



On the Efficiency of Covariance Localisation of the Ensemble Kalman Filter Using Augmented Ensembles

Alban Farchi* and Marc Bocquet

CEREA, Joint Laboratory École des Ponts ParisTech and EDF R&D, Université Paris-Est, Champs-sur-Marne, France

OPEN ACCESS

Edited by:

Ulrich Parlitz,
Max-Planck-Institute for Dynamics
and Self-Organisation, Max Planck
Society (MPG), Germany

Reviewed by:

Peter Jan Van Leeuwen,
University of Reading, United Kingdom
Xin Tong,
National University of Singapore,
Singapore

*Correspondence:

Alban Farchi
alban.farchi@enpc.fr

Specialty section:

This article was submitted to
Dynamical Systems,
a section of the journal
Frontiers in Applied Mathematics and
Statistics

Received: 08 November 2018

Accepted: 14 January 2019

Published: 26 February 2019

Citation:

Farchi A and Bocquet M (2019) On
the Efficiency of Covariance
Localisation of the Ensemble Kalman
Filter Using Augmented Ensembles.
Front. Appl. Math. Stat. 5:3.
doi: 10.3389/fams.2019.00003

Localisation is one of the main reasons for the success of the ensemble Kalman filter (EnKF) in high-dimensional geophysical data assimilation problems. It is based on the assumption that correlations between variables of a dynamical system decrease at a fast rate with the physical distance. In the EnKF, two types of localisation methods have emerged: domain localisation and covariance localisation. In this article, we explore possible implementations for covariance localisation in the deterministic EnKF using augmented ensembles in the analysis. We first discuss and compare two different methods to construct the augmented ensemble. The first method, known as modulation, relies on a factorisation property of the background covariance matrix. The second method is based on randomised singular value decomposition (svd) techniques, and has not been previously applied to covariance localisation. The qualitative properties of both methods are illustrated using a simple one-dimensional covariance model. We then show how localisation can be introduced in the perturbation update step using this augmented ensemble framework and we derive a generic ensemble square root Kalman filter with covariance localisation (LEnSRF). Using twin simulations of the Lorenz-1996 model we show that the LEnSRF is numerically more efficient when combined with the randomised svd method than with the modulation method. Finally we introduce a realistic extension of the LEnSRF that uses domain localisation in the horizontal direction and covariance localisation in the vertical direction. Using twin simulations of a multilayer extension of the Lorenz-1996 model, we show that this approach is adequate to assimilate satellite radiances, for which domain localisation alone is insufficient.

Keywords: ensemble Kalman filter, covariance localisation, modulation, random svd, satellite radiances, non-local observations

1. INTRODUCTION

The ensemble Kalman filter (EnKF, [1]) is an ensemble data assimilation (DA) method that has been applied with success to a wide range of dynamical systems in geophysics (see for example, [2, 3]). When the ensemble size is small, ensemble estimates are unreliable, which is why localisation techniques have been introduced in the EnKF. With a chaotic model, Bocquet and Carrassi [4] have shown that localisation is necessary when the ensemble size is smaller than the number of unstable and neutral modes of the dynamics.

Localisation is based on the assumption that correlations between variables in a dynamical system decrease at a fast rate with the physical distance. This assumption is used either to make

the assimilation of observations local (domain localisation; [5, 6]) or to artificially taper distant spurious correlations (covariance localisation, [7]). Although both approaches are based on the same rationale, each one has its own implementation and leads to different algorithms. EnKF algorithms using domain localisation are by construction embarrassingly parallel but cannot assimilate non-local observations without *ad hoc* approximations. By contrast, EnKF algorithms using covariance localisation rely on a single global analysis with a tapered (localised) background covariance, which is more complex to implement especially in a deterministic context. In practice, the two approaches coincide in the limit where the analysis is driven by the background statistics [8] and could differ otherwise.

The ability to assimilate non-local observations becomes increasingly important with the prominence of satellite observations [9]. EnKF algorithms using domain localisation have been adapted to the case of satellite radiances [see e.g., 10, 11]. In these algorithms, the shape of the weighting function associated to a specific satellite channel is used to give an approximate location to this channel (usually the function mode). However, using a realistic one-dimensional model with satellite radiances, Campbell et al. [9] have shown that this approach systematically yields higher errors than covariance localisation.

In this article, we focus on the implementation of covariance localisation in the deterministic EnKF and we put forward two difficulties: how to construct an accurate representation of the localised background covariance and how to efficiently update the perturbations using this representation.

Regarding the first issue, the EnKF literature shows a growing interest in using *augmented ensembles* during the analysis step, that is when the ensemble size during the analysis step is larger than during the forecast step. Buehner [12] has proposed a method to construct a *modulated* ensemble that follows the localised background covariance based on a factorisation property shown by Lorenc [13]. This method has then been leveraged upon by Bishop and Hodyss [14] and used in the literature to perform covariance localisation [15–19]. With an alternative point of view, Kretschmer et al. [20] have included localisation in the ensemble transform Kalman filter (ETKF, [21]) by the means of a *climatologically augmented* ensemble. Finally, Lorenc [22] has shown that the background error covariance matrix can be improved in hybrid ensemble variational data assimilation systems by using *time-lagged* and *time-shifted* perturbations.

Another possibility would be to construct the augmented ensemble using randomised singular value decomposition (svd) techniques [23]. Such a method will be detailed in this article.

With an augmented ensemble, standard formulae cannot be used for the perturbation update because the number of perturbations to propagate must be inferior to the augmented ensemble size. This issue is discussed by Bocquet [18], with proposed update formulae. The same update was later derived again by Bishop et al. [19] in a different but formally equivalent form. In this article, we will consider several alternatives for the perturbation update and we will eventually select the update of Bocquet [18], which we found to be the most adequate.

A brief introduction of the EnKF is given in section 2. In section 3, we explain how covariance localisation can be implemented in the deterministic EnKF using augmented ensembles. In section 4, we compare the accuracy of the methods designed to approximate the localised background covariance. Section 5 is dedicated to the numerical illustration of the resulting algorithms using the one-dimensional Lorenz 1996 [L96, 24] model. We explain in section 6 how these methods can be used to assimilate satellite radiances and give an illustration using a multilayer extension of the L96 model that mimics satellite radiances. Conclusions follow in section 7.

2. BACKGROUND AND MOTIVATION

Before getting to the matter of covariance localisation, we recall the basics of the EnKF and we introduce the notation.

2.1. The Kalman Filter Analysis

Consider the DA problem that consists in estimating a state vector $\mathbf{x}_k \in \mathbb{R}^{N_x}$ at discrete times $t_k, k \in \mathbb{N}$, through independent realisations of the observation vectors $\mathbf{y}_k \in \mathbb{R}^{N_y}$ given by

$$\mathbf{x}_k = \mathcal{M}_k(\mathbf{x}_{k-1}) + \mathbf{w}_k, \quad \mathbf{w}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_k), \quad (1)$$

$$\mathbf{y}_k = \mathcal{H}_k(\mathbf{x}_k) + \mathbf{v}_k, \quad \mathbf{v}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_k). \quad (2)$$

When the dynamical model \mathcal{M} and the observation operator \mathcal{H} are linear and when the initial probability density function (pdf) is Gaussian, the analysis pdf at all time is Gaussian with mean vector and error covariance matrix given by the dynamical Riccati equation.

In the following, we focus on one linear Gaussian analysis step. For simplicity we drop the time index k and the conditioning on the previous observations. The linear observation operator is written \mathbf{H} . The background and analysis pdfs are

$$p(\mathbf{x}) = \mathcal{N}(\mathbf{x}|\mathbf{x}_f, \mathbf{B}), \quad (3)$$

$$p(\mathbf{x}|\mathbf{y}) = \mathcal{N}(\mathbf{x}|\mathbf{x}_a, \mathbf{P}), \quad (4)$$

where \mathbf{x}_a and \mathbf{P} are given by

$$\mathbf{K} = \mathbf{B}\mathbf{H}^T (\mathbf{R} + \mathbf{H}\mathbf{B}\mathbf{H}^T)^{-1}, \quad (5)$$

$$\mathbf{x}_a = \mathbf{x}_f + \mathbf{K}(\mathbf{y} - \mathbf{H}\mathbf{x}_f), \quad (6)$$

$$\mathbf{P} = (\mathbf{I} - \mathbf{K}\mathbf{H})\mathbf{B}. \quad (7)$$

2.2. The EnKF Analysis

In the EnKF [25], the statistics are carried through by the ensemble $\{\mathbf{x}^i, i = 1 \dots N_e\}$. Let \mathbf{E} be the ensemble matrix, that is the $N_x \times N_e$ matrix whose columns are the ensemble members. Let $\bar{\mathbf{x}}$ be the ensemble mean and \mathbf{X} be the normalised anomaly matrix:

$$\bar{\mathbf{x}} = \frac{\mathbf{E}\mathbf{1}}{N_e}, \quad (8)$$

$$\mathbf{X} = \frac{\mathbf{E} - \bar{\mathbf{x}}\mathbf{1}^T}{\sqrt{N_e - 1}}, \quad (9)$$

where $\mathbf{1} \in \mathbb{R}^{N_e}$ is the vector whose components are 1. The main EnKF assumptions are

$$\mathbf{x}_f = \bar{\mathbf{x}}, \tag{10}$$

$$\mathbf{B} = \mathbf{X}\mathbf{X}^T. \tag{11}$$

This means that the background pdf $\mathcal{N}(\mathbf{x}_f, \mathbf{B})$ is represented by the empirical pdf $\mathcal{N}(\bar{\mathbf{x}}, \mathbf{X}\mathbf{X}^T)$. The ensemble update depends on the specific implementation of the EnKF. For example, in the ETKF it is given by

$$\mathbf{Y} = \mathbf{H}\mathbf{X}, \tag{12}$$

$$\bar{\mathbf{y}} = \frac{\mathbf{Y}\mathbf{1}}{N_e}, \tag{13}$$

$$\mathbf{x}_a = \bar{\mathbf{x}} + \mathbf{X} \left(\mathbf{I} + \mathbf{Y}^T \mathbf{R}^{-1} \mathbf{Y} \right)^{-1} \mathbf{Y}^T \mathbf{R}^{-1} (\mathbf{y} - \bar{\mathbf{y}}), \tag{14}$$

$$\mathbf{X}_a = \mathbf{X} \left(\mathbf{I} + \mathbf{Y}^T \mathbf{R}^{-1} \mathbf{Y} \right)^{-\frac{1}{2}} \mathbf{U}, \tag{15}$$

$$\mathbf{E} \leftarrow \mathbf{x}_a \mathbf{1}^T + \sqrt{N_e - 1} \mathbf{X}_a, \tag{16}$$

where \mathbf{U} is an arbitrary orthogonal matrix that verifies $\mathbf{U}\mathbf{1} = \mathbf{1}$ and the $\frac{1}{2}$ exponent denotes the square root for diagonalisable matrices with non-negative eigenvalues, defined as follows. Let \mathbf{M} be the matrix $\mathbf{G}\mathbf{D}\mathbf{G}^{-1}$ with \mathbf{G} an invertible square matrix and \mathbf{D} a diagonal matrix with non-negative elements. We define its square root to be the matrix $\mathbf{M}^{\frac{1}{2}} = \mathbf{G}\mathbf{D}^{\frac{1}{2}}\mathbf{G}^{-1}$.

