.org/10.1155/2013/485913>.
- Lange, H. and G. C. Craig, 2014: The Impact of Data Assimilation Length Scales on Analysis and Prediction of Convective Storms. *Mon. Wea. Rev.*, **142**, 37813808, doi:10.1175/MWR-D-13-00304.1.

- Lange, H. and T. Janjić, 2016: Assimilation of Mode-S EHS Aircraft Observations in COSMO-KENDA. *Monthly Weather Review*, **144** (5), 1697–1711, doi:10.1175/MWR-D-15-0112.1, URL <http://dx.doi.org/10.1175/MWR-D-15-0112.1>.
- LeCun Yann, Bengio Yoshua, and Hinton Geoffrey, 2015: Deep learning. *Nature*, **521**, 436, URL <https://doi.org/10.1038/nature14539>.
- Lin, Y.-L., R. D. Farley, and H. D. Orville, 1983: Bulk parameterization of the snow field in a cloud model. *Journal of Climate and Applied Meteorology*, **22** (6), 1065–1092, doi:10.1175/1520-0450(1983)022<1065:BPOTSF>2.0.CO;2, [https://doi.org/10.1175/1520-0450\(1983\)022<1065:BPOTSF>2.0.CO;2](https://doi.org/10.1175/1520-0450(1983)022<1065:BPOTSF>2.0.CO;2).
- Liu, C., Q. Xiao, and B. Wang, 2008: An Ensemble-Based Four-Dimensional Variational Data Assimilation Scheme. Part I: Technical Formulation and Preliminary Test. *Monthly Weather Review*, **136** (9), 3363–3373, doi:10.1175/2008MWR2312.1, URL <https://doi.org/10.1175/2008MWR2312.1>, <https://doi.org/10.1175/2008MWR2312.1>.
- Lorenc, A. C., 1986: Analysis methods for numerical weather prediction. *Quart. J. Roy. Meteor. Soc.*, **112**, 1177–1194.
- Lorenz, E. N., 1969: The predictability of a flow which possesses many scales of motion. *Tellus*, **XXI/3**, 289–307.
- Lynch, P., 2008: The origins of computer weather prediction and climate modeling. *Elsevier*, **227** (7), 3431–3444.
- Necker, T., M. Weissmann, and M. Sommer, 2018: The importance of appropriate verification metrics for the assessment of observation impact in a convection-permitting modelling system. *Quarterly Journal of the Royal Meteorological Society*, **144** (714), 1667–1680, doi:10.1002/qj.3390, <https://rmets.onlinelibrary.wiley.com/doi/pdf/10.1002/qj.3390>.
- Nielsen, M. A., 2015: *Neural networks and deep learning*. determination press.
- Nocedal, J. and S. Wright, 2006: *Numerical Optimization*. Springer.
- Owen, P. R. and W. R. Thomson, 1963: Heat transfer across rough surfaces. *Journal of Fluid Mechanics*, **15** (3), 321334, doi:10.1017/S0022112063000288.
- Pan, M. and E. F. Wood, 2006: Data Assimilation for Estimating the Terrestrial Water Budget Using a Constrained Ensemble Kalman Filter. *Journal of Hydrometeorology*, **7**, 534–547.
- Pathak, J., B. Hunt, M. Girvan, Z. Lu, and E. Ott, 2018a: Model-free prediction of large spatiotemporally chaotic systems from data: A reservoir computing approach. *Phys. Rev. Lett.*, **120**, 024102, doi:10.1103/PhysRevLett.120.024102, URL <https://link.aps.org/doi/10.1103/PhysRevLett.120.024102>.

