The Maximum Likelihood Ensemble Filter with State Space Localization

MILIJA ZUPANSKI[a]

[a] *Cooperative Institute for Research in the Atmosphere, Colorado State University, Fort Collins, Colorado*

ABSTRACT: A new method for ensemble data assimilation that incorporates state space covariance localization, global numerical optimization, and implied Bayesian inference is presented. The method is referred to as the MLEF with state space localization (MLEF-SSL) due to its similarity with the maximum likelihood ensemble filter (MLEF). One of the novelties introduced in MLEF-SSL is the calculation of a reduced-rank localized forecast error covariance using random projection. The Hessian preconditioning is accomplished via Cholesky decomposition of the Hessian matrix, accompanied with solving triangular system of equations instead of directly inverting matrices. For ensemble update, the MLEF-SSL system employs resampling of posterior perturbations. The MLEF-SSL was applied to Lorenz model II and compared to ensemble Kalman filter with state space localization and to MLEF with observation space localization. The observations include linear and nonlinear observation operators, each applied to integrated and point observations. Results indicate improved performance of MLEF-SSL, particularly in assimilation of integrated nonlinear observations. Resampling of posterior perturbations for an ensemble update also indicates a satisfactory performance. Additional experiments were conducted to examine the sensitivity of the method to the rank of random matrix and to compare it to truncated eigenvectors of the localization matrix. The two methods are comparable in application to low-dimensional Lorenz model, except that the new method outperforms the truncated eigenvector method in case of severe rank reduction. The random basis method is simple to implement and may be more promising for realistic high-dimensional applications.

KEYWORDS: Filtering techniques; Kalman filters; Optimization

1. Introduction

There are several desirable aspects of data assimilation that are desirable but challenging in realistic high-dimensional applications: 1) Bayesian inference, 2) nonlinear capability, and 3) assimilation of integrated observations. Bayesian inference (e.g., Jaynes 2003) is a process of updating probability density function (PDF) using new information. In terms of PDF moments, Bayesian inference implies updating all PDF moments in agreement with Bayes's theorem. Since commonly used data assimilation algorithms are based on the assumption of Gaussian errors, which reduces Bayesian approach to estimating the first two moments, namely the mean and the covariance, we also assume Gaussian errors. Nonlinearity in data assimilation is commonly addressed by using numerical optimization (e.g., Nocedal and Wright 1999). An important requirement for numerical optimization is global convergence, since in that case it is assured that the estimated state is indeed optimal for all points in the model domain. Vertically integrated observations are most often related to satellite radiances, which are the main source of information in meteorological applications.

Although this may not be immediately obvious, the assimilation of integrated observations can pose a challenge for data assimilation. To see that, consider ensemble data assimilation (Evensen 1994; Houtekamer and Mitchell 1998). Although in principle the

Bayesian inference is satisfied in terms of PDF moments and nonlinear capability can also be added (Zupanski 2005; Lorentzen and Naevdal 2011; Sakov et al. 2012; Evensen 2018). In high-dimensional applications, however, there is an insufficient number of ensembles to explain the high-dimensional analysis subspace. A common remedy is to apply covariance localization using a Schur product between the ensemble covariance and a predefined localization matrix (Houtekamer and Mitchell 1998; Houtekamer and Mitchell 2001; Hamill et al. 2001). As shown in Bocquet and Carrassi (2017) the localization is required when the ensemble size is smaller than the number of unstable and neutral modes of the dynamics. There are two typical approaches to covariance localization in ensemble data assimilation: (i) observation space localization and (ii) state (e.g., model) space localization. Both approaches are based on the assumption that correlations between variables decrease with distance and have been extensively investigated (Houtekamer and Mitchell 1998; Houtekamer and Mitchell 2001; Anderson 2007; Miyoshi and Yamane 2007; Nerger et al. 2012; Kepert 2009; Campbell et al. 2010; Janjić et al. 2011; Sakov and Bertino 2011; Greybush et al. 2011; Gasperoni and Wang 2015). A major difficulty of observation space localization unfolds in case of vertically integrated observations, since in that situation the vertical location of observation is undefined. Given the dominance of satellite radiance observations in meteorological data assimilation, it appears that state space localization is preferred. Unfortunately, in high-dimensional applications storing and using explicit localized error covariance in matrix form is computationally prohibitive, effectively preventing the exclusive use of ensemble data assimilation.

Variational data assimilation, originally introduced by Le Dimet and Talagrand (1986), can assimilate integrated observations, so the assimilation of satellite radiances is straightforward.

They are also equipped with numerical optimizations to address nonlinearities (e.g., Navon and Legler 1987). However, only the state is updated based on new information, while the error covariance is prescribed. This prevents a reliable estimate of analysis uncertainty and consequently does not allow Bayesian inference even under Gaussian PDF assumption.

Ensemble-variational (EnVar) methods and its hybrid extensions (e.g., Bannister 2017 and references therein) have emerged partially in an effort to address the above-mentioned three issues of realistic high-dimensional data assimilation. EnVar is a well-established scheme to address nonlinearities and state-space localization. Practical EnVar typically consists of two systems: (i) ensemble data assimilation algorithm used to obtain the flow-dependent uncertainty and (ii) variational data assimilation algorithm to produce the analysis. There are several advantages of such EnVar system, in that it produces the optimal state and its uncertainty, incorporates numerical optimization for addressing nonlinearities, and applies state-space localization for flow-dependent ensemble covariances, all in realistic high-dimensional systems. However, the use of two data assimilation systems, namely the ensemble and the variational, makes the estimate of the state and its uncertainty inconsistent (e.g., Auligne et al. 2016; Bannister 2017; Buehner et al. 2017), effectively preventing Bayesian inference. Resolving the Bayesian inference problem would require fundamental changes that may bring additional limitations and, therefore, may not be worthwhile. Since addressing nonlinearity and assimilation of integrated observations in practical EnVar is inherently connected to the use of variational methods, the uncertainty estimation consistent with Bayesian inference would have to be done through the variational component. However, finding a dependable method that can be used for uncertainty estimation in the context of high-dimensional variational algorithm is challenging. Although there may be other possibilities for estimating the analysis uncertainty in variational method, most often the approaches are based on using an inverse Hessian (e.g., Rabier and Courtier 1992; Fisher and Courtier 1995; LeDimet et al. 1997). While direct inversion of the Hessian matrix is practically impossible in realistic high dimensions, indirect estimation techniques such as partial SVD of Hessian, an update of inverse Hessian during minimization, or a second-order adjoint (e.g., Le Dimet et al. 2002) may be feasible. None of these techniques, however, have been used in practical EnVar, possibly because of added algorithmic complexity and still an excessive computational overhead.

In an apparent effort to preserve Bayesian inference of ensemble data assimilation while assimilating integrated observations, several research groups have been exploiting the use of state space localization. This includes the modulated ensemble approach (Buehner 2005; Leng et al. 2013; Bishop et al. 2015; Bishop et al. 2017), and the randomized singular value decomposition method (Farchi and Bocquet 2019). The modulated ensemble approach is based on developing the eigenvalue decomposition (EVD) of the localization matrix used in the Schur (elementwise) product and employing the truncated EVD for representing the augmented ensemble space. The randomized singular value decomposition (SVD) technique is described in

Halko et al. (2011) and is applied by Farchi and Bocquet (2019) to decompose the localized forecast error covariance. There are additional papers that demonstrate the benefits of state-space localization for assimilating vertically integrated satellite data in realistic applications (Lei et al. 2018; Shlyaeva et al. 2019; Steward et al. 2018). These methods are generally aimed at partitioning covariance localization to observation space localization in horizontal and to state space localization in vertical. While this is an efficient and practical way of dealing with vertically integrated observations in ensemble data assimilation, the observation space localization in horizontal prevents finding a global minimizing solution.

As implied from the above discussion, the difficulty in addressing and successfully resolving in high-dimensional applications all three desirable features of data assimilation come from the fact that they are closely connected so resolving one aspect may not be beneficial for another aspect of data assimilation. The intuitive approach is to try to consider all points simultaneously, but this is certainly not trivial given the complexity of realistic high-dimensional data assimilation.

There have been some recent attempts to develop a data assimilation system that can simultaneously address all the above three conditions in high-dimensional problems. One such method has been suggested by Nino-Ruiz et al. (2020), in which a Modified-Cholesky decomposition based on regularization of covariance matrices (Bickel and Levina 2008) is used to decompose the localized forecast error covariance. Although the modified-Cholesky method incorporates mathematically well-posed globally convergent numerical optimization, it may be less flexible for realistic high-dimensional applications and, therefore, adversely impact computational efficiency. This is because the modified Cholesky is calculated for the localized forecast error covariance defined in state space, which could be on the order of 10^8 in realistic models. Consequently, the efficiency of computations and storage of modified Cholesky becomes challenging in such situations.

Another such method is the ensemble-variational integrated localized (EVIL) method of Auligne et al. (2016). EVIL relies on capturing the leading Ritz vectors and values (approximate eigenvectors and eigenvalues) of the Hessian matrix as a by-product of the Lanczos minimization algorithm. The Hessian is also used for estimating the analysis error covariance, similar to Zupanski (2005). Although the idea of building eigenvectors of Hessian during minimization is commendable, the reconstruction of Hessian is hard-wired to the number of minimization iterations. Since at least one iteration of Lanczos minimization is required for producing one Ritz pair, the number of minimization iterations could be on the order of 10^6, which makes the method less favorable for realistic high-dimensional applications. The hard-wired link between Lanczos minimization and Hessian estimation is also preventing the use of alternative minimization algorithms that may be better suited for a particular nonlinear problem, making EVIL less robust.

Given the limitations of these algorithms, in this paper we present a data assimilation method that incorporates all the above desirable features in a computationally efficient way applicable to realistic high-dimensional problems. The new

method is related to maximum likelihood ensemble filter (MLEF; Zupanski 2005) in its idea to use a single algorithm to address nonlinearity while simultaneously estimating the analysis and its uncertainty, although the inclusion of state space localization results in important algorithmic differences. The main idea of the new method is to use the covariance localization as in EnVar (e.g., Buehner 2005), but reduce the rank of localized forecast error covariance matrix by creating a random projection to its range (e.g., Mahoney 2011, chapter 3). This effectively reduces the dimension of the analysis subspace so that a global numerical optimization with optimal Hessian preconditioning, which is essential in original MLEF, can be implemented in high-dimensional applications. The algorithmic details of the new method are designed to handle such large dimensions, in part by precalculating random projection, using highly efficient Cholesky decomposition of the Hessian for preconditioning, and employing a fast triangular system solver instead of direct inversion to handle the required matrix operations.

The new method results in several advantages over other existing data assimilation methods. The proposed method produces an uncertainty estimate consistent with the optimal state and, therefore, satisfies Bayesian inference. The proposed method can employ a wide range of numerical optimization algorithms, effectively expanding its capability to address nonlinear problems. While EVIL calculates an EVD of the Hessian matrix for preconditioning, like the original MLEF algorithm (Zupanski 2005) and ensemble transform Kalman filter (ETKF; Bishop et al. 2001), the method proposed here applies Cholesky factorization of the Hessian. The use of Hessian in minimization and for estimating analysis uncertainty is accomplished in EVIL and in original MLEF via direct inversion using eigenvector/eigenvalues, while in the proposed method a triangular system of equations is solved instead. Calculation of truncated eigenspace in EVIL is linked to the number of minimization iterations, therefore, requiring large number of minimization iterations in realistic high-dimensional applications to achieve a good approximation of analysis uncertainty. In the method proposed here, however, the calculation of the high-dimensional analysis subspace and uncertainty is practically independent of minimization.

The mathematical framework of the new method is presented in section 2, experimental design is given in section 3, followed by results and discussion in section 4, and summary and conclusions in section 5.

2. MLEF with state space localization

In this section we describe the new ensemble data assimilation algorithm which applies the state space covariance localization, referred here as the MLEF with state space localization (MLEF-SSL). The state space localization implies here that localized forecast error covariance is calculated explicitly by applying the Schur product between ensemble forecast error covariance and a predefined localization matrix. The new algorithm maintains some properties of the original MLEF algorithm (Zupanski 2005; Zupanski et al. 2008), such as a globally convergent numerical

optimization and an estimate of analysis uncertainty from the inverse Hessian at the minimum. The motivation for using the inverse Hessian method for estimating analysis uncertainty is mostly because of the relevance of Hessian for numerical optimization, including preconditioning (e.g., Gill et al. 1981; Axelsson and Barker 1984; Luenberger 1984). Using alternative means for estimating analysis uncertainty may also be possible. However, since at present there is no known clear advantage of a specific method, we proceed with the inverse Hessian method. The main appeal of the method is in its design that allows the three favorable features of data assimilation to be simultaneously satisfied in high-dimensional applications: (i) Bayesian inference, (ii) numerical optimization, and (iii) assimilation of integrated observations using state-space localization in all spatial directions.

There are three distinct components of the new algorithm: 1) calculation of localized forecast error covariance, 2) iterative minimization of the cost function to find optimal analysis, and 3) ensemble update required for continuing to the next assimilation cycle.

a. Forecast error covariance

Definition of the localized forecast error covariance is critical for understanding the new methodology and will be explained here in detail.