2.3. Rank Deficiency of the EnKF

The empirical error covariance matrix $\mathbf{X}\mathbf{X}^T$ has rank limited by $N_e - 1$, which is probably too small to accurately represent the full error covariance matrix of a high-dimensional system where $N_x \gg N_e$. Indeed, when the ensemble is too small, the empirical error covariance matrix is characterised by large sampling errors, which often take the form of spurious correlations at long distance.

To fix this, covariance localisation uses an $N_x \times N_x$ localisation matrix $\boldsymbol{\rho}$ and regularises the background error empirical covariance matrix by a Schur (element-wise) product

$$\mathbf{B} = \boldsymbol{\rho} \circ (\mathbf{X}\mathbf{X}^T). \tag{17}$$

The localisation matrix $\boldsymbol{\rho}$ is a short-range matrix that describes the correlation in the physical domain. If $\boldsymbol{\rho}$ is positive definite, then \mathbf{B} is positive semi-definite and therefore can be used as a covariance matrix [26]. This new background covariance matrix also has the following desirable properties: (i) if $\boldsymbol{\rho}$ is short-range then spurious correlations at long distance are removed in \mathbf{B} and (ii) the rank of \mathbf{B} is no longer limited by $N_e - 1$. In practice, \mathbf{B} is always full-rank (and hence positive definite).

3. IMPLEMENTING COVARIANCE LOCALISATION IN THE DETERMINISTIC ENKF

3.1. Principle

Covariance localisation as presented in Equation (17) is formulated in state space, while the ETKF ensemble update (Equations 12–16) is formulated in ensemble space. As is, these two approaches are irreconcilable. However, there are other variants of the deterministic EnKF in which covariance localisation can be easily integrated. In the “deterministic ensemble Kalman filter” [27], the ensemble update is based on the Kalman gain Equation (5) only, where covariance localisation can be included in \mathbf{B} . In the serial ensemble square root filter [28], the ensemble update is based on a modified scalar Kalman gain, for which the localisation matrix $\boldsymbol{\rho}$ can be applied entry-wise.

Another possibility to include covariance localisation in the EnKF is to use augmented ensembles. In this case, the EnKF analysis step would be split into three sub-steps:

1. compute a set of \widehat{N}_e perturbations $\widehat{\mathbf{X}}$ such that $\mathbf{B} = \boldsymbol{\rho} \circ \mathbf{X}\mathbf{X}^T \approx \widehat{\mathbf{X}}\widehat{\mathbf{X}}^T$;
2. apply an EnKF analysis step (e.g. the ETKF) using the background state $\bar{\mathbf{x}}$ and the \widehat{N}_e perturbations $\widehat{\mathbf{X}}$ to compute the analysis state \mathbf{x}_a and the \widehat{N}_e perturbations $\widehat{\mathbf{X}}_a$;
3. form N_e updated members using the analysis state \mathbf{x}_a and the \widehat{N}_e analysis perturbations $\widehat{\mathbf{X}}_a$.

Augmented ensembles are currently used in operational centres to implement localisation in four-dimensional ensemble variational methods [29–31].

In section 3.2, we present different methods that can be used to construct the augmented ensemble (sub-step 1) and in section 3.3 we discuss potential implementations of the perturbation update within this augmented ensemble context (sub-step 3).

3.2. Approximate Factorisation of the Prior Covariance Matrix

3.2.1. Mathematical Goal

Given the prior covariance matrix $\mathbf{B} = \boldsymbol{\rho} \circ (\mathbf{X}\mathbf{X}^T)$, we want an $N_x \times \widehat{N}_e$ matrix $\widehat{\mathbf{X}}$ such that

$$\widehat{\mathbf{X}}\widehat{\mathbf{X}}^T \approx \mathbf{B}, \tag{18}$$

$$\widehat{\mathbf{X}}\mathbf{1} = \mathbf{0}. \tag{19}$$

Note that, although $\widehat{\mathbf{X}}$ represents a set of \widehat{N}_e perturbations, we call it “augmented ensemble” for simplicity.

3.2.2. Approximation via Modulation

Suppose that we have an $N_x \times N_m$ matrix \mathbf{W} such that $\boldsymbol{\rho} \approx \mathbf{W}\mathbf{W}^T$. We define the $N_x \times N_m N_e$ matrix $\mathbf{W}\Delta\mathbf{X}$ by

$$[\mathbf{W}\Delta\mathbf{X}]_n^{jN_e+i} = [\mathbf{W}]_n^j [\mathbf{X}]_n^i, \tag{20}$$

which is a mix between a Schur product (for the state variable index n) and a tensor product (for the ensemble indices i and j).

$\mathbf{W}\Delta\mathbf{X}$ is the modulation product of \mathbf{X} and \mathbf{W} . As shown by Lorenc [13]:

$$(\mathbf{W}\Delta\mathbf{X})(\mathbf{W}\Delta\mathbf{X})^T = (\mathbf{W}\mathbf{W}^T) \circ (\mathbf{X}\mathbf{X}^T). \quad (21)$$

Moreover, it is easy to verify that, as long as $\mathbf{X}\mathbf{1} = \mathbf{0}$, we have $(\mathbf{W}\Delta\mathbf{X})\mathbf{1} = \mathbf{0}$. Therefore, $\widehat{\mathbf{X}} = \mathbf{W}\Delta\mathbf{X}$ is a solution to Equations (18) and (19) with $\widehat{N}_e = N_m N_e$ perturbations.

The name ‘‘modulation’’ was given by Bishop et al. [19]. It stems from the fact that the columns of \mathbf{W} should be the main modes of ρ .

Using Equation (20), we conclude that $\widehat{\mathbf{X}}$ is constructed with complexity:

$$T_{\text{mod}} = \mathcal{O}(N_x \widehat{N}_e). \quad (22)$$

In this equation, we excluded the cost of computing the matrix \mathbf{W} . Indeed, if the localisation matrix ρ is constant in time, then the same matrix \mathbf{W} can be used for all analysis steps and it only needs to be computed once. A fair comparison with the other methods must take into account this fact.

One remaining question is: how large must be N_m ? This question is largely discussed in the literature related to principal component analysis (see e.g., [32]). However, its answer highly depends on the spatial structure of ρ itself. In the numerical experiments of sections 4, 5, and 6, we illustrate how our performance criterion depends on the number of modes N_m . Yet at this point, it is not clear which degree of accuracy we need for the factorisation of \mathbf{B} . Finally note that, in high dimensional spaces, \mathbf{W} can be obtained using the random svd (Algorithm B) derived by Halko et al. [23] and described in **Appendix B**.

3.2.3. Including Balance in the Modulation

In this section, we describe a refinement of the modulation method, based on a new idea. When there is variability between the state variables, it could be interesting to remove part of this variability by transferring it to \mathbf{W} as follows. Let Λ be the $N_x \times N_x$ diagonal matrix containing the standard deviations of the ensemble:

$$\Lambda = \left[\text{diag}(\mathbf{X}\mathbf{X}^T) \right]^{\frac{1}{2}}. \quad (23)$$

The background error covariance matrix can then be written

$$\mathbf{B} = (\Lambda\rho\Lambda) \circ \left\{ (\Lambda^{-1}\mathbf{X})(\Lambda^{-1}\mathbf{X})^T \right\}. \quad (24)$$

Suppose now that the $N_x \times N_m$ matrix \mathbf{W} verifies $\Lambda\rho\Lambda \approx \mathbf{W}\mathbf{W}^T$. If we have an $N_x \times (N_m + \delta N_m)$ matrix \mathbf{W}_+ such that $\mathbf{W}_+\mathbf{W}_+^T \approx \rho$, then \mathbf{W} can be constructed as the N_m main modes of $\Lambda\mathbf{W}_+$. For the same reason as in section 3.2.2, $\widehat{\mathbf{X}} = \mathbf{W}\Delta(\Lambda^{-1}\mathbf{X})$ is a solution to Equations (18) and (19) with $\widehat{N}_e = N_m N_e$ perturbations.

In the transformed anomalies $\Lambda^{-1}\mathbf{X}$, all state variables have the same variability (namely 1). The variability transfer from \mathbf{X} to \mathbf{W} means that the matrix \mathbf{W} can be deformed and adapted to the current situation in order to yield a more accurate representation of the prior covariance matrix \mathbf{B} .

Using this method, the longest algorithmic operation consists in obtaining \mathbf{W} from the svd of $\Lambda\mathbf{W}_+$. Therefore, $\Lambda\mathbf{X}$ is constructed with approximate complexity:

$$T_{\text{mod, bal}} = \mathcal{O}(N_x(N_m + \delta N_m)^2). \quad (25)$$

Again, we excluded the cost of computing the matrix \mathbf{W}^+ because it only needs to be computed once.

3.2.4. Approximation via Truncated svd

Suppose that we have a truncated svd of \mathbf{B} given by

$$\mathbf{B} = \rho \circ (\mathbf{X}\mathbf{X}^T) \approx \mathbf{U}\Sigma\mathbf{U}^T, \quad (26)$$

where \mathbf{U} is an $N_x \times N_m$ orthogonal matrix and Σ is an $N_m \times N_m$ diagonal matrix. Since \mathbf{B} is symmetric positive definite, Equation (26) is a truncated eigendecomposition. Let $\widehat{\mathbf{X}}$ be an $N_x \times (N_m + 1)$ matrix such that

$$\widehat{\mathbf{X}}\widehat{\mathbf{X}}^T = \left(\mathbf{U}\Sigma^{\frac{1}{2}} \right) \left(\mathbf{U}\Sigma^{\frac{1}{2}} \right)^T, \quad (27)$$

$$\widehat{\mathbf{X}}\mathbf{1} = \mathbf{0}. \quad (28)$$

Appendix A describes a method to construct $\widehat{\mathbf{X}}$ from $\mathbf{U}\Sigma^{\frac{1}{2}}$. Then $\widehat{\mathbf{X}}$ is a solution to Equations (18) and (19) with $\widehat{N}_e = N_m + 1$ perturbations.

How can we efficiently obtain the truncated svd of \mathbf{B} Equation (26)? Since \mathbf{B} is an $N_x \times N_x$ matrix, its svd can be computed with complexity $\mathcal{O}(N_x^3)$. A more adequate solution is to use the random svd (Algorithm 2) derived by Halko et al. [23]. A brief description of this algorithm can be found in **Appendix B**. For a detailed description, we refer to the original article by Halko et al. [23].

Using this method, the longest algorithmic operations are empirically

1. applying the background error covariance matrix \mathbf{B} (steps 2, 5, 7, and 10 of Algorithm 2);
2. computing the QR factorisations (steps 3, 6, and 8 of Algorithm 2).