- Pathak, J., A. Wikner, R. Fussell, S. Chandra, B. R. Hunt, M. Girvan, and E. Ott, 2018b: Hybrid forecasting of chaotic processes: Using machine learning in conjunction with a knowledge-based model. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, **28** (4), 041 101, doi:10.1063/1.5028373, <https://doi.org/10.1063/1.5028373>.
- Piper, D., M. Kunz, F. Ehmele, S. Mohr, B. Mühr, A. Kron, and J. Daniell, 2016: Exceptional sequence of severe thunderstorms and related flash floods in May and June 2016 in Germany. Part I: Meteorological background. *Natural Hazards and Earth System Sciences*, **16**, 2835–2850, doi:10.5194/nhess-16-2835-2016.
- Posselt, D. J., D. Hodyss, and C. H. Bishop, 2014: Errors in Ensemble Kalman Smoother Estimates of Cloud Microphysical Parameters. *Monthly Weather Review*, **142**, 1631–1654, doi:10.1175/MWR-D-13-00290.1.
- Poterjoy, J., 2016: A Localized Particle Filter for High-Dimensional Nonlinear Systems. *American Meteorological Society*, **144**, 59–76.
- Rabin, R. M., S. Stadler, P. J. Wetzel, D. J. Stensrud, and M. Gregory, 1990: Observed effects of landscape variability on convective clouds. *Bulletin of the American Meteorological Society*, **71** (3), 272–280, doi:10.1175/1520-0477(1990)071<0272:OEOLVO>2.0.CO;2, [https://doi.org/10.1175/1520-0477\(1990\)071<0272:OEOLVO>2.0.CO;2](https://doi.org/10.1175/1520-0477(1990)071<0272:OEOLVO>2.0.CO;2).
- Raschendorfer, M., 2001: The new turbulence parametrization of LM. *COSMO News Letter*.
- Rasp, S. and S. Lerch, 2018: Neural networks for post-processing ensemble weather forecasts. *CoRR*, **abs/1805.09091**.
- Rasp, S., M. S. Pritchard, and P. Gentine, 2018: Deep learning to represent sub-grid processes in climate models. *Proceedings of the National Academy of Sciences*, **115** (39), 9684–9689, doi:10.1073/pnas.1810286115, <https://www.pnas.org/content/115/39/9684.full.pdf>.
- Reinhardt, T. and A. Seifert, 2006: A three-category ice scheme for LMK. *COSMO News Letter*.
- Rhodin, A., H. Lange, R. Potthast, and T. Janjić, 2013: *Documentation of the DWD Data Assimilation System*. Deutscher Wetterdienst (DWD).
- Richardson, L. F., 1922: *Weather prediction by numerical process*. Cambridge, The University press.
- Ritter, B. and J.-F. Geleyn, 1992: A comprehensive radiation scheme for numerical weather prediction models with potential applications in climate simulations. *Monthly Weather Review*, **120** (2), 303–325, doi:10.1175/1520-0493(1992)120<0303:ACRSFN>2.0.CO;2, [https://doi.org/10.1175/1520-0493\(1992\)120<0303:ACRSFN>2.0.CO;2](https://doi.org/10.1175/1520-0493(1992)120<0303:ACRSFN>2.0.CO;2).

- Roberts, N. M. and H. W. Lean, 2008: Scale-selective verification of rainfall accumulations from high-resolution forecasts of convective events. *Monthly Weather Review*, **136** (1), 78–97, doi:10.1175/2007MWR2123.1, <https://doi.org/10.1175/2007MWR2123.1>.
- Rotunno, R. and C. Snyder, 2008: A Generalization of Lorenz’s Model for the Predictability of Flows with Many Scales of Motion. *Journal of the Atmospheric Sciences*, **65** (3), 1063–1076, doi:10.1175/2007JAS2449.1, <https://doi.org/10.1175/2007JAS2449.1>.
- Ruckstuhl, Y. and T. Janjić, 2018: Parameter and state estimation with ensemble Kalman filter based algorithms for convectivescale applications. *Quart. J. Roy. Meteorol. Soc.*
- Ruiz, J. and M. Pulido, 2015: Parameter estimation using ensemble-based data assimilation in the presence of model error. *Monthly Weather Review*, **143** (5), 1568–1582, doi:10.1175/MWR-D-14-00017.1, URL <https://doi.org/10.1175/MWR-D-14-00017.1>, <https://doi.org/10.1175/MWR-D-14-00017.1>.
- Ruiz, J. J., M. Pulido, and T. Miyoshi, 2013a: Estimating Model Parameters with Ensemble-Based Data Assimilation: A Review. *Journal of the Meteorological Society of Japan*, **91** (2), 79–99.
- Ruiz, J. J., M. Pulido, and T. Miyoshi, 2013b: Estimating Model Parameters with Ensemble-Based Data Assimilation: Parameter Covariance Treatment. *Journal of the Meteorological Society of Japan*, **91** (4), 453–469.
- Saad, Y., 1996: *Iterative Methods for Sparse Linear Systems*. PWS Publishing Company.
- Sasaki, Y., 1970: Some basic formalisms in numerical variational analysis. *Monthly Weather Review*, **98** (12), 875–883, doi:10.1175/1520-0493(1970)098<0875:SBFINV>2.3.CO;2, [https://doi.org/10.1175/1520-0493\(1970\)098<0875:SBFINV>2.3.CO;2](https://doi.org/10.1175/1520-0493(1970)098<0875:SBFINV>2.3.CO;2).
- Scheck, L., P. Frérebeau, R. Buras-Schnell, and B. Mayer, 2016: A fast radiative transfer method for the simulation of visible satellite imagery. *Journal of Quantitative Spectroscopy and Radiative Transfer*.
- Schirber, S., D. Klocke, R. Pincus, J. Quaas, and J. L. Anderson, 2013: Parameter estimation using data assimilation in an atmospheric general circulation model: From a perfect toward the real world. *Journal of Advances in Modeling Earth Systems*, **5** (1), 58–70, doi:10.1029/2012MS000167, URL <https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2012MS000167>, <https://agupubs.onlinelibrary.wiley.com/doi/pdf/10.1029/2012MS000167>.
- Schraff, C., H. Reich, A. Rhodin, A. Schomburg, K. Stephan, A. Perriñez, and R. Potthast, 2016: Kilometre-Scale Ensemble Data Assimilation for the COSMO Model (KENDA). *Quart. J. Roy. Meteor. Soc.*, **142**, 1453–1472, doi:10.1002/qj.2748, URL <http://dx.doi.org/10.1002/qj.2748>.

