A Random Line-Search Optimization Method via Modified Cholesky Decomposition for Non-linear Data Assimilation

Elias D. Nino-Ruiz$^{(\boxtimes)}$ (ID)

Applied Math and Computer Science Lab, Department of Computer Science,
Universidad del Norte, Barranquilla 0800001, Colombia
`enino@uninorte.edu.co`
`https://sites.google.com/a/vt.edu/eliasnino/`

Abstract. This paper proposes a line-search optimization method for non-linear data assimilation via random descent directions. The iterative method works as follows: at each iteration, quadratic approximations of the Three-Dimensional-Variational (3D-Var) cost function are built about current solutions. These approximations are employed to build sub-spaces onto which analysis increments can be estimated. We sample search-directions from those sub-spaces, and for each direction, a line-search optimization method is employed to estimate its optimal step length. Current solutions are updated based on directions along which the 3D-Var cost function decreases faster. We theoretically prove the global convergence of our proposed iterative method. Experimental tests are performed by using the Lorenz-96 model, and for reference, we employ a Maximum-Likelihood-Ensemble-Filter (MLEF) whose ensemble size doubles that of our implementation. The results reveal that, as the degree of observational operators increases, the use of additional directions can improve the accuracy of results in terms of ℓ_2-norm of errors, and even more, our numerical results outperform those of the employed MLEF implementation.

Keywords: Ensemble Kalman filter · Line-search optimization · Modified Cholesky decomposition

1 Introduction

Data Assimilation is the process by which imperfect numerical forecasts are adjusted according to real observations [1]. In sequential methods, a numerical forecast $\mathbf{x}^b \in \mathbb{R}^{n \times 1}$ is adjusted according to an array of observations $\mathbf{y} \in \mathbb{R}^{m \times 1}$ where n and m are the number of model components and the number of observations, respectively. When Gaussian assumptions are made in prior and observational errors, the posterior mode $\mathbf{x}^a \in \mathbb{R}^{n \times 1}$ can be estimated via the minimization of the Three Dimensional Variational (3D-Var) cost function:

$$\mathcal{J}(\mathbf{x}) = \frac{1}{2} \cdot \left\| \mathbf{x} - \mathbf{x}^b \right\|_{\mathbf{B}^{-1}}^2 + \frac{1}{2} \cdot \left\| \mathbf{y} - \mathcal{H}(\mathbf{x}) \right\|_{\mathbf{R}^{-1}}^2 , \tag{1}$$

where $\mathbf{B} \in \mathbb{R}^{n \times n}$ and $\mathbf{R} \in \mathbb{R}^{m \times m}$ are the background error and the data error covariance matrices, respectively. Likewise, $\mathcal{H}(\mathbf{x}) : \mathbb{R}^{n \times 1} \to \mathbb{R}^{m \times 1}$ is a (non-) linear observation operator which maps vector states to observation spaces. The solution to the optimization problem

$$\mathbf{x}^a = \arg \min_{\mathbf{x}} \mathcal{J}(\mathbf{x}), \tag{2}$$

is immediate when $\mathcal{H}(\mathbf{x})$ is linear (i.e., closed-form expressions can be obtained to compute $\mathbf{x}^a$) but, for non-linear observation operators, numerical optimization methods such as Newton's one must be employed [2]. However, since Newton's step is derived from a second-order Taylor polynomial, it can be too large with regard to the actual step size. Thus, line search methods can be employed to estimate optimal step lengths among Newton's method iterations. A DA method based on this idea is the Maximum-Likelihood-Ensemble-Filter (MLEF), which performs the assimilation step onto the ensemble space. However, the convergence of this method is not guaranteed (i.e., as the mismatch of gradients cannot be bounded), and even more, analysis increments can be impacted by sampling noise. We think that there is an opportunity to enhance line-search methods in the non-linear DA context by employing random descent directions onto which analysis increments can be estimated. Moreover, the analysis increments can be computed onto the model space to ensure global convergence.

This paper is organized as follows: in Sect. 2, we discuss topics related to linear and non-linear data assimilation as well as line-search optimization methods. Section 3 proposes an ensemble Kalman filter implementation via random descent directions. In Sect. 4, experimental tests are performed to assess the accuracy of our proposed filter implementation by using the Lorenz 96 model. Conclusions of this research are stated in Sect. 5.