We begin by defining the three mathematical subspaces relevant for data assimilation: (i) *state space* S with dimension N_S, (ii) *observation* space O with dimension N_O, and (iii) *ensemble* space E with dimension N_E. Given that typical dimensions of state space and observation space are much larger than the ensemble space dimension, major MLEF-SSL (and MLEF) calculations are done in ensemble subspace. Adjustment of the code is possible if observation space is much smaller than ensemble space, but this will not be addressed in this paper. The ensemble forecast error covariance, $\mathbf{P}_E: S \rightarrow S$, is

$$\mathbf{P}_E = \sum_{i=1}^{N_E} \mathbf{p}_i \mathbf{p}_i^{\mathrm{T}}, \qquad (2.1)$$

where superscript T denotes the transpose and $\mathbf{p}_i$ is the ith column of the ensemble square root forecast error covariance $\mathbf{P}_E^{1/2}: E \rightarrow S$:

$$\mathbf{P}_E^{1/2} = (\mathbf{p}_1 \cdots \mathbf{p}_{N_E}). \qquad (2.2)$$

The localized forecast error covariance is typically calculated using a Schur product between the localizing matrix and the original ensemble forecast error covariance. If $\mathbf{L}: S \rightarrow S$ denotes the localizing matrix, then $\mathbf{P}_f = \mathbf{L} \circ \mathbf{P}_E$ denotes the localized forecast error covariance. Formally, the localization of ensemble forecast error covariance produces

$$\mathbf{L} \circ \mathbf{P}_E = \mathbf{L} \circ \sum_{i=1}^{N_E} \mathbf{p}_i \mathbf{p}_i^{\mathrm{T}} = \sum_{i=1}^{N_E} (\mathbf{L} \circ \mathbf{p}_i \mathbf{p}_i^{\mathrm{T}}). \qquad (2.3)$$

After using the Schur product identity,

$$\mathbf{L} \circ \mathbf{a}\mathbf{b}^{\mathrm{T}} = \mathrm{diag}(\mathbf{a}) \cdot \mathbf{L} \cdot \mathrm{diag}(\mathbf{b}), \qquad (2.4)$$

where **a** and **b** are vectors and "diag" denotes a diagonal matrix with vector elements on the main diagonal, the localized forecast error covariance is

$$\mathbf{L} \circ \mathbf{P}_E = \sum_{i=1}^{N_E} \mathbf{D}_i \cdot \mathbf{L} \cdot \mathbf{D}_i, \qquad (2.5)$$

where $\mathbf{D}_i = \mathrm{diag}(\mathbf{p}_i)$. The above formulation leads to the localized ensemble square root error covariance (e.g., Lorenc 2003; Buehner 2005), commonly used in EnVar methods. Reduction of the rank of the localizing matrix **L** is also possible (e.g., Buehner 2005; Buehner 2012; Buehner and Shlyaeva 2015), potentially reducing the computational cost of applying (2.5) in high-dimensional applications. However, following our approach to employ a single assimilation system and to calculate the inverse Hessian for preconditioning of minimization and for analysis uncertainty, storing and using matrix (2.5) still presents a computational challenge in high-dimensional applications.

Therefore, we proceed with further adjustments of the localized forecast error covariance (2.5) by introducing a sufficiently low-dimensional subspace of the high-dimensional state space. The strategy of MLEF-SSL is to form a reduced-rank localized forecast error covariance by a random projection to its range. Random projection method consists of postmultiplying the original matrix, in this case the localized error covariance, by a reduced-rank random matrix. This amounts to selecting linear combinations of the columns of the localized error covariance. Randomization methods have been used extensively in large scale problems for improving the efficiency of matrix operations (Mahoney 2011). In our application random projection is achieved by embedding a reduced-rank approximate identity matrix in the localized forecast error covariance:

$$\mathbf{L} \circ \mathbf{P}_E = (\mathbf{L} \circ P_E)^{1/2}(\mathbf{L} \circ P_E)^{\mathrm{T}/2} \approx (\mathbf{L} \circ P_E)^{1/2} \cdot \mathbf{I}_{\mathrm{RR}} \cdot (\mathbf{L} \circ P_E)^{\mathrm{T}/2}, \qquad (2.6)$$

where subscript "RR" refers to reduced rank and $\mathbf{I}_{\mathrm{RR}}: S \to S$ denotes the reduced-rank identity matrix. It is important to note that expression (2.6) approximates the original Schur product (2.5); however, it also is the one that potentially allows a consistent and satisfactory estimation of analysis uncertainty in the new system. The low-dimensional subspace of the state space is denoted S_{RR} and $S_{\mathrm{RR}} \subseteq S$. This subspace is spanned by basis vectors $\{\mathbf{u}_n \in S_{\mathrm{RR}}: n = 1,\dots, N_{\mathrm{RR}}\}$, where N_{RR} is the dimension of the reduced-rank subspace. The reduced-rank identity matrix is defined as an outer product of basis vectors:

$$\mathbf{I}_{\mathrm{RR}} = \sum_{n=1}^{N_{\mathrm{RR}}} \mathbf{u}_n \mathbf{u}_n^{\mathrm{T}}. \qquad (2.7)$$

The low-dimensional basis and the use of reduced-rank identity matrix effectively reduce the state space to manageable dimensions. The definition of "manageable" would certainly depend on dimensionality of the state space and on available computational resources. This could be better understood if we introduce the rank of the original localized covariance matrix

(2.5), denoted N_L, where $N_L \leq N_S$. The rank of the approximate matrix (2.6) is bounded by N_{RR}, as implied from (2.7). Although the upper bound of the rank is achieved only for an orthogonal basis in (2.7), to simplify the notation we will use N_{RR} to denote the rank of (2.7), and implicitly the rank of (2.6). The reduced-rank formulation of the localized error covariance (2.6) will make practical sense only if it can be larger than the dynamical ensemble size, i.e., if $N_E \leq N_{\mathrm{RR}}$. On the other end, introducing the reduced-rank localized error covariance will be meaningful only if the rank of the original localized error covariance (2.5) is sufficiently large, so that computing consistent estimates of the state and its uncertainty is computationally prohibitive, implying $N_E \leq N_{\mathrm{RR}} \leq N_L$. To achieve a true practical advantage of the reduced-rank formulation of localized error covariance, this relationship between ranks can be strengthened to $N_E \ll N_{\mathrm{RR}} \ll N_L$, where symbol "$\ll$" refers to a difference of at least one order of magnitude. In realistic problems $N_S \sim O(10^8)$, $N_L \sim O(10^6)$ $O(10^7)$, and $N_E \sim O(10^2)$. This puts the desired rank of the approximate localized error covariance (2.6) somewhere between $O(10^2)$ and $O(10^6)$, preferably $O(10^3) \leq N_{\mathrm{RR}} \leq O(10^5)$ to have a practical advantage. Note that reducing the rank of the localizing matrix **L** following Buehner (2012) could benefit the proposed method since in that case the rank N_L of the original localized formulation (2.5) would effectively be smaller and, therefore, easier to approximate by N_{RR}.

After embedding (2.7) in (2.6) the localized forecast error covariance is

$$\begin{aligned}\mathbf{P}_f &= (\mathbf{L} \circ \mathbf{P}_E)^{1/2}\left(\sum_n \mathbf{u}_n \mathbf{u}_n^{\mathrm{T}}\right)(\mathbf{L} \circ \mathbf{P}_E)^{\mathrm{T}/2} \\ &= \sum_n [(\mathbf{L} \circ \mathbf{P}_E)^{1/2}\mathbf{u}_n][(\mathbf{L} \circ \mathbf{P}_E)^{1/2}\mathbf{u}_n]^{\mathrm{T}}. \qquad (2.8)\end{aligned}$$

The square root of the localized ensemble error covariance (2.5) is

$$(\mathbf{L} \circ \mathbf{P}_E)^{1/2} = \left(\mathbf{D}_1 \mathbf{L}^{1/2} \cdots \mathbf{D}_{N_E}\mathbf{L}^{1/2}\right). \qquad (2.9)$$

Finally, using (2.9) in (2.8) gives the new localized forecast error covariance:

$$\mathbf{P}_f = \sum_{i=1}^{N_E}\sum_{n=1}^{N_{\mathrm{RR}}} (\mathbf{D}_i\mathbf{L}^{1/2}\mathbf{u}_n)(\mathbf{D}_i\mathbf{L}^{1/2}\mathbf{u}_n)^{\mathrm{T}}. \qquad (2.10)$$

The square root factorization $\mathbf{P}_f = \mathbf{F}\mathbf{F}^{\mathrm{T}}$

$$\mathbf{F} = (\mathbf{f}_1 \cdots \mathbf{f}_{N_E \times N_{\mathrm{RR}}}), \quad \mathbf{f}_k = \mathbf{D}_i\mathbf{L}^{1/2}\mathbf{u}_n \quad (k = 1,\dots, N_E \times N_{\mathrm{RR}}). \qquad (2.11)$$

Therefore, the row-dimension of the square root forecast error covariance can be considerably increased compared to the ensemble size. It is convenient to define the computational analysis subspace A of dimension $N_A = N_E = N_{\mathrm{RR}}$, with $\mathbf{F}: C \to S$, i.e., the subspace in which the analysis is computed. Note that the actual reduced-rank analysis subspace is of dimension N_{RR}, implied from the rank of $\mathbf{F}$. The formulation (2.11) shows that one needs to store only $N_E + N_{\mathrm{RR}}$ vectors in order to

produce $N_E + N_{RR}$ columns of **F**. Assuming that in realistic applications $N_{RR} \sim O(10^3)$, and $N_E \sim O(10^2)$, the dimension of the computational analysis subspace is $N_A \sim O(10^5)$.

As it can be seen from (2.10) and (2.11), a consequence of introducing (2.7) is that a matrix–vector product can be used to define the localization of the square root error covariance, which is computationally more efficient than using the matrix form (2.5). In addition, the potentially intensive computation of $\mathbf{L}^{1/2}\mathbf{u}_n$ can be precalculated since it does not depend on dynamical ensembles or observations, implying considerable computational savings.

Given that for increased efficiency and accuracy of calculations in (2.7) we are looking to find the smallest subset of basis vectors that can approximate the reduced-rank identity matrix, it appears that choosing orthogonal or near-orthogonal basis vectors $\{\mathbf{u}_n\}$ may be preferred. While the choice of orthogonal basis vectors is practically infinite, as a first step in developing the MLEF-SSL algorithm we define a random basis. Random basis is only near-orthogonal, though, but it approaches to orthogonality for large sample anticipated in realistic applications. The random basis option is also relatively simple to implement and computationally efficient.

Using random basis vectors, the identity matrix can be approximated by

$$\mathbf{I} \approx \frac{1}{N_{RR}-1} \sum_{n=1}^{N_{RR}} \mathbf{r}_n \mathbf{r}_n^T, \tag{2.12}$$

where $\mathbf{r} \sim N(0,1)$ is defined as an uncorrelated Gaussian random vector in state space ($\mathbf{r} \in S$) and N_{RR} is the number of samples. With this change

$$\mathbf{u}_n = \frac{1}{\sqrt{N_{RR}-1}} \mathbf{r}_n \quad (n = 1, \ldots, N_{RR}) \tag{2.13}$$

and the columns of the localized square root forecast error covariance are

$$\mathbf{f}_k = \frac{1}{\sqrt{N_{RR}-1}} \mathbf{D}_i \mathbf{L}^{1/2} \mathbf{r}_n \quad (k = 1, \ldots, N_E \times N_{RR}). \tag{2.14}$$

The precalculated vectors (2.13) can be used as long as the control variables, the analysis grid resolution, and the localization length are kept the same. Only if any of the mentioned three parameters has changed there will be a need for recalculation.

b. Analysis

For finding the optimal analysis solution and its uncertainty we consider cost function:

$$J(\mathbf{x}) = \frac{1}{2}(\mathbf{x} - \mathbf{x}^f)^T \mathbf{P}_f^{-1}(\mathbf{x} - \mathbf{x}^f) + \frac{1}{2}[\mathbf{y} - h(\mathbf{x})]^T \mathbf{R}^{-1}[\mathbf{y} - h(\mathbf{x})], \tag{2.15}$$

where $\mathbf{x}$ is state vector, superscript f defines the forecast/background, $\mathbf{y}$ is observation vector, $h: S \to O$ is a nonlinear observation operator, and $\mathbf{R}: O \to O$ is observation error covariance. The forecast error covariance is defined using (2.11).

We introduce the change of variable commonly used in data assimilation to make the new control variable scaled and nondimensional:

$$\mathbf{x} = \mathbf{x}^f + \mathbf{F}\mathbf{w}, \tag{2.16}$$

where $\mathbf{w}$ is the new control variable defined in analysis subspace. The cost function is transformed to

$$J(\mathbf{w}) = \frac{1}{2}\mathbf{w}^T\mathbf{w} + \frac{1}{2}[\mathbf{y} - h(\mathbf{x}^f + \mathbf{F}\mathbf{w})]^T \mathbf{R}^{-1}[\mathbf{y} - h(\mathbf{x}^f + \mathbf{F}\mathbf{w})]. \tag{2.17}$$

The cost function formulated in (2.17) is minimized in practice, not in the form (2.15), in order to avoid the inversion of the forecast error covariance.

It is convenient to define an $N_O \times N_E$ matrix:

$$\mathbf{Z}_f = \left(\mathbf{z}_1^f \cdots \mathbf{z}_{N_A}^f\right), \quad \mathbf{z}_k^f = \mathbf{R}^{-1/2}\mathbf{H}_f\mathbf{f}_k, \quad \mathbf{H}_f = \left(\frac{\partial h}{\partial \mathbf{x}}\right)_{\mathbf{x}^f}. \tag{2.18}$$

Although the above formulation is currently adopted in MLEF-SSL, one could alternatively use

$$\mathbf{z}_k^f = \mathbf{R}^{-1/2}[h(\mathbf{x}^f + \mathbf{f}_k) - h(\mathbf{x}^f)] \tag{2.19}$$

as in the original MLEF, with possible advantage due to extension to nonsmooth optimization (Zupanski et al. 2008). There are also computational reasons that may favor this definition. For the anticipated high-dimensional analysis subspace it is generally more efficient to compute the Jacobian coefficients only once and reuse them as needed following (2.18), instead of calculating the perturbed nonlinear operators $h(\mathbf{x}^f + \mathbf{f}_k)$ in (2.19) many times. Therefore, in the current MLEF-SSL the formulation (2.18) is chosen as a default. However, this situation may change if operator h is highly complex and requires considerable calculation and development. In addition, with the advent of Machine Learning (ML) and Deep Neural Networks (DNN) (e.g., LeCun et al. 2015 and references therein) it is possible to imagine a very fast calculation of the nonlinear operator h, favoring the definition (2.19). All those aspects need to be considered when calculating the matrix $\mathbf{Z}_f$ in real applications.

The gradient of the cost function (2.17) is

$$\mathbf{g} = \mathbf{w} - \mathbf{Z}_f^T \mathbf{R}^{-1/2}[\mathbf{y} - h(\mathbf{x}^f + \mathbf{F}\mathbf{w})] \tag{2.20}$$

and the Hessian (second derivative) is

$$\mathbf{Q} = \mathbf{I} + \mathbf{Z}_f^T \mathbf{Z}_f. \tag{2.21}$$

To proceed, the gradient and Hessian are used in iterative numerical optimization to find the optimal analysis. In each minimization iteration,

$$\mathbf{w}_{j+1} = \mathbf{w}_j + \alpha_j \mathbf{d}_j \quad (j = 0, \ldots, j_{max}), \tag{2.22}$$

where j is the iteration index, j_{max} is the maximum number of iterations, α is the step-size, and $\mathbf{d}$ is the descent direction vector.

The efficiency of numerical optimization can be improved by applying Hessian preconditioning (e.g., Gill et al. 1981; Luenberger 1984). The optimal Hessian preconditioning for quadratic cost function is defined (e.g., Axelsson and Barker 1984) as a transpose of the inverse square root of Hessian, which amounts to the second change of variable:

$$\mathbf{w} = (\mathbf{Q}^{1/2})^{-T}\boldsymbol{\zeta}, \tag{2.23}$$

where $\boldsymbol{\zeta}$ is the preconditioned control variable.