- Selz, T. and G. C. Craig, 2015: Upscale Error Growth in a High-Resolution Simulation of a Summertime Weather Event over Europe. *Monthly Weather Review*, **143** (3), 813–827, doi:10.1175/MWR-D-14-00140.1, <https://doi.org/10.1175/MWR-D-14-00140.1>.
- Simon, D., A. Samuelsen, L. Bertino, and S. Mouysset, 2015: Experiences in multiyear combined state-parameter estimation with an ecosystem model of the North Atlantic and Arctic Oceans using the Ensemble Kalman Filter. *Journal of Marine System*, **152**, 1–17.
- Simon, E. and L. Bertino, 2009: Application of the Gaussian anamorphosis to assimilation in a 3-D coupled physical-ecosystem model of the North Atlantic with the EnKF: A twin experiment. *Ocean Sci.*, **5**, 495–510, [www.ocean-sci.net/5/495/2009/](http://www.ocean-sci.net/5/495/2009/).
- Snyder, C., T. Bengtsson, P. Bickel, and J. L. Anderson, 2008: Obstacles in High-Dimensional Particle Filtering. *Monthly Weather Review*, **131**, 4629–4640.
- Sommer, M. and T. Janjić, 2018: A flexible additive inflation scheme for treating model error in ensemble Kalman filters. *Quarterly Journal of the Royal Meteorological Society*, **144** (716), 2026–2037, doi:10.1002/qj.3254, <https://rmets.onlinelibrary.wiley.com/doi/pdf/10.1002/qj.3254>.
- Sommer, M. and M. Weissmann, 2016: Ensemble-based approximation of observation impact using an observation-based verification metric. *Tellus A: Dynamic Meteorology and Oceanography*, **68** (1), 27885, doi:10.3402/tellusa.v68.27885, <https://doi.org/10.3402/tellusa.v68.27885>.
- Stephan, K., S. Klink, and C. Schraff, 2008: Assimilation of radar-derived rain rates into the convective-scale model COSMO-DE at DWD. *Quarterly Journal of the Royal Meteorological Society*, **134** (634), 1315–1326, doi:10.1002/qj.269, <https://rmets.onlinelibrary.wiley.com/doi/pdf/10.1002/qj.269>.
- Stull, R. B., 1988: *An Introduction to Boundary Layer Meteorology*. Dordrecht: Kluwer Academic Publishers.
- Tiedtke, M., 1989: A comprehensive mass flux scheme for cumulus parameterization in large-scale models. *Monthly Weather Review*, **117** (8), 1779–1800, doi:10.1175/1520-0493(1989)117<1779:ACMFSF>2.0.CO;2, [https://doi.org/10.1175/1520-0493\(1989\)117<1779:ACMFSF>2.0.CO;2](https://doi.org/10.1175/1520-0493(1989)117<1779:ACMFSF>2.0.CO;2).
- Tong, M. and M. Xue, 2008: Simultaneous Estimation of Microphysical Parameters and Atmospheric State with Simulated Radar Data and Ensemble Square Root Kalman Filter. Part II: Parameter Estimation Experiments. *Monthly Weather Review*, **136** (5), 1649–1668, doi:10.1175/2007MWR2071.1, <https://doi.org/10.1175/2007MWR2071.1>.
- van Leeuwen, P. J., 2009: Particle filtering in geophysical systems. *Monthly Weather Review*, **137** (12), 4089–4114, doi:10.1175/2009MWR2835.1, <https://doi.org/10.1175/2009MWR2835.1>.