2   Preliminaries

2.1   The Ensemble Kalman Filter

The Ensemble Kalman Filter (EnKF) is a sequential Monte-Carlo method for parameter and state estimation in highly non-linear models [3]. The popularity of the EnKF obeys to his simple formulation and relatively ease implementation. In the EnKF, an ensemble of model realizations is employed to estimate moments of the background error distribution [4]:

$$\mathbf{X}_k^b = \left[\mathbf{x}^{b[1]}, \mathbf{x}^{b[2]}, \ldots, \mathbf{x}^{b[N]} \right] \in \mathbb{R}^{n \times N} \tag{3}$$

where $\mathbf{x}^{b[e]} \in \mathbb{R}^{n \times 1}$ stands for the e-th ensemble member, for $1 \leq e \leq N$, at time k, for $0 \leq k \leq M$. Then, the ensemble mean:

$$\bar{\mathbf{x}}^b = \frac{1}{N} \cdot \sum_{e=1}^{N} \mathbf{x}^{b[e]} \in \mathbb{R}^{n \times 1}, \tag{4}$$

and the ensemble covariance matrix:

$$\mathbf{P}^b = \frac{1}{N-1} \cdot \boldsymbol{\Delta}\mathbf{X}^b \cdot \left[\boldsymbol{\Delta}\mathbf{X}^b\right]^T \in \mathbb{R}^{n \times n}, \tag{5}$$

act as estimates of the background state $\mathbf{x}^b$ and the background error covariance matrix $\mathbf{B}$, respectively, where the matrix of member deviations reads:

$$\boldsymbol{\Delta}\mathbf{X}^b = \mathbf{X}^b - \overline{\mathbf{x}}^b \cdot \mathbf{1}^T \in \mathbb{R}^{n \times N}. \tag{6}$$

Posterior members can be computed via the use synthetic observations:

$$\mathbf{X}^a = \mathbf{X}^b + \boldsymbol{\Delta}\mathbf{X}^a, \tag{7}$$

where the analysis increments can be obtained via the solution of the next linear system:

$$\left[\left[\mathbf{P}^b\right]^{-1} + \mathbf{H}^T \cdot \mathbf{R}^{-1} \cdot \mathbf{H}\right] \cdot \boldsymbol{\Delta}\mathbf{X}^a = \mathbf{H}^T \cdot \mathbf{R}^{-1} \cdot \mathbf{D}^s \in \mathbb{R}^{n \times N}, \tag{8}$$

and $\mathbf{D}^s \in \mathbb{R}^{m \times N}$ is the innovation matrix on the synthetic observations whose e-th column reads $\mathbf{y} - \mathbf{H} \cdot \mathbf{x}^{b[e]} + \boldsymbol{\varepsilon}^{[e]} \in \mathbb{R}^{m \times 1}$ with $\boldsymbol{\varepsilon}^{[e]} \sim \mathcal{N}\left(\mathbf{0}_m, \mathbf{R}\right)$. In practice, model dimensions range in the order of millions while ensemble sizes are constrained by the hundreds and as a direct consequence, sampling errors impact the quality of analysis increments. To counteract the effects of sampling noise, localizations methods are commonly employed [5], in practice. In the EnKF based on a modified Cholesky decomposition (EnKF-MC) [6] the following estimator is employed to approximate the precision covariance matrix of the background error distribution [7]:

$$\widehat{\mathbf{B}}^{-1} = \widehat{\mathbf{L}}^T \cdot \widehat{\mathbf{D}}^{-1} \cdot \widehat{\mathbf{L}} \in \mathbb{R}^{n \times n}, \tag{9}$$

where the Cholesky factor $\mathbf{L} \in \mathbb{R}^{n \times n}$ is a lower triangular matrix,

$$\left\{\widehat{\mathbf{L}}\right\}_{i,v} = \begin{cases} -\beta_{i,v} & , v \in P(i,r) \\ 1 & , i = v \\ 0 & , otherwise \end{cases}, \tag{10}$$

whose non-zero sub-diagonal elements $\beta_{i,v}$ are obtained by fitting models of the form,

$$\mathbf{x}_{[i]}^T = \sum_{v \in P(i,r)} \beta_{i,v} \cdot \mathbf{x}_{[v]}^T + \boldsymbol{\gamma}_i \in \mathbb{R}^{N \times 1}, 1 \leq i \leq n, \tag{11}$$

where $\mathbf{x}_{[i]}^T \in \mathbb{R}^{N \times 1}$ denotes the i-th row (model component) of the ensemble (3), components of vector $\boldsymbol{\gamma}_i \in \mathbb{R}^{N \times 1}$ are samples from a zero-mean Normal distribution with unknown variance σ^2, and $\mathbf{D} \in \mathbb{R}^{n \times n}$ is a diagonal matrix