This means that $\widehat{\mathbf{X}}$ is constructed with approximate complexity:

$$T_{\text{trunc svd}} = 2(q + 1)\widehat{N}_e T_{\mathbf{B}} + (2q + 1)\mathcal{O}(N_x \widehat{N}_e^2), \quad (29)$$

where $T_{\mathbf{B}}$ is the complexity of applying the matrix \mathbf{B} to a vector and the parameter q is the number of iterations performed in Algorithm 2. With a dense representation of the matrix \mathbf{B} , $T_{\mathbf{B}} = \mathcal{O}(N_x^2)$. However, for any vector \mathbf{v} of size N_x , we have the identity:

$$\mathbf{B}\mathbf{v} = \sum_{i=1}^{N_e} \mathbf{X}^i \circ (\rho(\mathbf{X}^i \circ \mathbf{v})), \quad (30)$$

where \mathbf{X}^i is the i -th column of matrix \mathbf{X} . If ρ is banded with non-zero coefficients on N_b diagonals, the matrix product $\mathbf{B}\mathbf{v}$ has complexity $T_{\mathbf{B}} = \mathcal{O}(N_e N_x N_b)$. Furthermore, if ρ is a circulant matrix (this corresponds to an invariance by translation

in physical space) it is diagonal in spectral space and $T_B = \mathcal{O}(N_e N_x \ln N_x)$.

Finally, as explained in details by Halko et al. [23] and recalled in **Appendix B.2**, the matrix multiplications implying B in the truncated svd method can be parallelised, which reduces $T_{\text{trunc svd}}$ to:

$$T_{\text{trunc svd}} = 2(q + 1) \frac{\widehat{N}_e}{N_{\text{thr}}} T_B + (2q + 1) \mathcal{O}(N_x \widehat{N}_e^2), \quad (31)$$

where N_{thr} is the number of available threads.

Again, the same question remains: how large must be N_m ? For the same reasons as in section 3.2.2, we cannot provide a clear answer at this point. However, in the numerical experiments of sections 4, 5, and 6, we illustrate how our performance criterion depends on the number of modes N_m .

3.3. Updating the Perturbations

Once the augmented ensemble \widehat{X} is constructed, we need to specify how we are going to update the perturbations. This is a non-trivial problem because the number of perturbations that compose \widehat{X} , \widehat{N}_e , is potentially different from (and most of the time greater than) the number of perturbations to update, N_e .

3.3.1. Updating the Perturbations Without Localisation

The perturbation update of the ETKF is given by

$$T_e = (\mathbf{I} + \mathbf{Y}^T \mathbf{R}^{-1} \mathbf{Y})^{-\frac{1}{2}}, \quad (32)$$

$$\mathbf{X}_a = \mathbf{X} T_e. \quad (33)$$

This is a simplified version of Equation (15) that rigorously satisfies

$$\mathbf{X}_a \mathbf{X}_a^T = \mathbf{P} = (\mathbf{I} - \mathbf{K} \mathbf{H}) \mathbf{X} \mathbf{X}^T. \quad (34)$$

Sakov and Bertino [8] have shown that Equation (33) is equivalent to

$$T_x = (\mathbf{I} + \mathbf{B} \mathbf{H}^T \mathbf{R}^{-1} \mathbf{H})^{-\frac{1}{2}}, \quad (35)$$

$$\mathbf{X}_a = T_x \mathbf{X}. \quad (36)$$

Note that $\mathbf{I} + \mathbf{B} \mathbf{H}^T \mathbf{R}^{-1} \mathbf{H}$ is not necessarily symmetric. However, if we suppose that \mathbf{B} is positive definite (the generalisation to positive semi-definite matrices is not fundamental) then $\mathbf{B} \mathbf{H}^T \mathbf{R}^{-1} \mathbf{H}$ is similar (in the matrix sense) to

$$\mathbf{B}^{-\frac{1}{2}} \mathbf{B} \mathbf{H}^T \mathbf{R}^{-1} \mathbf{H} \mathbf{B}^{\frac{1}{2}} = \mathbf{B}^{\frac{1}{2}} \mathbf{H}^T \mathbf{R}^{-1} \mathbf{H} \mathbf{B}^{\frac{1}{2}}, \quad (37)$$

which is symmetric positive semi-definite. Therefore, $\mathbf{B} \mathbf{H}^T \mathbf{R}^{-1} \mathbf{H}$ is diagonalisable with non-negative eigenvalues and $\mathbf{I} + \mathbf{B} \mathbf{H}^T \mathbf{R}^{-1} \mathbf{H}$ is diagonalisable with positive eigenvalues. This means that T_x is well-defined.

3.3.2. Updating the Perturbations With Localisation Using the Augmented Ensemble

The right-transform T_e is formulated in ensemble space. As a result, there is no way to enforce covariance localisation (formulated in state space) using this approach. By contrast, the left-transform T_x is formulated in state space and is therefore more adequate to covariance localisation.

Using the augmented ensemble constructed in section 3.2, the prior covariance matrix is approximated by Equation (18). This means that the left-transform can be approximated by

$$T_x \approx \widehat{T}_e = (\mathbf{I} + \widehat{\mathbf{X}} \widehat{\mathbf{Y}}^T \mathbf{R}^{-1} \mathbf{H})^{-\frac{1}{2}}, \quad (38)$$

where $\widehat{\mathbf{Y}} = \mathbf{H} \widehat{\mathbf{X}}$. Using this expression for the update still implies linear algebra in state space, which is problematic with high-dimensional systems. $\mathbf{I} + \widehat{\mathbf{X}} \widehat{\mathbf{Y}}^T \mathbf{R}^{-1} \mathbf{H}$ is an $N_x \times N_x$ matrix, it is therefore intractable with high-dimensional systems. However, Equation (25) of Bocquet [18] shows that

$$\widehat{T}_e = \mathbf{I} - \widehat{\mathbf{X}} \left(\mathbf{I} + \widehat{\mathbf{Y}}^T \mathbf{R}^{-1} \widehat{\mathbf{Y}} + \left(\mathbf{I} + \widehat{\mathbf{Y}}^T \mathbf{R}^{-1} \widehat{\mathbf{Y}} \right)^{\frac{1}{2}} \right)^{-1} \widehat{\mathbf{Y}}^T \mathbf{R}^{-1} \widehat{\mathbf{H}}, \quad (39)$$

where the linear algebra is now performed in the augmented ensemble space (that is, using $\widehat{N}_e \times \widehat{N}_e$ matrices). This perturbation update has been rediscovered by Bishop et al. [19] and included in their gain ETKF (GETKF) algorithm. However, the update formula used in the GETKF is prone to numerical cancellation errors as opposed to Equation (39).

In the rest of this article, we use the following perturbation update:

$$\mathbf{X}_a = \widehat{T}_e \mathbf{X}, \quad (40)$$

and we compute the left-transform \widehat{T}_e using Equation (39). Algorithm 1 summarises the analysis step of the resulting generic ensemble square root Kalman filter with covariance localisation (LEnSRF). Finally note that the consistency of the perturbation update in deterministic EnKF algorithms using covariance localisation is the subject of another study by Bocquet and Farchi (under review).

4. ACCURACY OF THE APPROXIMATE FACTORISATION

4.1. A Simple One-Dimensional Model for the Background Error Covariance Matrix

In this section, we investigate the accuracy of the factorisation Equation (18) obtained with the methods described in section 3.2. The background error covariance matrix is obtained as follows.

Let G be the piecewise rational function of Gaspari and Cohn [33]. Let d be the Euclidean distance over the set $\{1 \dots N_x\}$ with periodic boundary conditions. For any radius $r \in \mathbb{R}^+$, we define $C(r)$ as the $N_x \times N_x$ matrix whose coefficients satisfy

$$[C(r)]_{m,n} = G\left(\frac{d(m,n)}{r}\right). \quad (41)$$

Algorithm 1 Analysis step for a generic LEnSRF used in this article. The augmented ensemble at step 3 can be constructed using the modulation method with or without the balanced refinement or using the truncated svd method.

Require: $N_x \times N_e$ background ensemble matrix \mathbf{E} , observation vector $\mathbf{y} \in \mathbb{R}^{N_y}$ (inputs); multiplicative inflation factor λ (parameter).

- 1: $\bar{\mathbf{x}} = \frac{\mathbf{E}\mathbf{1}}{N_e}$.
- 2: $\mathbf{X} = \frac{\mathbf{E} - \bar{\mathbf{x}}\mathbf{1}^T}{\sqrt{N_e}}$.
- 3: Compute the augmented ensemble $\widehat{\mathbf{X}}$.
- 4: $\widehat{\mathbf{Y}} = \mathbf{H}\widehat{\mathbf{X}}$.
- 5: $\bar{\mathbf{y}} = \frac{\widehat{\mathbf{Y}}\mathbf{1}}{N_e}$.
- 6: $\delta = \mathbf{R}^{-\frac{1}{2}}(\bar{\mathbf{y}} - \mathbf{y})$.
- 7: $\widehat{\mathbf{S}} = \mathbf{R}^{-\frac{1}{2}}\widehat{\mathbf{Y}}$.
- 8: $\mathbf{x}_a = \bar{\mathbf{x}} + \widehat{\mathbf{X}}(\mathbf{I} + \widehat{\mathbf{S}}^T\widehat{\mathbf{S}})^{-1}\widehat{\mathbf{S}}^T\delta$.
- 9: $\mathbf{X}_a = \mathbf{X} - \widehat{\mathbf{X}}\left(\mathbf{I} + \widehat{\mathbf{S}}^T\widehat{\mathbf{S}} + (\mathbf{I} + \widehat{\mathbf{S}}^T\widehat{\mathbf{S}})^{\frac{1}{2}}\right)^{-1}\widehat{\mathbf{S}}^T\mathbf{R}^{-\frac{1}{2}}\mathbf{H}\mathbf{X}$.
- 10: $\mathbf{E} \leftarrow \mathbf{x}_a\mathbf{1}^T + \lambda\sqrt{N_e - 1}\mathbf{X}_a$.
- 11: **return** $N_x \times N_e$ analysis ensemble matrix \mathbf{E} .

For any vector $\mathbf{v} \in \mathbb{R}^{N_x}$, we define $\mathbf{D}(\mathbf{v})$ as the $N_x \times N_x$ diagonal matrix whose diagonal is precisely \mathbf{v} .

We draw a vector \mathbf{c} from the distribution $\mathcal{N}(\mathbf{1}, \alpha_c \mathbf{C}(r_c))$, where α_c and r_c are two parameters. We define:

$$\mathbf{C}_{\text{ref}} = \mathbf{D}(\mathbf{c}) \mathbf{C}(r_{\text{ref}}) \mathbf{D}(\mathbf{c}). \tag{42}$$

\mathbf{C}_{ref} plays the role of a reference covariance matrix in our model. We now draw a sample of N_e independent members from the distribution $\mathcal{N}(0, \mathbf{C}_{\text{ref}})$. Without loss of generality, we assume that the associated ensemble matrix \mathbf{X} is normalised by $\sqrt{N_e - 1}$. We define the background error covariance matrix as

$$\mathbf{B} = \rho \circ (\mathbf{X}\mathbf{X}^T), \tag{43}$$

with the localisation matrix $\rho = \mathbf{C}(r_{\text{ref}})$.