To apply the above-described Hessian preconditioning a way to compute the square root and the inverse is needed. The approach taken in MLEF-SSL is to calculate the Cholesky decomposition (e.g., Golub and van Loan 1989) of the Hessian matrix $\mathbf{Q}$:

$$\mathbf{Q} = \mathbf{GG}^T, \tag{2.24}$$

where $\mathbf{G}$ is the unique lower triangular matrix with $\mathbf{Q}^{1/2} = \mathbf{G}$. One could still consider using the EVD as in the original MLEF, but such approach may be computationally more challenging and numerically less stable than Cholesky decomposition (e.g., Golub and van Loan 1989) in high-dimensional applications.

Therefore, the square root of $\mathbf{Q}$ is obtained directly from Cholesky decomposition. Instead of calculating the inverse of $\mathbf{G}$, one can solve a triangular linear system of equations. For example, let a lower triangular matrix $\mathbf{G}$ and vector $\mathbf{t}$ be given and that an unknown vector $\mathbf{q} = \mathbf{G}^{-1}\mathbf{t}$ is needed. Instead of finding the inverse $\mathbf{G}^{-1}$ and directly solving for $\mathbf{q}$, one can alternatively solve the triangular linear system of equations $\mathbf{G}_q = \mathbf{t}$ for the unknown vector $\mathbf{q}$. Consequently,

$$\mathbf{q} = \mathbf{G}^{-1}\mathbf{t} \quad \Leftrightarrow \quad \mathbf{Gq} = \mathbf{t}. \tag{2.25}$$

The solution of the triangular linear system is quite efficient. The use of triangular system of equations instead of the inverse can be illustrated in calculations of descent direction using Newton equation that relates gradient and descent direction (Luenberger 1984), commonly used in the first iteration of minimization:

$$\mathbf{d} = -\mathbf{Q}^{-1}\mathbf{g} = -(\mathbf{G}^T)^{-1}\mathbf{g} = -\mathbf{G}^{-T}\mathbf{G}^{-1}\mathbf{g}. \tag{2.26}$$

The first matrix–vector multiplication produces the preconditioned gradient $\mathbf{g}_\zeta = \mathbf{G}^{-1}\mathbf{g}$ which could be obtained by solving the triangular system:

$$\mathbf{Gg}_\zeta = \mathbf{g}. \tag{2.27}$$

The next step is to obtain $\mathbf{d} = \mathbf{G}^{-T}\mathbf{g}_\zeta$ which can be accomplished by solving

$$\mathbf{G}^T\mathbf{d} = -\mathbf{g}_\zeta. \tag{2.28}$$

Similar reasoning can be applied to any other matrix–vector product that involves $\mathbf{G}^{-1}$ or $\mathbf{G}^{-T}$ that may be encountered in the minimization algorithm.

After the optimal analysis is found by iterative minimization, the analysis error covariance is required and is calculated here from the inverse Hessian matrix at the minimum, as in original MLEF. One should note that using inverse Hessian is not known to be better or worse than any other method for estimating analysis uncertainty, but it is used here to maintain the closeness with original MLEF and since Hessian is intrinsically connected to numerical optimization. A consequence of applying change of variable (2.16) is that Hessian matrix (2.21) has N_{RR} as an upper limit of its rank. For anticipated values $N_{RR} \ll N_L$ the Hessian will be approximate, also implying that the analysis error covariance estimation will be approximate. In relation to nonlinearity, the relationship between Hessian and analysis error covariance can still be an acceptable approximation for mildly nonlinear problems (Gejadze et al. 2013). In highly nonlinear problems the use of the "effective inverse Hessian" method of Gejadze et al. [2018, their Eq. (4.16)] or similar may be more adequate, although the cost/benefit ratio of such an estimate remains to be evaluated in realistic data assimilation system.

Inverse Hessian of the original cost function (2.15) estimated at the optimal analysis solution $\mathbf{x}^a$ is

$$\mathbf{P}_a = \mathbf{F}(\mathbf{I} + \mathbf{Z}_a^T\mathbf{Z}_a)^{-1}\mathbf{F}^T, \tag{2.29}$$

where

$$\mathbf{Z}_a = (\mathbf{z}_1^a \cdots \mathbf{z}_{N_A}^a), \quad \mathbf{z}_k^a = \mathbf{R}^{-1/2}\mathbf{H}_a\mathbf{f}_k, \quad \mathbf{H}_a = \left(\frac{\partial h}{\partial \mathbf{x}}\right)_{\mathbf{x}^a}. \tag{2.30}$$

As argued earlier for the Hessian preconditioning, one can also consider using the finite difference formulation $\mathbf{z}_k^a = \mathbf{R}^{-1/2}[h(\mathbf{x}^a + \mathbf{f}_k) - h(\mathbf{x}^a)]$ instead of (2.30). After calculating the optimal control vector $\mathbf{w}^a$ from iterative minimization (2.22) we apply the change of variable (2.16) to obtain the optimal state:

$$\mathbf{x}^a = \mathbf{x}^f + \mathbf{Fw}^a, \tag{2.31}$$

with analysis uncertainty implied from (2.29).

c. Ensemble update

To define the ensemble initial conditions used to evolve the analysis uncertainty to the next assimilation cycle, it is convenient to define a square root analysis error covariance matrix. After applying Cholesky decomposition,

$$\mathbf{I} + \mathbf{Z}_a^T\mathbf{Z}_a = \mathbf{G}_a\mathbf{G}_a^T, \tag{2.32}$$

where $\mathbf{G}_a$ is the unique lower triangular matrix, the square root analysis error covariance is

$$\mathbf{P}_a^{1/2} = \mathbf{FG}_a^{-T}, \tag{2.33}$$

with columns $[\mathbf{p}_i^a \in S : (i = 1, \ldots, N_A)]$. Note that $\mathbf{P}_a^{1/2} : A \to S$, implying that there are N_A columns of this matrix. However, the number of ensembles that needs to be evolved, i.e., the number of desired ensemble perturbations, is N_E.

The approach taken is the resampling of posterior perturbations used in the implicit particle filter (IPF; Chorin and Tu 2009; Chorin et al. 2010; Morzfeld et al. 2012), as well as in EVIL-R (Auligne et al. 2016):

$$\mathbf{x}_i = \mathbf{x}^a + \mathbf{P}_a^{1/2}\boldsymbol{\theta}_i \quad (i = 1, \ldots, N_E), \tag{2.34}$$

where $\mathbf{x}_i$ is the ith initial state for ensemble forecasting, and $\boldsymbol{\theta} \sim N(0, I)$ is a Gaussian random variable in analysis subspace ($\boldsymbol{\theta}_i \in A$). Note that the matrix–vector product $\mathbf{P}_a^{1/2}\boldsymbol{\theta}_i$ implies that all columns of the square root matrix are combined, with the coefficients of linear combination defined by random vectors:

$$\mathbf{P}_a^{1/2}\boldsymbol{\theta}_i = (\boldsymbol{\theta}_i)_1\mathbf{p}_1^a + \cdots + (\boldsymbol{\theta}_i)_{N_A}\mathbf{p}_{N_A}^a. \tag{2.35}$$

In addition to being algorithmically simple, this strategy may be especially beneficial when $N_E \ll N_A$, which will be the case in realistic high-dimensional applications.

Using (2.33) and (2.34) one must compute $\mathbf{FG}_a^{-T}\boldsymbol{\theta}_i$. The first part is to solve $\mathbf{G}_a^{-T}\boldsymbol{\theta}_i = \boldsymbol{\gamma}_i$ for the unknown $\boldsymbol{\gamma}_i$, which can be done by solving the triangular linear system

$$\mathbf{G}_a^T\boldsymbol{\gamma}_i = \boldsymbol{\theta}_i \tag{2.36}$$

for each realization $\boldsymbol{\theta}_i$. Finally, the ensemble initial state is

$$\mathbf{x}_i = \mathbf{x}^a + \mathbf{F}\boldsymbol{\gamma}_i. \tag{2.37}$$

The expression (2.37) updates all points, even if there are no observations nearby, consequently introducing a noise even in regions that are void of observations. This can have adverse impact on dynamical balances for sparsely observed systems. However, from (2.35)–(2.37) it follows that ensemble perturbations are defined in the same subspace as the square root forecast error covariance $\mathbf{F}$, which may help in constraining the introduced noise. The column vectors $\mathbf{F}\boldsymbol{\gamma}_i$ form the square root posterior covariance in the ensemble subspace E:

$$(\mathbf{P}_E^a)^{1/2} = (\mathbf{F}\boldsymbol{\gamma}_1 \cdots \mathbf{F}\boldsymbol{\gamma}_{N_E}). \tag{2.38}$$

d. Forecast

After completing the analysis and calculating the ensemble initial conditions update one proceeds by evolving in time the analysis and the analysis uncertainty. The initial guess (background) in the next analysis cycle is

$$\mathbf{x}^f = m(\mathbf{x}^a), \tag{2.39}$$

where now $\mathbf{x}^f$ refers to background vector in the *next* analysis cycle and m is a nonlinear prediction model. The forecast uncertainty in ensemble space is obtained as a difference between nonlinear forecasts

$$\mathbf{p}_i = m(\mathbf{x}^a + \mathbf{F}\boldsymbol{\gamma}_i) - m(\mathbf{x}^a) \quad (i = 1, \ldots, N_E) \tag{2.40}$$

forming the ensemble forecast error covariance [e.g., (2.1) and (2.2)] in the next cycle.

e. Algorithmic details

The MLEF-SSL equations described in previous sections are designed to maximize the efficiency of calculations in high-dimensional applications. In this section we will discuss algorithmic steps that form the MLEF-SSL algorithm. There is a

preparatory step and the actual calculations in each data assimilation cycle.

For preparatory calculations the localizing matrix $\mathbf{L}$ must be defined first. Due to efficiency of storing its elements and applying a matrix–vector product, we use a banded Toeplitz matrix with bandwidth corresponding to the localization length. In multivariate case this matrix would become a block symmetric matrix with each block being a banded Toeplitz. Second, choose the reduced-rank basis vectors and the dimension N_{RR}. In the current version of the MLEF-SSL system this implies calculating $\mathbf{s}_n = \mathbf{L}^{1/2}\mathbf{r}_n$ using random vectors. In high-dimensional applications it is recommended to make these calculations parallel and store only the local subset of vectors $\mathbf{s}_n$ on the current processor.

The index j refers to minimization iteration, index k to augmented ensemble, and index i to dynamical ensembles.

1) PREPARATORY CALCULATION

- $\mathbf{s}_n = \mathbf{L}^{1/2}\mathbf{r}_n(n = 1, \ldots, N_{RR})$

2) MLEF-SSL ALGORITHM

- Columns of a square root of a low-rank (N_{RR}) localized covariance matrix

$$\mathbf{f}_k = \mathbf{D}_i\mathbf{s}_n$$

- Initial iteration: $j = 0$

  Choose the starting point of minimization $\mathbf{x}_0 = \mathbf{x}^f$ implying $\mathbf{w}_0 = 0$
  $\mathbf{Z}f$: $\mathbf{Z}_f$: $z_k^f = \mathbf{R}^{-1/2}\mathbf{H}_f\mathbf{f}_k$
  Hessian and Cholesky: $\mathbf{Q} = \mathbf{I} + \mathbf{Z}_f^T\mathbf{Z}_f = \mathbf{GG}^T$
  Gradient: $\mathbf{g} = \mathbf{w} - \mathbf{Z}_f^T\mathbf{R}^{-1/2}[\mathbf{y} - h(\mathbf{x}^f + \mathbf{Fw})]$
  Solve for $\mathbf{d}_0$: $(\mathbf{GG}^T)\mathbf{d}_0 = -\mathbf{g}$
  Nonlinear line-search $[\alpha_0]$
  Update control: $\mathbf{w}_1 = \mathbf{w}_0 + \alpha_0\mathbf{d}_0$
  Update state: $\mathbf{x}_1 = \mathbf{x}_0 + \mathbf{Fw}_1$

  repeat

- $j = 1, \ldots, j_{max} = 0 - 1$:

  Gradient: $\mathbf{g}_{j+1} = \mathbf{w}_j - \mathbf{Z}_f^T\mathbf{R}^{-1/2}[\mathbf{y} - h(\mathbf{x}^f + \mathbf{Fw}_j)]$
  Solve for $\mathbf{d}_k$ [optimization algorithm dependent]
  Nonlinear line-search $[\alpha_j]$
  Update control variable: $\mathbf{w}_{j+1} = \mathbf{w}_j + \alpha_j\mathbf{d}_j$
  Update state variable: $\mathbf{w}_{j+1} = \mathbf{x}_j + \mathbf{Fw}_{j+1}$
  until convergence

- Analysis: $\mathbf{x}^a = \mathbf{x}_{jmax}$
- Ensemble update: $\mathbf{x}_i = \mathbf{x}^a + \mathbf{P}_a^{1/2}\boldsymbol{\theta}_i$

In high-dimensional problems it is recommended to make all these calculations parallel and store only the local subsets of vectors and matrices on the current processor. For example, the parallel flow of the algorithm could look like this:

- Local set of $\mathbf{f}_k$ column vectors is kept on a given CPU.
- Local calculations on the same CPU are done for $\mathbf{z}_k^f$ and $\mathbf{z}_k^a$.
- Local elements of Hessian are computed in preparation for parallel Cholesky.

- Parallel Cholesky produces only local elements of matrix **G**. Solution of the triangular linear system is also available locally, implying that only local gradients and descent directions are needed.
- Basic MPI reduction operation is needed to collect local information for line search.
- Local updates of the control and the state vectors.
- Analysis and ensemble update are local as well.

Highly efficient parallel Cholesky algorithms that store only the local matrix (e.g., using two-dimensional block-cyclic matrix distribution) are available for free download (e.g., ScaLAPACK, http://www.netlib.org/scalapack/, with CPU and GPU options).

3. Experimental design

a. Model

The prediction model used in the experiments is the model II of Lorenz (2005). The model configuration is similar to that one used by Bishop et al. (2015) and Huang et al. (2019). There are 240 model points ($N_S = 240$) with the smoothing parameter $K = 8$ and the forcing parameter $F = 15$. The model is applied with periodic boundary conditions. If the model domain is thought as a latitude circle, the model spatial resolution would be 1.5°. Time integration of the model employs the fourth-order Runge–Kutta scheme, and the time step is $dt = 0.025$ nondimensional units.