whose diagonal elements read,

$$\{\mathbf{D}\}_{i,i} = \widehat{\mathbf{var}} \left(\mathbf{x}_{[i]}^T - \sum_{v \in P(i,r)} \beta_{i,v} \cdot \mathbf{x}_{[j]}^T \right)^{-1} \tag{12}$$

$$\approx \mathbf{var}\left(\boldsymbol{\gamma}_i\right)^{-1} = \frac{1}{\sigma^2} > 0, \text{ with } \{\mathbf{D}\}_{1,1} = \widehat{\mathbf{var}}\left(\mathbf{x}_{[1]}^T\right)^{-1}, \tag{13}$$

where $\mathbf{var}(\bullet)$ and $\widehat{\mathbf{var}}(\bullet)$ denote the actual and the empirical variances, respectively. The analysis equations can then be written as follows:

$$\mathbf{X}^a = \mathbf{X}^b + \left[\widetilde{\mathbf{L}}^T \cdot \widetilde{\mathbf{D}}^{-1/2}\right]^{-1} \cdot \mathbf{E} \in \mathbb{R}^{n \times N}, \tag{14}$$

where

$$\widehat{\mathbf{A}}^{-1} = \widetilde{\mathbf{L}}^T \cdot \widetilde{\mathbf{D}}^{-1} \cdot \widetilde{\mathbf{L}} = \widehat{\mathbf{B}}^{-1} + \mathbf{H}^T \cdot \mathbf{R}^{-1} \cdot \mathbf{H} \tag{15}$$
$$= \widehat{\mathbf{L}}^T \cdot \widehat{\mathbf{D}}^{-1} \cdot \widehat{\mathbf{L}} + \mathbf{H}^T \cdot \mathbf{R}^{-1} \cdot \mathbf{H} \in \mathbb{R}^{n \times n},$$

is an estimate of the posterior precision covariance matrix while the columns of matrix $\mathbf{E} \in \mathbb{R}^{n \times N}$ are formed by samples from a standard Normal distribution, $\widetilde{\mathbf{L}}^T \in \mathbb{R}^{n \times n}$ is a lower triangular matrix (with the same structure as $\widehat{\mathbf{L}}$), and $\widetilde{\mathbf{D}}^{-1} \in \mathbb{R}^{n \times n}$ is a diagonal matrix. Given the special structure of the left-hand side in (14), the direct inversion of the matrix $\widetilde{\mathbf{L}} \cdot \widetilde{\mathbf{D}}^{-1/2} \in \mathbb{R}^{n \times n}$ can be avoided [8, Algorithm 1].

2.2   Maximum Likelihood Ensemble Filter (MLEF)

To handle non-linear observation operators during assimilation steps, optimization based methods can be employed to estimate analysis increments. A well-known method in this context is the Maximum-Likelihood-Ensemble-Filter (MLEF) [9,10]. This square-root filter employs the ensemble space to compute analysis increments, this is:

$$\overline{\mathbf{x}}^a - \overline{\mathbf{x}}^b \in \mathbf{range}\left\{\boldsymbol{\Delta}\mathbf{X}\right\},$$

which is nothing but a pseudo square-root approximation of $\mathbf{B}^{1/2}$. Thus, vector states can be written as follows:

$$\mathbf{x} = \overline{\mathbf{x}}^b + \boldsymbol{\Delta}\mathbf{X} \cdot \mathbf{w}, \tag{16}$$

where $\mathbf{w} \in \mathbb{R}^{N \times 1}$ is a vector in redundant coordinates to be computed later. By replacing (16) in (1) one obtains:

$$\mathcal{J}(\mathbf{x}) = \mathcal{J}\left(\overline{\mathbf{x}}^b + \boldsymbol{\Delta}\mathbf{X} \cdot \mathbf{w}\right) = \frac{N-1}{2} \cdot \|\mathbf{w}\|^2 + \frac{1}{2} \cdot \left\|\mathbf{y} - \mathcal{H}\left(\overline{\mathbf{x}}^b + \boldsymbol{\Delta}\mathbf{X} \cdot \mathbf{w}\right)\right\|_{\mathbf{R}^{-1}}^2. \tag{17}$$

The optimization problem to solve reads:

$$\mathbf{w}^* = \arg \min_{\mathbf{w}} \mathcal{J}\left(\overline{\mathbf{x}}^b + \boldsymbol{\Delta}\mathbf{X} \cdot \mathbf{w}\right). \tag{18}$$

This problem can be numerically solved via Line-Search (LS) and/or Trust-Region methods. However, convergence is not ensured since gradient approximations are performed onto a reduce space whose dimension is much smaller than that of the model one.