4.2. Experimental Setup

In the following test series, we use two matrices \mathbf{B}_1 and \mathbf{B}_2 . They are constructed using different realisations of the model described in section 4.1 with the parameters specified in **Table 1**. Note that we used short-range correlations ($r_{\text{ref}} = 20$) to construct \mathbf{B}_1 and mid-range correlations ($r_{\text{ref}} = 100$) to construct \mathbf{B}_2 . **Figure 1** displays the matrices \mathbf{B}_1 and \mathbf{B}_2 .

The modulation method described in sections 3.2.2 and 3.2.3 requires an approximate factorisation for $\rho = \mathbf{C}(r_{\text{ref}})$, that we precompute by keeping the first N_m or $N_m + \delta N_m$ (when using the balance refinement) modes in the svd of ρ . Since ρ is sparse, we use the random svd (Algorithm 2) to obtain this factorisation.

4.3. Results and Discussion

We apply the methods described in section 3.2 to obtain an approximate factorisation for \mathbf{B}_1 and \mathbf{B}_2 and we compute the

TABLE 1 | Parameters used to construct \mathbf{B}_1 and \mathbf{B}_2 with the model described in section.

Parameter	Value for \mathbf{B}_1	Value for \mathbf{B}_2
N_x	400	400
N_e	10	10
α_c	0.2	0.2
r_c	30	30
r_{ref}	20	100

normalised Frobenius error:

$$e_{F,i} = \frac{\|\mathbf{B}_i - \widehat{\mathbf{X}}\widehat{\mathbf{X}}^T\|_F}{\|\mathbf{B}_i\|_F}. \tag{44}$$

Using the Eckart–Young theorem [34], we conclude that the lowest normalised Frobenius error for a factorisation with rank $\widehat{N}_e - 1$ is

$$e_{F,i}^{\min} = \frac{\sqrt{\sum_{k=\widehat{N}_e}^{N_x} \sigma_k^2(\mathbf{B}_i)}}{\|\mathbf{B}_i\|_F}, \tag{45}$$

where $\sigma_k(\mathbf{M})$ is the k -th singular value of the matrix \mathbf{M} .

Figure 2 shows the evolution of the normalised Frobenius error $e_{F,i}$ as a function of the augmented ensemble size \widehat{N}_e when the factorisation is computed using the truncated svd method (section 3.2.4) or the modulation method with (section 3.2.3) or without (section 3.2.2) the balance refinement. The value reported for $e_{F,i}$ is the average value over 100 independent realisations of the random svd. For $q \geq 1$, the Frobenius error of the truncated svd method (not shown here) cannot be distinguished from the minimum possible value. For the modulation method, using the balance refinement with $\delta N_m > 10$ (not shown here) does not yield a clear improvement over the case $\delta N_m = 10$. The singular values of \mathbf{B}_2 (mid-range case) decay much faster than the singular values of \mathbf{B}_1 (short-range case). This explains why the normalised Frobenius errors $e_{F,2}$ are systematically lower than the normalised Frobenius errors $e_{F,1}$.

The modulation method is very fast but yields a poor approximation of the background error covariance matrix \mathbf{B} . With the balance refinement, the approximation is a bit better and the method is still very fast. By contrast, the truncated svd method is much slower but yields an approximation of \mathbf{B} close to optimal. At this point, it is not clear what level of precision is needed for \mathbf{B} . Yet, we can already conclude that, in a cycled DA context, we will have to find a balance between speed and accuracy in the construction of the augmented ensemble and in the perturbation update.

Finally, different matrix norms could have been used in this test series. Indeed, even if equivalent, two matrix norms give different informations. This is why we have computed the normalised spectral error (not shown here) and found quite similar results to those depicted in **Figure 2**. This shows that our results are not specific to the Frobenius norm.

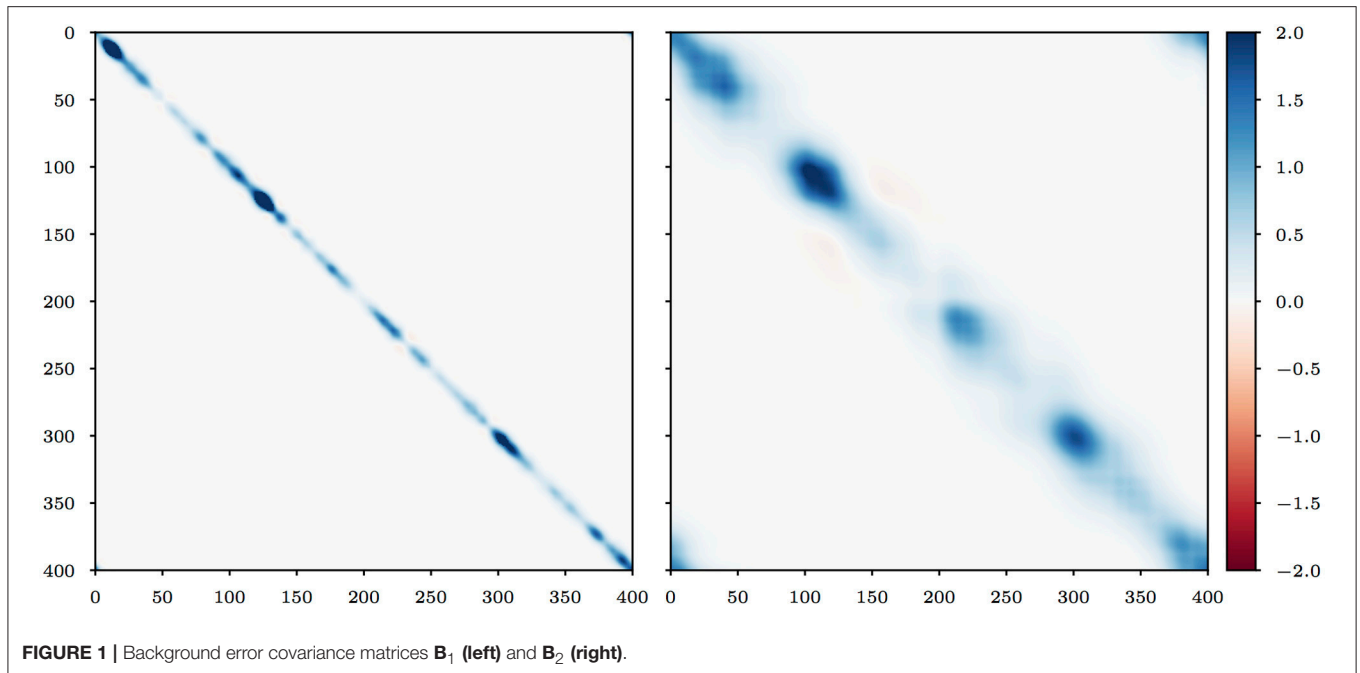


FIGURE 1 | Background error covariance matrices \mathbf{B}_1 (left) and \mathbf{B}_2 (right).

5. CYCLED DA EXPERIMENTS WITH A ONE-DIMENSIONAL MODEL

5.1. The Lorenz 1996 Model

The L96 model [24] is a low-order one-dimensional discrete model whose evolution is given by the following ordinary differential equations (ODEs):

$$\frac{dx_n}{dt} = (x_{n+1} - x_{n-2})x_{n-1} - x_n + F, \quad n = 1 \dots N_x, \quad (46)$$

where the indices are to be understood with periodic boundary conditions: $x_{-1} = x_{N_x-1}$, $x_0 = x_{N_x}$, $x_1 = x_{N_x+1}$ and where the system size N_x can take arbitrary values. These ODEs are integrated using a fourth-order Runge–Kutta method with a time step of 0.05 time unit.

The standard configuration $N_x = 40$ and $F = 8$ is widely used in DA to assess the performance of the DA algorithms. It yields chaotic dynamics with a doubling time around 0.42 time unit. In this section, we use an *extended* standard configuration where $N_x = 400$ and $F = 8$, which is essentially a repetition of ten times the standard configuration. The observations are given by

$$\mathbf{y} = \mathbf{x} + \mathbf{v}, \quad \mathbf{v} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}), \quad (47)$$

and the time interval between consecutive observations is $\Delta t = 0.05$ time unit, which represents 6 h of real time and corresponds to a model autocorrelation around 0.967. The standard deviation of the observation noise is approximately one tenth of the climatological variability of each observation.

For the localisation, we use the Euclidean distance over the set $\{1 \dots N_x\}$ with periodic boundary conditions (the same as the one that was used in section 4).

Note that we use $N_x = 400$ state variables instead of 40 such that typical local domains (with a localisation radius around $r = 20$ grid points) do not cover the whole domain.

5.2. Experimental Setup

In this section, we illustrate the performance of the LEnSRF Algorithm 1 using twin simulations of the L96 model in the extended configuration. The augmented ensemble (step 3 of Algorithm 1) is computed using the truncated svd method (section 3.2.4) or the modulation method with (section 3.2.3) or without (section 3.2.2) the balance refinement. Both methods use an ensemble of $N_e = 10$ members and a localisation matrix $\rho = \mathbf{C}(r)$, where the localisation radius r is a free parameter.

For the modulation method, the approximate factorisation of ρ is precomputed once for each twin experiment by keeping the first N_m or $N_m + \delta N_m$ (when using the balance refinement) modes in the svd of ρ .

For the truncated svd method, the matrix multiplications implying \mathbf{B} are computed using Equation (30) and ρ is applied in spectral space.

The runs are $2 \times 10^4 \Delta t$ long (with an additional $2 \times 10^3 \Delta t$ spin-up period) and our performance criterion is the time-average analysis root mean square error, hereafter called the RMSE.