The model setup includes (i) warm-up period of 960 time steps starting from random initial conditions, followed by (ii) climatology run of 800 time steps, to be used for estimating the statistics of forecast errors. After completing the total of 1760 time steps the last model forecast was used to define the initial conditions for the "true" prediction.

b. Observations

Observations are created by adding Gaussian random perturbations to the "true" model run. Given the observation error standard deviation σ_O and a Gaussian random number $\varepsilon \sim N(0, \sigma_O)$ as observation error, the observation is defined as

$$\mathbf{y} = h(\mathbf{x}_{\text{true}}) + \varepsilon, \tag{3.1}$$

where $\mathbf{x}_{\text{true}}$ denotes the forecast from the true run. The observation error standard deviation is defined as 20% of the climatological forecast error: $\sigma_O = 0.2\sigma_{\text{clim}}$, where σ_{clim} denotes the climatology error standard deviation. In our experiments $\sigma_{\text{clim}} = 6.29$, which implies the observation error standard deviation $\sigma_O = 1.258$.

The observation operator h consists of two mappings: h_1 and h_2, i.e., $h = h_2 h_1$. The operator h_1 is designed to take as observation (i) every mth grid point (e.g., pointwise operator), or (ii) the mean of 12 grid points (e.g., integrated operator). For point observations this means that every mth grid point is chosen to be the observation point:

$$[h_1(\mathbf{x}_{\text{true}})]_m = (\mathbf{x}_{\text{true}})_m, \tag{3.2}$$

while for integrated observations we define the sample mean at every mth gridpoint location as observation:

$$[h_1(\mathbf{x}_{\text{true}})]_m = \frac{1}{12} \sum_{k=m}^{k+12} (\mathbf{x}_{\text{true}})_k. \tag{3.3}$$

In our experiments we use every sixth grid point as observation location. The second component h_2 defines the transformation. Two transformations are considered in our experiments: (i) linear and (ii) hyperbolic tangent as an example of nonlinear transformation. The hyperbolic tangent function used in the experiments is defined as

$$\mathbf{y} = a \tanh(b\mathbf{y}_1), \tag{3.4}$$

where $\mathbf{y}_1 = h_1(\mathbf{x}_{\text{true}})$, $a = 20$, and $b = 0.08$. For calculation of matrix **Z** (2.22) the first derivative of this function will be needed, which is

$$\frac{\partial \mathbf{y}}{\partial \mathbf{y}_1} = \frac{ab}{[\cosh(b\mathbf{y}_1)]^2}, \tag{3.5}$$

with cosh denoting the hyperbolic cosine function. In case of integrated observations one also needs to include the tangent linear of (3.3). Since (3.3) is linear, its tangent linear has the same form.

c. Data assimilation setup

The observations are assimilated over 200 data assimilation cycles, each with assimilation window of 16 model time steps. This amounts to 3200 model time steps in data assimilation mode.

The initial state for data assimilation experiments is chosen to be shifted by 40 model time steps from the true initial conditions $\mathbf{x}_{\text{exp}}^0 = \mathbf{x}_{\text{true}}(t_0 - 40\,dt)$, where t_0 denotes the initial time of the true forecast run.

The initial uncertainty for data assimilation experiments is defined using lagged forecast method previously used in MLEF (e.g., Suzuki et al. 2017), in which a deterministic forecast centered at the experiment start time is integrated. The columns of the initial square root error covariance are chosen as a scaled difference between model outputs equally spaced over the integration period and the central forecast corresponding to the experiment start time.

d. Experiments

There are two main groups of experiments: 1) Basic evaluation of the MLEF-SSL performance, and 2) Sensitivity of the MLEF-SSL performance with respect to the choice of basis vectors $\{\mathbf{u}_n\}$ that span the low-dimensional subspace and to the dimension of the reduced-rank subspace N_{RR}.

Basic experiments include a comparison of MLEF-SSL with two referent systems: (i) MLEF with observation space localization (MLEF-OBS), and (ii) ensemble Kalman filter (EnKF) with state space localization (EnKF-SSL). As additional reference we also compute the no-observation experiment, which is essentially a long forecast integrated over the length of 200 data assimilation cycles, starting from the same initial

conditions as other data assimilation experiments. Since there is no data assimilation in such an experiment, we refer to it as the NODA experiment.

MLEF-OBS is developed from the original MLEF algorithm (Zupanski 2005; Zupanski et al. 2008) by including covariance localization in observation space as described by Hunt et al. (2007) and Miyoshi and Yamane (2007). Further mathematical details of the MLEF-OBS algorithm are given in appendix A. The implemented EnKF-SSL closely follows the EnKF with perturbed observations described in Houtekamer and Mitchell (1998, 2001) and Evensen (2003), however, using the forecast error covariance matrix localized in state space. It should be also mentioned that EnKF-SSL employs a linearized observation operator. Additional details of the EnKF-SSL algorithm are given in appendix B.

In all experiments we define 40 observations per cycle, i.e., $N_O = 40$. The number of ensembles is 10 for MLEF-SSL and MLEF-OBS ($N_E = 10$), while EnKF-SSL has 11 ensembles ($N_E = 11$). The additional ensemble member in EnKF-SSL compensates for the loss of one degree of freedom due to using the sample mean in calculation of the sample covariance. Each of the experiments includes 8 different sets of random seeds in order to reduce the impact of the random seed choice (Janjić et al. 2011; Bishop et al. 2015; Huang et al. 2019).

The banded Toeplitz matrix $\mathbf{L}$ is used for covariance localization, with elements calculated using the compactly supported fifth-order piecewise rational function defined in Gaspari and Cohn (1999).

The covariance inflation method developed by Zhang et al. (2004) is adopted for all data assimilation systems used in this paper, often referred to as the relaxation to prior perturbation (e.g., Kotsuki et al. 2017; Gharamti 2018):

$$(\mathbf{P}_a^{1/2})_{\text{new}} = \gamma \mathbf{P}_f^{1/2} + (1 - \gamma)\mathbf{P}_a^{1/2}, \quad (3.6)$$

where $\gamma \in [0, 1]$ is the inflation parameter that multiplies prior perturbation.

For all considered data assimilation systems and observation operators the optimal localization length and covariance inflation parameters are estimated empirically using trial-and-error, described in detail in appendix C. In all data assimilation systems the tuned localization length is equal to 12 grid points. Optimal inflation parameters were found to be (i) $\gamma = 0.7$ for EnKF, (ii) $\gamma = 0.1$ for MLEF-OBS, and (iii) $\gamma = 0$ for MLEF-SSL implying no inflation.

Two types of observation operators are considered: (i) linear, and (ii) nonlinear, defined using the hyperbolic tangent function. In addition, the nonlinear observation operator can be pointwise or integrated, as discussed in section 3b. Note that the linear observation operator experiment is equivalent to standard square root ensemble filters. The nonlinear observation operator experiment includes minimization with the nonlinear preconditioned conjugate gradient algorithm (Gill et al. 1981; Luenberger 1984), with five iterations. The line search method is quadratic interpolation, calculated from three points (Nocedal and Wright 1999).

1) BASIC EVALUATION EXPERIMENTS

This group of experiments is designed to evaluate the basic performance of MLEF-SSL against MLEF-OBS and EnKF-SSL.

TABLE 1. Comparison of capabilities of the considered data assimilation algorithms with respect to nonlinearity and integrated observations.

	Nonlinear capability	Integrated observation capability
EnKF-SSL	No	Yes
MLEF-OBS	Yes/No (no global convergence of minimization)	No
MLEF-SSL	Yes	Yes

The experiments include using pointwise and integrated, linear and nonlinear observation operators. MLEF-SSL used for the comparisons includes 100 random basis vectors in calculation of the localized forecast error covariance, i.e., following the formulation (2.14) with $N_{\text{RR}} = 100$.

The basic evaluation experiments are focusing on two main challenges of a data assimilation system that incorporates consistent estimates of the analysis and its uncertainty: (i) nonlinearity and (ii) integrated observations. As discussed earlier, nonlinearity should be best addressed using globally convergent minimization, while integrated observations should benefit from using state-space localization. Table 1 describes how considered algorithms compare in that respect. Only the MLEF-SSL has capability to handle both challenges, while EnKF-SSL and MLEF-OBS can only partially address them. Between EnKF-SSL and MLEF-OBS, EnKF may have a slight advantage since MLEF-OBS does not include globally convergent optimization and also it is not equipped to handle integrated observations. The experiments using pointwise linear observation operator serve as a reference for systems' performance, since it is expected that all systems should perform similarly. The experiments with integrated linear observations evaluate how the systems handle the challenge of integrated observations without the additional burden of nonlinearity. To examine how the systems handle nonlinear observations, but avoiding the potential challenges of covariance localization, the experiments with pointwise nonlinear observations are conducted. Finally, the systems are evaluated in most likely scenario when observations are both nonlinear and integrated.

2) SENSITIVITY EXPERIMENTS

In this group of experiments several MLEF-SSL assimilations are conducted to assess the sensitivity to the chosen basis $\{\mathbf{u}_n\}$ of the reduced-rank subspace S_{RR}. In particular we consider truncated eigenvectors of the localizing matrix $\mathbf{L}$. The truncated eigenvectors of $\mathbf{L}$ seem to be a natural choice since this approach leads to simplification of the calculation of the matrix–vector product $\mathbf{L}^{1/2}\mathbf{u}_n$. This was also the choice for the forecast error covariance specification in *modulated* ensemble approach (e.g., Leng et al. 2013; Bishop et al. 2015). The truncated eigenvector option introduces higher algorithmic complexity and requires more computations than the random basis option due to calculation of eigenvalue decomposition (EVD). Given that realistic state dimensions are on the order of 10^8, $\mathbf{L}$ is an $O(10^8) \times O(10^8)$ matrix and calculating its EVD
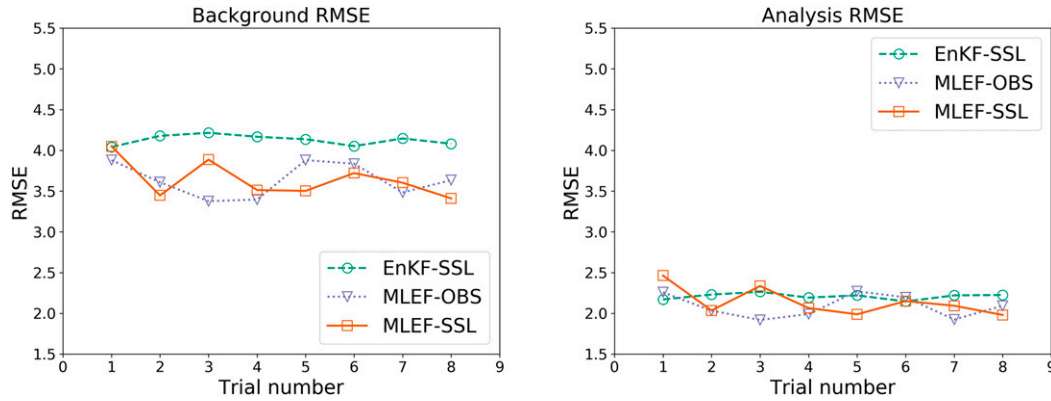
FIG. 1. RMSE results with pointwise linear observation operator for (left) the background and (right) the analysis. The dashed green line with circles represents EnKF-SSL, the dotted blue line with triangles represents MLEF-OBS, and MLEF-SSL is represented by the solid orange line with square markers.

is, therefore, quite challenging in high-dimensional applications. However, since $\mathbf{L}$ is predefined and practically a constant matrix, in principle its EVD could be precomputed. In applications to low-dimensional Lorenz model there are no computational issues, so we use it as an alternative to random basis vectors.

Since $\mathbf{L}$ is real and symmetric the EVD is

$$\mathbf{L} = \mathbf{V}\mathbf{\Lambda}\mathbf{V}^{\mathrm{T}} = \sum_{n=1}^{N_S} \boldsymbol{\lambda}_n \mathbf{v}_n \mathbf{v}_n^{\mathrm{T}}, \tag{3.7}$$

where $[(\boldsymbol{\lambda}_n, \mathbf{v}_n): n = 1, \ldots, N_S]$ are the pairs of eigenvalues and eigenvectors, respectively. The truncation up to N_{RR} vectors produces an eigenvector basis $(\mathbf{v}_n: n = 1, \ldots, N_{\mathrm{RR}})$ of the low-dimensional subspace. Since the square root matrix is $\mathbf{L}^{1/2} = \sum_{n=1}^{N_S} \boldsymbol{\lambda}_n^{1/2} \mathbf{v}_n \mathbf{v}_n^{\mathrm{T}}$ the matrix–vector product $\mathbf{L}^{1/2}\mathbf{v}_n = \boldsymbol{\lambda}_n^{1/2}\mathbf{v}_n$ implies that the columns of the square root localized forecast error covariance are

$$\mathbf{f}_k = \mathbf{D}_i(\boldsymbol{\lambda}_n^{1/2}\mathbf{v}_n) \quad (k = 1, \ldots, N_E \times N_{\mathrm{RR}}). \tag{3.8}$$

Each of the two methods is evaluated for the following dimensions of the reduced-rank subspace: (i) $N_{\mathrm{RR}} = 10$, (i) $N_{\mathrm{RR}} = 20$, (iii) $N_{\mathrm{RR}} = 100$, and (iv) $N_{\mathrm{RR}} = 500$. All these experiments are conducted using the nonlinear observation operator, deemed more challenging than using the linear observation operator.

4. Results and discussion

a. Basic evaluation of MLEF-SSL

As explained in section 3 we compare the results of MLEF-SSL with random sample of 100 against MLEF-OBS and EnKF-SSL.

1) LINEAR OBSERVATION OPERATOR

(i) Point observations

The results with linear pointwise observation operator are compared first. For each set of random seeds we compute the total root-mean-squared error (RMSE) with respect to truth,

over all 200 data assimilation cycles and all state points. In Fig. 1 we show the RMSE for each set of random seeds, for the background (Fig. 1, left panel) and the analysis (Fig. 1, right panel). The three lines describe the RMSE from the EnKF-SSL, MLEF-OBS, and MLEF-SSL experiments, the background RMSE, and the analysis RMSE. For reference, the NODA RMS is approximately equal to 9 (not shown). The background RMSE is the error of the short-range forecast between two cycles of the length of data assimilation window (e.g., 16 model time steps). It indicates the impact of data assimilation on short range forecast. The analysis RMSE is also shown since it shows the impact of observations in assimilation. From the perspective of numerical weather prediction (NWP) more relevant is the background RMSE than the analysis RMSE. This is because it is generally known that achieving good fit to observations, although desirable, does not necessarily guarantee the improvement of the forecast. All data assimilation algorithms produce generally similar results. While the analysis RMSE is almost identical for all experiments, the background RMSE is comparable for MLEF-SSL and MLEF-OBS and slightly larger for EnKF-SSL. This seems to indicate that in this example the analysis of MLEF-SSL and MLEF-OBS captures the growing perturbations somewhat better than EnKF-SSL. However, one can also notice that EnKF-SSL has much less variability of RMSE compared to other two algorithms, which seem to imply a more robust performance of EnKF-SSL.