2.3   Line Search Optimization Methods

The solution of optimization problems of the form (2) can be approximated via Numerical Optimization. In this context, solutions are obtained via iterations:

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \boldsymbol{\Delta}\mathbf{s}_k, \tag{19}$$

wherein k denotes iteration index, and $\boldsymbol{\Delta}\mathbf{s}_k \in \mathbb{R}^{n \times 1}$ is a descent direction, for instance, the gradient descent direction [11]

$$\boldsymbol{\Delta}\mathbf{s}_k = -\nabla \mathcal{J}\left(\mathbf{x}_k\right), \tag{20a}$$

the Newton's step [12],

$$\nabla^2 \mathcal{J}\left(\mathbf{x}_k\right) \cdot \boldsymbol{\Delta}\mathbf{s}_k = -\nabla \mathcal{J}\left(\mathbf{x}_k\right), \tag{20b}$$

or a quasi-Newton based method [13],

$$\mathbf{P}_k \cdot \boldsymbol{\Delta}\mathbf{s}_k = -\nabla \mathcal{J}\left(\mathbf{x}_k\right), \tag{20c}$$

where $\mathbf{P}_k \in \mathbb{R}^{n \times n}$ is a positive definite matrix. A concise survey of Newton based methods can be consulted in [14]. Since step lengths in (20) are based on first or second order Taylor polynomials, the step size can be chosen via line search [15] and/or trust region [16] methods. Thus, we can ensure global convergence of optimization methods to stationary points of the cost function (1). This holds as long as some assumptions over functions, gradients, and (potentially) Hessians are preserved [17]. In the context of line search, the following assumptions are commonly done:

C1 A lower bound of $\mathcal{J}(\mathbf{x})$ exists on $\Omega_0 = \{\mathbf{x} \in \mathbb{R}^{n \times 1}, \mathcal{J}(\mathbf{x}) \leq \mathcal{J}\left(\mathbf{x}^\dagger\right)\}$, where $\mathbf{x}^\dagger \in \mathbb{R}^{n \times 1}$ is available.

C2 There is a constant $\mathbf{L}$ such as:

$$\|\nabla \mathcal{J}(\mathbf{x}) - \nabla \mathcal{J}(\mathbf{z})\| \leq L \cdot \|\mathbf{x} - \mathbf{z}\|, \text{ for } \mathbf{x}, \mathbf{z} \in B, \text{ and } L > 0,$$

where B is an open convex set which contains Ω_0. These conditions together with iterates of the form,

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha \cdot \boldsymbol{\Delta}\mathbf{s}_k, \tag{21}$$

ensure global convergence [18] as long as α is chosen as an (approximated) minimizer of

$$\alpha^* = \arg \min_{\alpha \geq 0} \mathcal{J}\left(\mathbf{x}_k + \alpha \cdot \boldsymbol{\Delta}\mathbf{s}_k\right). \tag{22}$$

In practice, rules for choosing step-size such as the Goldstein rule [19], the Strong Wolfe rule [20], and the Halving method [21] are employed to partially solve (22). Moreover, soft computing methods can be employed for solving (22) [22].

3 Proposed Method: An Ensemble Kalman Filter Implementation via Line-Search Optimization and Random Descent Directions

In this section, we propose an iterative method to estimate the solution of the optimization problem (2). We detail our filter derivation, and subsequently, we theoretically prove the convergence of our method.

3.1 Filter Derivation

Starting with the forecast ensemble (3), we compute an estimate $\widehat{\mathbf{B}}^{-1}$ of the precision covariance $\mathbf{B}^{-1}$ via modified Cholesky decomposition. Then, we perform an iterative process as follows: let $\mathbf{x}_0 = \overline{\mathbf{x}}^b$, at iteration k, for $0 \leq k \leq K$, where K is the maximum number of iterations, we build a quadratic approximation of $\mathcal{J}(\mathbf{x})$ about $\mathbf{x}_k$

$$\mathcal{J}_k(\mathbf{x}) = \frac{1}{2} \cdot \|\mathbf{x} - \mathbf{x}_k\|_{\widehat{\mathbf{B}}^{-1}}^2 + \frac{1}{2} \cdot \left\|\mathbf{y} - \widehat{\mathcal{H}}_k(\mathbf{x})\right\|_{\mathbf{R}^{-1}}^2, \tag{23a}$$

where

$$\widehat{\mathcal{H}}_k(\mathbf{x}) = \mathcal{H}(\mathbf{x}_k) + \mathbf{H}_k \cdot [\mathbf{x} - \mathbf{x}_k],$$