5.3. Augmented Ensemble Size—RMSE Evolution

Figure 3 shows the evolution of the RMSE as a function of the augmented ensemble size \hat{N}_e . Both the truncated svd and the modulation methods are able to produce an estimate of the true state with an accuracy equivalent to that of the local ensemble transform Kalman filter (LETKF, [35]). As expected after the experiments in section 4, for a given level

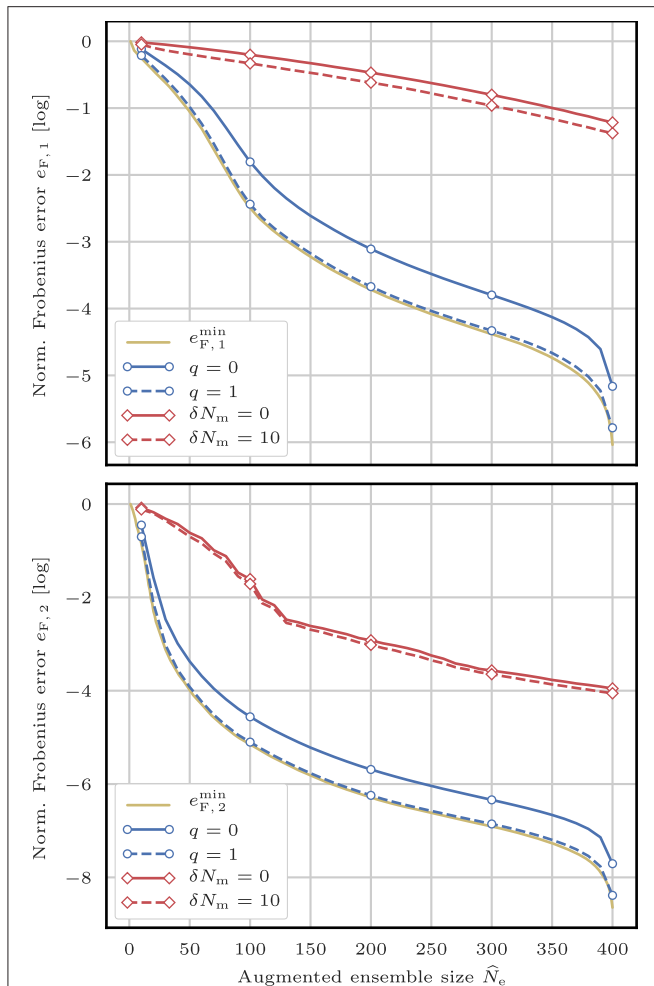


FIGURE 2 | Evolution of the normalised Frobenius errors $e_{F,1}$ (top) and $e_{F,2}$ (bottom) as a function of the augmented ensemble size \hat{N}_e . The approximate factorisation of \mathbf{B}_1 (top) and \mathbf{B}_2 (bottom) is computed either using the truncated svd method with several values for the parameter q for the random svd Algorithm 2 (blue lines) or using the modulation method with (solid red line) or without (dashed red line) the balance refinement. The lowest normalised achievable Frobenius errors $e_{F,i}^{\min}$ are plotted in yellow for both cases.

of RMSE score we need a much smaller augmented ensemble size \hat{N}_e when using the truncated svd method than when using the modulation method. However, before we conclude that the truncated svd method is more efficient, we need to take into account the fact that computing the augmented ensemble is much slower with this method than with the modulation method.

With the truncated svd method, the augmented ensemble size \hat{N}_e needs to be at least of the same order as the number of unstable and neutral modes of the model dynamics (around 133 here) in order to yield accurate results [4]. We have also tested $q > 1$ in the truncated svd method or $\delta N_m > 20$ in the modulation method (not shown here) and found that none of these parameters yields clear improvements in RMSE scores.

5.4. RMSE—Wall-Clock Time Evolution

The longest algorithmic operations in Algorithm 1 are:

1. constructing the augmented ensemble (step 3);
2. computing the svd of $\hat{\mathbf{S}}$ (required for steps 8 and 9);

When using the truncated svd method, some level of parallelisation can be included in the construction of the augmented ensemble as remarked in **Appendix B.2**. When using the modulation method (even with the balance refinement), constructing the augmented ensemble is almost instantaneous compared to computing the svd of $\hat{\mathbf{S}}$. Therefore, we only enable parallelisation when using the truncated svd method.

To compute the analysis state \mathbf{x}_a and analysis perturbations \mathbf{X}_a , we have tested several approaches and concluded that the most efficient is to compute the svd of $\hat{\mathbf{S}}$ with a direct svd algorithm, which cannot be parallelised.

Figure 3 shows the evolution of the wall-clock time of one analysis step as a function of the RMSE. All simulations are performed on the same double *Intel Xeon E5-2680* platform, which has a total of 24 cores. Parallelisation is enabled when possible using a fixed number of *OpenMP* threads. For a given level of RMSE score, the truncated svd method is clearly faster than the modulation method. This shows the advantage of using the truncated svd method over the modulation method, especially when parallelisation is possible. However, this result is specific to the problem considered in this section and may not generalise to all situations.

6. CYCLED DA EXPERIMENTS WITH SATELLITE RADIANCES

6.1. Is Covariance Localisation Viable With High-Dimensional Models?

In section 5, we have implemented covariance localisation in the EnKF and successfully applied the resulting algorithm to a one-dimensional DA problem with $N_x = 400$ state variables. With a high-dimensional system, covariance localisation in the EnKF will probably require a very large augmented ensemble size \hat{N}_e , too large to be affordable. In this case, the use of domain localisation will be mandatory.

When observations are local, domain localisation is simple to implement and yield efficient algorithms such as the LETKF. However, when observations are non-local, one must resort to *ad hoc* approximations to implement domain localisation in the EnKF, for example assigning an approximate location to each observation. In this section, we discuss the case of satellite radiances, which are non-local observations and we show how existing variants of the LETKF deal with such observations. We then give an extension of our LEnSRF Algorithm 1 designed to assimilate satellite radiances in a spatially extended model. Finally we introduce a multilayer extension of the L96 model that mimics satellite radiances and we illustrate the performance of our method using twin simulations of this model.

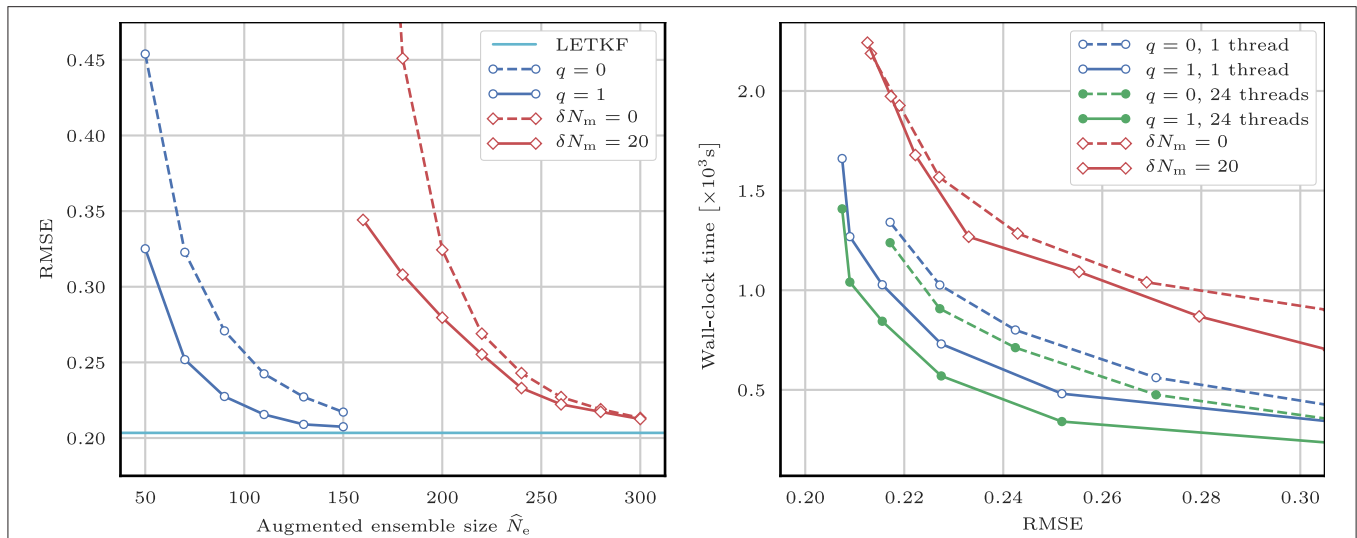


FIGURE 3 | Evolution of the RMSE as a function of the augmented ensemble size \hat{N}_e (left) and of the wall-clock analysis time for the 22×10^3 cycles as a function of the RMSE for the LEnSRF Algorithm 1. For each data point, the localisation radius r and the inflation factor λ are optimally tuned to yield the lowest RMSE. The augmented ensemble is computed either using the truncated svd method with several values for the parameter q in the random svd Algorithm 2 (green and blue lines) or using the modulation method with (solid red line) or without (dashed red line) the balance refinement. In the left panel, the horizontal solid cyan baseline is the RMSE of the LETKF with an ensemble of $N_e = 10$ members and optimally tuned localisation radius r and inflation factor λ .

6.2. The Case of Satellite Radiances

Suppose that the physical space consists of a multilayer space with P_z vertical levels of P_h state variables. For any $h \in \{1 \dots P_h\}$ and $z \in \{1 \dots P_z\}$, the state variable located at the h -th horizontal position and at the z -th vertical level is written $x_{(z, h)}$. For any state vector \mathbf{x} , the sub-vector of the components located at the h -th horizontal position at any level is written \mathbf{x}_h and called the h -th column of \mathbf{x} .

Suppose that each state vector column is observed independently via

$$\mathbf{y}_h = \mathbf{\Omega} \mathbf{x}_h, \tag{48}$$

where $\mathbf{\Omega}$ is a $P_c \times P_z$ weighting matrix and \mathbf{y}_h is the vector containing the P_c observations at the h -th horizontal position. P_c is the number of channels. The full observation vector \mathbf{y} is the concatenation of all \mathbf{y}_h , $h = 1 \dots P_h$. It has $N_y = P_c \times P_h$ components and for any $h \in \{1 \dots P_h\}$ and $c \in \{1 \dots P_c\}$, the observation located at the h -th horizontal position and corresponding to the c -th channel is written $y_{(c, h)}$.

This simple model describes a typical situation for satellite radiances. From these definitions, it is clear that each observation is attached to an horizontal position, but has no well-defined vertical position (unless the weighting matrix $\mathbf{\Omega}$ is diagonal). Several variants of the LETKF have been designed to assimilate such observations. When the weighting function of a channel has a single and well-located maximum, the vertical location of this maximum can play the role of an approximate height for this channel. This is the approach followed for example by Fertig et al. [11]. Based on these vertical positions, they use the channels to update “adjacent” vertical levels as long as the corresponding weighting function is above a threshold value. Campbell et al. [9]

has followed the same approach to define the approximate height of the channels. However their update formula includes a vertical tapering of the anomalies that depends on the vertical distance. When the weighting functions are flat, another possibility is to define the approximate height of a channel as the middle of the support of its weighting function [36]. Miyoshi and Sato [10] have proposed an alternative that does not require the definition of an approximate height of the channels: their update formula includes a vertical tapering of the anomalies that depends on the shape of the weight functions only. Finally, in the algorithm of Penny et al. [37], vertical localisation has been removed.

Using a realistic one-dimensional model with satellite radiances, Campbell et al. [9] have shown that *ad hoc* approaches based on domain localisation only systematically yield higher errors than covariance localisation. In a spatially extended model with satellite radiances, it seems natural to apply domain localisation in the horizontal direction, in which observations are local, while using covariance localisation in the vertical direction, in which observations are non-local.

6.3. Including Domain Localisation in the LEnSRF

Following the approach of Bishop et al. [19], we apply four modifications to the generic LEnSRF Algorithm 1 in order to include domain localisation in way similar to the LETKF.

1. We perform P_h local analyses instead of one global analysis. The aim of the h -th local analysis is to give an update for the P_z state variables that form the h -th column. The linear algebra must be adapted in consequence.