We further examine the performance by comparing the analysis skill and spread (Fig. 2) for the trial number 4. The skill is defined as analysis RMSE, while the spread is defined as the analysis error standard deviation. The RMSE defines the actual difference from truth, while the spread reflects what data assimilation algorithm "thinks" about the error. One would like these the two measures to be comparable.

Even though the RMSE and the analysis error standard deviation are comparable in general, one can notice that spreads in EnKF-SSL (Fig. 2, left panel) and MLEF-SSL (Fig. 2, right panel) are slightly overestimated, while in the MLEF-OBS experiment (Fig. 2, middle panel) the spread is adequate. Although this varies depending on the trial number, it suggests that ensemble perturbations calculated in MLEf-SSL using
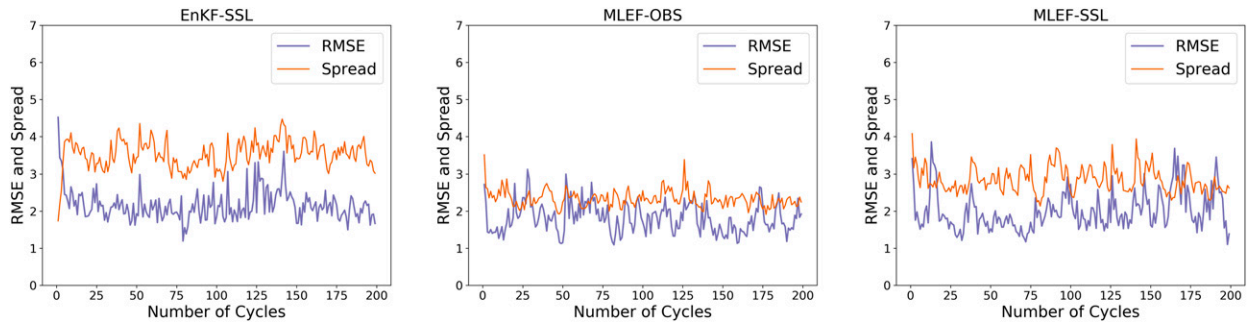
FIG. 2. RMSE and analysis error standard deviation in (left) EnKF-SSL, (middle) MLEF-OBS, and (right) MLEF-SSL experiments. RMSE is represented by the blue line, while the standard deviation is represented by the orange line. The horizontal axis represents data assimilation cycles.

resampling of the analysis perturbations introduce sufficient growth of errors in the forecast. This is achieved without a need for covariance inflation in MLEF-SSL, implying a need for better understanding of the impact of resampling and warrants further investigation in the future. Overall, the spread is equal or somewhat larger than skill in all experiments. These results are also summarized in Table 2 in terms of a time-averaged spread to skill ratio. There is no noticeable trend of either RMSE or spread, suggesting a stable performance of all algorithms.

(ii) Integrated observations

The results with linear integrated observation operator are compared next, as described by (3.3). Similarly to point observations, we show in Fig. 3 the RMSE for each set of random seeds, for the background (Fig. 3, left panel) and the analysis (Fig. 3, right panel). Since MLEF-OBS is not suited for integrated observations its RMSE shows the largest values among the three algorithms. The RMSE in MLEF-SSL is generally smaller than in EnKF-SSL. While the analysis RMS in MLEF-SSL has some comparable performances to EnKF-SSL for trials 1 and 2 (Fig. 3, right panel), the background RMSE in MLEF-SSL is clearly better than EnKF-SSL. This may again suggest that growing perturbations are better captured in the MLEF-SSL analysis.

The analysis skill and spread for trial number 4 can be found in Table 2, summarized in terms of a time-averaged spread to skill ratio. EnKF-SSL has the best match, while both MLEF-OBS and MLEF-SSL overestimate the spread. The overestimation of spread in MLEF-SSL, that was also noticed for point observations (Fig. 2, bottom panel), may be a consequence of ensemble update by resampling, but this would require more investigation before it can be confirmed. Although the cause of overestimated spread in MLEF-OBS is likely due to its reduced capability to assimilate integrated observations, the impact of overestimation on the performance was negative, unlike in MLEF-SSL.

2) NONLINEAR OBSERVATION OPERATOR

The more interesting results is the comparison using nonlinear observation operator. This is because the numerical optimization is performed globally (e.g., over all points) in MLEF-SSL, but can be done only locally in MLEF-OBS and there is no minimization in EnKF-SSL.

(i) Point observations

As done for the linear observation operator, for each set of random seeds we compute the total root mean squared error (RMSE) with respect to truth, over all 200 data assimilation cycles and all state points. In Fig. 4 we show the RMSE for each set of random seeds, for the background (Fig. 4, left panel) and for the analysis (Fig. 4, right panel). MLEF-SSL produces best results over all trial runs, for both the analysis and the forecast. MLEF-OBS and EnKF-SSL have mixed results: EnKF-SSL produces smaller RMSE of the analysis, but slightly larger RMSE of the forecast than MLEF-OBS. Recall that both algorithms have limitations in application to nonlinear observations: EnKF-SSL employs a linearized observation operator, while MLEF-OBS includes localized minimization that is consequently not globally convergent. It is also interesting to note for MLEF-OBS and EnKF-SSL that apparently a smaller analysis RMSE implies a larger forecast RMSE. A possible interpretation is that imperfections of the two algorithms (e.g., linearized observation operator in EnKF-SSL and only a locally convergent minimization in MLEF-OBS) create unrealistic adjustments in the analysis resulting in larger growth of errors for the analysis with smaller RMSE. On the other hand, since MLEF-SSL has a nonlinear capability and a globally convergent minimization, the analysis with smaller RMSE also produces a smaller forecast RMSE. The time-averaged spread to skill ratio is shown in Table 2. MLEF-SSL produces a relatively good spread to skill ratio slightly larger than one. At the same

TABLE 2. Time-averaged spread to skill ratio from the trial experiment 4. The results for the pointwise-linear observation operator correspond to Fig. 2.

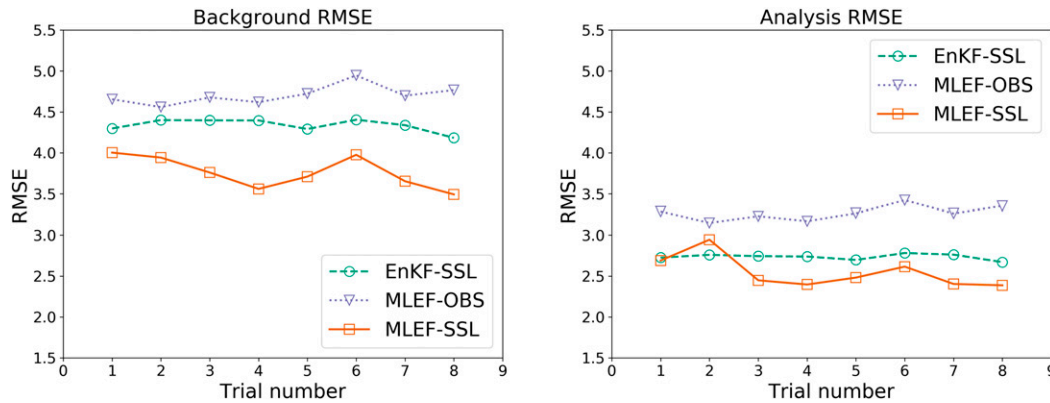| Observation operator	EnKF-SSL	MLEF-OBS	MLEF-SSL
Linear-pointwise	1.61	1.26	1.38
Linear-integrated	1.25	1.54	1.73
Nonlinear-pointwise	1.68	0.82	1.14
Nonlinear-integrated	1.47	1.17	1.45

Fig. 3. As in Fig. 1, but for the RMSE of the integrated linear observation operator.

time EnKF-SSL produces much larger spread to skill ratio, while MLEF-OBS has the ratio smaller than one.

The minimization of cost function in MLEF-OBS and MLEF-SSL experiments (EnKF-SSL does not include minimization of cost function) is relatively similar (not shown). From the RMSE scores (Fig. 4), a better fit to observations achieved by the MLEF-OBS cost function did not necessarily transfer into an improved short-range forecast. Although it is expected that local cost function in MLEF-OBS can be reduced more than global cost function in MLEF-SSL, this reduction must be dynamically balanced in order to result in forecast improvement.

(ii) Integrated observations

The RMSE for each set of random seeds is shown in Fig. 5, for the background (Fig. 5, left panel) and for the analysis (Fig. 5, right panel). There is a clear distinction between algorithms in this situation. The MLEF-SSL produces the smallest RMSE, for both the background and the analysis. The EnKF-SSL follows with somewhat larger RMSE, but MLEF-OBS is clearly underperforming due to its inability to extract information from integrated observations. The distinction between RMSE is more apparent than for the linear integrated

observation operator (Fig. 3), presumably because of the nonlinear advantage of MLEF-SSL over the other two algorithms, in addition to handling integrated observations due to state-space covariance localization. The time-averaged spread to skill ratio in Table 1 shows generally larger values for EnKF-SSL and MLEF-SSL than for the MLEF-OBS, but all values are greater than one implying an overestimation of the spread.

In conclusion, the results show that all systems have stable performances and that the analysis RMSE is generally smaller than the background RMSE, as expected in successful data assimilation. All results are summarized in Table 3. MLEF-SSL shows an improved performance over the other two methods, especially for integrated nonlinear observations. Inability of observation space localization to address integrated observations is clearly noticed in the MLE-OBS performance. The successful performance of the EnKF-SSL and MLEF-SSL algorithms in assimilation of integrated observations seem to suggest that correct treatment of integrated observations through state space localization is critical. There is also a general agreement between RMSE (skill) and the analysis standard deviation (spread) for all systems, although most often the spread is overestimated. Overall, the MLEF-SSL system produces better results in terms of RMSE than EnKF-SSL and MLEF-OBS.
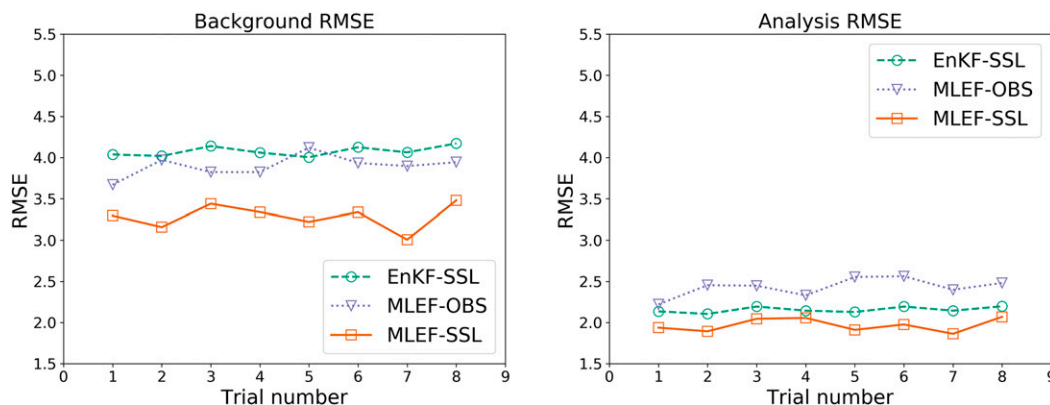

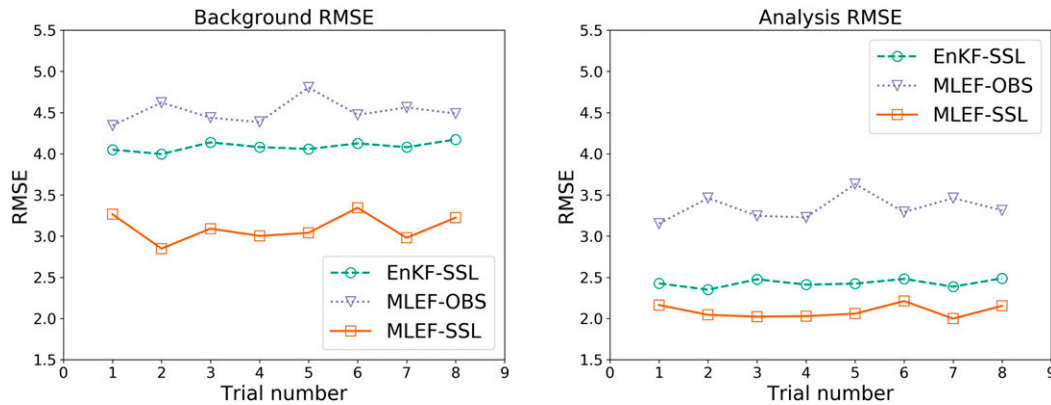Fig. 4. As in Fig. 1, but for the RMSE of the pointwise nonlinear observation operator.

FIG. 5. As in Fig. 1, but for the RMSE of the integrated nonlinear observation operator.

b. Sensitivity of MLEF-SSL performances

In this subsection we focus on the MLEF-SSL system only, with nonlinear observation operator. As in earlier experiments we conduct 200 data assimilation cycles with 8 different sets of random seeds. As explained in section 3d(2) we are interested in learning about the impact of the method used to define the approximate identity matrix, which is closely related to the dimension of the reduced-rank subspace.

First, we show the overall performance of various MLEF-SSL experiments for each trial, in terms of the total RMSE for the background and the analysis (Fig. 6). As expected, RMSE are in general much larger for smaller dimension of the reduced-rank subspace. Most notable differences in terms of RMSE can be seen for $N_{RR} = 10$ (Fig. 6, top-left panel), while for $N_{RR} = 20$ (Fig. 6, top-right panel), $N_{RR} = 100$ (Fig. 6, bottom left panel) and $N_{RR} = 500$ (Fig. 6, bottom-right panel) the RMSE differences are minor. Focusing on the background RMSE (blue lines with circle markers) one can see that for $N_{RR} = 10$ (Fig. 6, top-left panel) the random basis option (solid line) produces better results than truncated EVD (dotted line). There is no obvious advantage of either method for $N_{RR} = 20$ (Fig. 6, top-right panel), $N_{RR} = 100$ (Fig. 6, bottom-left panel), and for $N_{RR} = 500$ (Fig. 6, bottom-right panel). The same general conclusion can be drawn for the analysis RMSE.