and $\mathbf{H}_k$ is the Jacobian of $\mathcal{H}(\mathbf{x})$ at $\mathbf{x}_k$. The gradient of (23a) reads:

$$\begin{aligned}
\nabla \mathcal{J}_k(\mathbf{x}) &= \widehat{\mathbf{B}}^{-1} \cdot [\mathbf{x} - \mathbf{x}_k] - \mathbf{H}_k^T \cdot \mathbf{R}^{-1} \cdot [\mathbf{d}_k - \mathbf{H}_k \cdot \mathbf{x}] \\
&= \left[\widehat{\mathbf{B}}^{-1} + \mathbf{H}_k^T \cdot \mathbf{R}^{-1} \cdot \mathbf{H}_k\right] \cdot \mathbf{x} - \mathbf{H}_k^T \cdot \mathbf{R} \cdot \mathbf{d}_k \in \mathbb{R}^{n \times 1},
\end{aligned}$$

where $\mathbf{d}_k = \mathbf{y} - \mathcal{H}(\mathbf{x}_k) + \mathbf{H}_k \cdot \mathbf{x}_k \in \mathbb{R}^{m \times 1}$. Readily, the Hessian of (23a) is

$$\nabla^2 \mathcal{J}_k(\mathbf{x}) = \widehat{\mathbf{B}}^{-1} + \mathbf{H}_k^T \cdot \mathbf{R}^{-1} \cdot \mathbf{H}_k \in \mathbb{R}^{n \times n}, \tag{23b}$$

and therefore, the Newton's step can be written as follows:

$$\begin{aligned}
\mathbf{p}_k(\mathbf{x}) &= -\left[\widehat{\mathbf{B}}^{-1} + \mathbf{H}_k^T \cdot \mathbf{R}^{-1} \cdot \mathbf{H}_k\right]^{-1} \\
&\quad \cdot \left[\left[\widehat{\mathbf{B}}^{-1} + \mathbf{H}_k^T \cdot \mathbf{R}^{-1} \cdot \mathbf{H}_k\right] \cdot \mathbf{x} - \mathbf{H}_k^T \cdot \mathbf{R} \cdot \mathbf{d}_k\right], \\
&= -\mathbf{x} + \left[\widehat{\mathbf{B}}^{-1} + \mathbf{H}_k^T \cdot \mathbf{R}^{-1} \cdot \mathbf{H}_k\right]^{-1} \cdot \mathbf{H}_k^T \cdot \mathbf{R} \cdot \mathbf{d}_k. \tag{23c}
\end{aligned}$$

As we mentioned before, the step size (23c) is based on a quadratic approximation of $\mathcal{J}(\mathbf{x})$ and depending how highly non-linear is $\mathcal{H}(\mathbf{x})$, the direction (23c) can poorly estimate the analysis increments. Thus, we compute U random directions based on the Newton's one as follows:

$$\mathbf{q}_{u,k} = \Pi_u \cdot \mathbf{p}_k(\mathbf{x}_k) \in \mathbb{R}^{n \times 1}, \text{ for } 1 \leq u \leq U, \tag{23d}$$

where the matrices $\Pi_u \in \mathbb{R}^{n \times n}$ are symmetric positive definite and these are randomly formed with $\|\Pi_u\| = 1$. We constraint the increments to the space spanned by the vectors (23d), this is

$$\mathbf{x}_{k+1} - \mathbf{x}_k = \mathbf{range}\left\{\mathbf{Q}_k\right\},$$

where the u-th column of $\mathbf{Q}_k \in \mathbb{R}^{n \times U}$ reads $\mathbf{q}_{u,k}$. Thus,

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{Q}_k \cdot \boldsymbol{\gamma}^*, \tag{23e}$$

where $\boldsymbol{\gamma}^* \in \mathbb{R}^{U \times 1}$ is estimated by solving the following optimization problem

$$\boldsymbol{\gamma}^* = \arg \min_{\boldsymbol{\gamma}} \mathcal{J}\left(\mathbf{x}_k + \mathbf{Q}_k \cdot \boldsymbol{\gamma}\right). \tag{23f}$$

To solve (23f), we proceed as follows: generate Z random vectors $\boldsymbol{\gamma}_z \in \mathbb{R}^{U \times 1}$, for $1 \leq z \leq Z$, with $\|\boldsymbol{\gamma}_z\| = 1$. We then, for each direction $\mathbf{Q}_k \cdot \boldsymbol{\gamma}_z \in \mathbb{R}^{n \times 1}$, we solve the following one-dimensional optimization problem

$$\alpha_z^* = \arg \min_{\alpha_z} \mathcal{J}\left(\mathbf{x}_k + \alpha_z \cdot [\mathbf{Q}_k \cdot \boldsymbol{\gamma}_z]\right), \tag{23g}$$

and therefore, an estimate of the next iterate (23e) reads:

