ABSTRACT

Title of dissertation: NONLINEAR AND MULTIRESOLUTION ERROR COVARIANCE ESTIMATION IN ENSEMBLE DATA ASSIMILATION

Sabrina Rainwater
Doctor of Philosophy, 2012

Dissertation directed by: Professor Brian R. Hunt
Department of Mathematics and

Ensemble Kalman Filters perform data assimilation by forming a background covariance matrix from an ensemble forecast. The spread of the ensemble is intended to represent the algorithm's uncertainty about the state of the physical system that produces the data. Usually the ensemble members are evolved with the same model.

The first part of my dissertation presents and tests a modified Local Ensemble Transform Kalman Filter (LETKF) that takes its background covariance from a combination of a high resolution ensemble and a low resolution ensemble. The computational time and the accuracy of this mixed-resolution LETKF are explored and compared to the standard LETKF on a high resolution ensemble, using simulated observation experiments with the Lorenz Models II and III (more complex versions of the Lorenz 96 model). The results show that, for the same computation time, mixed resolution ensemble analysis achieves higher accuracy than standard ensemble analysis.

The second part of my dissertation demonstrates that it can be fruitful to rescale the ensemble spread prior to the forecast and then reverse this rescaling after the forecast. This technique, denoted “forecast spread adjustment” provides a tunable parameter that is complementary to covariance inflation, which cumulatively increases the ensemble spread to compensate for underestimation of uncertainty. As the adjustable parameter approaches zero, the filter approaches the Extended Kalman Filter when the ensemble size is sufficiently large. The improvement provided by forecast spread adjustment depends on ensemble size, observation error, and model error. The results indicate that it is most effective for smaller ensembles, smaller observation errors, and larger model error, though the effectiveness depends significantly on the type of model error.

NONLINEAR AND MULTIRESOLUTION ERROR COVARIANCE ESTIMATION IN ENSEMBLE DATA ASSIMILATION

by

Sabrina Rainwater

Dissertation submitted to the Faculty of the Graduate School of the
University of Maryland, College Park in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
2012

Advisory Committee:
Professor Brian R. Hunt, Chair
Professor Kayo Ide
Professor Eugenia Kalnay
Professor Takemasa Miyoshi
Professor Ross Salawitch, Dean’s representative

Table of Contents

List of Tables

List of Figures

List of Abbreviations

EnKF	ensemble Kalman filter
ETKF	Ensemble Transform Kalman Filter
mETKF	mixed resolution ETKF
ηETKF	ETKF with forecast spread adjusted by η
LETKF	Local Ensemble Transform Kalman Filter
mLETKF	mixed resolution LETKF
ηLETKF	LETKF with forecast spread adjusted by η
OSSE	Observing System Simulation Experiment
rms	root mean square function
RMSE	time averaged root mean square analysis error
XKF	Extended Kalman Filter

Chapter 1

Introduction

1.1 Background: Weather Forecasting and Data Assimilation

To predict the weather, we require both a numerical model for the partial differential equations governing the evolution of weather and initial conditions for the current atmospheric state. Since weather is chaotic, the forecast must periodically be corrected to more closely match the observations. This corrected model state is called the analysis. The process of forecasting a model state and modifying it according to observations is called data assimilation.

Neither the forecast nor the observations exactly coincide with the true state of the weather. Data assimilation balances the uncertainties in the forecast (also called the "background" state) and the observations, and produces an "analysis" state deemed to be the most likely state given the available information.

1.1.1 The Kalman Filter

A common paradigm for data assimilation is the Kalman filter. The Kalman filter analysis can be described by $x^a = x^b + K\left(y^o - Hx^b\right)$, where x^a and x^b are vectors representing the analysis and background states, y^o represents the observations, H is a linear operator (e.g., an interpolation operator) which transforms model vectors into expected observations, $K = P^b H^\top \left(HP^b H^\top + R^o\right)^{-1}$ is called the Kalman gain matrix, and P^b and R^o represent the background and observation error covariance matrices.

The Kalman filter is optimal provided the dynamics are linear, the observed quantities are linear functions of the model state, and the errors are Gaussian. Even when not optimal, the Kalman filter often works well for nonlinear models and nonlinear observation operators by using a linear approximation.

Advanced data assimilation techniques require an estimate of the uncertainty in the background state. The background uncertainty at a particular time is related to, but differs from the uncertainty at previous times. Nonlinear Kalman filters (e.g. XKF, ETKF, LETKF) approximately evolve the background error covariance but tend to underestimate it. In ensemble Kalman filters, this is typically compensated for by multiplicative inflation of the background error covariance prior to analysis.

In the Extended Kalman Filter (XKF) (Jazwinski 1970; Evensen 1992), the estimated background error covariance is evolved by linearizing the model around the background state, which is also being evolved by the model. With the complex weather prediction models however, the computation cost of the XKF makes it impractical.

Another way to estimate the background error covariance is through the use of a collection of forecasts, called an ensemble. According to sample statistics, the background error covariance is approximated by $P^b \approx \frac{1}{k-1} X^b {X^b}^\top$, where k is the size of the ensemble, and X^b is a matrix whose columns are the ensemble perturbations, i.e., the difference between the model state and the mean. Most ensemble techniques use the mean of the background ensemble to provide the background state. Rather than output a single, optimal analysis state, the ensemble is regenerated as k suboptimal states centered around the optimal analysis.

The Ensemble Transform Kalman Filter (ETKF) (Bishop et al. 2001; Wang et al. 2004) is one algorithm that regenerates the ensemble after performing the analysis. However, forecasting a large ensemble can be costly. Furthermore, artificial correlations between model values at distant grid points are induced by computing P^b from sample statistics. These difficulties are circumvented by performing an analysis for each grid point, considering only nearby observations when computing the update. This method is called localization and several localization methods have been proposed (Houtekamer and Mitchell 1998; Ott et al. 2004; Hamill et al. 2001; Whitaker and Hamill 2002). The Local Ensemble Transform Kalman Filter (LETKF) is described in Hunt et al. (2007). Though my research can be applied to other ensemble methods, most of my research is performed with ETKF the or the LETKF.

1.1.2 Experimental Methods and Models

I used Observing System Simulation Experiments (OSSEs) to test the accuracy of the methods I developed. In OSSEs, one forecasts a model state designated the “truth” and generates observations by adding error. Then, pretending not to know the truth, data assimilation is performed, and the analysis is compared to the truth to measure the accuracy in terms of the time averaged, spatial root mean square error. Improvements to data assimilation are generally tested on simple models before being implemented and tested in more realistic weather models because with a simple model, OSSEs can be run quickly and at low computation cost.

The simple models I used for testing are the Lorenz (2005) Models II, and III . These are generalizations of the well known Lorenz 96 model (Lorenz 1996). The domain for the Lorenz 96 model is a set of grid points (typically 40) describing a circle. The model consists of three terms

$$\frac{dx_n}{dt} = \left(x_{n+1} - x_{n-2}\right) x_{n-1} - x_n + F \tag{1.1}$$

simulating the chaotic effects of advection, diffusion, and forcing respectively. Model II smooths the Lorenz 96 model over a domain of K times as many grid points (i.e. increasing the resolution by a factor of K):

$$\frac{dx_n}{dt} = \left(x_{n+K} x^{\text{avg}}_{n-K}\right)^{\text{avg}} - x^{\text{avg}}_{n-2K} x^{\text{avg}}_{n-K} - x_n + F\,, \tag{1.2}$$

where q^{avg}_j refers to the average value of q over the region K grid points in length centered at grid point j. Model III simulates even greater resolution, sufficient to describe coupled small scale activity. Model III evolves the variables $z_n = x_n + y_n$ according to

$$\begin{aligned}\frac{dz_n}{dt} &= \left(x_{n+K} x^{\text{avg}}_{n-K}\right)^{\text{avg}} - x^{\text{avg}}_{n-2K} x^{\text{avg}}_{n-K} - x_n + F \\ &\quad + b^2 \left(y_{n+1} - y_{n-2}\right) y_{n-1} - b y_n + c\left(x_{n+1} y_{n-1} - y_{n-2} x_{n-1}\right),\end{aligned}$$

where x_n and y_n represent the large scale and small scale components of z_n respectively (defined precisely in Section 2.3.2, equation (2.25)), b controls the amplitude and speed of the small scale activity and c controls the coupling.

1.2 Objectives

Ensemble Kalman filters are limited by the computational cost of evolving a large, high-resolution ensemble. Larger ensembles correspond to a greater sample size, which allows better statistical analysis. On the other hand, high model resolution is desirable because it improves the individual forecasts at all scales and better resolves the small scale activity. So traditional ensemble Kalman filters compromise between the resolution of the model and the size of the ensemble. Much of the literature on ensemble Kalman filters uses the ensemble mean as the best estimate for the truth, but Gao and Xue (2008) proposed updating a high resolution forecast with the background error covariance estimated from a low resolution ensemble.

The goal of the first part of my doctoral research (Chapter 2) is to develop and test a technique for mixed-resolution ensemble data assimilation that incorporates information from both a high resolution ensemble and a low resolution ensemble. This alleviates the trade-off between model resolution and ensemble size, allowing a large low-resolution ensemble to bolster a small high-resolution ensemble. My results demonstrate that mixed-resolution ensemble data assimilation outperforms single-resolution data assimilation *for the same computational cost*.

The algorithm I developed for mixed-resolution ensemble data assimilation effectively estimates the covariance as a weighted average of the background error covariances estimated from high and low resolution ensembles. I discovered that the results are not very sensitive to exact tuning of the weight parameter. Exploration of the reasons for this lead to the second part of my dissertation research.

The goal of the second part of my doctoral research (Chapter 3) is to improve ensemble data assimilation for nonlinear models by developing and testing an algorithm in which the spread of the ensemble during the forecast does not correspond to the amount of uncertainty. With forecast spread adjustment I also aim to connect many aspects of data assimilation, specifically the ensemble Kalman filters and the extended Kalman filter, background and analysis error covariance inflation, and forecast spread and observation error covariance inflation. In the ensemble Kalman filters, the background mean can be thought of as a random variable estimating the true model state, with the background ensemble representing random sampling according to the distribution of this random variable. In other words, we assume that the spread of the ensemble corresponds to the uncertainty in the mean. However, I have found advantages to ensemble data assimilation when forecasting an ensemble with a scaled spread, provided that the ensemble spread is rescaled prior to analysis. I call this technique forecast spread adjustment.

For the idealized case of data assimilation with a linear model, forecast spread adjustment has no effect, i.e. the algorithm with forecast spread adjustment simplifies to the standard algorithm without forecast spread adjustment. Thus a discussion of the effects of forecast spread adjustment is only relevant and meaningful for a nonlinear model such as numerical weather models.

If the ensemble is clustered too closely about the mean, the true model state could lie outside of the spread of the ensemble, especially when model error is substantial. Model nonlinearity can further accentuate the discrepancy between the ensemble and the truth. For a large ensemble with infinitesimal spread, the rescaled perturbations describe a linear approximation for the error similar to that of the XKF. Compared to the XKF, ensemble Kalman filters can better track nonlinear effects as the ensemble is spread over a larger region of the attractor. I show that further expanding the ensemble spread can provide additional advantage, especially in the presence of model error.

An alternate way of describing forecast spread adjustment is in terms of inflating the observation error covariance; though, as Sections 3.3.2 and 3.4.3 demonstrate, in the cases I examined, only the forecast spread adjustment interpretation correctly ascribes the observation, background, and analysis error estimates while the error estimates ascribed by the observation error covariance inflation interpretation exaggerate or underestimate the true error.

Though the algorithms given in the literature on ensemble data assimilation with multiplicative inflation typically inflate the background error covariance, inflating the analysis error covariance instead is often mentioned as being equally viable. Forecast spread adjustment can be used to transform an algorithm that inflates the background error covariance into an algorithm that inflates the analysis error covariance or into an algorithm that inflates both by different amounts. Furthermore, multiplicative inflation (including possibly deflation) of both the background error covariance and the analysis error covariance separately is yet another way of describing ensemble data assimilation with forecast spread adjustment and background error covariance inflation.

1.3 Organization

My dissertation consists of two research projects, both of which I am submitting for publication. They are written as separate articles, so Chapter 2 and Chapter 3 are complete and discrete in and of themselves. This introduces some redundancy in my dissertation, especially between the introductions to Chapters 2 and 3. Chapter 2 describes mixed-resolution ensemble data assimilation. Chapter 3 describes ensemble data assimilation with an adjusted forecast spread.

In Chapter 2, I start by introducing ensemble data assimilation and motivate mixed-resolution ensemble data assimilation. I describe the ETKF (Ensemble Transform Kalman Filter) algorithm (Bishop et al. 2001; Wang et al. 2004) and I describe how I adapted the ETKF algorithm to accommodate ensembles of two resolutions (mETKF). In the limit as two ensembles with the same number of members approach the same resolution I describe how the mETKF (on two ensembles with the same resolution) compares to the ETKF on the entire ensemble. I also describe the cost function that is minimized by my mETKF algorithm and how this could be extended to hybrid systems. I actually implemented a mixed resolution LETKF (Local ETKF, (Hunt et al. 2007)), in which I perform a mETKF analysis at each grid point, only allowing nearby observations to influence the analysis. I tested the mLETKF with OSSEs using Lorenz (2005) Models II and III as the low and high resolution models respectively. I compared the root mean square difference between the analysis mean and the truth for various mixed-resolution ensembles using the mLETKF and for various single-resolution ensembles using the LETKF respectively. I tested the mLETKF in perfect model scenarios and in scenarios with model error. My results show the trade-off between forecast computation time and time averaged root mean square analysis error for my experiments with the mLETKF and the LETKF, and I discuss how using mixed resolution ensembles allows more accurate results for the same forecast computation time.

In Chapter 3, I start by introducing data assimilation, the ensemble Kalman filters (specifically ETKF and LETKF), and the extended Kalman filter. I describe ensemble data assimilation with forecast spread adjustment. I demonstrate how this reduces to standard data assimilation when the model is linear. I show that this is equivalent to observation error covariance inflation for an appropriately initialized ensemble and discuss why forecast spread adjustment is the more natural interpretation for my experiments. Forecast spread adjustment with multiplicative covariance inflating can also be described as multiplicative inflation of the analysis error covariance followed by multiplicative deflation or inflation of the background error covariance. Furthermore in the limit of a large ensemble with a small forecast spread, ensemble data assimilation with forecast spread adjustment approaches the extended Kalman filter. I tested the LETKF with forecast spread adjustment via OSSEs on the Lorenz (2005) Model II for different amounts of model error, different ensemble sizes, and different amounts of observation error. I also tested how the ETKF with forecast spread adjustment compares to the XKF for large ensembles and small spread. I discuss the results of these experiments and the potential benefits of forecast spread adjustment.

In Chapter 4, I discuss my conclusions and future directions for research.

Chapter 2

Mixed Resolution Ensemble Data Assimilation

2.1 Introduction.

Numerical weather prediction uses a numerical model of atmospheric physics to predict the future states of the atmosphere given an estimate for the current atmospheric state. Uncertainty about the current state, along with inaccuracies in the model dynamics, leads to (greater) uncertainty in the forecast. Quantifying this uncertainty is important for interpreting the forecast and for data assimilation. In data assimilation, information from a short term forecast is combined with information from recent observations, resulting in an estimate of the current atmospheric state used to initialize subsequent forecasts.

One tool for assessing forecast uncertainty is an “ensemble prediction system” (Leith 1974) in which an ensemble of initial conditions are evolved by the model. Ideally, the ensemble of forecasts samples the future atmospheric state, thinking of it as a random variable. Weather services generally make a single forecast at a high resolution, determined by their computational and operational constraints, and forecast an ensemble at lower resolution (Toth and Kalnay 1993; Molteni et al. 1996; Toth and Kalnay 1997; Buizza et al.

2005). The resolution may change for different forecast lead times, e.g., forecast 7 days ahead, then forecast another 7 days at reduced resolution (Buizza et al. 2007). However, at a given lead time all of the ensemble members generally have the same resolution.

In this chapter, we will consider potential advantages in describing forecast uncertainty of an ensemble with multiple members at each of two different resolutions. We will assess the advantage by proposing a mixed-resolution ensemble data assimilation system and comparing its results to the analogous single-resolution assimilation system.

A common paradigm for data assimilation is the Kalman filter. The Kalman filter cycles between the analysis phase and the forecast phase. The "analysis" generated by the Kalman filter is a function of the forecast estimate, called the "background" state, the forecast uncertainty quantified as a background error covariance, new observations, and the observation uncertainty quantified as an observation error covariance. The resulting analysis state is an updated version of the background state, which better fits the observations. Forecasting the analysis provides the background for the next cycle.

Accurately estimating and evolving the background error covariance is a time consuming process. There is growing interest in estimating the background error covariance using sample statistics on ensemble forecasts, either independently (as in ensemble Kalman filters), or in combination with variational methods (e.g., Lorenc 2003). In general, for a k member ensemble of background states

$$\mathbb{X}^b = \begin{bmatrix} \mathbb{X}_1^b & \mathbb{X}_2^b & \dots & \mathbb{X}_k^b \end{bmatrix} \tag{2.1}$$

with mean $\bar{x}^b$ and perturbations

$$X^b = \begin{bmatrix} \mathbb{X}_1^b - \bar{x}^b & \mathbb{X}_2^b - \bar{x}^b & \dots & \mathbb{X}_k^b - \bar{x}^b \end{bmatrix}, \tag{2.2}$$

the background error covariance is estimated by

$$P^b = \frac{1}{k-1} X^b {X^b}^\top. \tag{2.3}$$

The number of members composing the ensemble and the resolution of the model states being evolved both contribute to the accuracy of the forecast uncertainty estimated with the ensemble and also to the computation time of the forecast.

Most ensemble data assimilation literature, e.g. Houtekamer and Mitchell (1998), Anderson and Anderson (1999), Bishop et al. (2001), Whitaker and Hamill (2002), Ott et al. (2004), assumes that all ensemble members have the same resolution. While forecasting the background ensemble at a lower resolution substantially reduces the computational cost involved, it also decreases the accuracy of the analysis. We argue that an ensemble composed of forecasts evolved at two different resolutions may better characterize forecast uncertainty than a single resolution ensemble, within given computation constraints.

Du (2004) proposed combining low resolution ensemble perturbations with a high resolution forecast as part of an ensemble forecast system. Gao and Xue (2008) propose data assimilation with a single high resolution forecast and a low resolution ensemble. In what they call dual resolution data assimilation, the low resolution ensemble is evolved and independently of the high resolution forecast, and the background error covariance estimated from the low resolution ensemble is also used to update a single high resolution forecast. Gao et al. (2010) extend this technique to use the hybrid method of Lorenc (2003). The ensemble perturbations are updated separately, but the ensemble analysis mean can be shifted toward or replaced by the high resolution analysis, as in Zhang and Zhang (2012).