To better understand the reasons for such behavior one has to consider the range of the localizing matrix **L** and the fact that random basis approaches orthogonality only for large samples. We are considering only the range of **L** (as opposed to the null space), since for zero eigenvalues the columns of localized error covariance (3.8) become zero and, therefore, have no impact in data assimilation. In Fig. 7 we show the eigenvalues of matrix **L**, with markers defining the truncation for different values of N_{RR}. Given the relationship between eigenvalues of a real symmetric banded Toeplitz matrix and a circulant matrix (Arbenz and Golub 1988), only the largest and the smallest eigenvalues of **L** are simple, while all other eigenvalues are defined in pairs (Arbenz 1991). This explains small "steps" seen in Fig. 7.

The dimension of the range can be estimated numerically as the number of eigenvalues larger than zero (e.g., Golub and van Loan 1989), in this case approximately equal to 35.

The true rank may be somewhat larger, since there are additional very small nonzero eigenvalues that are difficult to distinguish from Fig. 7. Therefore, for $N_{RR} = 10$ the range of **L** is severely underestimated, implying the difficulty of the EVD approach with such a small number of eigenvectors. The random basis also has difficulties with such small basis dimension, indicated by relatively large values of RMSE, as seen in Fig. 6 (top left panel). Although for $N_{RR} = 20$ the larger eigenvalues are still not fully captured, the range is well approximated implying acceptable results of the EVD approach. Even though the random basis may still have issues with such a small sample, the results are comparable to EVD (Fig. 6, top right panel). Finally, for $N_{RR} = 100$ and $N_{RR} = 500$ we can assume that the EVD approach captures the full range, producing even more improved RMSE and getting closer to its optimal performance. The random basis can still improve since even 500 random vectors may not be sufficient to accurately represent the relevant subspace. However, the RMSE results from EVD and random basis approaches are becoming much closer (Figs. 6, bottom row).

We further analyze the impact of the reduced-rank subspace generation method as a function of dimension N_{RR}, by plotting the central column (e.g., at $N_S = 120$) of the "static" matrix $\sum_n (\mathbf{L}^{1/2}\mathbf{u}_n)(\mathbf{L}^{1/2}\mathbf{u}_n)^T$, where for $\mathbf{u}_n$ we use the random basis and the truncated eigenvectors of **L**. This is represented in Fig. 8. Note that these plots can be also interpreted as a response to single observation located at the center of model domain, with matrix $\sum_n (\mathbf{L}^{1/2}\mathbf{u}_n)(\mathbf{L}^{1/2}\mathbf{u}_n)^T$ as the forecast error covariance. It is

TABLE 3. Summarized basic evaluation experiments. The performance of the algorithms is graded using numbers from 1 to 3, with 1 denoting the best performance and 3 denoting the worse performance.

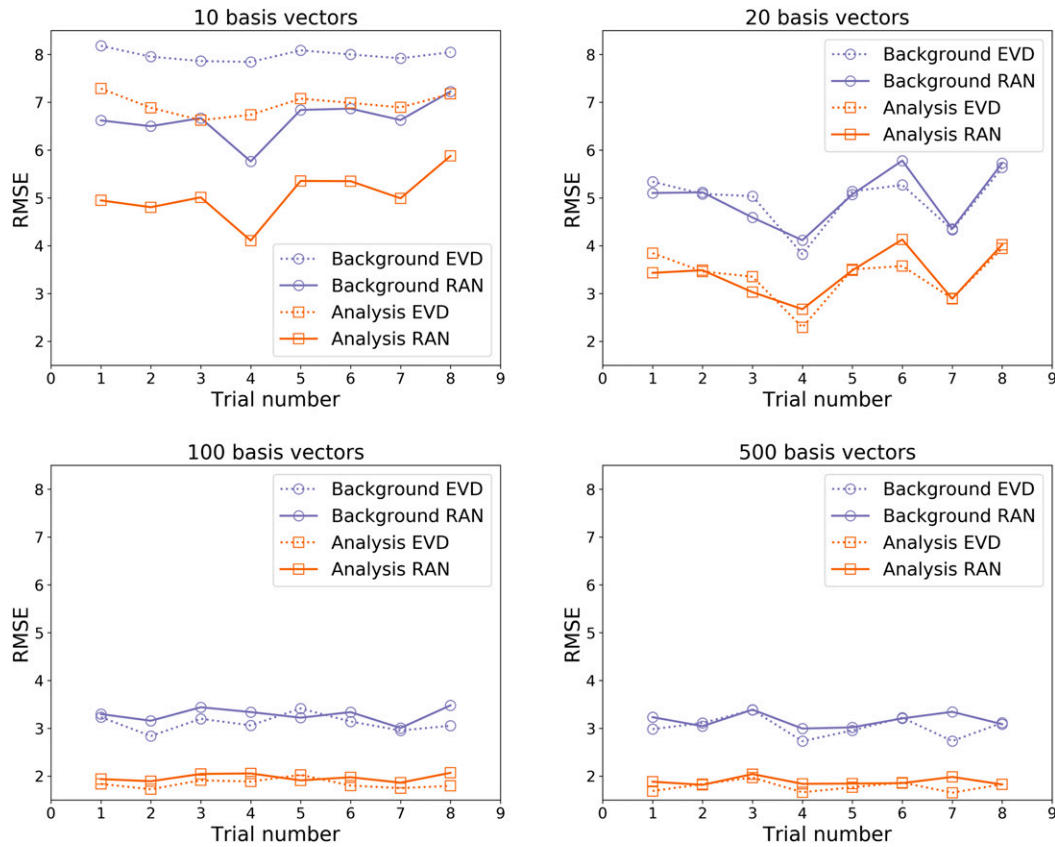|	Pointwise observation operator		Integrated observation operator	
	Linear	Nonlinear	Linear	Nonlinear
EnKF-SSL	1–3	2–3	2	2
MLEF-OBS	1–3	2–3	3	3
MLEF-SSL	1–3	1	1	1

FIG. 6. MLEF-SSL RMSE from eight trial runs with (top left) $N_{RR} = 10$, (top right) $N_{RR} = 20$, (bottom left) $N_{RR} = 100$, and (bottom right) $N_{RR} = 500$. The dotted line denotes the experiments using the truncated EVD, while the solid line denotes the experiments that use random basis for the localized forecast error covariance. The background RMSE are in blue color with circle markers, while the analysis RMSE are in orange color with square markers.

apparent that truncated eigenvectors show better response than random basis vectors. This confirms our earlier discussion that suggests that the eigenvectors of **L** are more efficient in representing the reduced-rank subspace. Even for severely truncated case $N_{RR} = 10$ the truncated eigenvectors show a better response than the random basis vectors (Fig. 8, top row). The response from the eigenvectors is considerably improved for $N_{RR} = 20$ (Fig. 8, second row, left panel), while the random basis response is still questionable (Fig. 8, second row, right panel). The eigenvectors show an almost perfect localized response at $N_{RR} = 100$ (Fig. 8, third row, left panel), while even for $N_{RR} = 500$ the response from random basis vectors is still not completely localized (Fig. 8, bottom row, right panel). However, we also notice that even with a less-than-perfect response of the random basis vectors (Fig. 8, third row, right panel) the RMSE results are comparable with the eigenvectors approach.

The above analysis suggests that random basis vectors and truncated eigenvectors of **L** are comparable in application to low-dimensional Lorenz model. The only difference comes for very small dimensions of the low-dimensional subspace ($N_{RR} = 10$), but this may not be relevant for realistic high-dimensional

applications. However, it is still not clear if a feasible truncation will be sufficient in realistic high-dimensional case. One can expect that in that situation the dimension of the reduced-rank subspace is significantly smaller than the dimension of the state space, even though it could be of the order of thousands or more. In that situation the random vector basis is almost orthogonal, therefore, efficient in describing the spanned subspace, while the truncated eigenvectors may still severely underestimate the range of **L**. Given the algorithmic simplicity of random basis implementation and comparable performance to truncated eigenvectors of **L** in presented low-dimensional applications, it seems that random basis may be a satisfactory choice for preliminary implementations of MLEF-SSL in realistic high-dimensional systems.

5. Summary and conclusions

MLEF-SSL method for nonlinear ensemble data assimilation with state space localization is presented. This system maintains some of the characteristics of the original MLEF, such as the use of global numerical optimization of cost function to find the analysis and estimating the posterior error
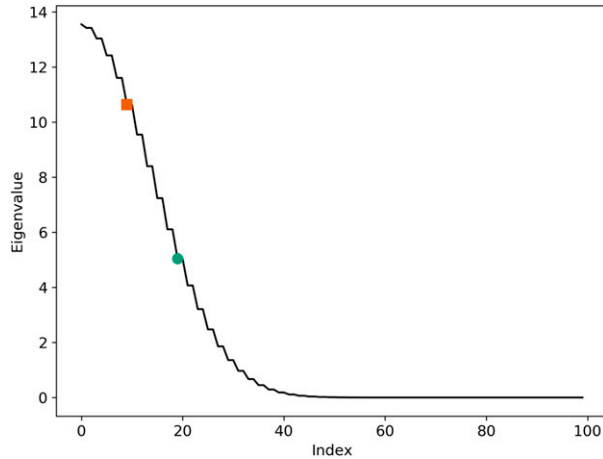
FIG. 7. Eigenvalues of the localizing matrix **L**. The markers define the eigenvalue truncation for $N_{RR} = 10$ (orange square) and $N_{RR} = 20$ (green circle).

covariance from the inverse Hessian at the minimum. One of the unique features of MLEF-SSL is the explicit formulation of the localized forecast error covariance that incorporates basis vectors of a low-dimensional subspace of the state space to reduce the analysis subspace dimension. Such formulation allows some flexibility in choosing the feasible dimensions of the analysis subspace. For ensemble update the MLEF-SSL system employs the resampling of posterior perturbations, which entails to the analysis uncertainty in low-dimensional dynamical ensemble space defined as a random linear combination of all columns of the high-dimensional localized square root posterior error covariance.

The matrix rank of (2.6) is clearly limited by the rank of I_{RR}. This also implies that the posterior uncertainty estimation will be limited. Therefore, it remains to be seen how big the actual analysis dimension N_{RR} must be to get good results in practice.

One of the main computational savings in MLEF-SSL is achieved by performing offline the potentially expensive calculation of the matrix–vector product between the localization matrix and a basis vector, in the preparatory step of data assimilation. For given model dimensions, control variables, and localization length, this product is computed only once. In addition, most numerical operations in MLEF-SSL are embarrassingly parallel, therefore, greatly reducing the computational requirements in high-dimensional systems.

The MLEF-SSL was applied to the Lorenz model II. The results indicate that MLEF-SSL is generally performing better than MLEF-OBS and EnKF-SSL, particularly for integrated nonlinear observations. The ensemble update is creating sufficient spread even without covariance inflation. Regarding the choice of basis vectors, the use of random basis to define low-dimensional subspace appears to be comparable or advantageous to using truncated eigenvectors of localizing matrix **L**. In future development the MLEF-SSL system will be evaluated in realistic high-dimensional application.

APPENDIX A

MLEF with Observation Space Localization (MLEF-OBS)

Before describing the MLEF with observation space localization it would be helpful to introduce relevant notation. In localized approach one defines a local analysis point at each grid point. In multivariate problems there may be M state points defined at each local grid point, where M denotes the number of control variable types. This implies that one can distinguish between state space S (with dimension N_S) and grid space G (with dimension N_G). Clearly $N_G \leq N_S$. For simplicity of presentation we assume $N_S = N_G$. While in the original MLEF the analysis is obtained by minimizing global cost function, in localized approach the analysis is obtained by minimizing local cost function defined at each grid point.

In the original MLEF a global cost function is defined as follows:

$$J(\mathbf{x}) = \frac{1}{2}(\mathbf{x} - \mathbf{x}^f)^T \mathbf{P}_E^{-1}(\mathbf{x} - \mathbf{x}^f) + \frac{1}{2}[\mathbf{y} - h(\mathbf{x})]^T \mathbf{R}^{-1}[\mathbf{y} - h(\mathbf{x})],$$

(A.1)

where $\mathbf{P}_E$ is the ensemble forecast error covariance, $\mathbf{R}$ is the observation error covariance, and h is a nonlinear observation operator. The state vector is $\mathbf{x}$ and $\mathbf{y}$ is the observation vector. A common change of variable that transforms the control variable from dimensional form ($\mathbf{x}$) to a nondimensional ($\mathbf{w}$) vector is introduced as follows:

$$\mathbf{x} = \mathbf{x}^f + \mathbf{u}; \quad \mathbf{u} = \mathbf{P}_E^{1/2}\mathbf{w}, \tag{A.2}$$

where $\mathbf{u} \in S$ is the analysis update vector, resulting in a transformed cost function that is minimized in practice:

$$J(\mathbf{w}) = \frac{1}{2}\mathbf{w}^T\mathbf{w} + \frac{1}{2}[\mathbf{y} - h(\mathbf{x}^f + \mathbf{u})]^T \mathbf{R}^{-1}[\mathbf{y} - h(\mathbf{x}^f + \mathbf{u})]. \quad (A.3)$$