2. We taper the anomalies related to each observation with respect to the *horizontal* distance to the h -th column. This is usually implemented in $\mathbf{R}^{-\frac{1}{2}}$.
3. Observations whose horizontal position is far from the h -th column (i.e., observations that are not in the local domain) do not contribute to the update. These observations are therefore omitted in the local analysis in order to save some computational time.
4. Since observations that are outside of the local domain are omitted, we only need to compute an augmented ensemble for the state variables inside the local domain. Since covariance localisation is only applied in the vertical direction, the augmented ensemble $\widehat{\mathbf{X}}^\ell$ must have empirical covariance matrix given by

$$\widehat{\mathbf{X}}^\ell (\widehat{\mathbf{X}}^\ell)^T \approx \rho_v \circ (\mathbf{X}^\ell (\mathbf{X}^\ell)^T), \quad (49)$$

where \mathbf{X}^ℓ is the set of perturbations that are located within the local domain and ρ_v is the vertical localisation matrix, whose coefficients only depend on the vertical layer indices.

This modified LEnSRF, hereafter called local analysis LEnSRF (L^2 EnSRF), implements domain localisation in the horizontal direction and covariance localisation in the vertical direction and therefore can be used to assimilate vertically non-local observations such as satellite radiances.

6.4. The Multilayer L96 Model

We now introduce a multilayer extension of the L96 model, hereafter called mL96 model. This multilayer extension is used to test and illustrate the performance of the L^2 EnSRF algorithm.

The mL96 model consists of P_z coupled layers of the one-dimensional L96 model with P_h variables. Keeping the notations defined in section 6.2, the evolution of the model is given by the following $P_z P_h$ ODEs:

$$\begin{aligned} \frac{dx_{(z,h)}}{dt} &= (x_{(z,h+1)} - x_{(z,h-2)})x_{(z,h-1)} - x_{(z,h)} + F_z \\ &+ \delta_{\{z>0\}} \Gamma (x_{(z-1,h)} - x_{(z,h)}) \\ &+ \delta_{\{z \leq P_z\}} \Gamma (x_{(z+1,h)} - x_{(z,h)}). \end{aligned} \quad (50)$$

The first line of Equation (50) corresponds to the original L96 ODE Equation (46), with a forcing term that may depend on the vertical layer index z . The second and third lines correspond to the coupling between adjacent layers, with a constant intensity Γ . As for the L96, the horizontal indices are to be understood with periodic boundary conditions: $x_{(z,-1)} = x_{(z,P_h-1)}$, $x_{(z,0)} = x_{(z,P_h)}$, and $x_{(z,1)} = x_{(z,P_h+1)}$. These ODEs are integrated using a fourth-order Runge–Kutta method with a time step of 0.05 time unit.

For this experiment, we use $P_z = 32$ layers and $P_h = 40$ to mimic the standard configuration of the L96 model. The forcing term linearly decreases from $F_1 = 8$ at the lowest level to $F_{P_z} = 4$ at the highest level. Without the coupling, these values would render the lower levels dynamics chaotic and the higher levels

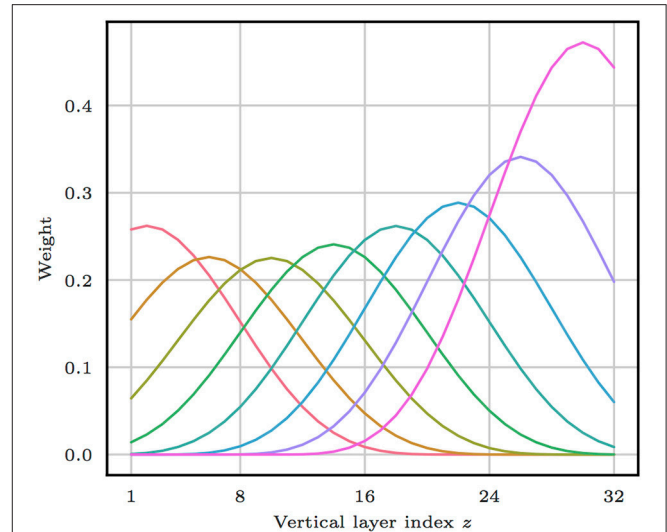


FIGURE 4 | Observation operator \mathbf{H} . Each line represents the weighting function of a different channel, corresponding to a row of the weighting matrix $\mathbf{\Omega}$. Every channel has a single maximum and is relatively broad (half-width around 10 vertical layers). The sum of the weights has been adjusted individually such that every channel yields an observation with approximately the same climatological variability.

dynamics laminar, which is a typical behaviour in the atmosphere. Finally, we take $\Gamma = 1$ such that adjacent layers are highly correlated (correlation around 0.87). To be more specific, the correlation between the z -th level and the $(z + \delta z)$ -th level first rapidly decreases with δz . It reaches approximately -0.1 for $\delta z = 6$ and then it starts increasing. Finally, its absolute value is below 10^{-2} when $\delta z > 10$. This model is chaotic and the dimension of the unstable or neutral subspace is around 50.

The observation operator \mathbf{H} follows the model described in section 6.2. We use $P_c = 8$ channels and a weighting matrix $\mathbf{\Omega}$ designed to mimic satellite radiances as shown in **Figure 4**. The observations are given by

$$\mathbf{y} = \mathbf{H}\mathbf{x} + \mathbf{v}, \quad \mathbf{v} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}), \quad (51)$$

and the time interval between consecutive observations is the same as the one used with the L96 model, $\Delta t = 0.05$ time unit. Once again, the standard deviation of the observation noise is approximately one tenth of the climatological variability of each observation.

For the horizontal localisation, we use the Euclidean distance d_h over the set $\{1 \dots P_h\}$ with periodic boundary conditions. For the vertical localisation, we use the Euclidean distance d_v over the set $\{1 \dots P_z\}$.

6.5. Experimental Setup

In this section, we give some details on how localisation is implemented in the L^2 EnSRF algorithm for the mL96 model. We then described which approximations are made to implement an LETKF algorithm. For each algorithm, the runs are $10^4 \Delta t$ long

(with an additional $10^3 \Delta t$ spin-up period) and our performance criterion is the time-average analysis RMSE.

6.5.1. Horizontal Localisation

Let r_h be the horizontal localisation radius. During the h_1 -th local analysis, the anomalies related to observation $y_{(c, h_2)}$ are tapered by a factor

$$\sqrt{G\left(\frac{d_h(h_1, h_2)}{r_h}\right)}, \tag{52}$$

where G is the piecewise rational function of Gaspari and Cohn [33]. This means that the h -th local domain consists of the columns $\{h - \lfloor r_h \rfloor \dots h + \lfloor r_h \rfloor\}$ where the indices are to be understood with P_h periodic boundary conditions and $\lfloor r_h \rfloor$ is the integer part of r_h .

6.5.2. Vertical Localisation

Let r_v be the vertical localisation radius. The vertical localisation matrix ρ_v has coefficients given by

$$[\rho_v]_{(z_1, h_1), (z_2, h_2)} = G\left(\frac{d_v(z_1, z_2)}{r_v}\right). \tag{53}$$

The local domain gathers $P_h^\ell = 2 \lfloor r_h \rfloor + 1$ columns, hence ρ_v is a $P_z P_h^\ell \times P_z P_h^\ell$ block-diagonal matrix. Since its coefficients only depend on the vertical layer indices, it can also be seen as a $P_z \times P_z$ matrix.

The $P_z P_h^\ell \times \widehat{N}_e$ matrix $\widehat{\mathbf{X}}^\ell$ of the augmented ensemble is computed using the truncated svd method (section 3.2.4) or the modulation method with (section 3.2.3) or without (section 3.2.2) the balance refinement. Both methods use an ensemble of $N_e = 8$ members and the localisation matrix ρ_v .

For the modulation method, the approximate factorisation of ρ_v is precomputed once for each twin experiment by keeping the first N_m or $N_m + \delta N_m$ (when using the balance refinement) modes in the svd of the $P_z \times P_z$ matrix ρ_v .

For the truncated svd method, the matrix multiplications implying \mathbf{B} are computed using Equation (30). Because the coefficients of the localisation matrix ρ_v only depends on the vertical layer indices, applying the $P_z P_h^\ell \times P_z P_h^\ell$ matrix ρ_v to a vector with $P_z P_h^\ell$ components reduces to applying the $P_z \times P_z$ matrix ρ_v to a vector with P_z components. It should be relatively quick and therefore we do not perform this product in spectral space. This means that the implementation can be straightforwardly adapted to the general case where the vertical layers are not equally distributed in height.

6.5.3. Ad hoc Approximations for the LETKF

We define an approximate height z_c for the c -th channel:

$$z_c = \frac{\sum_{z=1}^{P_z} z [\boldsymbol{\Omega}]_{c, z}}{\sum_{z=1}^{P_z} [\boldsymbol{\Omega}]_{c, z}} \in [1, P_z]. \tag{54}$$

We did not define z_c as the vertical position of the maximum of the c -th weight function because we wanted to account for the fact that our weight functions are skewed in the vertical direction.

In the LETKF algorithm, we perform $P_z P_h$ local analyses (one for each state variable). In the local analysis for the state variable $x_{(z, h_1)}$, the anomalies related to observation $y_{(c, h_2)}$ are tapered by a factor

$$\sqrt{G\left(\sqrt{\left(\frac{\delta h}{r_h}\right)^2 + \left(\frac{\delta z}{r_v}\right)^2}\right)}, \tag{55}$$

where

$$\delta h = \min\{|h_2 - h_1|, P_h - |h_2 - h_1|\}, \tag{56}$$

$$\delta z = |z - z_c|, \tag{57}$$

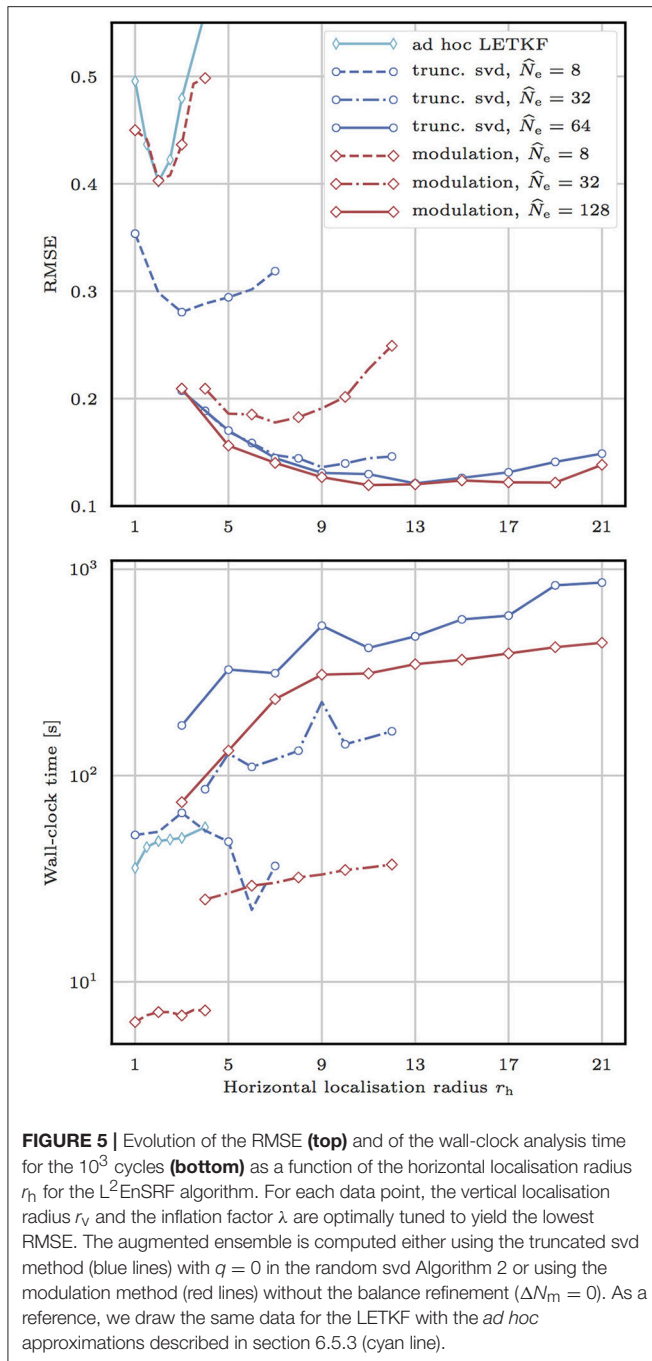
and r_h and r_v are the horizontal and vertical localisation radii, respectively.