- Weir, B., R. N. Miller, and Y. H. Spitz, 2013: Implicit estimation of ecological model parameters. *Bull. Math. Biol.*, **75**, 223–257.
- Weston, P. P., W. Bell, and J. R. Eyre, 2014: Accounting for correlated error in the assimilation of high-resolution sounder data. *Q.J.R. Meteorol. Soc.*, **140**, 2420–2429, doi:10.1002/qj.2306.
- Wiener, N., 1949: *Extrapolation, interpolation and smoothing of stationary time series with engineering applications*. M.I.T. Press.
- Wilks, D., 2006: *Statistical Methods in the Atmospheric Sciences*. New York: Academic Press.
- Würsch, M. and G. C. Craig, 2014: A simple dynamical model of cumulus convection for data assimilation research. *Meteorologische Zeitschrift*, **23**, 483–490.
- Yang, R. and M. A. Friedl, 2003: Determination of Roughness Lengths for Heat and Momentum Over Boreal Forests. *Boundary-Layer Meteorology*, **107** (3), 581–603, doi: 10.1023/A:1022880530523, URL <https://doi.org/10.1023/A:1022880530523>.
- Yang, S.-C., E. Kalnay, B. Hunt, and N. E. Bowler, 2009: Weight interpolation for efficient data assimilation with the Local Ensemble Transform Kalman Filter. *Quarterly Journal of the Royal Meteorological Society*, **135** (638), 251–262, doi:10.1002/qj.353, <https://rmets.onlinelibrary.wiley.com/doi/pdf/10.1002/qj.353>.
- Zängl, G., D. Reinert, P. Rípodas, and M. Baldauf, 2015: The ICON (icosahedral non-hydrostatic) modelling framework of DWD and MPI-M: Description of the non-hydrostatic dynamical core. *Quarterly Journal of the Royal Meteorological Society*, **141** (687), 563–579, doi:10.1002/qj.2378, <https://rmets.onlinelibrary.wiley.com/doi/pdf/10.1002/qj.2378>.
- Zeng, Y., 2014: Efficient Radar Forward Operator for Operational Data Assimilation within the COSMO-model. Ph.D. thesis, KIT.
- Zeng, Y., U. Blahak, and D. Jerger, 2016: An efficient modular volume-scanning radar forward operator for NWP models: description and coupling to the COSMO model. *Quarterly Journal of the Royal Meteorological Society*, **142** (701), 3234–3256, doi:10.1002/qj.2904, <https://rmets.onlinelibrary.wiley.com/doi/pdf/10.1002/qj.2904>.
- Zeng, Y., U. Blahak, M. Neuper, and D. Jerger, 2014: Radar Beam Tracing Methods Based on Atmospheric Refractive Index. *Journal of Atmospheric and Oceanic Technology*, **31** (12), 2650–2670, doi:10.1175/JTECH-D-13-00152.1, <https://doi.org/10.1175/JTECH-D-13-00152.1>.
- Zeng, Y. and T. Janjić, 2016: Study of conservation laws with the Local Ensemble Transform Kalman Filter. *Quarterly Journal of the Royal Meteorological Society*, **142** (699), 2359–2372, doi:10.1002/qj.2829, URL <http://dx.doi.org/10.1002/qj.2829>.

- Zeng, Y., T. Janjić, A. de Lozar, U. Blahak, H. Reich, C. Keil, and A. Seifert, 2019: Representation of Model Error in Convective-Scale Data Assimilation: Additive Noise, Relaxation Methods, and Combinations. *Journal of Advances in Modeling Earth Systems*, **10** (11), 2889–2911, doi:10.1029/2018MS001375, <https://agupubs.onlinelibrary.wiley.com/doi/pdf/10.1029/2018MS001375>.
- Zeng, Y., T. Janjić, Y. Ruckstuhl, and M. Verlaan, 2017: Ensemble-type Kalman filter algorithm conserving mass, total energy and enstrophy. *Quarterly Journal of the Royal Meteorological Society*, **143** (708), 2902–2914, doi:10.1002/qj.3142, <https://rmets.onlinelibrary.wiley.com/doi/pdf/10.1002/qj.3142>.
- 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. *Monthly Weather Review*, **132**, 1238–1253.