One of the disadvantages of pure ensemble methods is rank deficiency of the covariance, especially for small ensembles. Localization allows smaller ensembles as we discuss in Section 2.3.1, but the rank deficiency can still be a disadvantage when the ensemble size is too small. The background error covariance used in the ensemble based hybrid methods is a weighted average of a full rank static covariance (such as that from a variational method) and a reduced rank ensemble estimated covariance (Hamill and Snyder 2000). However, the ensemble perturbations are updated separately with the reduced rank ensemble estimated error covariance and data assimilation with single resolution ensembles must still balance the available computational resources between the size and resolution of the ensemble.

To achieve the accuracy of a high resolution ensemble for a lower computational cost, we have developed a modification of the Local Ensemble Transform Kalman Filter (LETKF) (Hunt et al. 2007) for a mixed resolution ensemble composed of a small high resolution ensemble and a larger low resolution ensemble. Similarly to the hybrid method, e.g. Hamill and Snyder (2000), we form the background covariance as a linear combination of the sample covariances of the two ensembles. To limit the effects of interpolation between the two model spaces, we use the combined background covariance in observation space as much as possible.

In contrast to hybrid methods where the static covariance does not influence the analysis ensemble perturbations, in mixed resolution ensemble data assimilation both the high and low resolution analysis ensemble perturbations are influence by the combined background error covariance. Additionally, the mixed resolution combined background error covariance could be easily incorporated into hybrid methods e.g. Hamill and Snyder (2000) including frameworks where the cost function is preconditioned e.g. Lorenc (2003). These cost functions are described in Section 2.2.4.

We remark that our method allows the high resolution ensemble to have just one member, with the background covariance formed entirely from the low resolution ensemble, as proposed e.g. by Du (2004), Zupanski (2005), Gao and Xue (2008). When the high resolution ensemble has multiple members however, both ensembles contribute to the background covariance.

We test this hypothesis by comparing analysis errors in a mixed resolution ensemble Kalman filter scheme. In particular, we test the mixed resolution LETKF (mLETKF) on two chaotic models designed by Lorenz (2005). The low resolution model (Model II) is similar to the Lorenz 96 model (Lorenz 1996; Lorenz and Emanuel 1998) but is smoother in that the values at adjacent grid points have a strong positive correlation. The high resolution model (Model III) includes short wave coupling in addition to the smoothing.

In Section 2.2 we motivate and describe the algorithm for performing the mLETKF. For simplicity, we discuss the method mainly in the context of no localization, using the ETKF (Bishop et al. 2001; Wang et al. 2004) as formulated in Hunt et al. (2007). We give a side-by-side comparison of the two algorithms (ETKF and mETKF). In Section 2.3 we describe the full algorithms that we use in our experiments, complete with localization, interpolation, and reduced impact for distant observations. We describe the low and high resolution models (Lorenz (2005) Models II and III) that we use in our experiments, and we describe our experiments. We test the mLETKF most extensively with a simple scenario of limited observations, testing the LETKF for high resolution ensembles of various sizes and testing the mLETKF for mixed resolution ensembles varying the size of both the high and low resolution components of the ensemble, including ensembles with only one high resolution member. We also investigate the potential advantages or disadvantages of the mLETKF for more accurate observations with full coverage and more frequent assimilation, sufficient to resolve the small scale variability, and in less ideal conditions where model error was introduced. We discuss the parameters, which we tuned for best accuracy, and we discuss balancing limited computation resources for best accuracy. We give our results at the end of Section 2.3. In Section 2.4 we conclude that mixed resolution ensemble analysis is beneficial in all of the scenarios tested, in that a greater accuracy can be attained for the same computation time.

2.2 Methodology.

Below we describe in detail how we perform a single analysis step with a mixed-resolution ensemble, in comparison with an ETKF analysis step. We also describe the mETKF in terms of minimizing a cost function and discuss how to incorporate the mETKF within hybrid systems.

In Sections 2.2.1 and 2.2.2 we present the global Ensemble Transform Kalman Filter (ETKF) algorithm and the global mixed-resolution ETKF algorithm (mETKF). However, we used the Local Ensemble Transform Kalman Filter (LETKF) algorithm and the mLETKF algorithm in our experiments. In the LETKF (or mLETKF), a separate ETKF (or mETKF) analysis is done for each model grid point, using only observations from a region local to that grid point. Once the choice of observations is made, the LETKF analysis is equivalent to (though formulated differently than) the Ensemble Transform Kalman Filter (Bishop et al. 2001) with a centered spherical simplex ensemble (Wang et al. 2004). It is also equivalent to the local analysis of LEKF (Ott et al. 2004). We discuss localization and other minor but practical modifications to our algorithm in Section 2.3.1.

2.2.1 ETKF

The Ensemble Transform Kalman Filter (ETKF) algorithm (Bishop et al. 2001; Wang et al. 2004) is designed to generate an analysis ensemble of suboptimal state estimates commensurate with the analysis error covariance indicated by the Kalman filter and centered around the Kalman filter optimal analysis state, while shifting the background ensemble as little as possible to form the analysis ensemble (Ott et al. 2004).

We represent the background ensemble with k members by the matrix $\mathbb{X}^b$, with each column standing for a different ensemble member, $\mathbb{X}_i^b$ (see equation (2.1)). The mean of the ensemble, $\bar{x}^b$, is used as the background state. We constructed the perturbations via equation (2.2), writing each ensemble member as $\mathbb{X}_i^b = \bar{x}^b + X_i^b$. We denote by X^b the matrix of perturbations from the mean (see equation (2.2)). The observations and the observation error covariance are y^o and R^o respectively. Instead of using the background error covariance P^b explicitly in the computations, we represent the implicit background error covariance by a square root matrix $U^b = (k-1)^{-\frac{1}{2}} X^b$, where $P^b = U^b {U^b}^\top$ (equation (2.3)).

The number of elements (rows) in y^o is equal to the number of observations, N_o, and the number of rows in $\mathbb{X}^b$ is equal to N_m, the number of grid points in the model space. For an ensemble of k members, $\mathbb{X}^b$ will have k columns.

Let $h(x)$ be the forward operator that maps from model space into observation space. The matrix $\mathbb{Y}^b$ represents the background ensemble in observation space, i.e., $\mathbb{Y}^b_i = h\left(\mathbb{X}^b_i\right)$. The vector $\bar{y}^b$ and the matrix Y^b represent the mean and perturbations of the background ensemble in observation space so that $\mathbb{Y}^b_i = \bar{y}^b + Y^b_i$. If $h(x)$ is linear then it can be represented by a $[N_o \times N_m]$ matrix H, $h(x) = Hx$. Three intermediary matrices, V^b, C, and M^a, are created by the algorithm for efficiency during computation; V^b has size $[N_o \times k]$, C has size $[k \times N_o]$, and M^a has size $[k \times k]$. Similar to U^b, the matrix U^a is a square root of the analysis error covariance $P^a = U^a {U^a}^\top$. The background ensemble $\mathbb{X}^b$, the analysis ensemble $\mathbb{X}^a$, and their respective implicit error covariances U^b and U^a all have size $[N_m \times k]$.

We have split the ETKF algorithm into 3 major steps. In Step (1) we construct the inputs needed for the ETKF. In Step (2) we create intermediate matrices used in the analysis. In Step (3) we compute the analysis ensemble.

1. (a) $U^b = \frac{1}{\sqrt{k-1}} X^b$

   (b) $V^b = \frac{1}{\sqrt{k-1}} Y^b$

2. (a) $C = {V^b}^\top \left(R^o\right)^{-1}$

   (b) $M^a = \left(\frac{1}{\rho} I + C V^b\right)^{-1}$

3. (a) $\bar{x}^a = \bar{x}^b + U^b M^a C \left(y^o - h\left(\bar{x}^b\right)\right)$

   (b) $U^a = U^b \left(M^a\right)^{\frac{1}{2}}$

   (c) $X^a = \sqrt{k-1} U^a$

   (d) $\mathbb{X}_i^a = \bar{x}^a + X_i^a$

In Step (3c) the square root of M^a is defined as the unique symmetric positive definite matrix whose square is M^a. Using this square root of M^a gives an analysis ensemble with members that are the nearest to their background ensemble counterparts when compared to the analysis ensembles computed using any other square root (Ott et al. 2004).

With $\rho = 1$, the analysis mean $\bar{x}^a$ and covariance $P^a = U^a {U^a}^\top$ are consistent with the Kalman filter applied to $\bar{x}^b$ and P^b. For a variety of reasons including model nonlinearity and limited ensemble size, the background error covariance estimated from an ensemble forecast tends to underestimate the size of the error in the background mean (Houtekamer and Mitchell 1998; Anderson and Anderson 1999; Whitaker and Hamill

2002). The inflation factor ρ in Step (2b) effectively multiplies the background error covariance matrix by ρ. The value of ρ is often determined by tuning to optimize performance in a given scenario (Anderson and Anderson 1999; Whitaker and Hamill 2002). One can also adjust ρ adaptively (Anderson 2007; Li et al. 2009; Miyoshi 2011). Adaptive inflation allows ρ to depend on location, which can be important when the model and observations are spatially heterogeneous.

2.2.2 mETKF

High resolution models are more accurate than low resolution models yet take more time to run. Among ensembles of the same resolution, ensembles with more members provide a better estimate for the background mean and covariance. However, increasing either the size or the resolution of the ensemble also increases the computational cost. Thus, within given computational constraints, a single-resolution ensemble compromises between the size of the ensemble and the resolution of the model. Combining the information in a large, low resolution ensemble with a small, high resolution ensemble could produce a better analysis than either resolution produces by itself. Described below is a method for mixed resolution ensemble analysis as it applies to the Ensemble Transform Kalman Filter (we abbreviate mixed ETKF by mETKF).

The high resolution background ensemble $\mathbb{X}_h^b$ has size $N_h \times k_h$ (grid size × ensemble size). In the same fashion the low resolution background ensemble $\mathbb{X}_\ell^b$ has size $N_\ell \times k_\ell$. The high and low resolution background means are $\bar{x}_h^b$ and $\bar{x}_\ell^b$ respectively. In the mixed algorithm, X_h^b (X_ℓ^b) stands for the high (low) resolution background perturbations from the high (low) resolution ensemble mean:

$$\left[\mathbb{X}_h^b\right]_i = \bar{x}_h^b + \left[X_h^b\right]_i \tag{2.4}$$

$$\left[\mathbb{X}_\ell^b\right]_i = \bar{x}_\ell^b + \left[X_\ell^b\right]_i . \tag{2.5}$$

In observation space, members of the high and low background ensembles are denoted by $\left[\mathbb{Y}_{(h/\ell)}^b\right]_i = h_{h/\ell}\left(\left[X_{h/\ell}^b\right]_i\right) = \bar{y}_{(h/\ell)}^b + \left[Y_{(h/\ell)}^b\right]_i$. The operators h_h and h_ℓ transform high and low resolution model states to observation space. We use parentheses around the subscripts h and ℓ to indicate objects that are related to one of the ensembles but do not lie in the corresponding model state space.

In various equations below, two matrices combine to form a larger augmented matrix. For matrices A and B with the same number of rows, the augmented matrix

$$\left[\begin{array}{c|c} A & B \end{array}\right] = \left[\begin{array}{cccccccc} A_1 & A_2 & \ldots & A_{k_A} & B_1 & B_2 & \ldots & B_{k_B} \end{array}\right]. \tag{2.6}$$

is the matrix with leftmost columns encompassing A and rightmost columns encompassing B.

When doing analysis on a mixed ensemble, we occasionally need to compare vectors and matrices of different resolutions. To create a joint (augmented) ensemble of the background perturbations, we interpolate the low resolution background perturbations onto the high resolution grid by means of the cubic spline with circular boundary conditions. The function $\operatorname{spline}()$ denotes an operator interpolating a low resolution field onto the high resolution grid. Similarly, high resolution fields are projected onto the low resolution grid via the operator $\operatorname{proj}()$. We remark that if the high resolution grid points are a subset of the low resolution grid points, then $\operatorname{proj}()$ is truly a projection operator; otherwise it involves some interpolation as well.

In the mETKF, as in the ETKF, the background error covariance is never explicitly computed. Instead we estimate U^b, the implicit background error covariance ($P^b = U^b U^{b\top}$), and V^b, the implicit background error covariance as expressed in observation space ($R^b = V^b V^{b\top}$), from a weighted combination of the high and low resolution perturbations:

$$U^b = \left[\sqrt{\tfrac{\alpha}{k_\ell - 1}} \operatorname{spline}\left(X_\ell^b\right) \middle| \sqrt{\tfrac{1-\alpha}{k_h - 1}} X_h^b \right], \tag{2.7}$$

$$V^b = \left[\sqrt{\tfrac{\alpha}{k_\ell - 1}} Y_{(\ell)}^b \middle| \sqrt{\tfrac{1-\alpha}{k_h - 1}} Y_{(h)}^b \right]. \tag{2.8}$$

Equivalently, R^b, can be written as a weighted average of the high and low resolution background error covariances

$$R^b_{(h/\ell)} = \frac{1}{k_{h/\ell} - 1} Y^b_{(h/\ell)} {Y^b_{(h/\ell)}}^\top \tag{2.9}$$

$$R^b = \alpha R^b_{(\ell)} + (1-\alpha) R^b_{(h)} = V^b {V^b}^\top. \tag{2.10}$$

The scalar α determines the relative weight of each ensemble. When $\alpha = 1$, only the low resolution perturbations are used to determine the analysis update, i.e. $R^b = R^b_{(\ell)}$. Setting $\alpha = 1$ is necessary when the high resolution ensemble contains only a single member. Even though the high resolution perturbations are ignored, the high resolution single member ensemble will still be updated in the analysis. Similarly when $\alpha = 0$, the low resolution perturbations are ignored and $R^b = R^b_{(h)}$.

The mETKF algorithm is shown below with steps numbered in a corresponding manner to the steps of the ETKF algorithm in Section 2.2.1. The input parameters for the mETKF analysis step are: the background ensembles ($\mathbb{X}^b_h$ and $\mathbb{X}^b_\ell$), the weight parameter α, the inflation factor ρ, and the observations y^o with their error covariance R^o. We construct the augmented implicit background error covariance matrices U^b and V^b in Step (1). Step (2) is useful for efficient computation. In Step (3) we perform the analysis. We remark that the analysis mean for the low (high) resolution ensemble is updated from the background mean for the low (high) resolution ensemble.

1. (a) $U^b = \left[\begin{array}{c|c} \sqrt{\frac{\alpha}{k_\ell - 1}}\text{spline}\left(X_\ell^b\right) & \sqrt{\frac{1-\alpha}{k_h-1}} X_h^b \end{array} \right]$

   (b) $V^b = \left[\begin{array}{c|c} \sqrt{\frac{\alpha}{k_\ell - 1}} Y_{(\ell)}^b & \sqrt{\frac{1-\alpha}{k_h-1}} Y_{(h)}^b \end{array} \right]$

2. (a) $C = V^{b^\top} \left(R^o\right)^{-1}$

   (b) $M^a = \left(\dfrac{1}{\rho} I + C V^b\right)^{-1}$

3. (a) i. $\bar{x}_h^a = \bar{x}_h^b + U^b M^a C \left(y^o - h_h\left(\bar{x}_h^b\right)\right)$

   ii. $\bar{x}_\ell^a = \bar{x}_\ell^b + \text{proj}\left(U^b\right) M^a C \left(y^o - h_\ell\left(\bar{x}_\ell^b\right)\right)$

   (b) $U^a = U^b \left(M^a\right)^{\frac{1}{2}}$

   (c) split U^a into components corresponding to the low and high resolution ensembles: $U^a = \left[\begin{array}{c|c} U_{(\ell)}^a & U_{(h)}^a \end{array} \right]$

   i. $X_h^a = \sqrt{\frac{k_h - 1}{1-\alpha}} U_{(h)}^a$

   ii. $X_\ell^a = \sqrt{\frac{k_\ell - 1}{\alpha}} \text{proj}\left(U_{(\ell)}^a\right)$

   (d) i. $\left[\mathbb{X}_h^a\right]_i = \bar{x}_h^a + \left[X_h^a\right]_i$

   ii. $\left[\mathbb{X}_\ell^a\right]_i = \bar{x}_\ell^a + \left[X_\ell^a\right]_i$

Step (3d) is done for each ensemble member, i.e. $1 \leq i \leq k_{h/\ell}$.

We remark that because proj$\left(\right)$ inverts spline$\left(\right)$, the X_ℓ^a computed by (3c) is the same as if we had formed U^b in low-resolution model space (projecting X_h^b), performed Step (3b) on the low-resolution ensemble, and selected the columns of the low-resolution U^a that correspond to the low-resolution ensemble.

When the high resolution ensemble only has one member, the perturbations from the mean are zero. Since the background error covariance of the high resolution part cannot be computed, no weight should be given to the high resolution component of the covariance. This can be done by setting $\alpha = 1$, using 0 instead of $\sqrt{\frac{1-\alpha}{k_h-1}}$ in Step (1), and setting $X_h^a = \overrightarrow{0}$ in Step (3c). Both the low resolution ensemble and the high resolution member will be updated from the information in the low resolution perturbations.

As the low resolution analysis mean ignores the high resolution background mean, when $\alpha = 1$ the low resolution analysis ensemble from the mETKF is identical to that produced from the ETKF on the low resolution ensemble. Thus a code for mixed resolution ensemble data assimilation can also accommodate single resolution ensembles by setting $\alpha = 1$ and $\mathbb{X}_h^b = \overrightarrow{0}$ (or $\mathbb{X}_h^b = \varnothing$) for a low resolution ETKF or by setting $\alpha = 0$ and $\mathbb{X}_\ell^b = \overrightarrow{0}$ for a high resolution ETKF.

With some models it might be desirable to compute a single analysis mean for both the high and low resolution ensembles, replacing Step 3(a)ii with $\bar{x}_\ell^a = \text{proj}(\bar{x}_h^a)$ and replacing $\bar{x}_h^a$ in Step 3(a)i with $\bar{x}^a$ representing a state estimate better than $\bar{x}_h^a$ if such exists. However, our results were marginally worse when we replaced $\bar{x}_\ell^a$ with $\text{proj}(\bar{x}_h^a)$. Du (2004) explores the forecasting advantages this provides when the high resolution ensemble has only one member.