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{Q}_k \cdot [\alpha_k^* \cdot \boldsymbol{\gamma}_k], \tag{23h}$$

where the pair $(\alpha_k^*, \boldsymbol{\gamma}_k)$ is chosen as the duple $(\alpha_z^*, \boldsymbol{\gamma}_z)$ which provide the best profit (minimum value) in (23g), for $1 \leq z \leq Z$. The overall process detailed in equations (23) is repeated until some stopping criterion is satisfied (i.e., we let a maximum number of iterations K).

Based on the iterations (23h), we estimate the analysis state as follows:

$$\overline{\mathbf{x}}^a = \overline{\mathbf{x}}^b + \sum_{k=1}^{K} \mathbf{Q}_k \cdot [\alpha_k^* \cdot \boldsymbol{\gamma}_k] = \mathbf{x}_K. \tag{24}$$

The inverse of the Hessian (23b) provides an estimate of the posterior error covariance matrix. Thus, posterior members (analysis ensemble) can be sampled as follows:

$$\mathbf{x}^{a[e]} \sim \mathcal{N}\left(\overline{\mathbf{x}}^a, \left[\nabla^2 \mathcal{J}_K\left(\overline{\mathbf{x}}^a\right)\right]^{-1}\right). \tag{25}$$

To efficiently perform the sampling process (25) the reader can consult [23]. Afterwards, the analysis members are propagated in time until a new observation is available. We name this formulation the Random Ensemble Kalman Filter (RAN-EnKF).

3.2   Convergence of the Analysis Step in the RAN-EnKF

For proving the convergence of our method, we consider the assumptions C1, C2, and

$$\nabla \mathcal{J}\left(\mathbf{x}_k\right)^T \cdot \mathbf{q}_{u,k} < 0, \ \text{ for } 1 \leq u \leq U. \tag{26}$$

The next Theorem states the necessary conditions in order to ensure global convergence of the analysis step in the RAN-EnKF.

Theorem 1. *If* (2.3), (2.3), *and* (26) *hold, then the RSLS-RD with exact line search generates an infinite sequence* $\{\mathbf{x}_k\}_{u=0}^{\infty}$, *then*

$$\lim_{k \to \infty} \left[\frac{-\nabla \mathcal{J}\left(\mathbf{x}_k\right)^T \cdot \mathbf{Q}_k \cdot \boldsymbol{\gamma}^*}{\|\mathbf{Q}_k \cdot \boldsymbol{\gamma}^*\|} \right]^2 = 0 \tag{27}$$

holds.

Proof. By Taylor series and the Mean Value Theorem we know that,

$$\mathcal{J}\left(\mathbf{x}^{(u)} + \alpha^* \cdot \mathbf{Q}_k \cdot \boldsymbol{\gamma}^*\right) = \mathcal{J}\left(\mathbf{x}_k\right)$$
$$+ \alpha^* \cdot \int_0^1 \nabla \mathcal{J}\left(\mathbf{x}_k + \alpha^* \cdot t \cdot \mathbf{Q}_k \cdot \boldsymbol{\gamma}^*\right)^T$$
$$\cdot \mathbf{Q}_k \cdot \boldsymbol{\gamma}^* \cdot dt,$$

and therefore,

$$\mathcal{J}\left(\mathbf{x}_k\right) - \mathcal{J}\left(\mathbf{x}_{k+1}\right) \geq -\alpha^* \cdot \int_0^1 \nabla \mathcal{J}\left(\mathbf{x}_k + \alpha^* \cdot t \cdot \mathbf{Q}_k \cdot \boldsymbol{\gamma}^*\right)^T$$
$$\cdot \mathbf{Q}_k \cdot \boldsymbol{\gamma}^* \cdot dt$$

for any $\mathbf{x}_{k+1}$ on the ray $\mathbf{x}_k + \alpha \cdot \mathbf{Q}_k \cdot \boldsymbol{\gamma}^*$, with $\alpha \in [0, 1]$, we have

$$\mathcal{J}\left(\mathbf{x}_k\right) - \mathcal{J}\left(\mathbf{x}_{k+1}\right) \geq \mathcal{J}\left(\mathbf{x}_k\right) - \mathcal{J}\left(\mathbf{x}_k + \alpha^* \cdot \mathbf{Q}_k \cdot \boldsymbol{\gamma}^*\right),$$