6.6. Results

Figure 5 shows the evolution of the RMSE and of the wall-clock time of one analysis step as a function of the horizontal localisation radius r_h for the L^2 EnSRF and the LETKF. All simulations are performed on the same double *Intel Xeon E5-2680 platform*, which has a total of 24 cores. Parallelisation is enabled for the P_h independent local analyses using a fixed number of *OpenMP* threads $N_{thr} = 20$. In the L^2 EnSRF algorithm, the augmented ensemble is computed either using the truncated svd method with $q = 0$ in the random svd (Algorithm 2) or using the modulation method without the balance refinement ($\delta N_m = 0$). Preliminary experiments with $q > 0$ or $\delta N_m > 0$ (not shown here) did not display clear improvements in RMSE score over the cases $q = 0$ and $\delta N_m = 0$.

The LETKF produces rather high RMSE scores (compare to the standard deviation of the observation noise, which is 1), while not completely failing to reconstruct the true state. Although domain localisation in the horizontal direction is a powerful tool, vertical localisation is necessary in this DA problem. Because the weight functions of the channels are quite broad, observations cannot be considered local and domain localisation in the vertical direction is inefficient. By contrast, with a reasonable augmented ensemble size \widehat{N}_e , the L^2 EnSRF yields significantly lower RMSEs. This shows that combining domain localisation in the horizontal direction and covariance localisation in the vertical direction is an adequate approach to assimilate satellite radiances.

The comparison between the truncated svd and the modulation methods is not as simple as it was in the L96 test series. As expected, for a given augmented ensemble size \widehat{N}_e , the truncated svd method yields lower RMSE scores. However, for a given level of RMSE score, using the truncated svd method is not always the fastest approach. For example, the RMSE of the truncated svd method with $\widehat{N}_e = 64$ is approximately the same as the RMSE of the modulation method with $\widehat{N}_e = 128$, but in this case the modulation method is faster by a factor 1.5 on average. This can be explained by two factors. First, in the truncated svd method the vertical localisation matrix is not



applied in spectral space. Second, both the truncated svd and the modulation method benefit from parallelisation, but the parallelisation potential of the truncated svd method is not fully exploited here because our computational platform has a limited number of cores. This would change if we could use several threads for each local analysis. Finally, these results confirm that, for high dimensionals DA problems where the memory requirement is an issue, the truncated svd method is the best approach to obtain accurate results while using only a limited augmented ensemble size \hat{N}_e .

7. CONCLUSIONS

Localisation is widely used in DA to make the EnKF viable in high-dimensional systems. In the EnKF, two different approaches can be used to include localisation: domain localisation or covariance localisation. In this article, we have explored possible implementations for covariance localisation in the deterministic EnKF using augmented ensembles in the analysis step. We have discussed the two main difficulties related to augmented ensembles: how to construct the augmented ensemble and how to update the perturbations.

We have used two different methods to construct the augmented ensemble. The first one is based on a factorisation property of the background error covariance matrix. It is already widespread in the geophysical DA literature under the name *modulation*. For this method, we also introduced a *balance* refinement in order to smooth some variability between the state variables. As an alternative, we proposed a second method based on randomised svd techniques, which are very efficient when the localisation matrix is easy to apply. The random svd algorithm is commonly used in the statistical literature but it had never been applied in this context. We have called this approach *truncated svd* method.

We have shown how covariance localisation can be included in the perturbation update using the augmented ensemble framework. The resulting update formula [18] uses linear algebra in the augmented ensemble space. It is included in the generic LEnSRF detailed in this article.

Using numerical simulations of a very simple one-dimensional covariance model with 400 state variables, we have shown that the truncated svd method yields a much more accurate approximation of the background covariance than the modulation method. This result has been confirmed by twin simulations of the one-dimension L96 model with 400 variables. In a standard DA setup, we have found that the balance between fast computation of the augmented ensemble and fast perturbation update is in favor of the truncated svd method. In other words, for a given level of RMSE score, it is worth spending more time to construct a smaller but more reliable augmented ensemble with the truncated svd method and then use a fast perturbation update.

We have defined the L^2 EnSRF algorithm as an extension of the LEnSRF suited to assimilate satellite radiances in spatially extended models. It implements domain localisation in the horizontal direction in a similar way as the LETKF and covariance localisation in the vertical direction. Such an extension had been discussed by Bishop et al. [19] but without numerical illustration.

Finally, we have constructed a simple multilayer extension of the L96 model, called mL96 model. We have performed twin simulations of this model using a satellite-like observation operator. As expected, the LETKF hardly reconstructs the true state. By contrast, the L^2 EnSRF yields an estimate of the true state with an acceptable accuracy. We have concluded that using

domain localisation in the horizontal direction and covariance localisation in the vertical direction is an adequate approach to assimilate satellite radiances in a spatially extended model. For a given level of RMSE score, the modulation method is the fastest approach in this DA problem. This result is however mitigated by the fact that our computational setup does not use the full parallelisation potential of the truncated svd method. However, when the augmented ensemble size \widehat{N}_e is limited, the truncated svd method is the best approach to obtain accurate results.

REFERENCES

1. Evensen G. Sequential data assimilation with a nonlinear quasi-geostrophic model using Monte Carlo methods to forecast error statistics. *J Geophys Res.* (1994) **99**:10143–62. doi: 10.1029/94JC00572
2. Houtekamer PL, Mitchell HL, Pellerin G, Buehner M, Charron M, Spacek L, et al. Atmospheric data assimilation with an ensemble Kalman filter: results with real observations. *Mon Weather Rev.* (2005) **133**:604–20. doi: 10.1175/MWR-2864.1
3. Sakov P, Counillon F, Bertino L, Lisæter KA, Oke PR, Korabev A. TOPAZ4: an ocean-sea ice data assimilation system for the North Atlantic and Arctic. *Ocean Sci.* (2012) **8**:633–56. doi: 10.5194/os-8-633-2012
4. Bocquet M, Carrasi A. Four-dimensional ensemble variational data assimilation and the unstable subspace. *Tellus B* (2017) **69**:1304504. doi: 10.1080/16000870.2017.1304504
5. Houtekamer PL, Mitchell HL. A sequential ensemble Kalman filter for atmospheric data assimilation. *Mon Weather Rev.* (2001) **129**:123–37. doi: 10.1175/1520-0493(2001)129<0123:ASEKFF>2.0.CO;2
6. Ott E, Hunt BR, Szunyogh I, Zimin AV, Kostelich EJ, Corazza M, et al. A local ensemble Kalman filter for atmospheric data assimilation. *Tellus B* (2004) **56**:415–28. doi: 10.1111/j.1600-0870.2004.00076.x
7. Hamill TM, Whitaker JS, Snyder C. Distance-dependent filtering of background error covariance estimates in an ensemble Kalman filter. *Mon Weather Rev.* (2001) **129**:2776–90. doi: 10.1175/1520-0493(2001)129<2776:DDFOBE>2.0.CO;2
8. Sakov P, Bertino L. Relation between two common localisation methods for the EnKF. *Computat Geosci.* (2011) **15**:225–37. doi: 10.1007/s10596-010-9202-6
9. Campbell WF, Bishop CH, Hodyss D. Vertical covariance localization for satellite radiances in ensemble Kalman filters. *Mon Weather Rev.* (2010) **138**:282–90. doi: 10.1175/2009MWR3017.1
10. Miyoshi T, Sato Y. Assimilating satellite radiances with a local ensemble transform Kalman filter (LETKF) applied to the JMA global model (GSM). *SOLA* (2007) **3**:37–40. doi: 10.2151/sola.2007-010
11. Fertig EJ, Hunt BR, Ott E, Szunyogh I. Assimilating non-local observations with a local ensemble Kalman filter. *Tellus B* (2007) **59**:719–30. doi: 10.1111/j.1600-0870.2007.00260.x
12. Buehner M. Ensemble-derived stationary and flow-dependent background-error covariances: evaluation in a quasi-operational NWP setting. *Q J Roy Meteor Soc.* (2005) **131**:1013–43. doi: 10.1256/qj.04.15
13. Lorenc AC. The potential of the ensemble Kalman filter for NWP—a comparison with 4D-Var. *Q J Roy Meteor Soc.* (2003) **129**:3183–203. doi: 10.1256/qj.02.132
14. Bishop CH, Hodyss D. Ensemble covariances adaptively localized with ECO-RAP. Part 2: a strategy for the atmosphere. *Tellus B* (2009) **61**:97–111. doi: 10.1111/j.1600-0870.2008.00372.x
15. Brankart JM, Cosme E, Testut CE, Brasseur P, Verron J. Efficient local error parameterizations for square root or ensemble Kalman filters: application to a basin-scale ocean turbulent flow. *Mon Weather Rev.* (2011) **139**:474–93. doi: 10.1175/2010MWR3310.1

AUTHOR CONTRIBUTIONS

All authors have made a substantial, direct and intellectual contribution to the work, and approved it for publication.

ACKNOWLEDGMENTS

CEREA is a member of Institut Pierre–Simon Laplace (IPSL). MB acknowledges early discussions with N. Bousserez on the randomised svd techniques.