2.2.3 A side by side comparison

As a thought experiment, consider splitting a $2k$-member ensemble into two equal parts, yielding two k-member ensembles of the same resolution. In this section we will compare the ETKF on the $2k$-member ensemble to the mETKF on the two k-member ensembles. Since both k-member ensembles have the same size and resolution, we weigh them equally, using $\alpha = \frac{1}{2}$. For each step, the ETKF step is given on the left and the comparable mETKF step is given on the right. We label one k-member ensemble with subscript ℓ and the other with subscript h, even though in the present scenario both ensembles are in the same model space, and there is no need for interpolation or projection.

1. (a) $U^b = \frac{1}{\sqrt{2k-1}} X^b$

$$\left[\begin{array}{c|c} \frac{1}{\sqrt{2k-2}} X^b_\ell & \frac{1}{\sqrt{2k-2}} X^b_h \end{array} \right] = U^b$$

   (b) $V^b = \frac{1}{\sqrt{2k-1}} Y^b$

$$\left[\begin{array}{c|c} \frac{1}{\sqrt{2k-2}} Y^b_{(\ell)} & \frac{1}{\sqrt{2k-2}} Y^b_{(h)} \end{array} \right] = V^b$$

2. (a) $C = {V^b}^\top {R^o}^{-1}$

$${V^b}^\top {R^o}^{-1} = C$$

   (b) $M^a = \left(\dfrac{1}{\rho} I + C V^b \right)^{-1}$

$$\left(\frac{1}{\rho} I + C V^b \right)^{-1} = M^a$$

3. (a) $\bar{x}^a = \bar{x}^b + U^b M^a C \left(y^o - h\big(\bar{x}^b\big) \right)$

$$\bar{x}^b_{h/\ell} + U^b M^a C \left(y^o - h \left(\bar{x}^b_{h/\ell} \right) \right) = \bar{x}^a_{h/\ell}$$

(b) $U^a = U^b \left(M^a \right)^{\frac{1}{2}}$

$$U^b \left(M^a \right)^{\frac{1}{2}} = U^a = \left[\begin{array}{c|c} U^a_{(\ell)} & U^a_{(h)} \end{array} \right]$$

(c) $X^a = \sqrt{2k-1} U^a$

$$\sqrt{2k-2} U^a_{(h/\ell)} = X^a_{h/\ell}$$

(d) $\mathbb{X}^a_i = \bar{x}^a + X^a_i$

$$\bar{x}^a_{h/\ell} + \left[X^a_{h/\ell} \right]_i = \left[\mathbb{X}^a_{h/\ell} \right]_i$$

The differences are as follows. First, in the mETKF separate means $\bar{x}_h$ and $\bar{x}_\ell$ are computed for the two ensembles; these means are used explicitly in Steps (3a) and (3d), and also in forming the ensemble perturbations in Step (1). Second, the rank of the mETKF combined background error covariance is $2k - 2$, while the rank of the corresponding ETKF background error covariance is slightly larger ($2k - 1$). Thus, for large k the mETKF and the ETKF should produce similar results, but for small k, the mETKF will be more influenced by sampling error. We note that the ensemble Kalman filter proposed by Houtekamer and Mitchell (1998) also uses two equal-size ensembles and updates their means separately, but compensates for sampling error by updating the mean from one ensemble using only the background covariance computed by the other ensemble. While the combined background covariance of mETKF is probably detrimental in this thought-experiment scenario, it is an essential feature of mETKF in the mixed-resolution scenario.

2.2.4 Cost Function

The ETKF and the mETKF can also be considered in terms of minimizing a cost function. We also show the cost function for a mixed ensemble based hybrid scheme.

2.2.4.1 The ETKF Cost Function

For the ETKF this cost function is

$$J^*(w) = \rho^{-1} w^\top w + \left(y^o - \bar{y}^b - V^b w\right)^\top {R^o}^{-1} \left(y^o - \bar{y}^b - V^b w\right), \tag{2.11}$$

where $\bar{y}^b = h\big(\bar{x}^b\big)$ and ρ is the background error covariance inflation. If h is linear then $h\big(\bar{x}^b + U^b w\big) = \bar{y}^b + V^b w$, and it follows that if $w^a \in \mathbb{R}^k$ minimizes J^* then $\bar{x}^a = \bar{x}^b + U^b w^a$ minimizes the alternative cost function

$$J(x) = \left(x - \bar{x}^b\right)^\top \left(\rho P^b\right)^{-1} \left(x - \bar{x}^b\right) + \left(y^o - h(x)\right)^\top {R^o}^{-1} \left(y^o - h(x)\right) \tag{2.12}$$

in the space spanned by the ensemble (Hunt et al. 2007).

2.2.4.2 The mETKF Cost Function

A similar cost function applies to each resolution in the mETKF:

$$J^*_{h/\ell}(w) = \rho^{-1}w^\top w + \left(y^o - \bar{y}^b_{(h/\ell)} - V^b w\right)^\top {R^o}^{-1} \left(y^o - \bar{y}^b_{(h/\ell)} - V^b w\right), \tag{2.13}$$

where $\bar{y}^b_{(h/\ell)} = h_{h/\ell}\Big(\bar{x}^b_{h/\ell}\Big)$. Similarly, when $h_{h/\ell}$ is linear, then $\bar{x}^a_h = \bar{x}^b_h + U^b w^a$ and $\bar{x}^a_\ell = \bar{x}^b_\ell + \text{proj}\left(U^b\right) w^a$ minimize the alternative cost function

$$J_{h/\ell}(x) = \left(x - \bar{x}^b_{h/\ell}\right)^\top \left(\rho P^b\right)^{-1} \left(x - \bar{x}^b_{h/\ell}\right) + \left(y^o - h_{h/\ell}(x)\right)^\top {R^o}^{-1} \left(y^o - h_{h/\ell}(x)\right) \tag{2.14}$$

in the space spanned by the ensemble, provided $w^a \in \mathbb{R}^{k_{h+\ell}}$ minimizes $J^*_{h/\ell}$.

The minimizer w^a for J^*_h is

$$w^a = \left(\tfrac{1}{\rho}I + {V^b}^\top {R^o}^{-1} V^b\right)^{-1} {V^b}^\top {R^o}^{-1}\left(y^o - h_h\left(\bar{x}^b_h\right)\right), \tag{2.15}$$

and the equation for the high resolution analysis mean from Step 3(a)i of Section 2.2.2 gives

$$\bar{x}^a_h = \bar{x}^b_h + U^b w^a, \tag{2.16}$$

For linear h_h, $\bar{x}^a_h$ (equation (2.16) and Step 3(a)i) minimizes the cost function J_h (equation 2.14) in the space spanned by the ensemble. A similar derivation applies to the low resolution analysis $\bar{x}^a_\ell$.

In summary, the mETKF analysis minimizes separate cost functions for the high and low resolution ensembles. The two ensembles are coupled by the combined perturbation matrix V^b in both cost functions (equation 2.13), or equivalently by the combined background covariance matrix P^b (equation 2.14).

2.2.4.3 The Hybrid mETKF Cost Function

As we have not explored mixed resolution ensemble based hybrid methods, this section describes a generalized theoretical cost function. We consider a hybrid method where a control forecast x^b_h is performed at high resolution and ensembles are forecast at mid resolution $\mathbb{X}^b_m$ and low resolution $\mathbb{X}^b_\ell$. We assume that the control forecast is the most accurate and hence the best prior estimate.

The hybrid mETKF cost function similar to that given in Lorenc (2003) and Gao et al. (2010) is given by

$$\begin{aligned} J^*_{\text{hybrid}}(v,w) &= v^\top v + \rho^{-1} w^\top w \\ &+ \left(y^o - \bar{y}^b - \beta_1 h\left(\left(P^b_{\text{static}}\right)^{\frac{1}{2}} v\right) - \beta_2 V^b w\right)^\top R^{o^{-1}} \left(y^o - \bar{y}^b - \beta_1 h\left(\left(P^b_{\text{static}}\right)^{\frac{1}{2}} v\right) - \beta_2 V^b w\right), \end{aligned} \tag{2.17}$$

where $y^b_h = h\left(x^b_h\right)$ and $\beta_1^2 + \beta_2^2 = 1$.

As in the previous sections (2.2.4.1 and 2.2.4.2), when h is linear, then $x_h^a = x_h^b + \beta_1 \left(P^b_{\text{static}}\right)^{\frac{1}{2}} v^a + \beta_2 U^b w^a$ minimizes the alternative cost function similar to that given in Hamill and Snyder (2000):

$$J(x) = \left(x - \bar{x}^b\right)^\top \left(\beta_1^2 P^b_{\text{static}} + \beta_2^2 \rho U^b {U^b}^\top\right)^{-1} \left(x - \bar{x}^b\right) + \left(y^o - h(x)\right)^\top {R^o}^{-1} \left(y^o - h(x)\right). \tag{2.18}$$

Further details on the equivalence between Hamill and Snyder (2000) and Lorenc (2003) are provided in Wang et al. (2007).

2.3 Numerical Experiments and Results.

To test the mixed resolution LETKF, we used two chaotic models designed by Lorenz (2005) to generate the high and low resolution ensembles. Both models are extensions of the Lorenz 96 model (Lorenz 1996; Lorenz and Emanuel 1998). The latter model is called Model I in Lorenz (2005), and is smoothed to produce Model II, which we use as our low resolution model. The high resolution Model III introduces short wave coupling in addition to the smoothing.

We tested the mixed resolution LETKF and the standard LETKF in several cases with Observing System Simulation Experiments (OSSEs). In these types of experiments, a computer is not only used to generate the forecasts, it is also used to generate simulated observations. It creates a trajectory that we call the “truth” and from this it generates observations by adding random error to the truth at selected points. In most of our experiments, we use a “limited observation” network consisting of one simulated observation every 24 Model III grid points. We also consider a “small-scale resolving” scenario in which we have observations at every grid point.

2.3.1 Modifications to the Algorithms.

In the LETKF, the ETKF analysis is performed for each grid point, using only the observations in a neighborhood surrounding the grid point. This eliminates long-range background correlations that may be spurious due to limited ensemble size. In addition, global errors may grow in many directions in model space, but fewer directions are needed to describe local errors. The result of local analysis is that one member of the ensemble may be heavily weighted near one grid point, yet lightly weighted near another, so a smaller ensemble can be used to cover all of the directions of error growth. See Greybush et al. (2011) for further discussion of localization. For a mixed ensemble, the mLETKF localization is done in the same fashion as the LETKF, i.e. the mETKF is done on each grid point, using only the observations in a neighborhood surrounding the grid point. In all cases we use a localization radius of 32 Model III grid points, i.e. 2 or 3 observations per local region in our limited observation network.

When observations are sparse, edge effects can cause adjacent grid points to have significantly different analyses. For instance, consider the extreme case of a grid point whose local region contains no observations, which is adjacent to a grid point with one observation on the very edge of its local region. The analysis at the first grid point would be the background at that grid point, since no observations means no update. However, the second grid point would be updated to more closely align with the observation quite a distance from it. To reduce this unrealistic choppiness, in our limited observation scenarios we tapered the influence of distant observations by increasing the observation error covariance associated with observations near the edge of a local region (Greybush et al. 2011). Hunt et al. (2007) suggest multiplying the elements of the inverse observation error covariance matrix $(R^o_{local})^{-1}$ by a weight function, which is 1 for observations in the center of the local region and dwindles to 0 for observations beyond the edge of the local region. Our tapering weight function tapered the elements of $(R^o_{local})^{-1}$ in a trapezoid-like fashion. For observations on the very edge of a local region, we increased the observation error covariance by 4 times. For observations in the exact center of the tapered region, we doubled the observation error covariance. For observations located elsewhere in the tapered region, we multiplied the observation error covariance by an interpolated power of 2. For observations in the central half of the local region, we did not change the observation error covariance.

With the LETKF, the analyses at different grid points can be done in parallel. If grid points are closely spaced, it can be advantageous to calculate the "weights" $w^a = M^a C\left(y^o - h\left(\bar{x}^b\right)\right)$ (see equation (2.15)) and $W^a = (M^a)^{\frac{1}{2}}$ only on a subset of the grid points and interpolate these onto rest of the grid points (Yang et al. 2009). These weights multiply the background ensemble perturbation matrix U^b in Steps (3a) and (3b) of our ETKF formulation given in Section 2.2.1. The analysis ensemble at a particular grid point j is computed from the interpolated $w^a_{(j)}$ and $W^a_{(j)}$ by finding the analysis mean and covariance

$$\bar{x}^a_{[j]} = \bar{x}^b_{[j]} + U^b_{[j]} w^a_{(j)} \tag{2.19}$$

$$U^a_{[j]} = U^b_{[j]} W^a_{(j)}, \tag{2.20}$$

and computing $\mathbb{X}^a_{[j]}$ from Steps (3c) and (3d). Here the subscript $[j]$ denotes the rows corresponding to grid point j. We performed the analysis at every fourth Model III grid point (i.e. the Model II grid) and used weight interpolation to determine the analysis at the other Model III grid points.

2.3.2 Models.

Following Lorenz (2005), we use a circular grid of 240 points for Model II, and for Model III we use 960 points that refine the Model II grid by a factor of 4. To avoid confusion, we introduce the *unit distance* defined as the distance between one Model III grid point and the next. Thus the circle is 960 units in circumference and Model II grid points are 4 units apart. A snapshot of both models, overlaid for comparison, is given in Figure 2.1.

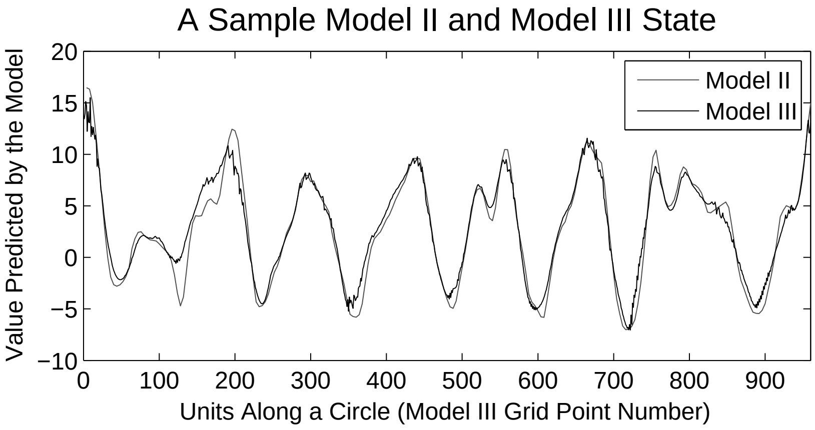

Figure 2.1: Sample Model II and Model III states shown simultaneously. Model III has short wave coupling at high resolution, while Model II, with a lower resolution, is smooth.

Both models use an averaging function on nearby grid points, where the smoothing parameter K determines the distance over which the average is taken. When K is odd, the averaging function $\langle\rangle_{K;n}$ refers to the arithmetic mean over the K grid points nearest (and including) the grid point indexed by the number n, i.e. grid points $n - \frac{K-1}{2}$ to $n + \frac{K-1}{2}$. When K is even, $\langle\rangle_{K;n}$ is the average over the region of length K grid points, centered at grid point number n, with boundary points weighted half as much as interior points, i.e. considering the nearest $K+1$ grid points, the $K-1$ interior grid points are weighted by

$\frac{1}{K}$ and the two most distant grid points ($n - \frac{K}{2}$ and $n + \frac{K}{2}$) are weighted by $\frac{1}{2K}$.

The averaging function for an even valued smoothing parameter is related to the $\sum'$ notation of Lorenz (2005) according to the the equation

$$\langle x \rangle_{K,n} = \frac{1}{K} \sum_{j=-K/2}^{K/2}{}' x_{n+j}, \tag{2.21}$$

where

$$\sum_{j=-J}^{J}{}' x_{n+j} = \frac{(x_{n-J} + x_{n+J})}{2} + \sum_{j=-J+1}^{J-1} x_{n+j}. \tag{2.22}$$

We use the $\sum'$ notation when describing Model II and Model III and also when defining the long wave component of Model III.

Model II is given by

$$\begin{aligned} \frac{dx_n}{dt} &= \tfrac{1}{K} \sum_{j=-K/2}^{K/2}{}' x_{n+K+j} \cdot \langle x \rangle_{K;n-K+j} \\ &\quad - \langle x \rangle_{K;n-2K} \langle x \rangle_{K;n-K} - x_n + F \end{aligned} \tag{2.23}$$

when K is even; when K is odd, $\sum_{j=-(K-1)/2}^{(K-1)/2}$ replaces $\sum' {}_{j=-K/2}^{j=K/2}$. F represents the forcing constant in this differential equation. When $K = 1$, no averaging takes place and Model II reduces to the Lorenz 96 model (Lorenz 1996; Lorenz and Emanuel 1998). Following suggestions in Lorenz (2005), all of our experiments use a smoothing parameter of $K = 8$

(corresponding to a smoothing over 32 Model III units) and most use a forcing constant of $F = 15$. Some experiments use different values of F to simulate model error.

For our small-scale resolving scenario, we performed data assimilation every 0.005 time units and in all other cases we performed data assimilation every 0.05 time units. We integrated Model II via the fourth order Runge-Kutta scheme with a time step size of 0.005 or $0.05/2$ respectively.

In our experiments the truth and the high resolution ensemble were both simulated with the high resolution model, Model III, given by the equation

$$\begin{aligned}\frac{dz_n}{dt} &= \tfrac{1}{K}\sum_{j=-K/2}^{K/2}{}' x_{n+K+j} \cdot \langle x \rangle_{K;n-K+j} \\ &\quad - \langle x \rangle_{K;n-2K} \langle x \rangle_{K;n-K} - x_n + F \\ &\quad + b^2 \left(y_{n+1} - y_{n-2} \right) y_{n-1} - b y_n \\ &\quad + c \left(x_{n+1} y_{n-1} - y_{n-2} x_{n-1} \right) \qquad (2.24)\end{aligned}$$

where $\sum_{j=-(K-1)/2}^{(K-1)/2}$ replaces $\sum{}'^{\,j=K/2}_{\,j=-K/2}$ when K is odd and x and y are defined as follows in terms of z.

$$\begin{aligned}x_n &= \sum_{i=-I}^{I}{}' \left(\frac{3I^2+3}{2I^3+4I} - \frac{2I^2+1}{I^4+2I^2}|i| \right) z_{n+i} \qquad (2.25) \\ y_n &= z_n - x_n\end{aligned}$$

The coefficients b^2 and c are parameters of the model that drive the coupling. The number I is chosen so that x will not include short waves; the coefficients in the equation defining x_n filter out non-quadratic fluctuations between $n-I$ and $n+I$ (Lorenz 2005). The vectors x and y in equation (2.24) then correspond to the long and short wave components of z respectively.