hence:

$$\mathcal{J}\left(\mathbf{x}_k\right) - \mathcal{J}\left(\mathbf{x}_{k+1}\right) \geq -\alpha^* \cdot \nabla \mathcal{J}\left(\mathbf{x}_k\right)^T \cdot \mathbf{Q}_k \cdot \boldsymbol{\gamma}^*$$
$$- \alpha^* \cdot \int_0^1 \left[\nabla \mathcal{J}\left(\mathbf{x}_k + \alpha^* \cdot t \cdot \mathbf{Q}_k \cdot \boldsymbol{\gamma}^*\right) - \nabla \mathcal{J}\left(\mathbf{x}_k\right)\right]^T$$
$$\cdot \mathbf{Q}_k \cdot \boldsymbol{\gamma}^* \cdot dt,$$

by the Cauchy Schwarz inequality we have

$$\mathcal{J}\left(\mathbf{x}_k\right) - \mathcal{J}\left(\mathbf{x}_{k+1}\right) \geq -\alpha^* \cdot \nabla \mathcal{J}\left(\mathbf{x}_k\right)^T \cdot \mathbf{Q}_k \cdot \boldsymbol{\gamma}^*$$
$$- \alpha^* \cdot \int_0^1 \|\nabla \mathcal{J}\left(\mathbf{x}_k + \alpha^* \cdot t \cdot \mathbf{Q}_k \cdot \boldsymbol{\gamma}^*\right) - \nabla \mathcal{J}\left(\mathbf{x}_k\right)\|$$
$$\cdot \|\mathbf{Q}_k \cdot \boldsymbol{\gamma}^*\| \cdot dt$$
$$\geq -\alpha^* \cdot \nabla \mathcal{J}\left(\mathbf{x}_k\right)^T \cdot \mathbf{Q}_k \cdot \boldsymbol{\gamma}^*$$
$$- \alpha^* \cdot \int_0^1 L \cdot \|\alpha^* \cdot t \cdot \mathbf{Q}_k \cdot \boldsymbol{\gamma}^*\| \cdot \|\mathbf{Q}_k \cdot \boldsymbol{\gamma}^*\| \cdot dt$$
$$= -\alpha^* \cdot \nabla \mathcal{J}\left(\mathbf{x}_k\right)^T \cdot \mathbf{Q}_k \cdot \boldsymbol{\gamma}^*$$
$$- \alpha^* \cdot L \cdot \|\mathbf{Q}_k \cdot \boldsymbol{\gamma}^*\| \cdot \int_0^1 \|t \cdot \alpha^* \cdot \mathbf{Q}_k \cdot \boldsymbol{\gamma}^*\| \cdot dt$$
$$= -\alpha^* \cdot \nabla \mathcal{J}\left(\mathbf{x}_k\right)^T \cdot \mathbf{Q}_k \cdot \boldsymbol{\gamma}^* - \frac{1}{2} \cdot \alpha^{*2} \cdot L \cdot \|\mathbf{Q}_k \cdot \boldsymbol{\gamma}^*\|^2,$$

choose

$$\alpha^* = -\frac{\nabla \mathcal{J}\left(\mathbf{x}_k\right)^T \cdot \mathbf{Q}_k \cdot \boldsymbol{\gamma}^*}{L \cdot \|\mathbf{Q}_k \cdot \boldsymbol{\gamma}^*\|^2},$$

therefore,

$$
\begin{aligned}
\mathcal{J}\left(\mathbf{x}_k\right) - \mathcal{J}\left(\mathbf{x}_{k+1}\right) &\geq \frac{\left[\nabla \mathcal{J}\left(\mathbf{x}_k\right)^T \cdot \mathbf{Q}_k \cdot \boldsymbol{\gamma}^*\right]^2}{L \cdot \|\mathbf{Q}_k \cdot \boldsymbol{\gamma}^*\|^2} \\
&\quad - \frac{1}{2} \cdot \frac{\left[-\nabla \mathcal{J}\left(\mathbf{x}_k\right)^T \cdot \mathbf{Q}_k \cdot \boldsymbol{\gamma}^*\right]^2}{L \cdot \|\mathbf{Q}_k \cdot \boldsymbol{\gamma}^*\|^2} \\
&= \frac{1}{2 \cdot L} \cdot \left[-\frac{\nabla \mathcal{J}\left(\mathbf{x}_k\right)^T \cdot \mathbf{Q}_k \cdot \boldsymbol{\gamma}^*}{\|\mathbf{Q}_k \cdot \boldsymbol{\gamma}^*\|}\right]^2.
\end{aligned}
$$

By (2.3), and (26), it follows that $\{\mathcal{J}\left(\mathbf{x}_k\right)\}_{k=0}^{\infty}$ is a monotone decreasing number sequence and it has a bound below, therefore $\{\mathcal{J}\left(\mathbf{x}_k\right)\}_{k=0}^{\infty}$ has a limit, and consequently (27) holds.