# Acknowledgement

It is no secret that mountains drove me to Munich. The PhD position I was offered seemed interesting and all, but it was my passion for the outdoors that made me most excited to move here. Over the past couple of years this has changed. Not because I lost interest in mountains. On the contrary, I love them more than ever. But because of my experience at the LMU. Science, discussing and socializing with colleagues, conferences, being part of W2W, all of that made me fall in love with my job. I would therefore like to thank everyone who discussed scientific ideas with me, helped me fix technical stuff that I was too stupid to fix myself, endured my “sleepy drunkenness” behavior, made me laugh, explored exiting countries with me, basically everyone who contributed to making these past 3,5 years some of the happiest of my life. But there are a few persons that stand out.

**Tijana**, without you this PhD thesis would not exist. One always hears of people getting stuck at some point during their PhD, but I never felt that way. The reason is that you were always available to discuss and propose new ideas to further my (or actually our) work. You always valued my opinions and encouraged me to follow my own ideas. I can't express how much I appreciated this. I learned a lot from you, both scientifically and personally and I truly feel blessed to have had you as my supervisor. I look forward to continue working with you in the second phase of W2W.

**Matthias** (Sommer), even though you left us, we haven't forgotten you. You definitely steepened my learning curve during my first few months by being refreshingly blunt when I was doing something that doesn't make sense. I learned a lot from our scientific discussions and soon realized you were kinda fun to hang with too. We miss you!

**Yuefei**, I don't know how I would have completed this work without you. You always were able to fix my COSMO-KENDA related problem, for which I am eternally grateful. On top of that you often bring me weird (and sometimes delicious) food back from China to sustain me throughout the long afternoons between lunch and dinner.

I should also mention my appreciation for the endurance level of my colleagues **Robert** and **Leo**. I often bothered you with stupid computer questions and you were both very capable of hiding your annoyance. I would like to sincerely thank you for that and also apologize in advance for any future questions, because I will stick around for a while. Muahaha (evil laugh).

**Florian**, **Federico** and **Phillip**, we had some fun in the office didn't we? The occasional espresso, strange conversations and the awesome office decorations, created a work environment that I always looked forward to get to in the morning. Too bad our Christ-

mas tree was taken from us... I can only hope I will remain this lucky with office mates throughout my career!

**Kevin, Mirjam, Tobi** (Selz), **Tobi** (Necker), **Matthias** (Schindler), **Stephan, Julia, Heiner** and the afore mentioned colleges, I owe a great deal to all of you. From intense scientific discussions to plotting the perfect murder to extended conferences, our lunches and social activities always made the breaks from work interesting and fun.

Thanks to my colleagues at the **German Weather Service**, who granted me access to their software.

I would also like to thank my **W2W** colleagues for organizing great meetings, which are always full of science and fun.

I cannot thank everyone who made my time as a PhD student so pleasant without mentioning my family in particular, most of whom have been PhD students themselves. Their support and guidance has been very valuable. **Mamsk**, soms liet ik iets te lang niks van mij horen, maar gelukkig durf je dan eindelijk te bellen of een berichtje te sturen en kletsen we bij. In stressvolle periodes word ik graag afgeleid door een lieve moeder die mij een hart onder de riem steekt. Niet alleen een moeder trouwens. **Kar, Nic**, maar ook zeker **Chiel** en **Gid**, jullie weten dat ik heel graag weekendjes plan om jullie te zien, zelfs in drukke periodes. **Suus** en **Viet**, voor jullie geld hetzelfde. Jullie zijn immers zo goed als familie. **Gwen, Kiek, Lilly, Chloe**, wanneer ik met jullie ben, of als foto's en filmpjes van jullie op telegram verschijnen, dan ben ik zo vrolijk dat de rest van de wereld wel even kan wachten. Een zeer effectieve stress remedie. **Orpa, Henk**, bedankt voor jullie begrip dat we naar het buitenland zijn vertrokken voor mijn promotie onderzoek. In plaats van klagen, maken jullie graag gebruik van jullie super busje om ons op te komen zoeken. Ook met jullie zijn weekendjes weg altijd gezellig!

My deepest gratitude goes to my husband, whom I love dearly and always ensures I have a healthy work-life balance. **Jur**, "ons leven is awesome", zoals we onszelf vaak herinneren. Bergen, toffe vakanties, geweldige vrienden en familie, en natuurlijk een super baan met flexibele werktijden. Ik ben zo dankbaar voor het leven dat we leiden.

Thank you all!