For Model III, we use a smoothing parameter of $K = 32$ (smoothing over the same length scale as Model II) and a forcing constant of $F = 15$. Our other parameters were $I = 12$, $b = 10$, and $c = 2.5$. This gives short waves that vary about 10 times faster than the long waves with about $1/10$ the amplitude (Lorenz 2005). Integration was done via the fourth order Runge-Kutta scheme with a time step size of $0.05/24$ or $0.005/3$. While the Model III can run without analysis with a larger time step size of $0.05/12$ as in Lorenz (2005), we found that data assimilation with Model III requires a smaller time step due to the instability of evolving with short waves of artificially large amplitude. All of our other parameters are consistent with those recommended by Lorenz (2005).

2.3.3 Experiments.

As mentioned above, our experiments took place on a gridded circle of size 960 units, where adjacent Model III grid points are 1 unit apart and adjacent Model II grid points are 4 units apart. We tested the mLETKF with a limited observation network of 40

simulated observations spaced evenly every 24 units around the entire circle, and with a dense observation network of 960 simulated observations, one at each Model III grid point. We simulated observations by adding uncorrelated Gaussian noise to a model run that we regard as the "truth". In all scenarios below we used the same truth run generated from Model III with forcing constant $F = 15$. In our small-scale resolving scenario we used the dense observation network, with observations and analyses every 0.005 time units, and observation errors with standard deviation 0.3, which is about the size of the small scale variability (Yoon and Ott 2010). In all other cases, we used the limited observation network, with observations and analyses every 0.05 time units, and observation errors with standard deviation 2. We remark that observing and performing data assimilation every 0.05 time units was not sufficient to resolve the small-scale variables y_n, which vary 10 times faster than the large-scale variables x_n; this is consistent with the findings of Ballabrera-Poy et al. (2009) for a similar model.

Our basic limited observations scenario and our small-scale resolving scenario have a "perfect" high resolution model ($F = 15$ for both truth and data assimilation). To consider scenarios where both models are imperfect, we used a different forcing constant for the ensemble forecasts than for the truth run. We tested two model error scenarios. In one scenario we used a forcing constant of $F = 14$ for both the high and low resolution forecasts. In the other scenario we used a forcing constant of $F = 12.5$. Introducing model error detracts from the high resolution model's capability to resolve the small scales, so we only tested our small-scale resolving scenario within the perfect model framework.

We tested our three limited observation scenarios (basic, imperfect model $F = 14$, and imperfect model $F = 12.5$) with the mLETKF on three different sizes of mixed resolution ensembles and with the LETKF on three different sizes of high resolution ensembles. In the majority of our mixed resolution experiments, the low resolution component of the ensemble contained $k_\ell = 30$ members and only the size of the high resolution component was varied with $k_h = 1$, 2, and 3 members. The high resolution ensembles tested contained $k = 3$, 4, and 5 members. In addition, we tested our basic scenario on numerous other ensemble size combinations. Similarly, we tested our small-scale resolving scenario with the mLETKF on three different sizes of mixed resolution ensembles and with the LETKF on four different sizes of high resolution ensembles (15, 16, 17 or 18 members). In each of the mixed cases, the low resolution component of the ensemble contained 30 members and the high resolution component contained 9, 12, or 15 members. We present these results in Section 2.3.6.

We measure the accuracy of each experiment as the root mean square difference between the analysis and the truth at the Model III grid points, averaged over 4500 assimilation steps. We ran each experiment for a total of 5000 assimilation steps, but we ignored the first 500 assimilation steps, giving the ensembles time to spin-up. Specifically, the accuracy of each experiment is given by

$$\text{RMSE} = \tfrac{1}{4500} \sum_{i=501}^{5000} \text{rms}\left(x^a(t_i) - x^t(t_i)\right), \tag{2.26}$$

where rms is the root-mean-square function over the 960 grid points, $x^t(t_i)$ refers to the true model state at the i^{th} assimilation step, and $x^a(t_i)$ refers to the analysis state (best estimate) at the i^{th} assimilation step in Model III space, i.e. $x^a(t_i) = \bar{x}^a_h(t_i)$ for mixed resolution experiments, $x^a(t_i) = \bar{x}^a(t_i)$ for high resolution experiments, and $x^a(t_i) =$ spline $(\bar{x}^a(t_i))$ for low resolution experiments.

2.3.4 Parameters Tuned.

For each result below we tuned the covariance inflation, ρ, in order to minimize the analysis RMSE. While tuning ρ substantially improves the accuracy, we found that the results were generally less sensitive to changes in α. Recall that α determines the weight given to the high resolution ensemble versus the low resolution ensemble when determining the implicit background error covariance.

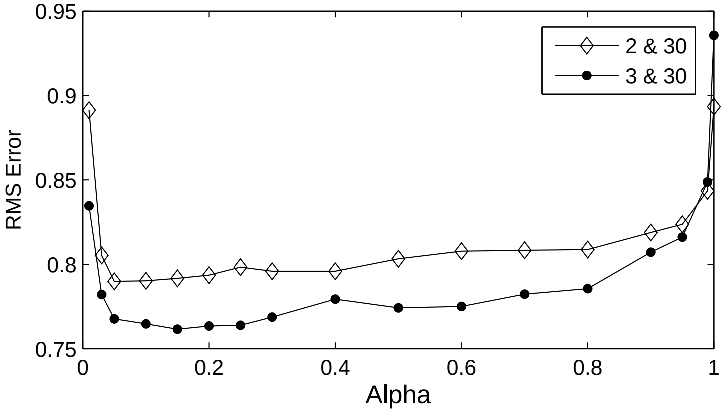

Figure 2.2: Dependence of the analysis RMSE on α for the 2 & 30 (circle) and 3 & 30 (diamond) mixed cases in our basic scenario. The results are averaged over 4500 assimilation steps after a spin up of 500 assimilation steps. When $\alpha = 1$ only the low resolution perturbations are used to determine the analysis update. The flatness of the graph indicates that α does not need extensive tuning. Notice that $\alpha = .2$ is close to optimal in both cases shown, and is well separated from the large errors we observe for $\alpha < 0.05$.

The representative example shown in Figure 2.2 indicates that the RMSE does not change substantially for any value of α between $\alpha = 0.1$ and $\alpha = 0.9$. When $\alpha = 1$, only the low resolution perturbations are used to determine the analysis update. Similarly when $\alpha = 0$, only the high resolution perturbations contribute to the analysis update. Therefore, in the mixed case with only one high resolution member, we necessarily use $\alpha = 1$. We tuned α for the four mixed cases of 2 or 3 high resolution and 30 low resolution ensemble members both with our basic scenario and with model error $F = 12.5$. Since $\alpha = 0.2$ was close to ideal for each of these cases, in all of our basic and model error results below, we use $\alpha = 0.2$. In our small-scale resolving experiments α tuned to 0.8.

Though we discuss the joint covariance as a weighted average of the high and low resolution covariances, the flatness of Figure 2.2 for intermediate values of α and the larger tuned value of α for our small-scale experiments indicates that this interpretation may be too simplistic. Changing the value of α also changes, after many assimilation cycles, the relative variances of the high and low resolution ensembles. Larger values of α result in smaller low resolution perturbations and larger high resolution perturbations. For example, in our basic scenario with 3 high resolution members and 30 low resolution members, increasing α from $\frac{1}{3}$ to $\frac{2}{3}$ roughly halves the variance of the low resolution ensemble and increases the variance of the high resolution ensemble by 30%:

$$P_\ell^b(\alpha = \tfrac{2}{3}) \approx \tfrac{1}{2} P_\ell^b(\alpha = \tfrac{1}{3}) \tag{2.27}$$

$$P_h^b(\alpha = \tfrac{2}{3}) \approx 1.3 \cdot P_h^b(\alpha = \tfrac{1}{3}). \tag{2.28}$$

As a result, the low resolution part of the joint covariance $P^b = \alpha P_\ell^b + (1 - \alpha) P_h^b$ stays about the same as α changes from $\frac{1}{3}$ to $\frac{2}{3}$, while the high resolution part decreases (but by a smaller factor than $1 - \alpha$ decreases). We have not fully explored the relationship between α and ensemble spread.

In each limited observation experiment, we tried values of ρ differing in the hundredths place and our results below use the value of ρ that gave the most accurate RMSE.

Values of ρ were between 1.04 and 1.13 where the higher values were optimal when the forecast model was less accurate or when the ensemble size was small. In mixed resolution ensembles, ρ tuned to slightly lower values than in single resolution ensembles of comparative computational cost. In our small-scale resolving experiments, ρ tuned to a much larger values, even as large as 1.45, so we tuned ρ along a coarser mesh of size 0.05.

2.3.5 Balancing Limited Computational Resources.

Computation speed limits the accuracy obtainable in practice. The time taken by the forecast is directly proportional to the size of the ensemble, provided that the same model is being used to forecast the entire ensemble. More generally, the time taken to evolve the ensemble forecast equals the sum of the times taken to evolve each member. Though the forecast can be done much more quickly in parallel, computational resources still limit the maximum ensemble size.

The forecast computation times shown follow a linear proportionality with the ensemble size. In our experiments, forecasting each high resolution member took about 100 times longer than forecasting each low resolution member. Therefore, when forecasting a mixed ensemble, a relatively large low resolution ensemble can be used without much impact on the total forecast computation time.

However, the analysis time hinges more upon the size of the total ensemble, which in our experiments is approximately the size of the large, low resolution ensemble. Generally speaking, the forecast time limits the size of the high resolution ensemble we can use, and the analysis time inhibits how large of a low resolution ensemble we can use. The relationship between analysis time and forecast time is situation dependent however, so we only give results in terms of forecast time, which is relevant to both ensemble forecasting to assess forecast uncertainty as well data assimilation.

In Table 2.1 we compare results for our basic scenario from mixed resolution ensembles with high resolution components of 1, 2, and 3 members coupled with low resolution components of 5, 10, 20, 30, and 60 members. We found that accuracy does not diminish extensively even for as few as 10 low resolution members. In later results, we chose to use 30 low resolution members as it seemed a good balance between accuracy and time for our limited observation scenarios. In our small-scale resolving scenario, we also chose to use 30 low resolution members.

	$k_h = 1$	$k_h = 2$	$k_h = 3$
$k_\ell = 5$	0.98	0.85	0.80
$k_\ell = 10$	0.92	0.83	0.78
$k_\ell = 20$	0.91	0.80	0.76
$k_\ell = 30$	0.90	0.79	0.76
$k_\ell = 60$	0.87	0.79	0.76

Table 2.1: Varying the size of the low & high resolution ensembles. This table gives the analysis RMSEs for our basic scenario with various mixed ensembles. A low resolution component with 20 or 30 members does nearly as well as with 60.

2.3.6 Results.

In order to estimate the ideal accuracy obtainable using the LETKF with the high resolution model, we tested our basic scenario and our small-scale resolving scenario on increasingly greater ensemble sizes. We summarize our scenarios in Table 2.2. Similarly for the low resolution model, we tested our basic scenario for various ensemble sizes. These results are graphed in Figure 2.3. The lowest obtainable RMSE in our small-scale resolving scenario for assimilation with a low resolution is 0.17 and only when results are smaller than that do we say that they resolve the small scale variability.

Table 2.2: Summary of scenarios. The limited observation network consists of 40 observations and the dense observation network has full coverage (960 observations). The true forcing constant is $F^t = 15$; forcing constants different from $F = 15$ simulate model error. The observation error refers to the standard deviation of the error in the simulated observations.

Scenario	observation network	Forcing constant	observation error
basic	limited	$F = 15$	$\sigma = 2$
$F = 14$	limited	$F = 14$	$\sigma = 2$
$F = 12.5$	limited	$F = 12.5$	$\sigma = 2$
small-scale	dense	$F = 15$	$\sigma = 0.3$

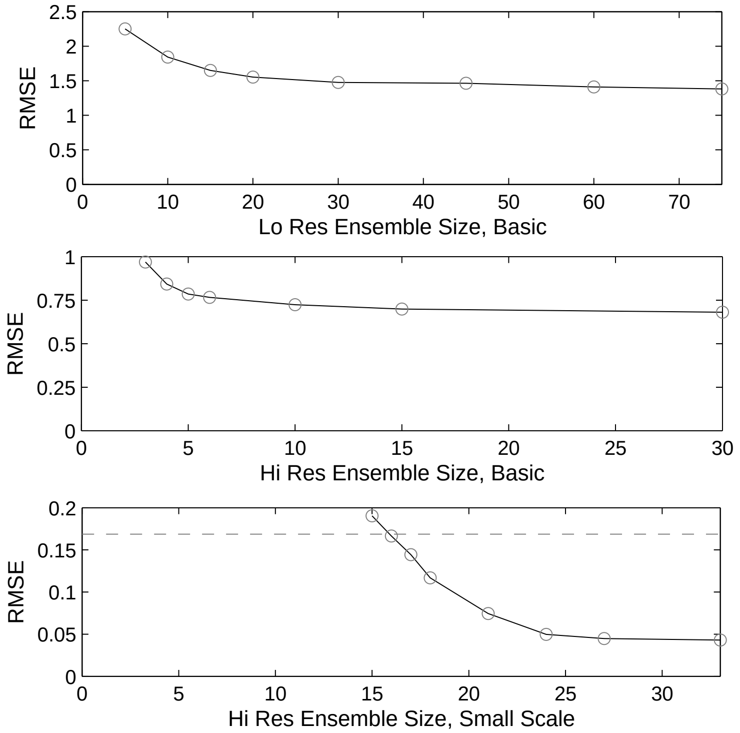

Figure 2.3: *Top*: the RMSE of the analysis generated in our basic scenario with a low-resolution model, averaged over 4500 assimilation steps after a 500 time step spin-up period. *Middle*: the RMSE of the analysis generated in our basic scenario with a high-resolution model, averaged over 4500 assimilation steps after a 500 assimilation step spin-up period. *Bottom*: the RMSE of the analysis generated in our small-scale resolving scenario with a high-resolution model, averaged over 4500 assimilation steps after a 500 assimilation step spin-up period. The dashed line indicates the small scale variability, i.e. the minimum RMSE with a low resolution model for our small-scale resolving scenario.

Our remaining numerical results represent the accuracy of the mLETKF in each of our scenarios in terms of the RMSE (equation (2.26)). We compared these mixed resolution results to the accuracy of the LETKF with various ensembles sizes at both low and high resolution. At each assimilation time, we compared our high resolution forecast mean with the truth and found the root mean square error over the grid (see equation (2.26)). Starting from a random ensemble of model states, we ran each experiment for 5000 assimilation steps. Figure 2.4 shows a plot of the root mean square error for the first 1000 assimilation steps, using the 2 high resolution and 30 low resolution mixed ensemble in our basic scenario. The RMSE behaves similarly continuing out to 10^5 assimilation steps (the maximum we tested), and in other cases and scenarios. Though spin-up appears to occur quickly, we chose to be cautious, using a longer spin-up of 500 assimilation steps.

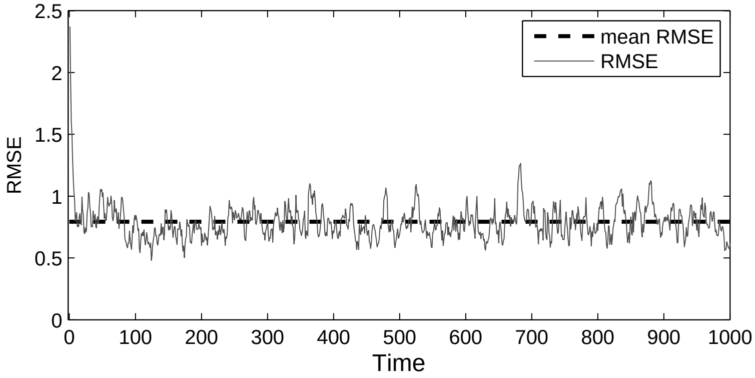

Figure 2.4: The root mean square error at each assimilation step for the initial 1000 assimilation steps. This data is taken from the run of the 2 high resolution and 30 low resolution basic scenario.

Excluding this spin-up period, we took the mean and standard deviation of the root mean square error over the 4500 remaining assimilation steps. The time averaged errors are shown in Tables 2.3 and 2.5 and the standard deviation of the fluctuations about that mean over time are shown in Tables 2.4 and 2.5. These fluctuations, as depicted in Figure 2.4 are correlated in time and do not allow a simple calculation of the sampling error in the RMSEs given in Tables 2.3 and 2.5. To roughly assess the sampling error, we took the standard deviation of a 500 assimilation step moving average, with results summarized in the caption of Table 2.3. The standard deviation of the moving average in most cases was between 0.01 and 0.03. However, the 3 member high resolution cases and the 4 member large model error case ($F = 12.5$) were not as precise, with a standard deviation of the moving average between 0.04 and 0.07. This suggests that the sampling error in the average RMSEs is at most 1 or 2 in the second decimal place.

The short term fluctuations in root mean square error for the mixed resolution cases are consistently small in all cases (Tables 2.4 and 2.5). For our high resolution cases with limited observations, these fluctuations are significantly larger, even for similar sizes of average RMSE. Greater error in the model induces larger fluctuations, whereas increasing the size of the ensemble decreases the size of the fluctuations.

2.4 Summary and Discussion

In Section 2.2, we discussed standard ensemble Kalman filter algorithms that estimate the background error covariance with the matrix expression:

$$\left(\frac{1}{\sqrt{k-1}}X^b\right)\left(\frac{1}{\sqrt{k-1}}X^b\right)^\top \tag{2.29}$$

where k is the ensemble size and X^b are perturbations from the background mean (equation (2.3)). With an ensemble of mixed resolution, we estimated the background error covariance (equation (2.7)) in a similar manner as $U^b {U^b}^\top$, where

$$U^b = \left[\ \sqrt{\tfrac{\alpha}{k_\ell - 1}}\text{spline}\left(X^b_\ell\right)\ \middle|\ \sqrt{\tfrac{1-\alpha}{k_h - 1}}X^b_h\ \right]. \tag{2.30}$$

Here k_ℓ and k_h are the low and high resolution ensemble sizes respectively, X^b_ℓ and X^b_h are the low and high resolution perturbations from the means of their respective ensembles, “spline” represents interpolation from the low resolution to the high resolution grid, and α is a weight based upon the accuracies of the two ensembles and dictating the influence of each ensemble in the analysis. We gave the mixed resolution algorithm associated with the Local Ensemble Transform Kalman Filter and a comparison to the parent algorithm.