16. Bishop CH, Hodyss D. Adaptive ensemble covariance localization in ensemble 4D-VAR state estimation. *Mon Weather Rev.* (2011) **139**:1241–55. doi: 10.1175/2010MWR3403.1
17. Leng H, Song J, Lu F, Cao X. A new data assimilation scheme: the space-expanded ensemble localization Kalman filter. *Adv Meteorol.* (2013) **2013**:410812. doi: 10.1155/2013/410812
18. Bocquet M. Localization and the iterative ensemble Kalman smoother. *Q J Roy Meteor Soc.* (2016) **142**:1075–89. doi: 10.1002/qj.2711
19. Bishop CH, Whitaker JS, Lei L. Gain form of the ensemble transform Kalman filter and its relevance to satellite data assimilation with model space ensemble covariance localization. *Mon Weather Rev.* (2017) **145**:4575–92. doi: 10.1175/MWR-D-17-0102.1
20. Kretschmer M, Hunt BR, Ott E. Data assimilation using a climatologically augmented local ensemble transform Kalman filter. *Tellus B* (2015) **67**:26617. doi: 10.3402/tellusa.v67.26617
21. Bishop CH, Etherton BJ, Majumdar SJ. Adaptive sampling with the ensemble transform Kalman filter. Part I: theoretical aspects. *Mon Weather Rev.* (2001) **129**:420–36. doi: 10.1175/1520-0493(2001)129<0420:ASWTET>2.0.CO;2
22. Lorenc AC. Improving ensemble covariances in hybrid variational data assimilation without increasing ensemble size. *Q J Roy Meteor Soc.* (2017) **143**:1062–72. doi: 10.1002/qj.2990
23. Halko N, Martinsson PG, Tropp JA. Finding structure with randomness: probabilistic algorithms for constructing approximate matrix decompositions. *SIAM Rev.* (2011) **53**:217–88. doi: 10.1137/090771806
24. Lorenz EN, Emanuel KA. Optimal sites for supplementary weather observations: simulation with a small model. *J Atmos Sci.* (1998) **55**:399–414. doi: 10.1175/1520-0469(1998)055<0399:OSFSWO>2.0.CO;2
25. Evensen G. *Data Assimilation: The Ensemble Kalman Filter*. Berlin; Heidelberg: Springer-Verlag (2009).
26. Horn RA, Johnson CR. *Matrix Analysis*. New York, NY: Cambridge University Press (2012).
27. Sakov P, Oke PR. A deterministic formulation of the ensemble Kalman filter: an alternative to ensemble square root filters. *Tellus B* (2008) **60**:361–71. doi: 10.1111/j.1600-0870.2007.00299.x
28. Whitaker JS, Hamill TM. Ensemble data assimilation without perturbed observations. *Mon Weather Rev.* (2002) **130**:1913–24. doi: 10.1175/1520-0493(2002)130<1913:EDAWPO>2.0.CO;2
29. Desroziers G, Camino JT, Berre L. 4D-EnVar: link with 4D state formulation of variational assimilation and different possible implementations. *Q J Roy Meteor Soc.* (2014) **140**:2097–110. doi: 10.1002/qj.2325
30. Desroziers G, Arbogast É, Berre L. Improving spatial localization in 4D-EnVar. *Q J Roy Meteor Soc.* (2016) **142**:3171–85. doi: 10.1002/qj.2898
31. Arbogast É, Desroziers G, Berre L. A parallel implementation of a 4D-EnVar ensemble. *Q J Roy Meteor Soc.* (2017) **143**:2073–83. doi: 10.1002/qj.3061
32. Peres-Neto PR, Jackson DA, Somers KM. How many principal components? Stopping rules for determining the number of non-trivial axes revisited. *Comput Stat Data Ann.* (2005) **49**:974–97. doi: 10.1016/j.csda.2004.06.015
33. Gaspari G, Cohn SE. Construction of correlation functions in two and three dimensions. *Q J Roy Meteor Soc.* (1999) **125**:723–57. doi: 10.1002/qj.49712555417

34. Eckart C, Young G. The approximation of one matrix by another of lower rank. *Psychometrika* (1936) **1**:211–8. doi: 10.1007/BF02288367
35. Hunt BR, Kostelich EJ, Szunyogh I. Efficient data assimilation for spatiotemporal chaos: a local ensemble transform Kalman filter. *Phys D* (2007) **230**:112–26. doi: 10.1016/j.physd.2006.11.008
36. Anderson J, Lei L. Empirical localization of observation impact in ensemble Kalman filters. *Mon Weather Rev.* (2013) **141**:4140–53. doi: 10.1175/MWR-D-12-00330.1
37. Penny SG, Behringer DW, Carton JA, Kalnay E. A hybrid global ocean data assimilation system at NCEP. *Mon Weather Rev.* (2015) **143**:4660–77. doi: 10.1175/MWR-D-14-00376.1

Conflict of Interest Statement: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Copyright © 2019 Farchi and Bocquet. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.

APPENDIX

A. RECENTRE THE PERTURBATIONS

Let \mathbf{X} be an $N_x \times (N_e - 1)$ matrix. We want to construct a matrix that has the same empirical covariance matrix and which is centred. Since \mathbf{X} has rank at most $N_e - 1$, we need to find an $N_x \times N_e$ matrix \mathbf{Z} such that

$$\mathbf{Z}\mathbf{Z}^T = \mathbf{X}\mathbf{X}^T, \tag{A1}$$

$$\mathbf{Z}\mathbf{1} = \mathbf{0}. \tag{A2}$$

For $\epsilon \in \{-1, 1\}$, we define $\lambda = \sqrt{N_e}(\sqrt{N_e} - \epsilon)^{-1}$ and \mathbf{Q}_ϵ as the $N_e \times N_e$ matrix whose coefficients are

$$[\mathbf{Q}_\epsilon]_{i,j} = \begin{cases} \frac{\epsilon}{\sqrt{N_e}} & \text{if } i = 1 \text{ or } j = 1 \\ 1 - \frac{\lambda}{N_e} & \text{if } i = j \geq 2 \\ -\frac{\lambda}{N_e} & \text{else} \end{cases}. \tag{A3}$$

It can be easily checked that $\mathbf{Q}_\epsilon \mathbf{Q}_\epsilon^T = \mathbf{I}$ (i.e., \mathbf{Q}_ϵ is an orthogonal matrix) and $\mathbf{Q}_\epsilon \mathbf{1} = \mathbf{e}_1$, the first basis vector.

Let \mathbf{W} be the $N_x \times N_e$ matrix whose first column is zero and whose other columns are those of \mathbf{X} , that is

$$\mathbf{W} = [\mathbf{0}, \mathbf{X}]. \tag{A4}$$

By construction $\mathbf{Z} = \mathbf{W}\mathbf{Q}_\epsilon$ is a solution of Equations (A1) and (A2).

B. A RANDOM SVD ALGORITHM

B.1. The Algorithm

Algorithm 2 describes the random svd algorithm proposed by Halko et al. [23]. The objective of this algorithm is to efficiently compute an approximate truncated svd with P columns of the $M \times N$ matrix \mathbf{A} as a parallelisable alternative to Lanczos techniques.

The random svd algorithm is based on two ideas. First, suppose that there is a matrix \mathbf{Q} with P orthonormal columns which approximates the range of the matrix \mathbf{A} . In other words $\mathbf{A} \approx \mathbf{Q}\mathbf{Q}^T\mathbf{A}$. Then, an approximate truncated svd can be obtained for \mathbf{A} using the svd of the smaller matrix $\mathbf{Q}^T\mathbf{A}$. Second, the matrix \mathbf{Q} can be constructed using random draws. Indeed, if $\mathcal{X} = \{\mathbf{x}_1 \dots \mathbf{x}_P\}$ is a set of random vectors, then it is most likely a linearly independent set. Therefore, the set $\mathcal{Y} = \{\mathbf{A}\mathbf{x}_1 \dots \mathbf{A}\mathbf{x}_P\}$ is most likely linearly independent, which means that it spans the range of \mathbf{A} .

One major contribution of Halko et al. [23] and the references therein is that they have provided a mathematical justification of these ideas. In particular, they have given statistical performance bounds for the random svd algorithm and emphasised the fact that, on average, the (spectral or Frobenius) error of the resulting truncated svd with P columns should be close to the minimal error for a truncated svd with P columns.

Finally, Halko et al. [23] have introduced two elements to improve the numerical stability and efficiency of the random svd algorithm. The first element is a loop over $i \in \{1 \dots q\}$, which forces the algorithm to construct singular vectors of $(\mathbf{A}\mathbf{A}^T)^q \mathbf{A}$ instead of \mathbf{A} . However, $(\mathbf{A}\mathbf{A}^T)^q \mathbf{A}$ and \mathbf{A} share the same singular vectors. Moreover, the singular values of $(\mathbf{A}\mathbf{A}^T)^q \mathbf{A}$ decay faster than those of \mathbf{A} , which means that this technique enables a better approximation of the decomposition as shown by Corollary 10.10 of Halko et al. [23]. The second element is to include QR factorisations to make the algorithm less vulnerable to round-off errors. Both elements have been taken into account in Algorithm 2.

Algorithm 2 Random svd algorithm.

Require: $M \times N$ matrix \mathbf{A} (input); number of columns P in the truncated svd, number of iterations q (parameters).

Draw a $N \times P$ Gaussian random matrix Ω .

2: $\mathbf{B}_0 = \mathbf{A}\Omega$.

Compute the QR factorisation $\mathbf{Q}_0\mathbf{R}_0 = \mathbf{B}_0$.

4: **for** $i = 1$ **to** q **do**

$\widehat{\mathbf{B}}_i = \mathbf{A}^T\mathbf{Q}_{i-1}$.

6: Compute the QR factorisation $\widehat{\mathbf{Q}}_i\widehat{\mathbf{R}}_i = \widehat{\mathbf{B}}_i$.

$\mathbf{B}_i = \mathbf{A}\widehat{\mathbf{Q}}_i$.

8: Compute the QR factorisation $\mathbf{Q}_i\mathbf{R}_i = \mathbf{B}_i$.

end for

10: $\mathbf{B} = \mathbf{Q}_q^T\mathbf{A}$.

Compute the svd $\widehat{\mathbf{U}}\widehat{\Sigma}\mathbf{V}^T = \mathbf{B}$.

12: $\mathbf{U} = \mathbf{Q}_q\widehat{\mathbf{U}}$.

return Approximate truncated svd with P columns $\mathbf{A} \approx \mathbf{U}\widehat{\Sigma}\mathbf{V}^T$.

B.2. Application to the Prior Covariance Matrix

In section 3.2.4, we need to compute the truncated svd Equation (26) of the prior covariance matrix. To do this, we can apply Algorithm 2 using the input matrix $\mathbf{A} = \rho \circ (\mathbf{X}\mathbf{X}^T)$. The prior covariance matrix is a large $N_x \times N_x$ matrix. However, Algorithm 2 can work with the map

$$\begin{cases} \mathbb{R}^{N_x} \rightarrow \mathbb{R}^{N_x} \\ \mathbf{v} \mapsto (\rho \circ (\mathbf{X}\mathbf{X}^T)) \mathbf{v} \end{cases}, \tag{A5}$$

which can be efficiently computed using Equation (30). Steps 2, 5, 7, and 10 of Algorithm 2 can even be parallelised by applying $\mathbf{A} = \rho \circ (\mathbf{X}\mathbf{X}^T)$ independently to each column.

Finally, the approximate truncated svd

$$\mathbf{A} = \rho \circ (\mathbf{X}\mathbf{X}^T) \approx \mathbf{U}\widehat{\Sigma}\mathbf{V}^T, \tag{A6}$$

resulting from Algorithm 2 does not necessarily satisfy $\mathbf{U} = \mathbf{V}$ even though the input matrix is symmetric positive definite. The simplest fix is to make the additional approximation

$$\mathbf{U}\widehat{\Sigma}\mathbf{V}^T \approx \mathbf{U}\widehat{\Sigma}\mathbf{U}^T. \tag{A7}$$