In preparation for observation space localization, one can identify a local state subspace $S_k \subseteq S$ and a related ensemble space $E_k \subseteq E$, both corresponding to the kth grid point of the model domain. Using such local indexing one could rewrite the change of variable (A.2):

$$\mathbf{x}_k = \mathbf{x}_k^f + \mathbf{u}_k; \quad \mathbf{u}_k = (\mathbf{P}_E^{1/2}\mathbf{w})_k (k = 1, \ldots, N_G), \tag{A.4}$$

where

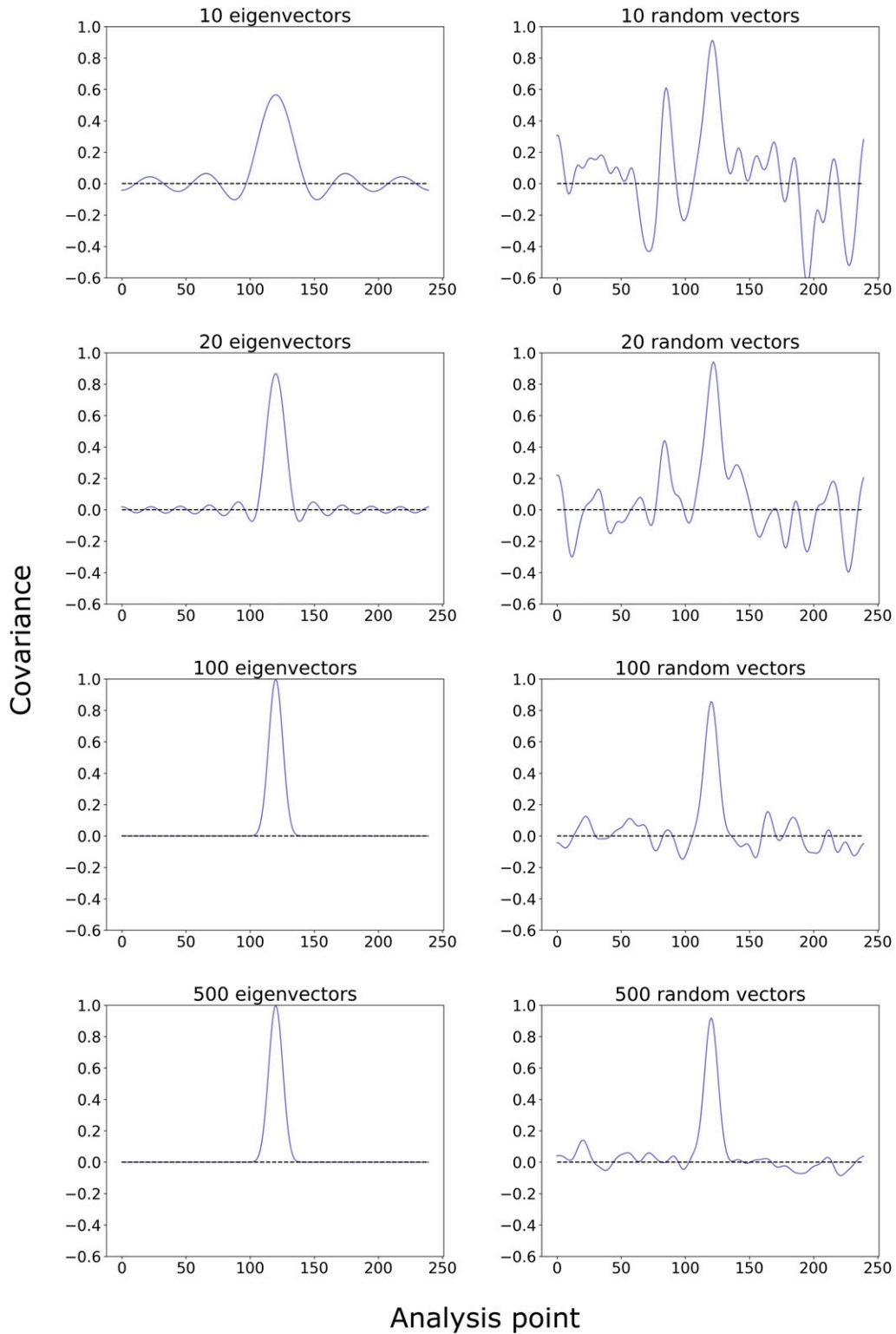$$\mathbf{x}_k = (x_{k,1} \cdots x_{k,M})^T. \tag{A.5}$$

FIG. 8. Columns of the matrix $\sum_n (\mathbf{L}^{1/2}\mathbf{u}_n)(\mathbf{L}^{1/2}\mathbf{u}_n)^{\mathrm{T}}$ at $N_S = 120$. (left) Obtained with truncated eigenvectors and (right) obtained using random basis. (from top to bottom) $N_{\mathrm{RR}} = 10$, $N_{\mathrm{RR}} = 20$, $N_{\mathrm{RR}} = 100$, and $N_{\mathrm{RR}} = 500$.

In MLEF-OBS the formulation of the analysis update vector in (A.4) is changed to

$$\mathbf{u}_k = [\mathbf{P}_E^{1/2}]_k \mathbf{w}_k, \tag{A.6}$$

where $\mathbf{w}_k \in E_k$. Note that $[\mathbf{P}_E^{1/2}]_k$ is same as $\mathbf{P}_E^{1/2}$ for elements corresponding to the kth analysis point, implying that all correlation in the global forecast error covariance matrix are maintained in the localized approach. Finally, a local transformed cost function is defined as follows:

$$J(\mathbf{w}_k) = \frac{1}{2}\mathbf{w}_k^T \mathbf{w}_k + \frac{1}{2}[\mathbf{y} - h(\mathbf{x}^f + \mathbf{u})]^T [\mathbf{A}_k \circ \mathbf{R}]^{-1}[\mathbf{y} - h(\mathbf{x}^f + \mathbf{u})], \tag{A.7}$$

where $\mathbf{u}$ is defined as in (A.6), $\mathbf{A}_k$ is a distance-dependent localizing covariance matrix in observation space centered at gridpoint k, and symbol $\circ$ denotes the Schur product. Note that matrix $\mathbf{A}_k$ is required to have the same dimensions as $\mathbf{R}$ in order to apply the Schur product. In typical case of diagonal $\mathbf{R}$ this implies that $\mathbf{A}_k$ is also diagonal, with elements calculated using a Gaussian function (e.g., Miyoshi and Yamane 2007) or a compactly supported correlation function (e.g., Gaspari and Cohn 1999).

Following the original MLEF approach, one can define matrix $\mathbf{Z}$ for each grid point k:

$$\mathbf{Z}_k = (\mathbf{z}_k^1 \cdots \mathbf{z}_k^{N_E}), \quad \mathbf{z}_k^i = (\mathbf{A}_k \circ \mathbf{R})^{-1/2}\{h[\mathbf{x} + (\mathbf{P}_E^{1/2})_i] - h(\mathbf{x})\}$$

$$(i = 1, \ldots, N_E). \tag{A.8}$$

The gradient and Hessian of (A.7) are

$$\mathbf{g}_k = \mathbf{w}_k - \mathbf{Z}_k^T (\mathbf{A}_k \circ \mathbf{R})^{-1/2}[\mathbf{y} - h(\mathbf{x}^f + \mathbf{u})], \tag{A.9}$$

$$\mathbf{Q}_k = \mathbf{I} + \mathbf{Z}_k^T \mathbf{Z}_k. \tag{A.10}$$

The optimal Hessian preconditioning requires another change of variable:

$$\mathbf{w}_k = \mathbf{Q}_k^{-T/2}\boldsymbol{\zeta}_k, \tag{A.11}$$

which is accomplished by using the eigenvalue decomposition:

$$\mathbf{I} + \mathbf{Z}_k^T \mathbf{Z}_k = \mathbf{U}_k \boldsymbol{\Lambda}_k \mathbf{U}_k^T, \tag{A.12}$$

so that

$$\mathbf{Q}_k^{-T/2} = \mathbf{U}_k \boldsymbol{\Lambda}_k^{-1/2} \mathbf{U}_k^T \quad (k = 1, \ldots, N_G). \tag{A.13}$$

The minimizing solution to (A.7) is obtained using an iterative minimization algorithm, such as the nonlinear conjugate-gradient method (e.g., Luenberger 1984). The local minimization framework is similar to global minimization, except for two distinctions: 1) local gradient and Hessian preconditioning are used [e.g., (A.8)–(A.13)], and 2) after each iteration of minimization a global analysis update is created from local contributions (A.6).

After the minimization is completed the analysis error covariance is estimated from local Hessian preconditioning

(A.13), however, this time calculated at the optimal analysis point:

$$\mathbf{A}_k = (\mathbf{a}_1^k \cdots \mathbf{a}_{N_E}^k), \quad \mathbf{A}_k = [\mathbf{P}_E^{1/2}]_k \mathbf{Q}_k^{-T/2}. \tag{A.14}$$

The global analysis error covariance is formed from local contributions:

$$\mathbf{P}_a^{1/2} = (\mathbf{p}_1^a \cdots \mathbf{p}_{N_E}^a), \quad \mathbf{P}_a^{1/2} = \begin{pmatrix} \mathbf{A}_1 \\ \vdots \\ \mathbf{A}_{N_G} \end{pmatrix}. \tag{A.15}$$

Finally, the initial conditions for ensemble forecasting used to transport uncertainty to the next data assimilation cycle are

$$\mathbf{x}_i^a = \mathbf{x}^a + \mathbf{p}_i^a \tag{A.16}$$

APPENDIX B

EnKF with State Space Localization

The motivation for using EnKF with state space localization is to compare MLEF-SSL to a more standard algorithm, such as the EnKF with perturbed observations, however with localization of forecast error covariance that is more similar to the proposed method. Although such direct approach to state space covariance localization in EnKF is not computationally efficient in realistic high-dimensional applications, but it was possible in the idealized application to Lorenz model conducted here. In practice, EnKF is often applied with observation space localization (e.g., Houtekamer and Mitchell 1998, 2001; Evensen 2003), although more practical square root EnKF algorithms without perturbed observations have been also developed (Shlayeva and Whitaker 2018).

The main distinction of the EnKF algorithm used here is explained by the localized Kalman gain matrix in Houtekamer and Mitchell [2001, their Eq. (5)], in which a Schur product $\mathbf{L} \circ \mathbf{P}_E$ is used instead of $\mathbf{P}_E$. All other components are essentially same as in the standard EnKF with perturbed observations. The analysis equation is

$$\mathbf{x}_i^a = \mathbf{x}_i^f + (\mathbf{L} \circ \mathbf{P}_E)[\mathbf{H}(\mathbf{L} \circ \mathbf{P}_E)\mathbf{H}^T + \mathbf{R}]^{-1}[\mathbf{y}_i - h(\mathbf{x}_i^f)], \tag{B.1}$$

where $i = 1, \ldots, N_E$ denotes the ensemble member. The first derivative of observation operator h, denoted $\mathbf{H}$, is linearized around the ensemble forecast mean. The perturbed observation vector is $\mathbf{y}_i = \mathbf{y} + \mathbf{R}^{1/2}\boldsymbol{\varepsilon}$, where $\boldsymbol{\varepsilon} \sim N(0,1)$.

APPENDIX C

Estimation of Optimal Localization and Inflation Parameters

For all considered data assimilation systems and observation operators the optimal localization length and covariance inflation parameters are estimated empirically using trial-and-error. The measures of success were calculated as an average

over 200 data assimilation cycles and include (i) analysis root-mean-square error calculated with respect to truth (e_{RMS}), (ii) square root of analysis error variance (σ_a), and (iii) relative error of the skill-spread ratio (e_{SSR}). The first two measures define common goals of data assimilation to obtain optimal posterior state that best approximates the truth and also achieves the minimum uncertainty, and both have been used for tuning localization and inflation (e.g., Gharamti 2018; Attia and Constantinescu 2019, and references thereof). Since smaller RMSE and reduced uncertainty may not be always achieved simultaneously (e.g., Kotsuki et al. 2017, their Figs. 2 and 6), an additional measure is introduced. Given the relevance of the relationship between skill and ensemble spread (e.g., Houtekamer 1993; Whitaker and Loughe 1998; Grimit and Mass 2007), the additional measure is defined as the error of the analysis skill-spread ratio with respect to the optimal value of one. In each data assimilation cycle one can define the skill–spread ratio:

$$e_{\mathrm{SSR}} = \frac{e_{\mathrm{RMS}}}{\sigma_a}, \tag{C.1}$$

where

$$e_{\mathrm{RMS}} = \sqrt{\frac{1}{N_S}\sum_j [\mathbf{x}^a - \mathbf{x}_{\mathrm{true}}]_j^2} \tag{C.2}$$

is the analysis root-mean-square error, index j denotes the state point, and

$$\sigma_a = \sqrt{\frac{1}{N_s}\mathrm{Tr}(\mathbf{P}_a)} \tag{C.3}$$

is the average variance of analysis error covariance matrix, here represented using matrix trace. The values (C.1)–(C.3) are calculated in each cycle and then averaged over all data assimilation cycles to form the final measures:

$$E_{\mathrm{SSR}} = \sqrt{\frac{1}{N_{\mathrm{cycle}}}\sum_k [(e_{\mathrm{SSR}})_k - 1]^2}, \tag{C.4}$$

$$E_{\mathrm{RMS}} = \sqrt{\frac{1}{N_{\mathrm{cycle}}}\sum_k (e_{\mathrm{RMS}})_k^2}, \tag{C.5}$$

$$\Sigma_a = \sqrt{\frac{1}{N_{\mathrm{cycle}}}\sum_k (\sigma_a)_k^2}, \tag{C.6}$$

where index k denotes data assimilation cycle and N_{cycle} is the number of data assimilation cycles.

The localization lengths used in tuning are 8, 12, 16, and 20 gridpoint units. For each of these localization lengths a set of inflation parameters γ is ran with each data assimilation algorithm (MLEF-SSL, MLEF-OBS, and EnKF), with some preliminary runs to assess the most likely range of acceptable values.

For all experiments the localization length of 12 grid points indicated best overall performance in terms of the above described measures. The optimal inflation parameters were estimated to be $\gamma = 0.0$ for MLEF-SSL, $\gamma = 0.1$ for

MLEF-OBS, and $\gamma = 0.7$ for EnKF. While these choices are empirical estimates and clearly nonunique, they provide a satisfactory performance of the considered data assimilation algorithms.

REFERENCES

Anderson, J. L., 2007: Exploring the need for localization in ensemble data assimilation using a hierarchical ensemble filter. *Physica D*, **230**, 99–111, https://doi.org/10.1016/j.physd.2006.02.011.

Arbenz, P., 1991: Computing eigenvalues of banded symmetric Toeplitz matrices. *SIAM J. Sci. Stat. Comput.*, **12**, 743–754, https://doi.org/10.1137/0912039.

——, and G. H. Golub, 1988: On the spectral decomposition of Hermitian matrices modified by low rank perturbations with applications. *SIAM J. Matrix Anal. Appl.*, **9**, 40–58, https://doi.org/10.1137/0609004.

Attia, A., and E. Constantinescu, 2019: An optimal experimental design framework for adaptive inflation and covariance localization for ensemble filters. https://arxiv.org/abs/1806.10655.

Auligne, T., B. Menetrier, A. C. Lorenc, and M. Buehner, 2016: Ensemble-variational integrated localized data assimilation. *Mon. Wea. Rev.*, **144**, 3677–3696, https://doi.org/10.1175/MWR-D-15-0252.1.

Axelsson, O., and V. A. Barker, 1984: *Finite-Element Solution of Boundary-Value Problems: Theory and Computation.* Academic Press, 432 pp.

Bannister, R. N., 2017: A review of operational methods of variational and ensemble-variational data assimilation. *Quart. J. Roy. Meteor. Soc.*, **143**, 607–633, https://doi.org/10.1002/qj.2982.

Bickel, P. J., and E. Levina, 2008: Regularized estimation of large covariance matrices. *Ann. Stat.*, **36**, 199–227, https://doi.org/10.1214/009053607000000758.

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, https://doi.org/10.1175/1520-0493(2001)129<0420:ASWTET>2.0.CO;2.