We tested this mixed resolution ensemble analysis with the Lorenz Models II & III described in Section 2.3.2. The low resolution model, Model II, smoothly transitions from one grid point to the next. The high resolution model, Model III, also contains a short wave coupling component.

We performed experiments with three different limited observation scenarios: basic, model error $F = 14$, and model error $F = 12.5$. The latter two cases simulate situations where the true system dynamics are not perfectly known. We also performed experiments with dense, frequent, and more accurate observations to simulate a situation where it is possible to resolve the small scale variability.

We tested ensembles of various sizes. In the high resolution, limited observation experiments, we tested ensembles of size 3, 4, 5, and 6. In the mixed resolution experiments, we reported results with 30 low resolution ensemble members and 1, 2, or 3 high resolution members. In our small-scale resolving experiments, we tested high resolution ensembles of size 15, 16, 17, and 18 and we tested mixed resolution ensembles with 30 low resolution members and 9, 12, and 15 high resolution members. We also conducted experiments with different numbers of low resolution ensemble members. We summarize the forecast computation time and the analysis RMSEs for our basic and small-scale resolving scenarios using different numbers of ensemble members in Figure 2.5.

In our mixed resolution and high resolution OSSEs (observing system simulation experiments), forecasting the high resolution ensemble took up the bulk of the overall forecast computation time. However, introducing an additional high resolution member

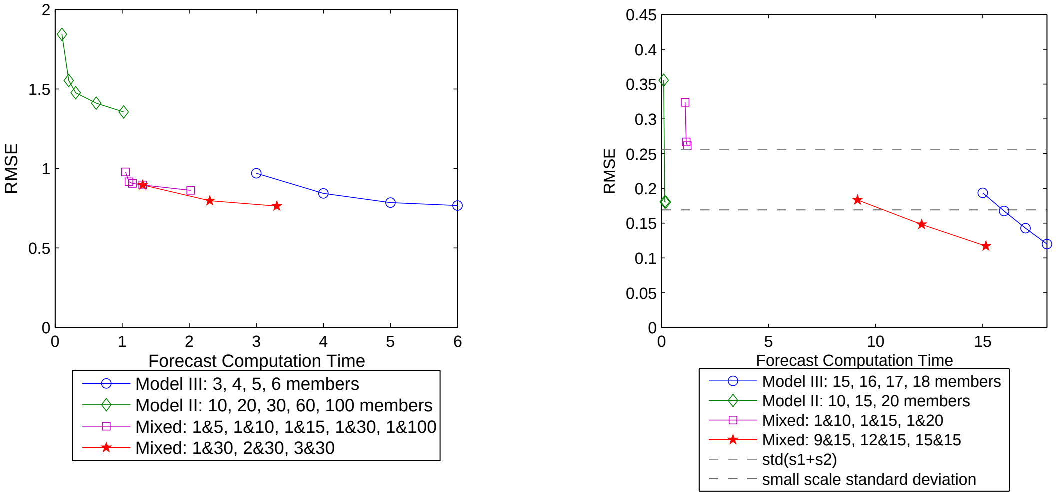

Figure 2.5: The average forecast computation time taken for each assimilation time step versus the average root mean square error over 4500 assimilation steps for *left*: our basic scenario and *right*: our small-scale resolving scenario. Better computation times and better RMSE (lower error) are both closer to the origin. *Left*: Data is shown for high resolution analysis with 3, 4, 5, and 6 members, low resolution analysis with 10, 20, 30, and 60 members, mixed resolution analysis on 1 high resolution member with a low resolution ensemble of 5, 10, 15, 30, and 60, and mixed resolution analysis on a mixed ensemble of 30 low resolution members and 1, 2, and 3 high resolution members. *Right*: Data is shown for high resolution analysis with 15, 16, 17, and 18 members, low resolution analysis with 10, 15, and 20 members, mixed resolution analysis on 1 high resolution member with a low resolution ensemble of 10, 15, and 20, and mixed resolution analysis on a mixed ensemble of 30 low resolution members and 9, 12, and 15 high resolution members. The dashed black line represents the variation in the small scales and the dashed grey line corresponds to twice as much variation in the small scales.

can be more cost effective than increasing the size of the low resolution ensemble, as indicated by the star and square data points respectively in Figure 2.5. Comparison between the circle and star data points shows in Figure 2.5 shows that mixed resolution analysis provides an improvement over analysis at high resolution, in the sense that mixed resolution analysis provides greater accuracy for the same overall forecast computation time. Mixed resolution analysis with multiple high resolution members also improves upon results with only one high resolution member. In our small-scale resolving experiments, the mixed ensemble with only one high resolution member is unable to resolve the small scale variability in the truth and the uncorrelated small scale variability of the high resolution member increases the error.

In our experiments with model error, we found a greater advantage of mixed resolution analysis over single resolution analysis than in the corresponding perfect model experiment (Table 2.3). We also found that mixed resolution analysis reduces the temporal volatility of analysis errors (Table 2.4).

We remark that our technique could be easily extended to ensembles of 3 or more different resolution, albeit with the introduction of more parameters to tune. While we haven't fully explored the effects of tuning α or varying the size of the low resolution ensemble, the results given here give an indication of what one can expect in applying this method.

In conclusion, we found that this mixing technique is beneficial and cost-efficient in a simple model.

Time Averaged RMSE	Mixed Resolution Cases			High Resolution Cases		
	1 high +30 low	2 high +30 low	3 high +30 low	3 high	4 high	5 high
basic	0.90	0.79	0.76	0.97	0.84	0.78
$F = 14$	0.95	0.84	0.82	1.10	0.93	0.89
$F = 12.5$	1.06	1.00	0.98	1.53	1.28	1.18

Table 2.3: The time averaged RMSE over 4500 assimilation steps for various mixed and high resolution ensembles in various scenarios. The standard deviation of a 500 assimilation step moving average in most cases was between 0.01 and 0.03. The 4 member large model error case ($F = 12.5$) and all of the 3 member high resolution cases were not as precise, with a standard deviation of the moving average between 0.04 and 0.07.

Standard Deviation of Fluctuations	Mixed Resolution Cases			High Resolution Cases		
	1 high +30 low	2 high +30 low	3 high +30 low	3 high	4 high	5 high
basic	0.13	0.11	0.11	0.19	0.15	0.13
$F = 14$	0.14	0.12	0.11	0.24	0.15	0.13
$F = 12.5$	0.13	0.11	0.11	0.27	0.19	0.16

Table 2.4: Standard deviation of fluctuations in the RMSE with time, over the same 4500 assimilation steps used in Table 2.3.

Small-Scale Scenario	Mixed Resolution Cases			High Resolution Cases			
	9 high +30 low	12 high +30 low	15 high +30 low	15 high	16 high	17 high	18 high
RMSE	0.19	0.15	0.12	0.19	0.17	0.14	0.12
Fluctuations	0.04	0.03	0.02	0.05	0.05	0.05	0.05

Table 2.5: Time averaged RMSE and standard deviation of fluctuations for our small-scale resolving scenario. Mixed resolution RMSEs are accurate to ± 0.01 and high resolution RMSEs are accurate to ± 0.02 as measured by the standard deviation of a 500 assimilation step moving average.

Chapter 3

Ensemble Data Assimilation with an Adjusted Forecast Spread

3.1 Introduction.

Data assimilation determines initial conditions for weather forecasts from the information provided by atmospheric observations. Comparing a prior weather forecast to observations and taking into account the uncertainties of each, data assimilation attempts to provides an optimal estimate for the current state of the atmosphere. Modifying the forecast for an optimal fit to the observations is called the analysis phase of data assimilation. The analysis (optimal estimate) is then evolved during the forecast phase.

In weather prediction via data assimilation, the observation uncertainty is typically characterized as additive errors distributed according to a known Gaussian distribution whose covariance is constant in time. Forecast errors are often treated as Gaussian too, but the forecast error covariance changes in time and can be difficult to estimate. Ensemble data assimilation forecasts an ensemble to provide a statistical sampling of the forecast, estimating the forecast error covariance through sample statistics. Ensemble Kalman filters fit the ensemble mean to the observations and provide an ensemble of sub-optimal estimates to represent the remaining uncertainty (Burgers et al. 1998; Anderson and Anderson 1999; Bishop et al. 2001; Ott et al. 2004; Wang et al. 2004; Evensen 1994; Houtekamer and Mitchell 1998). Forecasting this analysis ensemble provides the sample forecast uncertainty for the next assimilation cycle.

The Kalman filter (Kalman 1960) is optimal for linear models with Gaussian errors. Extensions to nonlinear models, including ensemble Kalman filters, are suboptimal and can diverge even if there is no model error (Jazwinski 1970; Anderson and Anderson 1999). One reason is that model nonlinearity causes ensemble Kalman filters to underestimate the forecast error covariance relative to the uncertainty in the initial conditions (Whitaker and Hamill 2002). Insufficient ensemble covariance can also be caused by unquantified errors in the forecast and observation models. One common way to compensate for covariance underestimation is to inflate the forecast error covariance by a multiplicative factor larger than one during each analysis step (Anderson and Anderson 1999; Whitaker and Hamill 2002; Miyoshi 2011).

Typically the ensemble is evolved with spread representative of the approximate covariance as described above. However, the estimated covariance is only utilized during the analysis phase of data assimilation. In this chapter we investigate the potential advantages of evolving an ensemble whose spread is *not* commensurate with the approximate covariance by rescaling the ensemble perturbations by a factor η before the forecast step and rescaling them by $1/\eta$ after the forecast step. We call this technique forecast spread adjustment and we remark that this has no net effect with a linear model.

One reason we expect improvement by varying η is related to the advantage sometimes observed in ensemble prediction of the mean of an ensemble forecast versus the forecast of the ensemble mean. If the mean of the forecast from an unscaled ensemble ($\eta = 1$) is more accurate (on average) than the forecast of the mean (corresponding to $\eta \to 0$), then the mean of the forecast from an ensemble scaled by $\eta \neq 1$ may be better still.

This method of forecast spread adjustment allows us to relate many aspects of data assimilation which were previously considered independently.

- As we will later show, for a linear observation operator forecast spread adjustment has the same long-term effect on the analysis means as rescaling the observation error covariance (e.g., Stroud and Bengtsson 2007) by η^2, but it results in a different (and in our experiments, more appropriate) analysis covariance.

- Multiplicative covariance inflation, described above, is applied sometimes by rescaling the forecast ensemble perturbations and sometimes by rescaling the analysis ensemble perturbations (e.g., Bonavita et al. 2008, Kleist 2012); when using forecast spread adjustment in addition, both ensembles are rescaled by amounts that can be varied independently.

- The limit $\eta \to 0$ discussed above also corresponds to evolving the ensemble covariance according to the linear approximation about the mean, as in the Extended Kalman Filter (e.g., Jazwinski 1970, Evensen 1992).

In Section 3.2 we describe various data assimilation algorithms based on the Kalman filter, specifically the Extended Kalman Filter (which we abbreviate XKF) and the ensemble Kalman filters ETKF (Bishop et al. 2001; Wang et al. 2004) and LETKF (Hunt et al. 2007). In Section 3.3.1 we describe forecast spread adjustment. We compare forecast spread adjustment to alternative formulations that can produce the same analysis means, specifically: inflating the observation error covariance with the forecast error covariance estimated from the evolved (un-adjusted) forecast ensemble (Section 3.3.2) and inflating the analysis error covariance in addition to or instead of the background (forecast) error covariance (Section 3.3.3). The main difference between these approaches is that inflating the observation error covariance results in a larger background and analysis error covariance than forecast spread adjustment. We find (Section 3.4.3) that the analysis error covariance specified by forecast spread adjustment better reflects the actual analysis errors for our numerical experiments, in which the correct observation error covariance is known. Section 3.3.4 compares the Ensemble Transform Kalman Filter (ETKF) and the Extended Kalman Filter (XKF) and discusses how forecast spread adjustment can transition from the XKF to the ETKF and beyond. In Section 3.4.1 we describe the Lorenz (2005) Model

II used for our experiments, and in Section 3.4.2 we describe our experiments. In Section 3.4.3 we describe how we tuned the forecast spread adjustment parameter η, along with the multiplicative covariance inflation factor, in order to minimize analysis errors. We also discuss the possibility of adaptively inflating both parameters with techniques similar to Li et al. (2009). Section 3.4.4 depicts and discusses the results of our experiments. We discuss our conclusions and summarize our findings in Section 3.5.

3.2 Kalman Filters.

The Kalman filter is an optimal algorithm for data assimilation with a linear model with Gaussian errors. As with data assimilation in general, the Kalman filter can be described by the cycle of forecasting the previous analysis (optimal estimate) to get the background (forecast) and generating a new analysis by updating the background to more closely match the observations. Thus we describe the Kalman filter in terms of two phases: the forecast phase and the analysis phase. Most data assimilation algorithms for weather prediction (ensemble Kalman filters and variational methods) are approximations to the Kalman filter.

The analysis state according to the Kalman filter with no model error assumed is given by

$$x^a = x^b + K\left(y^o - Hx^b\right) \tag{3.1}$$

$$K = P^b H^\top \left(H P^b H^\top + R^o\right)^{-1} \tag{3.2}$$

where K is the Kalman gain matrix, x^a is the analysis state, x^b is the forecast or background state, P^b is the background error covariance, R^o is the observation error covariance, y^o is the vector of observations, and H is matrix that transforms model space into observation space. The error covariance of the analysis state x^a is given by

$$P^a = (I - KH)\, P^b. \tag{3.3}$$

Having determined x^a and P^a at a certain time t, the forecast phase evolves these to the background state x^{b^+} and its error covariance P^{b^+} at time $t^+ > t.$ Specifically, in the forecast phase from time t to time t^+, the model M evolves the analysis state at time t to the background state at time t^+. The forecast phase is given by

$$x^{b^+} = M(x^a) \tag{3.4}$$

$$P^{b^+} = M P^a M^\top. \tag{3.5}$$

These Kalman filter equations (3.1)-(3.5) are only optimal for perfect model scenarios with linear models. Model error reduces the accuracy of the background mean beyond that accounted for by the evolved background error covariance. Moreover, even without model error, a nonlinear model tends to cause the ensemble to underestimate the background error covariance (Whitaker and Hamill 2002). One way to partially compensate for this underestimation of the background error covariance is to inflate P^b. Ensemble Kalman filters typically inflate P^b with a multiplicative covariance inflation factor ρ, replacing P^b in (3.1)-(3.3) with $P^b_{\text{inflated}} = \rho P^b_{\text{initial}}$ (Anderson and Anderson 1999; Whitaker and Hamill 2002). When $\rho = 1$ no inflation is performed.

3.2.1 Ensemble Kalman Filters.

In this section we discuss the class of data assimilation techniques known as the ensemble Kalman filters (EnKFs). In the ensemble techniques, rather than evolving the covariance directly as done in the Kalman filter and the Extended Kalman Filter (XKF), an ensemble $\left\{\mathbb{X}_i^b — 1 \leq i \leq k\right\}$ is chosen to represent a statistical sampling of a Gaussian background distribution, with the sample mean $\bar{x}^b$ of the background ensemble approximating the background state x^b in equation (3.1). The subscript i here refers to the index of the ensemble member, and all members of the ensemble estimate the background at the same time.

Each ensemble member $\mathbb{X}_i^b$ can be expressed as a sum $\mathbb{X}_i^b = \bar{x}^b + X_i^b$ of the mean and a perturbation X_i^b. Letting $\mathbb{X}^b$ and X^b be the matrices with columns $\mathbb{X}_i^b$ and X_i^b respectively, we write

$$\mathbb{X}^b = \bar{x}^b + X^b, \tag{3.6}$$

where by adding a vector and a matrix we mean adding the vector to each column of the matrix. The columns of X^b represent the ensemble of background perturbations.

The background error covariance is computed from the background ensemble according to sample statistics, namely

$$P^b = \frac{\rho}{k-1} X^b X^{b\top}, \tag{3.7}$$

where k is the size of the ensemble and ρ is the multiplicative covariance inflation factor.

We use similar notation for the analysis ensemble $\mathbb{X}^a$ with mean $\bar{x}^a$ and perturbations X^a. Ensemble Kalman filters are designed to satisfy the Kalman filter equations (3.1)-(3.5) with x^b and x^a replaced by $\bar{x}^b$ and $\bar{x}^a$. Thus, the analysis ensemble perturbations must correspond a square root of the reduced rank analysis error covariance matrix in equation (3.3), i.e.,

$$\frac{1}{k-1} X^a X^{a\top} = P^a. \tag{3.8}$$

Different ensemble Kalman filters select different solutions X^a to this equation. In all ensemble Kalman filters, the ensemble members are forecast independently. For a deterministic model M representing a forecast from time t to time t^+, the background ensemble members at time t^+ are given by

$$\mathbb{X}_i^{b+} = M\left(\mathbb{X}_i^a\right). \tag{3.9}$$

3.2.1.1 ETKF

We tested ensemble data assimilation with forecast spread adjustment on the Ensemble Transform Kalman Filter (ETKF) (Bishop et al. 2001; Wang et al. 2004) and the Local Ensemble Transform Kalman Filter (LETKF) (Ott et al. 2004; Hunt et al. 2007). We implemented both methods according to the formulation in Hunt et al. (2007). For speedy computation, in these algorithms the analysis phase of the Kalman filter is performed in the space of the observations, with the background ensemble transformed into that space for comparison.

In the ETKF, the analysis mean is given by

$$\bar{x}^a = \bar{x}^b + \tfrac{1}{k-1} X^b U {Y^b}^\top {R^o}^{-1} \left(y^o - h(x^b)\right) \tag{3.10}$$

$$U = \left(\tfrac{1}{\rho} I + \tfrac{1}{k-1} {Y^b}^\top {R^o}^{-1} Y^b\right)^{-1}, ^1 \tag{3.11}$$

where h is the (possibly nonlinear) observation operator, and Y^b are the background per-

[1] The U in equation (3.11) is different from the notation in Chapter 2; $U_{\text{Ch3}} = M^a_{\text{Ch2}}$.

turbations in observation space derived by transferring the ensemble $\mathbb{X}^b$ to observation space and subtracting the mean $\bar{y}^b = \overline{h\left(\mathbb{X}^b\right)}$, $Y^b = h\left(\mathbb{X}^b\right) - \bar{y}^b$. For a linear observation operator, the ETKF analysis mean $\bar{x}^a$ computed in equation (3.10) equals the Kalman filter analysis estimate x^a of equation (3.1) assuming $x^b = \bar{x}^b$ and $P^b = \frac{\rho}{k-1} X^b {X^b}^\top$ (equation (3.7)).

The ETKF analysis perturbations are given by

$$X^a = X^b U^{\frac{1}{2}}, \tag{3.12}$$

where the exponent $\frac{1}{2}$ represents taking the symmetric square root. These perturbations differ from the background perturbations as little as possible while still remaining a square root of P^a (Ott et al. 2004). For a full discussion of the equivalence and relationship between (3.10)-(3.12) and (3.1)-(3.3) see Hunt et al. (2007).

3.2.1.2 LETKF

Essentially, the Local Ensemble Transform Kalman Filter (LETKF) performs an ETKF analysis at every model grid point, excluding distant observations from the analysis. At each grid point, it computes y^o_{local} and R^o_{local}, truncating y^o and R^o to include only the observations within a certain distance from the grid point. The LETKF computes the local analysis for each grid point from the ETKF analysis equations (3.10)-(3.12) and restrict the resulting analysis to that grid point alone. Observation localization is motivated and explored in (Houtekamer and Mitchell 1998; Ott et al. 2004; Hunt et al. 2007; Greybush et al. 2011). We provide a limited motivation here.

One problem with estimating P^b (equation (3.7)) from an ensemble is that sampling error can introduce artificial correlations between the background error at distant grid points. As the analysis increment $\bar{x}^a - \bar{x}^b$ in equation (3.10) is computed in observation space, it is possible to eliminate these spurious correlations (along with any real correlations) for the analysis at a particular model grid point by truncating the observation space beyond a specified distance from the grid point.

Other methods of localization (e.g., Whitaker and Hamill 2002, Bishop and Hodyss 2009) modify the background error covariance directly. For all these approaches, localization is found to greatly reduce the number of ensemble members needed to produce reasonable analyses for models with large spatial domains.

3.2.2 Extended Kalman Filter

In the Extended Kalman Filter (XKF) (see e.g., Jazwinski 1970), the model evolves the analysis x^a as in equation 3.4 to determine the subsequent background x^{b^+}, and the background error covariance P^{b^+} is determined from P^a by linearizing the model at x^a. For more information on the XKF and how it compares to ensemble Kalman filters, see

Evensen (2003) and Kalnay (2003).

Like the Kalman filter, the XKF is usually formulated with an additive model error term that increases the background covariance. However, for ease of comparison to the ETKF, our implementation of the XKF uses multiplicative inflation instead of additive inflation.

One major difficulty with the Extended Kalman Filter is the computational cost of evolving the covariance matrix (Evensen 1994; Burgers et al. 1998; Kalnay 2003). This makes the model unfeasible for high-dimensional models in a realistic setting. However, when using simple test models such as the Lorenz (2005) Model II, other data assimilation techniques can be compared to the XKF.

Another potential drawback of the XKF is its linear approximation when evolving the covariance (Evensen 2003). As we find in our experiments, this can be disadvantageous compared to the nonlinear evolution of both mean and covariance provided by an ensemble.

3.3 Methodology and Theory.

In this section we formally describe the method of forecast spread adjustment for ensemble Kalman filters. One way of implementing the forecast spread adjustment is to multiply the analysis perturbations by η prior to forecasting and, after the forecast, multiply the forecast perturbations by $\frac{1}{\eta}$ (Section 3.3.1). This is only beneficial when the model is nonlinear. Other authors (e.g., Stroud and Bengtsson 2007) have found benefits from multiplicative observation error covariance inflation, motivated by observation errors other than measurement error. While forecast spread adjustment is equivalent to multiplicative observation error covariance inflation by η^2 for an appropriately initialized ensemble (Section 3.3.2), our experiments are performed with observations lying on the model grid points and perfect knowledge of the exact observation error covariance, so inflating the observation error covariance would be mathematically unmotivated. However as adjusting the forecast spread by η and $\eta^2 R^o$ inflation have similar accuracies after spin-up, perhaps some of the benefits of inflating R^o stem from the forecast spread adjustment effect.

3.3.1 Ensemble data assimilation with forecast spread adjustment.

This section describes the technique of shrinking (or expanding if $\eta > 1$) the ensemble perturbations for the forecast and expanding (or shrinking) the ensemble for the analysis. After [1]finding the analysis through any ensemble data assimilation techniques, the background perturbations are obtained by: [2]multiplying the analysis perturbations by η, [3]evolving the adjusted analysis, and [4]multiplying the forecast perturbations by $\frac{1}{\eta}$.

Mathematically, this can be expressed with a matrix transformation S corresponding to an expansion factor of η:

$$\mathbb{X}^{b+} = M(\mathbb{X}^a S)\, S^{-1} \tag{3.13}$$

where

$$S = \eta I_{k\times k} + (1-\eta)\left(\tfrac{1}{k}\mathbf{1}_{k\times k}\right) \tag{3.14}$$

$$\mathbf{1}_{k\times k} = \begin{bmatrix} 1 & \cdots & 1 \\ \vdots & \ddots & \vdots \\ 1 & \cdots & 1 \end{bmatrix}. \tag{3.15}$$

The model integration of the matrix $\mathbb{X}^a S$ in equation (3.13) refers to forecasting each ensemble member (column of $\mathbb{X}^a S$) separately, i.e.

$$M\left(\mathbb{X}^a S\right) := \begin{bmatrix} M(\{\mathbb{X}^a S\}_1)\,, & \cdots\,, & M(\{\mathbb{X}^a S\}_k) \end{bmatrix}. \tag{3.16}$$

For an ensemble of size k, each column of the $[k \times k]$ matrix $\mathbf{1}_{k\times k}$ right transforms an ensemble into its row sum so that $\frac{1}{k}\mathbb{X}\mathbf{1}_{k\times k} = \begin{bmatrix} \bar{x} & \ldots & \bar{x} \end{bmatrix}$. Thus right multiplication by S transforms each ensemble member $\mathbb{X}^a_i$ into a weighted average of itself (prior to adjustment) and the mean of the ensemble

$$\{\mathbb{X}^a S\}_i = \eta \bar{x}^a + (1-\eta)\,\mathbb{X}^a_i. \tag{3.17}$$

Viewed another way, right multiplication by S scales the ensemble perturbations by η, producing a new ensemble with the same mean

$$\{\mathbb{X}^a S\}_i = \bar{x}^a + \eta X_i^a. \tag{3.18}$$

Notice that

$$S^{-1} = \frac{1}{\eta} I_{k\times k} + \left(1 - \frac{1}{\eta}\right) \left(\tfrac{1}{k}\mathbf{1}_{k\times k}\right), \tag{3.19}$$

so S^{-1} effectively rescales ensemble perturbations by η^{-1}. We call a forecast contraction and analysis re-expansion *forecast spread adjustment*.

The full Kalman filter cycle from $\mathbb{X}^b$ to $\mathbb{X}^{b+}$ with the forecast phase described by equation (3.13) is given in the following algorithm.

1. Perform analysis on $\mathbb{X}^b$ to find $\mathbb{X}^a$.

2. Left multiply $\mathbb{X}^a$ by S to scale the perturbations X^a by a factor of η, $\left(\text{i.e. } X^{a,f} = \eta X^a\right)$

$$\mathbb{X}^{a,f} = \mathbb{X}^a S. \tag{3.20}$$

3. Evolve each member of the resulting ensemble:

$$\mathbb{X}_i^{b+,f} = M\left(\mathbb{X}_i^{a,f}\right). \tag{3.21}$$

4. Left multiply $\mathbb{X}^{b+,f}$ by S^{-1} to rescale the perturbations of the evolved ensemble by $\frac{1}{\eta}$:

$$\mathbb{X}^{b+} = \mathbb{X}^{b+,f} S^{-1}. \tag{3.22}$$

Here and elsewhere $\mathbb{X}^{b,f}$ and $\mathbb{X}^{a,f}$ refer to the ensemble during the forecast phase. We abbreviate the ETKF with forecast spread adjustment and the LETKF with forecast spread adjustment (via a scaling factor of η) as the ηETKF and the ηLETKF respectively.

We remark that when $\eta = 1$, ensemble data assimilation with forecast spread adjustment reduces to standard ensemble data assimilation. Furthermore, if the model is linear, forecast spread adjustment has no net effect. When the model is nonlinear, forecast spread adjustment could affect the background mean, the background perturbations, and (minimally) the analysis phase background spread. We expect that the effect on the mean will be the most significant.

3.3.2 Relation to inflation of the observation error covariance.

Some authors have recommended scaling the reported observation error covariance, assuming that it is misrepresented (e.g., Stroud and Bengtsson 2007, Li et al. 2009). As we will show below, inflating the observation error covariance is closely related to forecast spread adjustment, and thus in many cases both approaches can improve the analyses. Indeed, if the observation operator h is linear then we will show that both approaches, if initialized with the same forecast ensemble, will produce the same analysis means. Having the same forecast ensembles implies that during the analysis phase, the two approaches will use different ensemble perturbations (related by a factor of η) and hence different covariances.

3.3.2.1 ηETKF without R^o inflation.

Let $\mathbb{X}^b$ be the initial background ensemble. In this case, we perform ensemble data assimilation with forecast spread adjustment to find $\mathbb{X}^{b^+}$. The ηETKF analysis ensemble $\mathbb{X}^a = \bar{x}^a + X^a$ is given by equations (3.10)-(3.12) repeated here for convenience

$$\bar{x}^a = \bar{x}^b + \tfrac{1}{k-1} X^b U Y^b {R^o}^{-1} \left(y^o - \bar{y}^b \right) \tag{3.23}$$

$$U = \left(\tfrac{1}{\rho} I + \tfrac{1}{k-1} {Y^b}^\top {R^o}^{-1} Y^b \right)^{-1} \tag{3.24}$$

$$X^a = X^b U^{\frac{1}{2}}. \tag{3.25}$$

The next background ensemble $\mathbb{X}^{b^+}$is given by equation (3.13)

$$\mathbb{X}^{b^+} = M(\, \mathbb{X}^a S \,) \, S^{-1}. \tag{3.26}$$

3.3.2.2 ETKF with R^o inflated to $\eta^2 R^o$.

Let the initial background ensemble be $\mathbb{X}^{b_2}$ and assume that $\mathbb{X}^{b_2} = \mathbb{X}^b S$. Along with $\mathbb{X}^{b,f} := \mathbb{X}^b S$ defined based on equations (3.20) and (3.22), this implies the additional equivalences $\bar{x}^{b_2} = \bar{x}^b$ and $X^{b_2} = X^{b,f} = \eta X^b$. We perform standard ensemble data assimilation to find $\mathbb{X}^{b_2^+}$ assuming an observation error covariance of η^2 times its value in Case 3.3.2.1. We compare $\mathbb{X}^{b_2^+}$ with $\mathbb{X}^{b+}$and $\mathbb{X}^{b+,f} = \mathbb{X}^{b+} S$.

In this case (3.3.2.2), the analysis ensemble is given by equations (3.10)-(3.12), with $R^{o_2} = \eta^2 R^o$ replacing R^o. After algebraic manipulation, these equations are

$$U_2 = \left(\frac{1}{\rho} I + \frac{1}{k-1} \eta Y^{b_2^\top} R^{o^{-1}} \eta Y^{b_2}\right)^{-1} \tag{3.27}$$

$$\bar{x}^{a_2} = \bar{x}^{b_2} + \frac{1}{k-1} \eta X^{b_2} U_2 \eta Y^{b_2} R^{o^{-1}} \left(y^o - \bar{y}^{b_2}\right) \tag{3.28}$$

$$X^{a_2} = X^{b_2} U_2^{\frac{1}{2}}. \tag{3.29}$$

Substituting $X^{b_2} = \eta X^b$ and assuming a linear observation operator so that $Y^{b_2} = \eta Y^b$, we find that

$$U_2 = U \tag{3.30}$$

$$\bar{x}^{a_2} = \bar{x}^a \tag{3.31}$$

$$X^{a_2} = \eta X^a. \tag{3.32}$$

Thus $\mathbb{X}^{a_2} = \mathbb{X}^a S$. As we define $\mathbb{X}^{a,f} := \mathbb{X}^a S$, the ensembles are identical during evolution.

The next background ensemble is given by

$$\begin{aligned} \mathbb{X}^{b_2^+} &= M(\mathbb{X}^{a_2}) \\ &= M(\mathbb{X}^a S)\,. \end{aligned} \tag{3.33}$$

Comparing this to equation (3.26), we see that $\mathbb{X}^{b_2^+} = \mathbb{X}^{b+} S =: \mathbb{X}^{b+,f}$; therefore $\bar{x}^{b_2^+} = \bar{x}^{b+}$. In other words, the means remain the same and the perturbations maintain the same ratio.

We remark that although both cases evolve the same ensemble, the error covariances in the two cases differs by the factor η^2.This is demonstrated in Figure 3.1.

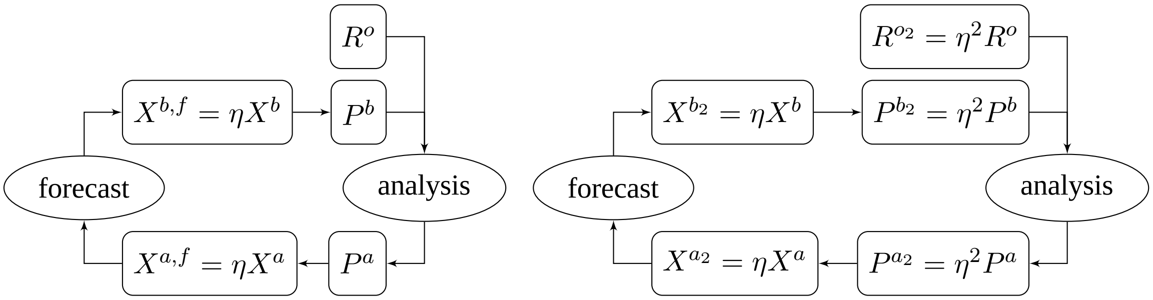

Figure 3.1: *Left*: data assimilation with a forecast spread adjustment of η. *Right*: standard data assimilation with R^o inflated by η^2. The two data assimilation cycles depicted above are initialized during the forecast phase, i.e. $X^{b_2} = X^{b,f}$.

3.3.3 Which to inflate, P^a or P^b?

As discussed in the beginning of Section 3.2, due to nonlinearity and model error ensemble data assimilation tends to underestimate the evolved error covariance. This is typically corrected by inflating the background error covariance prior to analysis, e.g. Anderson and Anderson (1999), Whitaker and Hamill (2002). However it would be just as theoretically valid to inflate the (posterior) analysis error covariance, e.g. Bonavita et al. (2008). With forecast spread adjustment, it no longer becomes a question of inflating one or the other; instead forecast spread adjustment provides a continuum connecting P^b multiplicative inflation and P^a multiplicative inflation.

Consider the standard data assimilation cycle as it applies to P^a and P^b with multiplicative analysis and background covariance inflation of ρ_a and ρ_b respectively (Figure 3.2 *Left*). Following the algorithms given in Hunt et al. (2007), in our implementation of the ηETKF, the ηLETKF, we only inflate the background error covariance (Figure 3.2 *Right*). By setting $\eta = \sqrt{\rho}$, we transform a data assimilation algorithm that inflates the

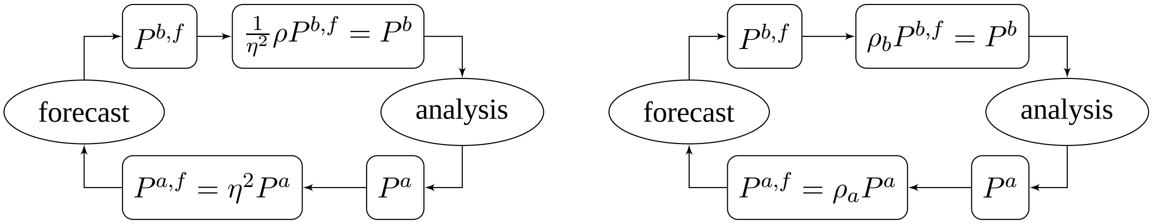

Figure 3.2: *Left*: the data assimilation cycle on P^b and P^a including a forecast spread adjustment of η and inflation of P^b with ρ. *Right*: the data assimilation cycle on P^b and P^a including inflation of P^b with ρ_b and inflation of P^a with ρ_a.

background error covariance P^b into a data assimilation algorithm that inflates the analysis error covariance P^a. Furthermore, this comparison demonstrates that ensemble data assimilation with an adjusted forecast spread and with multiplicative background error covariance inflation can be alternatively interpreted as standard data assimilation, inflating both the background and analysis error covariances. In particular, adjusting the forecast spread by η and inflating the background error covariance by ρ is equivalent to inflating the background error covariance by $\rho_b = \rho/\eta^2$ and inflating the analysis error covariance by $\rho_a = \eta^2$.

Those who have found more accurate results when inflating P^a in preference to P^b likely will benefit from forecast spread adjustment with $\eta > 1$. Similarly, if one has previously obtained more accurate results with P^b inflation in preference to P^a inflation, forecast spread adjustment with $\eta < 1$ might prove beneficial.

3.3.4 Comparison to the XKF.

The Ensemble Transform Kalman Filter (ETKF) (Bishop et al. 2001; Wang et al. 2004) and the Extended Kalman Filter (XKF) (Jazwinski 1970; Evensen 1992) are similar in that they both evolve the covariance matrix in time. The XKF evolves the best estimate along with its error covariance. The ETKF evolve a surrounding ensemble which approximates the error covariance and does not explicitly evolve the best estimate, which is the mean of the ensemble.

Previous articles, e.g. Burgers et al. (1998), demonstrate the equivalence between the analysis phases of the Extended Kalman Filter and the ensemble Kalman filter (EnKF), provided the ensemble is sufficiently large. However, this equivalence does not continue into the forecast phase. Starting from the same analysis state $x^a_{\text{XKF}} = \bar{x}^a$ and the same analysis error covariance $P^a_{\text{XKF}} = \frac{1}{k-1} X^a {X^a}^\top$, the XKF and the EnKF will evolve different states $x^{b+}_{\text{XKF}} \neq \bar{x}^{b+}$ with different error covariances $P^{b+}_{\text{XKF}} \neq \frac{1}{k-1} X^{b+} {X^{b+}}^\top$.

Varying the ensemble spread changes the evolved ensemble in a nonlinear fashion, affecting the mean and perturbations (in both magnitude and direction). In the limit as $\eta \to 0$, the ensemble perturbations become infinitesimal and the forecast evolves the ensemble mean and the perturbations evolve according to the tangent linear model. If the ensemble is large enough that the perturbations span the model space, then a full-rank covariance is evolved according to the tangent linear model, just as in the XKF. Therefore for a sufficiently large ensemble, the ηETKF should approach the XKF as η tends to zero; otherwise, it approaches a reduced-rank XKF. Furthermore, when $\eta = 1$, the ηETKF is the standard ensemble transform Kalman filter. Thus for intermediate values of η, the ηETKF can be considered a hybrid of the XKF and the ETKF.