——, B. Huang, and X. Wang, 2015: A nonvariational consistent hybrid ensemble filter. *Mon. Wea. Rev.*, **143**, 5073–5090, https://doi.org/10.1175/MWR-D-14-00391.1.

——, J. S. Whitaker, and L. Lei, 2017: Gain form of the ensemble transform Kalman filter and its relevance to satellite data assimilation with model space ensemble covariance localization. *Mon. Wea. Rev.*, **145**, 4575–4592, https://doi.org/10.1175/MWR-D-17-0102.1.

Bocquet, M., and A. Carrassi, 2017: Four-dimensional ensemble variational data assimilation and the unstable subspace. *Tellus*, **69A**, 1304504, https://doi.org/10.1080/16000870.2017.1304504.

Buehner, M., 2005: Ensemble-derived stationary and flow-dependent background-error covariances: Evaluation in a quasi-operational NWP setting. *Quart. J. Roy. Meteor. Soc.*, **131**, 1013–1043, https://doi.org/10.1256/qj.04.15.

——, 2012: Evaluation of a spatial/spectral covariance localization approach for atmospheric data assimilation. *Mon. Wea. Rev.*, **140**, 617–636, https://doi.org/10.1175/MWR-D-10-05052.1.

——, and A. Shlyaeva, 2015: Scale-dependent background-error covariance localization. *Tellus*, **67A**, 28027, https://doi.org/10.3402/tellusa.v67.28027.

——, R. McTaggart-Cowan, and S. Heilliette, 2017: An ensemble Kalman filter for numerical weather prediction based on variational data assimilation: VarEnKF. *Mon. Wea. Rev.*, **145**, 617–635, https://doi.org/10.1175/MWR-D-16-0106.1.

Campbell, W. F., C. H. Bishop, and D. Hodyss, 2010: Vertical covariance localization for satellite radiances in ensemble Kalman filters. *Mon. Wea. Rev.*, **138**, 282–290, https://doi.org/10.1175/2009MWR3017.1.

Chorin, A. J., and X. Tu, 2009: Implicit sampling for particle filters. *Proc. Natl. Acad. Sci. USA*, **106**, 17 249–17 254, https://doi.org/10.1073/pnas.0909196106.

——, M. Morzfeld, and X. Tu, 2010: Implicit particle filters for data assimilation. *Appl. Math. Comput. Sci.*, **5**, 221–240, https://doi.org/10.2140/camcos.2010.5.221.

Evensen, G., 1994: Sequential data assimilation with a nonlinear quasi-geostrophic model using Monte Carlo methods to forecast error statistics. *J. Geophys. Res.*, **99**, 10 143–10 162, https://doi.org/10.1029/94JC00572.

——, 2003: The ensemble Kalman filter: Theoretical formulation and practical implementation. *Ocean Dyn.*, **53**, 343–367, https://doi.org/10.1007/s10236-003-0036-9.

——, 2018: Analysis of iterative ensemble smoothers for solving inverse problems. *Comput. Geosci.*, **22**, 885–908, https://doi.org/10.1007/s10596-018-9731-y.

Farchi, A., and M. Bocquet, 2019: On the efficiency of covariance localisation of the ensemble Kalman filter using augmented ensembles. *Front. Appl. Math. Stat.*, **5**, 15 pp., https://doi.org/10.3389/fams.2019.00003.

Fisher, M., and P. Courtier, 1995: Estimating the covariance matrix of analysis and forecast error in variational data assimilation. ECMWF Tech. Memo. 220, 28 pp.

Gaspari, G., and S. E. Cohn, 1999: Construction of correlation functions in two and three dimensions. *Quart. J. Roy. Meteor. Soc.*, **125**, 723–757, https://doi.org/10.1002/qj.49712555417.

Gasperoni, N. A., and X. Wang, 2015: Adaptive localization for the ensemble-based observation impact estimate using regression confidence factors. *Mon. Wea. Rev.*, **143**, 1981–2000, https://doi.org/10.1175/MWR-D-14-00272.1.

Gejadze, I. Y., V. Shutyaev, and F.-X. Le Dimetc, 2013: Analysis error covariance versus posterior covariance in variational data assimilation. *Quart. J. Roy. Meteor. Soc.*, **139**, 1826–1841, https://doi.org/10.1002/qj.2070.

——, ——, and F.-X. Le Dimet, 2018: Hessian-based covariance approximations in variational data assimilation. *Russ. J. Numer. Anal. Math. Model.*, **33**, 25–39, https://doi.org/10.1515/rnam-2018-0003.

Gharamti, M., 2018: Enhanced adaptive inflation algorithm for ensemble filters. *Mon. Wea. Rev.*, **146**, 623–640, https://doi.org/10.1175/MWR-D-17-0187.1.

Gill, P. E., W. Murray, and M. H. Wright, 1981: *Practical Optimization*. Academic Press, 401 pp.

Golub, G. H., and C. F. van Loan, 1989: *Matrix Computations*. 2nd ed. The Johns Hopkins University Press, 642 pp.

Greybush, S. J., E. Kalnay, T. Miyoshi, K. Ide, and B. R. Hunt, 2011: Balance and ensemble Kalman filter localization techniques. *Mon. Wea. Rev.*, **139**, 511–522, https://doi.org/10.1175/2010MWR3328.1.

Grimit, E. P., and C. F. Mass, 2007: Measuring the ensemble spread–error relationship with a probabilistic approach: Stochastic ensemble results. *Mon. Wea. Rev.*, **135**, 203–221, https://doi.org/10.1175/MWR3262.1.

Halko, N., P. G. Martinsson, and J. A. Tropp, 2011: Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions. *SIAM Rev.*, **53**, 217–288, https://doi.org/10.1137/090771806.

Hamill, T. M., J. S. Whitaker, and C. Snyder, 2001: Distance-dependent filtering of background error covariance estimates in an ensemble Kalman filter. *Mon. Wea. Rev.*, **129**, 2776–2790, https://doi.org/10.1175/1520-0493(2001)129<2776:DDFOBE>2.0.CO;2.

Houtekamer, P. L., 1993: Global and local skill forecasts. *Mon. Wea. Rev.*, **121**, 1834–1846, https://doi.org/10.1175/1520-0493(1993)121<1834:GALSF>2.0.CO;2.

——, and H. L. Mitchell, 1998: Data assimilation using an ensemble Kalman filter technique. *Mon. Wea. Rev.*, **126**, 796–811, https://doi.org/10.1175/1520-0493(1998)126<0796:DAUAEK>2.0.CO;2.

——, and ——, 2001: A sequential ensemble Kalman filter for atmospheric data assimilation. *Mon. Wea. Rev.*, **129**, 123–137, https://doi.org/10.1175/1520-0493(2001)129<0123:ASEKFF>2.0.CO;2.

Huang, B., X. Wang, and C. H. Bishop, 2019: The high-rank ensemble transform Kalman filter. *Mon. Wea. Rev.*, **147**, 3025–3043, https://doi.org/10.1175/MWR-D-18-0210.1.

Hunt, B., E. Kostelich, and I. Szunyogh, 2007: Efficient data assimilation for spatiotemporal chaos: A local ensemble transform Kalman filter. *Physica D*, **230**, 112–126, https://doi.org/10.1016/j.physd.2006.11.008.

Janjić, T., L. Nerger, A. Albertella, J. Schroter, and S. Skachko, 2011: On domain localization in ensemble-based Kalman filter algorithms. *Mon. Wea. Rev.*, **139**, 2046–2060, https://doi.org/10.1175/2011MWR3552.1.

Jaynes, E. T., 2003: *Probability Theory: The Logic of Science*. Cambridge University Press, 727 pp.

Kepert, J. D., 2009: Covariance localization and balance in an ensemble Kalman filter. *Quart. J. Roy. Meteor. Soc.*, **135**, 1157–1176, https://doi.org/10.1002/qj.443.

Kotsuki, S., Y. Ota, and T. Miyoshi, 2017: Adaptive covariance relaxation methods for ensemble data assimilation: Experiments in the real atmosphere. *Quart. J. Roy. Meteor. Soc.*, **143**, 2001–2015, https://doi.org/10.1002/qj.3060.

LeCun, Y., Y. Bengio, and G. Hinton, 2015: Deep learning. *Nature*, **521**, 436–444, https://doi.org/10.1038/nature14539.

Le Dimet, F.-X., and O. Talagrand, 1986: Variational algorithms for analysis and assimilation of meteorological observations: Theoretical aspects. *Tellus*, **38A**, 97–110, https://doi.org/10.1111/j.1600-0870.1986.tb00459.x.

——, H. E. Ngodock, B. Luong, and J. Verron, 1997: Sensitivity analysis in variational data assimilation. *J. Meteor. Soc. Japan*, **75**, 245–255, https://doi.org/10.2151/jmsj1965.75.1B_245.

——, I. M. Navon, and D. N. Daescu, 2002: Second-order information in data assimilation. *Mon. Wea. Rev.*, **130**, 629–648, https://doi.org/10.1175/1520-0493(2002)130<0629:SOIIDA>2.0.CO;2.

Lei, L., J. S. Whitaker, and C. Bishop, 2018: Improving assimilation of radiance observations by implementing model space localization in an ensemble Kalman filter. *J. Adv. Model. Earth Syst.*, **10**, 3221–3232, https://doi.org/10.1029/2018MS001468.

Leng, H., J. Song, F. Lu, and X. Gao, 2013: A new data assimilation scheme: The space-expanded ensemble localization Kalman filter. *Adv. Meteor.*, **2013**, 410812, https://doi.org/10.1155/2013/410812.

Lorenc, A. C., 2003: The potential of the ensemble Kalman filter for NWP—A comparison with 4D-Var. *Quart. J. Roy. Meteor. Soc.*, **129**, 3183–3203, https://doi.org/10.1256/qj.02.132.

Lorentzen, R. J., and G. Naevdal, 2011: An iterative ensemble Kalman filter. *IEEE Trans. Autom. Control*, **56**, 1990–1995, https://doi.org/10.1109/TAC.2011.2154430.

Lorenz, E. N., 2005: Designing chaotic models. *J. Atmos. Sci.*, **62**, 1574–1587, https://doi.org/10.1175/JAS3430.1.

Luenberger, D. L., 1984: *Linear and Non-Linear Programming*. Addison-Wesley, 491 pp.

Mahoney, M. W., 2011: Randomized algorithms for matrices and data. *Found. Trends Mach. Learn.*, **3**, 123–224, https://doi.org/10.1561/2200000035.

Miyoshi, T., and S. Yamane, 2007: Local ensemble transform Kalman filtering with an AGCM at a T159/L48 resolution. *Mon. Wea. Rev.*, **135**, 3841–3861, https://doi.org/10.1175/2007MWR1873.1.

Morzfeld, M., X. Tu, E. Atkins, and A. J. Chorin, 2012: A random map implementation of implicit filters. *J. Comput. Phys.*, **231**, 2049–2066, https://doi.org/10.1016/j.jcp.2011.11.022.

Navon, I. M., and D. M. Legler, 1987: Conjugate-gradient methods for large-scale minimization in meteorology. *Mon. Wea. Rev.*, **115**, 1479–1502, https://doi.org/10.1175/1520-0493(1987)115<1479:CGMFLS>2.0.CO;2.

Nerger, L., T. Janjic, J. Schroter, and W. Hiller, 2012: A regulated localization scheme for ensemble-based Kalman filters. *Quart. J. Roy. Meteor. Soc.*, **138**, 802–812, https://doi.org/10.1002/qj.945.

Nino-Ruiz, E. D., A. Mancilla-Herrera, S. Lopez-Restrepo, and O. Quintero-Montoya, 2020: A maximum likelihood ensemble filter via a modified Cholesky decomposition for non-Gaussian data assimilation. *Sensors*, **20**, 877, https://doi.org/10.3390/s20030877.

Nocedal, J., and S. J. Wright, 1999: *Numerical Optimization*. Springer-Verlag, 634 pp.

Rabier, F., and P. Courtier, 1992: Four dimensional assimilation in the presence of baroclinic instability. *Quart. J. Roy. Meteor. Soc.*, **118**, 649–672, https://doi.org/10.1002/qj.49711850604.

Sakov, P., and L. Bertino, 2011: Relation between two common localisation methods for the EnKF. *Comput. Geosci.*, **15**, 225–237, https://doi.org/10.1007/s10596-010-9202-6.

——, D. S. Oliver, and L. Bertino, 2012: An iterative EnKF for strongly nonlinear systems. *Mon. Wea. Rev.*, **140**, 1988–2004, https://doi.org/10.1175/MWR-D-11-00176.1.

Shlyaeva, A., and J. S. Whitaker, 2018: Using the linearized observation operator to calculate observation-space ensemble perturbations in ensemble filters. *J. Adv. Model. Earth Syst.*, **10**, 1414–1420, https://doi.org/10.1029/2018MS001309.

——, ——, and C. Snyder, 2019: Model-space localization in serial ensemble filters. *J. Adv. Model. Earth Syst.*, **11**, 1627–1636, https://doi.org/10.1029/2018MS001514.

Steward, J. L., J. E. Roman, A. L. Davina, and A. Aksoy, 2018: Parallel direct solution of the covariance-localized ensemble square root Kalman filter equations with matrix functions. *Mon. Wea. Rev.*, **146**, 2819–2836, https://doi.org.10.1175/MWR-D-18-0022.1.

Suzuki, K., M. Zupanski, and D. Zupanski, 2017: A case study involving single observation experiment performed over snowy Siberia using a coupled atmosphere-land modeling system. *Atmos. Sci. Lett.*, **18**, 106–111, https://doi.org/10.1002/asl.730.

Whitaker, J. S., and A. F. Loughe, 1998: The relationship between ensemble spread and ensemble mean skill. *Mon. Wea. Rev.*, **126**, 3292–3302, https://doi.org/10.1175/1520-0493(1998)126<3292:TRBESA>2.0.CO;2.

Zhang, F., C. Snyder, and J. Sun, 2004: Impacts of initial estimate and observation availability on convective-scale data assimilation with an ensemble Kalman filter. *Mon. Wea. Rev.*, **132**, 1238–1253, https://doi.org/10.1175/1520-0493(2004)132<1238:IOIEAO>2.0.CO;2.

Zupanski, M., 2005: Maximum likelihood ensemble filter: Theoretical aspects. *Mon. Wea. Rev.*, **133**, 1710–1726, https://doi.org/10.1175/MWR2946.1.

——, I. M. Navon, and D. Zupanski, 2008: The maximum likelihood ensemble filter as a non-differentiable minimization algorithm. *Quart. J. Roy. Meteor. Soc.*, **134**, 1039–1050, https://doi.org/10.1002/qj.251.