In the scenarios we explored where tuning η was especially effective, η tuned to values greater than one, enhancing the advantages of the ensemble Kalman filter over the

Extended Kalman Filter.

3.4 Experiments and Results

Our experiments are observed system simulation experiments (OSSEs). In OSSEs, we evolve a truth with the model, adding error at a limited number of points to simulate observations. Treating the truth as unknown, we use data assimilation to estimate it. Comparing the analysis to the truth, we estimate the accuracy of the data assimilation techniques being tested. Thus we can evaluate and compare the accuracy of data assimilation techniques. We assessed accuracy via the root mean square error (RMSE) difference between the analysis mean and the truth evaluated at every grid point.

3.4.1 Models

We tested forecast spread adjustment on the Lorenz (2005) Model II with a smoothing parameter of $K = 2$ (smoothing the Lorenz 96 model (Lorenz 1996; Lorenz and Emanuel 1998) over twice as many grid points). For a forcing constant F and smoothing parameter K, the model at grid point j is evolved according to

$$\frac{dx_n}{dt} = \left(x_{n+K} x^{\text{avg}}_{n-K}\right)^{\text{avg}} - x^{\text{avg}}_{n-2K} x^{\text{avg}}_{n-K} - x_n + F, \tag{3.34}$$

where x_n^{avg} refers to averaging x_n the model state at grid point n with the model state at nearby grid points.

When $K = 2$, the averaging includes only the grid point itself and the immediately adjacent grid points namely

$$x_n^{\text{avg}} = \frac{1}{4}x_{n-1} + \frac{1}{2}x_n + \frac{1}{4}x_{n+1}. \tag{3.35}$$

Note that the avg function is applied to $(x_j x_i^{\text{avg}})$ by
$(x_n x_m^{\text{avg}})^{\text{avg}} = \frac{1}{4}x_{n-1}x_{m-1}^{\text{avg}} + \frac{1}{2}x_n x_m^{\text{avg}} + \frac{1}{4}x_{n+1}x_{m+1}^{\text{avg}}$.

Model II is evolved on a circular grid, in our experiments a circular grid with 60 grid points. We evolved the forecasts according to the Runge-Kutta fourth order method over a time step size of $\delta t = 0.05$, performing an analysis every time step.

To generate the observations, we evolved the truth with a forcing constant of $F^t = 12$ and added independent Gaussian errors with mean 0 and standard deviation σ at every other grid point. Thus at each time step, we have 30 evenly spaced observations with known error covariance $R^o = \sigma^2 I_{30\times30}$. Figure 3.3 gives a snapshot of these observations for $\sigma = 1$ along with the truth.

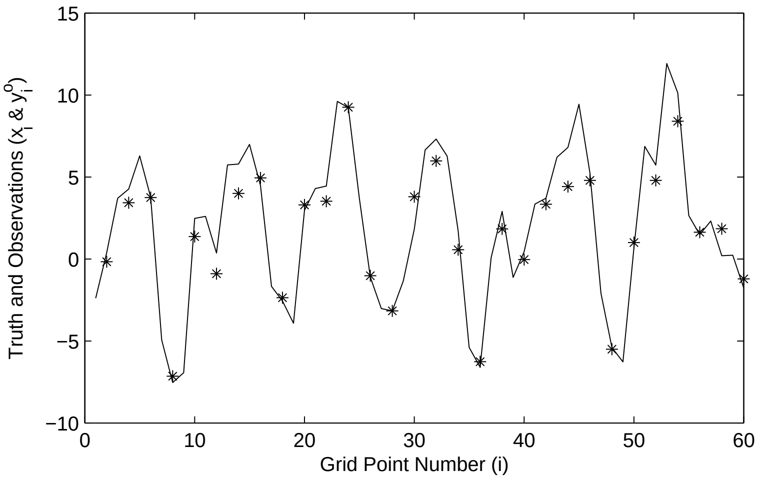

Figure 3.3: Sample Model II output with observations for 60 grid points, half of which are observed (*), observation error $\sigma = 1$, smoothing parameter $K = 2$, and forcing constant $F = 12$.

3.4.2 Experimental Design

We explored the effectiveness of tuning η in various scenarios with the LETKF. Our default parameters for data assimilation are: an observation error $\sigma = 1$, an ensemble of $k = 10$ members, and a forcing constant of $F = 14$ for the ensemble forecast to simulate model error. In all cases we use a localization radius of 3 grid points, i.e. 3 to 4 observations in a local region.

Keeping the other two parameters constant at their default value, we varied the observation error, the ensemble size, and the forcing constant. For instance, we varied $\sigma = \frac{1}{2}$, 1, 2, keeping $k = 10$ and $F = 14$. We tested ensemble sizes of 5, 10, 20, and 40.

We varied the forcing constant from $F = 9$ to $F = 15$.

In each experiment we tuned ρ for $\eta = 1$ and plotted the RMSE for different values of η:

$$\text{RMSE} = \tfrac{1}{4500} \sum_{i=501}^{5000} \text{rms}\left(\bar{x}^a(t_i) - x^t(t_i)\right), \tag{3.36}$$

where $x^t(t_i)$ is the truth at time t_i, $\bar{x}^a(t_i)$ is the analysis ensemble mean at time t_i, and rms is the root mean square function over the 60 grid points, i.e. the RMSE is the time averaged root mean square difference between the analysis mean and the truth. In each case we compared the minimum RMSE among the various values of η tested to the RMSE when $\eta = 1$ and plotted the percent improvement

$$\text{\% Imprv.} = \frac{\text{RMSE}(\eta = 1) - \min\limits_{0<\eta<\infty} \text{RMSE}(\eta)}{\text{RMSE}(\eta = 1)}. \tag{3.37}$$

We also tested how the ηETKF compares to the XKF for large ensembles and small η. Specifically, we tested the ηETKF for $\eta = 1$, $\frac{1}{2}$, $\frac{1}{4}$, $\frac{1}{8}$, and 10^{-6} and ensembles of size $k = 40$, 60, and 80. For $\eta = 10^{-6}$ and $k = 80$ we expect the RMSE from the ηETKF to be very similar to the RMSE from the XKF.

3.4.3 Tuning η and ρ

Since the two cases are identical in the linear setting, forecast spread adjustment could be advantageous in certain nonlinear settings. However, the best value of η is not specified mathematically. Hence we tune η to get the lowest RMSE during data assimilation. Similarly, we also tune the multiplicative covariance inflation factor ρ. Figure 3.4 depicts the ensemble spread both before (red) and after (blue) adjustment as a function of η.

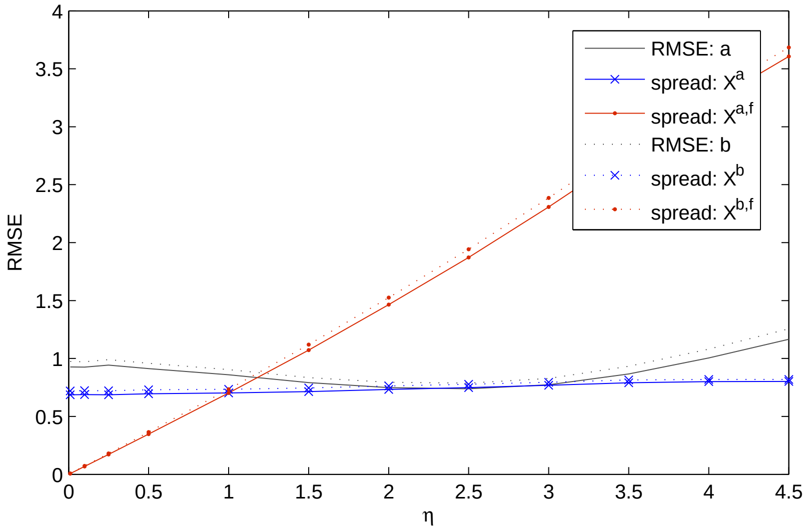

Figure 3.4: Tuning η (grey) and the effect of η on ensemble spread during the forecast phase (red) and analysis phase (blue) for our default parameters: $k = 10$, $F = 14$ ($F^t = 12$), $\sigma = 1$. Comparison of the ensemble spread during the analysis phase (blue) to the actual RMSE for various values of η (grey). The solid grey curve shows that η tunes to 2.5. We tuned η using the tuned value of ρ for $\eta = 1$.

We define the ensemble spread by

$$\sqrt{\frac{1}{N}\text{trace}\left(\frac{1}{k-1}XX^{\top}\right)}, \tag{3.38}$$

where $N = 60$ is the length of the circular grid and $\frac{1}{k-1}XX^{\top}$ is an NxN matrix approximating the error covariance as described in equations (3.7) and (3.8). When $\eta = 1$, the ensemble spread (equation (3.38)) is intended to estimate the actual RMSE (equation (3.36)). As demonstrated in Figure 3.4, when $\eta \neq 1$, the ensemble spread only estimates the actual RMSE during the analysis phase. In other words, the forecast spread adjustment parameter η only strongly affects the ensemble spread during the forecast phase (red). During the analysis phase (blue), the ensemble spread does not vary significantly with η. The minimum of the solid grey curve corresponds to the tuned value of η, which for our default parameters is $\eta = 2.5$.

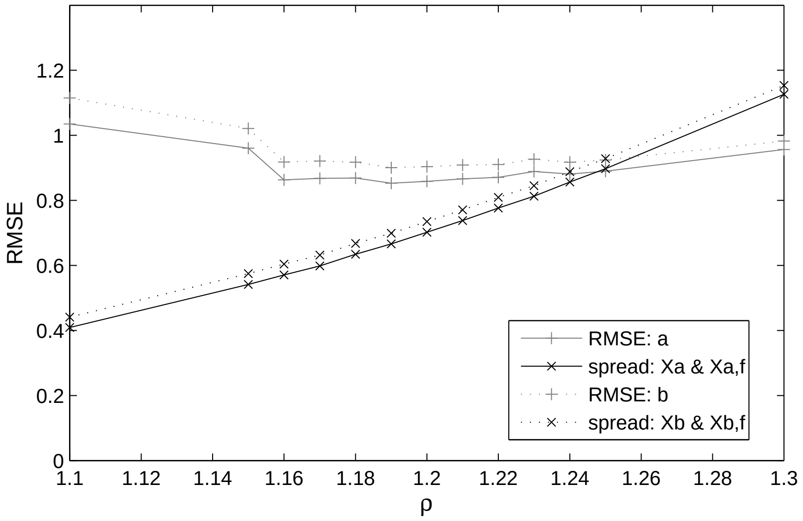

Figure 3.5: Tuning ρ and the effect of ρ on ensemble spread for our default parameters with $\eta = 1$. The dotted lines correspond to the background and the solid lines correspond to the analysis. The spread is indicated in black and the RMSE in grey.

Recall (Section 3.3.2) that with observation error covariance inflation, the analysis perturbations X^{a_2} are the same as the perturbations $X^{a,f}$ for the forecast spread adjustment. Thus, in this scenario, when η is tuned to a value different than one, the ensemble spread X^a for forecast spread adjustment corresponds much better to the actual analysis errors than the spread X^{a_2} for observation error covariance inflation.

Figure 3.5 (black) demonstrates that multiplicative background error covariance inflation by ρ (Anderson and Anderson 1999; Whitaker and Hamill 2002) carries through to both phases of data assimilation. We remark that in this example ρ tunes to 1.20 (grey), even though the ensemble spread underestimates the RMSE. However, in this case (Figure

3.4) we find that tuning η as well as ρ decreases the error to be commensurate with the ensemble spread. To tune ρ in our LETKF experiments we let $\eta = 1$ and tested values of ρ differing by 0.01, choosing the value of ρ associated with the lowest analysis RMSE. When comparing the ηETKF to the XKF, we tuned $\rho = 1.29$ for our XKF experiment and used that ρ for our ηETKF experiments as well.

In our experiments we determined the optimal values of ρ and η through tuning. However, there are methods that adaptively determine the values of ρ (Anderson 2007; Li et al. 2009; Miyoshi 2011). In particular, Li et al. (2009) simultaneously estimate values for a background error covariance factor and for an observation error covariance factor. When determining the analysis mean, these two parameters are equivalent in some sense to ρ and η (see Section 3.3.2). Thus, the adaptive techniques explored by Li et al. (2009) could potentially be applied to simultaneous adaptive estimation of η and ρ. On the other hand, some adjustment may be necessary if η is mainly compensating for the nonlinearity and model bias as opposed to a misspecification of the observation error covariance.

3.4.4 Results

Our results are given graphically as both η versus RMSE; as well as k (ensemble size), F (forcing constant), and σ (observation error) versus percent improvement in the RMSE. The RMSEs are generally accurate out to the hundredths place (± 0.01); the RM-

SEs from five independent trials with our default parameters and $\eta = 1$ are 0.86, 0.86, 0.87, 0.88, and 0.88. When $k = 5$, the RMSEs with $\eta = 1$ and with the tuned value of $\eta = 3.5$ are less precise with accuracy approximately ± 0.04. In all of our other experiments (LETKF varying k, F, or σ, and ηETKF vs. XKF) the RMSEs are accurate to about ± 0.01.

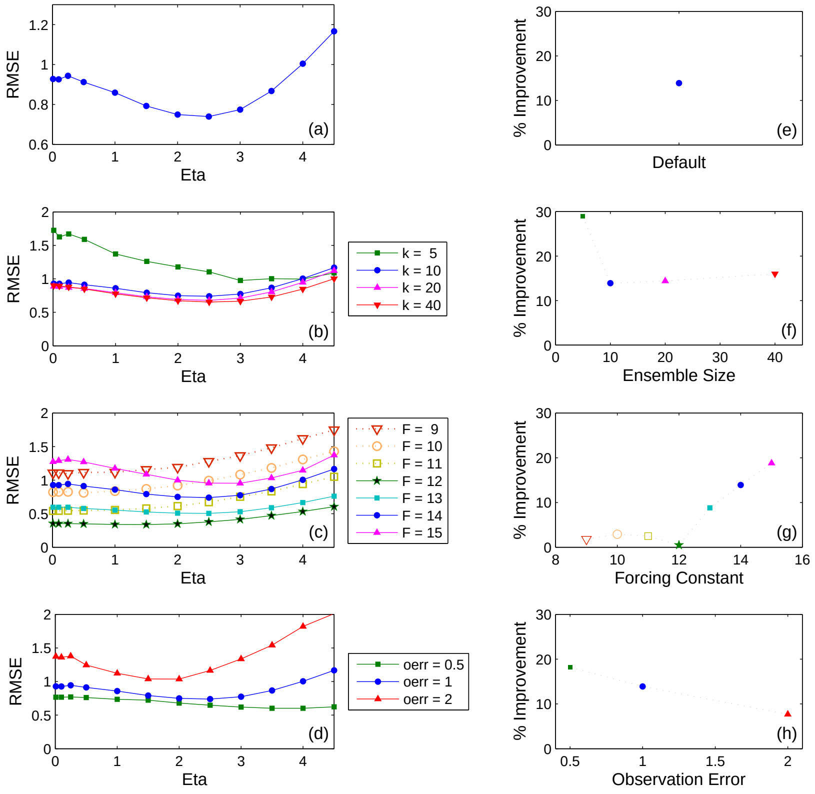

Figure 3.6: *Left*: The RMSE as a function of η for (a) our default parameters: an ensemble with $k = 10$ members, model error due to the forcing constant $F = 14$ where the true forcing is $F^t = 12$, and observation error of 1; (b) various ensemble sizes: $k = 5, 10, 20,$ and 40 members; (c) different amounts of model error: $F = 9, 10, 11, 12, 13, 14,$ and 15 with $F^t = 12$; (d) different amounts of observation error: $\sigma = 0.5, 1,$ and 2. In each of (b), (c) and (d), one of the parameters from (a) ($k = 10$, $F = 14$, $\sigma = 1$) is varied, keeping the other two parameters at their default values; the curve graphed in (a) appears in (b), (c) and (d) as well. We restrict the range of the y-axis in (a) to better show the structure of the curve. *Right*: Comparing the RMSE when $\eta = 1$ to the minimum RMSE when η is allowed to vary. Subplot (e) shows the percent improvement in the RMSE when varying η with our default parameters, i.e. setting $\eta = 2.5$ gives results with an RMSE that is 14% smaller than when $\eta = 1$. Also shown are the percent improvements versus (f) various ensemble sizes, (g) different forcing constants, and (h) different amounts of observation error. Each right hand figure (e), (f), (g), and (h) is derived from the same data as the figure immediately to its left, i.e. (a), (b), (c), and (d) respectively.

3.4.4.1 Default Parameters

When $k = 10$, $F = 14$ ($F^t = 12$), and $\sigma = 1$, we obtain the most accurate results when $\eta = 2.5$, providing a 14% improvement over the RMSE when $\eta = 1$ (0.74 vs. 0.87). Although we generally did not retune ρ after tuning η, we remark than doing so further improves the RMSE; setting $\eta = 2.5$ and $\rho = 1.16$ improves the RMSE by 19% over $\eta = 1$ and $\rho = 1.20$.

Figure 3.6(a) shows the RMSE for various values of η assuming the default parameters. The curve is minimized when $\eta = 2.5$. Figures 3.6(b), 3.6(c), and 3.6(d) show the effect of varying one of the default parameters (k, F, σ), while keeping the other two constant. Thus the curve in 3.6(a) also appears in 3.6(b), 3.6(c), and 3.6(d).

3.4.4.2 Varying the Ensemble Size

Figures 3.6(b) and 3.6(f) shows how varying the ensemble size, k, affects the RMSE and the tuned value of η. We find that forecast spread adjustment improves results more with smaller ensembles than with large ensembles. We expect that this is because increasing the ensemble size generally improves the accuracy of the results, so small ensembles have more room to improve their accuracy. For ensembles with $k = 10$ or more members, the RMSE improvement due to tuning η levels out at around 15%.

3.4.4.3 Varying the Model Error

We simulated model error by forecasting the ensemble with a different forcing constant than $F^t = 12$ used for our truth run. Figures 3.6(c) and 3.6(g) show how varying the amount of model error (i.e. varying F) can change the effectiveness of tuning η. Forecast spread adjustment is most helpful in the presence of model errors that result in to larger amplitude oscillations than the truth, such as when $F = 13$, 14, or 15 and $F^t = 12$. Averaging the forecast ensemble to produce the background mean tends to reduce the amplitude of these oscillations, and by increasing η we intensify this reduction. We believe this is compensating for some of the model error. We did not find benefits to tuning η in the perfect model scenario or when the forcing constant for the ensemble was smaller than the true forcing constant.

3.4.4.4 Varying the Observation Error

Figures 3.6(d) and 3.6(e) show that effectiveness of forecast spread adjustment and the tuned value of η depend on the size of the observation error σ. As with smaller ensembles, larger observation errors imply larger analysis errors and hence greater room for improvement. The tuned values of η for observation errors of $\sigma = 0.5$, 1, and 2 are η=4, 2.5, and 1.75 respectively. Thus more observation error corresponds to a smaller tuned value of η. The ensemble spread in each of these three cases is about 1.8. The ensemble spread roughly determines the size of the difference between the mean of the ensemble forecast and the forecast of the ensemble mean; thus, our finding of an ideal forecast ensemble spread independent of observation error size is consistent with the hypothesis above that ensemble averaging is compensating for some of the model error present.

3.4.4.5 Approximating the XKF with the ηETKF

Table 3.1 lists the RMSE from the ηETKF with forecast spread adjustment parameter $\eta = 1$, $\frac{1}{2}$, $\frac{1}{4}$, $\frac{1}{8}$, and 10^{-6} and ensembles of size $k = 40$, 60, and 80.

	$k\!=\!40$	$k\!=\!60$	$k\!=\!80$	XKF
$\eta = 1$	0.88	0.77	0.76	
$\eta = 1/2$	0.95	0.82	0.81	
$\eta = 1/4$	0.97	0.83	0.82	
$\eta = 1/8$	0.97	0.83	0.83	
$\eta = 10^{-6}$	0.96	0.83	0.83	
XKF				0.83

Table 3.1: Comparison between the XKF RMSE and the ηETKF RMSEs for $\eta = 1$, $\frac{1}{2}$, $\frac{1}{4}$, $\frac{1}{8}$, and 10^{-6} and $k = 40$, 60, and 80 with our default parameters.

We remark that the RMSE associated with the ηETKF approaches the XKF-RMSE from below (increasing the error) as η tends to zero and from above (decreasing the error) as k becomes large.

3.5 Summary and Conclusions

Forecast spread adjustment is an add-on to ensemble data assimilation. After the analysis phase we expand the ensemble perturbations about their mean via multiplication by the factor η, forecasting the adjusted ensemble. Prior to the next analysis, we readjust the perturbations around the background mean in order to form the background error covariance matrix.

For linear models, ensemble data assimilation with forecast spread adjustment by any amount reduces to standard ensemble data assimilation ($\eta = 1$). For nonlinear models however, forecast spread adjustment can be beneficial. Furthermore, a ensemble data assimilation with an adjusted forecast spread of $\eta = 1$ is equivalent to standard ensemble data assimilation. Thus ensemble data assimilation with tuned forecast spread adjustment will always perform at least as well as standard ensemble data assimilation even if the parameter η is only roughly tuned.

Forecast spread adjustment affects the ensemble spread during the forecast phase, so for nonlinear models, an ensemble with a different spread will evolve to a different mean with different perturbation amounts and directions.

Some authors (e.g., Stroud and Bengtsson 2007) have found benefits to inflating the observation error covariance, assuming errors in addition to measurement error. In addition to correcting the underestimation of the observation error covariance, they might also be benefiting from the effects of forecast spread adjustment. For an ensemble initialized immediately after the model evolution, forecast spread adjustment and observation error covariance inflation will produce the same analysis mean and will evolve the same ensemble. This is because both maintain the same the ratio between the background and observation error covariances. The difference is that when inflating the observation error covariance, the size of the assumed errors (P^b, R^o, and P^a) is η^2 times bigger than those assumed with forecast spread adjustment.

Due to nonlinear effects and model error, an ensemble which appropriately describes the uncertainty at the analysis time will underestimate the uncertainty after it is evolved in time. To compensate for this, many algorithms implement multiplicative covariance inflation on either the background (e.g., Anderson and Anderson 1999, Whitaker and Hamill 2002, Hunt et al. 2007) or the analysis (e.g., Bonavita et al. 2008). Either technique is valid and forecast spread adjustment can transition between them. Furthermore, forecast spread adjustment can be described as doing both, i.e. inflating the analysis error covariance by η^2, evolving the ensemble, then deflating the background error covariance by ρ/η^2.

We demonstrated that the ηETKF provides a continuum from a reduced-rank Extended Kalman Filter ($\eta = 0$) to the ordinary ETKF ($\eta = 1$) and beyond. If the ensemble is sufficiently large, we recover the full-rank Extended Kalman Filter. On the other hand, we generally found the lowest analysis errors when η was close to or larger than 1.

With our standard parameters, the LETKF with forecast spread adjustment improves by 14% upon the standard LETKF. In comparison, a 10 member ensemble with forecast spread adjustment performs better than a 40 member ensemble without forecast spread adjustment. The parameter η tunes to larger than 1 in all scenarios where we saw improvements. This corresponds to evolving an ensemble with a larger spread than when $\eta = 1$. Our standard parameters include model error induced by changing the forcing constant used when evolving the ensemble to a larger value than the true forcing constant. Forecast spread adjustment proved even more effective in our tests with an even larger forcing constant. However, in perfect model scenarios and when the ensemble forcing constant was smaller than the true forcing, tuning η did not significantly improve the RMSE. In comparison to the RMSE when $\eta = 1$, tuning η was just as effective in improving the RMSE with ensembles of 40 members as it was with ensembles of 10 members, and was even more effective with ensembles of 5 members. Lower observation errors increased the tuned value of η in such a way as to suggest that there is an ideal ensemble spread during the forecast phase which is relatively independent of the true error covariance but depends upon the accuracy of the model.

In conclusion, forecast spread adjustment with $\eta > 1$ provided significant improvement compared to $\eta = 1$ (no adjustment) in some of our experiments with a simple model.

Forecast spread adjustment was particularly effective for smaller ensembles, smaller observation errors, and (in some cases) larger model errors. Our interpretation of why $\eta > 1$ improves results when $F > 12$ but not when $F \leq 12$ is that larger values of F are associated with higher amplitude oscillations and larger values of η tend to reduce these oscillations in the background mean by averaging over a larger ensemble spread.

Chapter 4

Summary and Future Work

I developed and tested two techniques for improving ensemble data assimilation. The first, which I call mixed resolution ensemble data assimilation, incorporates ensembles of two different resolutions,combining the accuracy of a high-resolution model with the larger ensemble size feasible for a low-resolution model. The second, which I call forecast spread adjustment, allows the ensemble spread during the model evolution to be different from the forecast uncertainty, increasing or decreasing the nonlinear effects. This can improve upon previous techniques that restrict the ensemble spread to be commensurate with the forecast uncertainty during both the forecast and analysis phases of data assimilation.

Chapter 1 introduced data assimilation, in particular ensemble Kalman filters. With an ensemble of forecasts, as opposed to a single forecast, we can estimate the uncertainty in the forecast. Data assimilation balances the uncertainties in the "background" (forecast) and in the observations and produces a better "analysis" estimate of the true model state. The uncertainties in the background, observations, and analysis are quantified as error covariances.

Current ensemble methods assume the ensemble is evolved at the same resolution, with possibly one higher resolution control forecast. In Chapter 2, I developed and tested an ensemble data assimilation technique which incorporates ensembles of two resolutions. Higher resolution is desirable for more accurate forecasts; a larger ensemble is desirable for improved sample statistics when estimating the error covariance. However, due to limited computational resources, single resolution ensembles compromise between the ensemble size and resolution. My research shows that combining a small high-resolution ensemble with a large low-resolution ensemble improves the trade-off and improves the accuracy of the analysis *for the same computation cost* when compared to single resolution ensembles or even ensembles which include one high resolution control forecast. I compared the forecast computation time to the RMSE for different ensemble sizes of both mixed and single resolution, assimilating with the mLETKF (mixed-resolution LETKF) and the LETKF (Local Ensemble Transform Kalman Filter) respectively. I tested perfect model scenarios and scenarios with model error. I also designed and tested a scenario where it was possible to resolve the small scale variability of the high resolution model. In all cases I showed improvement for mixed resolution ensembles over both single resolution ensembles and ensemble with only one high resolution member. Future possibilities for expanding upon this research include: applying it to operational weather models; allowing the weight parameter α that controls the relative influence between the two subensembles to vary with scale, e.g., allowing the large scale component of the covariance to come equally from the high and low resolution ensembles while the small scale component comes only from the high resolution ensemble (C. Bishop, personal communication); and more fully exploring the relationship between α and the ensemble spread.

In current ensemble methods, the ensemble not only provides background error covariance used to determine the analysis, but the ensemble spread also continues to correspond to the uncertainty as it is evolved. However, the ensemble only needs to be commensurate with the background uncertainty during the analysis phase of data assimilation. In Chapter 3 I explored the possibility of evolving an ensemble whose spread is not commensurate with the uncertainty during its evolution, readjusting the spread to be commensurate with the uncertainty during the analysis phase. I call this technique forecast spread adjustment. In addition to testing this technique, I also discussed relationships between forecast spread adjustment and other techniques. When the forecast adjustment parameter η is small and the ensemble size is large, the ensemble Kalman filter (EnKF) will evolve the same covariance as the Extended Kalman Filter (XKF). Previous authors have demonstrated the equivalence between the analysis phases of EnKF and the XKF (e.g. Burgers et al. 1998); my research connects them during their forecast phases as well. I also show how forecast spread adjustment can be used to transform background error covariance inflation into analysis error covariance inflation or to combine the two approaches. Some authors have discussed inflating the observation error covariance as well, assuming that it is underestimated (Stroud and Bengtsson 2007) or misrepresented (Li et al. 2009). Forecast spread adjustment produces the same background and analysis means as observation error covariance inflation for ensembles that are initialized during the forecast phase. When the observation error is correctly specified however, the background and analysis errors will be more commensurate with the covariances specified by forecast spread adjustment rather than by observation error covariance inflation.

My results also show that forecast spread adjustment can lead to significant improvements in the accuracy of ensemble data assimilation when model error is present, though the improvement depends on the nature of the model error. In my experiments, improvement occurred in cases where ensemble averaging compensates for model biases. Future research may include further investigation of the reasons behind improvements due to forecast spread adjustment, extensions of the method to help in cases where forecast spread adjustment shows little improvement, development of adaptive methods for tuning η, investigation of forecast spread adjustment in conjunction with ensemble forecasting.

Bibliography

Anderson, J.L, 2007: Exploring the need for localization in ensemble data assimilation using a hierarchical ensemble filter. *Physica D*, **230**, 99-111.

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.

Ballabrera-Poy J., E. Kalnay, and S-C. Yang, 2009: Data assimilation in a system with two scales–combining two initialization techniques. *Tellus A*, **61**, 539-549.

Bishop, C.H., B.J. Etherton, and S.J. Majumdar, 2001: Adaptive sampling with the ensemble transform Kalman filter. Part I: Theoretical aspects. *Mon. Wea. Rev.*, **129**, 420-436.

Bishop, C.H., Hodyss, D., 2009: Ensemble covariances adaptively localized with ECO-RAP. Part 1: tests on simple error models. *Tellus A* **61**, 84-96.

Bonavita, M., L. Torrisi, and F. Marcucci, 2008: The ensemble Kalman filter in an operational regional NWP system: Preliminary results with real observations. *Q.J.R. Meteorol. Soc.*, **134**, 1733-1744.

Buizza, R., P.L. Houtekamer, Z. Toth, G. Pellerin, M. Wei, and Y. Zhu, 2005: A Comparison of the ECMWF, MSC, and NCEP Global Ensemble Prediction Systems. *Mon. Wea. Rev.*, **133**, 1076-1097.

Buizza, R., J.-R. Bidlot, N. Wedi, M. Fuentes, M. Hamrud, G. Holt, and F. Vitart, 2007: The new ECMWF VAREPS (Variable Resolution Ensemble Prediction System). *Q.J.R. Meteorol. Soc.*, **133**, 681-695.

Burgers, G., P.J. van Leeuwen, G. Evensen, 1998: Analysis scheme in the ensemble Kalman filter. *Mon. Wea. Rev.*, **126**, 1719-1724.

Du,J., 2004: Hybrid ensemble prediction system: A new ensembling approach. Preprints. *Symp. 50th Anniversary Operational Numerical Weather Prediction*, College Park, MD, Amer. Meteor. Soc., Paper 4.2.

Evensen, G., 1992: Using the Extended Kalman Filter with a Multilayer Quasi-Geostrophic Ocean Model. *J. Geophys. Res.* **97**, 17905-17924.

Evensen, G., 1994: Sequential data assimilation with a nonlinear quasigeostrophic model using Monte Carlo methods to forecast error statistics. *J. Geophys. Res.* **99**, 10143-10162.

Evensen, G., 2003: The Ensemble Kalman Filter: theoretical formulation and practical implementation. *Ocean Dyn.*, **53**, 343-367.

Gao, J., and M. Xue, 2008: An efficient dual-resolution approach for ensemble data assimilation and tests with assimilated Doppler radar data. *Mon. Wea. Rev.*, **136**, 945-963.

Gao, J., M. Xue, and D. J. Stensrud, 2010: The development of a hybrid EnKF-3DVar algorithm for storm-scale data assimilation. *25th Conf. Several Local Storms*, Denver, CO, Amer. Meteor. Soc., Paper 7.4.

Greybush, S.J., E. Kalnay, T. Miyoshi, K. Ide, B.R. Hunt, 2011: Balance and Ensemble Kalman Filter Localization Techniques. *Mon. Wea. Rev.*, **139**, 511-522.

Hamill, T.M., and C. Snyder, 2000: A Hybrid Ensemble Kalman Filter-3D Variational Analysis Scheme. *Mon. Wea. Rev.*, **128**, 2905-2919.

Hamill, T.M., J.S. Whitaker, C. Snyder, 2001: Distance-dependent filtering of background error covariance estimates in an ensemble Kalman filter, *Mon. Wea. Rev.*, **129**, 2776-2790.

Houtekamer, P.L., H.L. Mitchell, 1998: Data assimilation using an ensemble Kalman filter technique. *Mon. Wea. Rev.*, **126**, 796-811.

Hunt, B.R., E. Kostelich, and I. Szunyogh, 2007: Efficient data assimilation for spatiotemporal chaos: a local ensemble transform Kalman filter. *Physica D*, **230**, 112-126.

Jazwinski, A.H., 1970: *Stochastic Processes and Filtering Theory*. Academic Press, 376 pp. Reprint by Dover Publications in 2007.

Kalman, R.E., A new approach to linear filtering and prediction problems, *Trans. ASME, Ser. D, J.Basic Eng.*, **82**, 35-45.

Kalnay, E., 2003: *Atmospheric Modeling, Data Assimilation and Predictability*. Cambridge University Press, 341 pp.

Kleist, D., 2012: *An evaluation of hybrid variation-ensemble data assimilation for the NCEP GFS.* PhD thesis, University of Maryland College Park.

Leith, C. E., 1974: Theoretical skill of Monte Carlo forecasts, *Mon. Wea. Rev.*, **102**, 409-418.

Li, H., E. Kalnay, and T. Miyoshi, 2009: Simultaneous estimation of covariance inflation and observation errors within an ensemble Kalman filter. *Q.J.R. Meteorol. Soc.*, **135**, 523-533.

Lorenc, A.C., 2003: The potential of the ensemble Kalman filter for NWP—A comparison with 4D-VAR. *Q.J.R. Meteorol. Soc.*, **129**, 3183-3203.

Lorenz, E.N., 1996: Predictability—A problem partly solved. *Proc. Seminar on Predictability*, Vol. 1, ECMWF, Reading, Berkshire, UK, 1-18.

Lorenz, E.N., 2005: Designing Chaotic Models. *J. Atmos. Sci.*, **62**, 1574-1587.

Lorenz, E.N., and K.A. Emanuel, 1998: Optimal sites for supplementary weather observations: Simulation with a small model. *J. Atmos. Sci.*, **55**, 399-414.

Miyoshi, T., 2011: The Gaussian approach to adaptive covariance inflation and its implementation with the Local Ensemble Transform Kalman Filter. *Mon. Wea. Rev.*, **139**, 1519-1535.

Molteni, F., R. Buizza, T. N. Palmer, and T. Petroliagis, 1996: The ECMWF ensemble prediction system: Methodology and validation. *Q.J.R. Meteorol. Soc.,* **122**, 73-119.

Ott, E., B.R. Hunt, I. Szunyogh, A.V. Zimin, E.J. Kostelich, M. Corazza, E. Kalnay, D.J. Patil, and J.A. Yorke, 2004: A local ensemble Kalman filter for atmospheric data assimilation. *Tellus A*, **56**, 415-428.

Toth, Z., and E. Kalnay, 1993: Ensemble Forecasting at NMC: The Generation of Perturbations. *Bull. Amer. Meteorol. Soc.,* **74**, 2317-2330.

Toth, Z., and E. Kalnay, 1997: Ensemble forecasting at NCEP and the breeding method. *Mon. Wea. Rev.*, **125**, 3297-3319.

Stroud, J.R., and T. Bengtsson, 2007: Sequential State and Variance Estimation within the Ensemble Kalman Filter. *Mon. Wea. Rev.*, **135**, 3194-3208.

Wang, X., C.H. Bishop, and S.J. Julier, 2004: Which is better, an ensemble of positive-negative pairs or a centered spherical simplex ensemble?. *Mon. Wea. Rev.*, **132**, 1590-1605.

Wang, X., C. Snyder, and T. M. Hamill, 2007: On the theoretical equivalence of differently proposed ensemble/3D-Var hybrid analysis schemes. *Mon. Wea. Rev.*, **135**, 222-227.

Whitaker, J.S., and T.M. Hamill, 2002: Ensemble data assimilation without perturbed observations. *Mon. Wea. Rev.*, **130**, 1913-1924.

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. *Q.J.R. Meteorol. Soc.*, **135**, 251-262.

Yoon, Y-N, and E. Ott, 2010: On the Propagation of Information and the Use of Localization in Ensemble Kalman Filtering. *J. Atmos. Sci.*, **67**, 3823-3834.

Zhang, M., and F. Zhang, 2012: E4DVar: Coupling an Ensemble Kalman Filter with Four-Dimensional Variational Data Assimilation in a Limited-Area Weather Prediction Model. *Mon. Wea. Rev.*, **140**, 587-600.

Zupanski, M., 2005: Maximum likelihood ensemble filter: Theoretical aspects. *Mon. Wea. Rev.*, **133**, 1710-1726.