We are now ready to test our proposed method numerically.

4 Experimental Results

For the experiments, we consider non-linear observation operators, a current challenge in the context of DA [6,24]. We make use of the Lorenz-96 model [25] as our surrogate model during the experiments. The Lorenz-96 model is described by the following set of ordinary differential equations [26]:

$$\frac{dx_j}{dt} = \begin{cases} (x_2 - x_{n-1}) \cdot x_n - x_1 + F & \text{for } j = 1, \\ (x_{j+1} - x_{j-2}) \cdot x_{j-1} - x_j + F & \text{for } 2 \leq j \leq n-1, \\ (x_1 - x_{n-2}) \cdot x_{n-1} - x_n + F & \text{for } j = n, \end{cases} \quad (28)$$

where F is external force and $n = 40$ is the number of model components. Periodic boundary conditions are assumed. When $F = 8$ units the model exhibits chaotic behavior, which makes it a relevant surrogate problem for atmospheric dynamics [27,28]. A time unit in the Lorenz-96 represents 7 days in the atmosphere. We create the initial pool $\widehat{\mathbf{X}^b}_0$ of $\widehat{N} = 10^4$ members. The error statistics of observations are as follows:

$$\mathbf{y}_k \sim \mathcal{N}\left(\mathcal{H}_k\left(\mathbf{x}_k^*\right), [\epsilon^o]^2 \cdot \mathbf{I}\right), \text{ for } 0 \leq k \leq M,$$

where the standard deviations of observational errors $\epsilon^o = 10^{-2}$. The components are randomly chosen at the different assimilation cycles. We use the

non-smooth and non-linear observation operator [29]:

$$\{\mathcal{H}(\mathbf{x})\}_j = \frac{\{\mathbf{x}\}_j}{2} \cdot \left[\left(\frac{\left| \{\mathbf{x}\}_j \right|}{2} \right)^{\beta-1} + 1 \right], \tag{29}$$

where j denotes the j-th observed component from the model state. Likewise, $\beta \in \{1, 3, 5, 7, 9\}$. Since the observation operator (29) is non-smooth, gradients of (1) are approximated by using the ℓ_2-norm. A full observational network is available at assimilation steps. The ensemble size for the benchmarks is $N = 20$. These members are randomly chosen from the pool $\widehat{\mathbf{X}^b}_0$ for the different experiments in order to form the initial ensemble $\mathbf{X}_0^b$ for the assimilation window. Evidently, $\mathbf{X}_0^b \subset \widehat{\mathbf{X}^b}_0$. The ℓ_2-norm of errors are utilized as a measure of accuracy at the assimilation step k,

$$\mathcal{E}(\mathbf{x}_k, \mathbf{x}^*) = \sqrt{[\mathbf{x}^* - \mathbf{x}_k]^T \cdot [\mathbf{x}^* - \mathbf{x}_k]}, \tag{30}$$

where $\mathbf{x}^*$ and $\mathbf{x}_k$ are the reference and current solution at iteration k, respectively. The initial background error, in average, reads $\epsilon^b \approx 31.73$. By convenience, this value is expressed in the log scale: $\log(\epsilon^b) = 3.45$. We consider a single assimilation cycle for the experiments. We try sub-spaces of dimensions $U \in \{10, 20, 30\}$ and number of samples from those spaces of $Z \in \{10, 30, 50\}$. We set a maximum number of iterations of 40. We compare our results with those obtained by the MLEF with $N = 40$, note that, the ensemble size in the MLEF doubles the ones employed by our method.

We group the results in Figs. 1 and 2 by sub-space size and sample size (sub-space dimension), respectively. As can be seen, the RAN-EnKF outperforms the MLEF in terms of ℓ_2-norm of errors, for all cases. Note that the error differences between the compared filter implementations are given by order of magnitudes. This can be explained as follows: the MLEF method performs the assimilation step onto a space given by the ensemble size; this is equivalent to perform an assimilation process by using the sample covariance matrix (5) whose quality is impacted by sampling errors. Contrarily, in our formulation, we employ sub-spaces whose basis vectors rely on the precision covariance (9) and, therefore, the impact of sampling errors is mitigated during optimization steps. As the degree β of the observation operator increases, the accuracy of the MLEF degrades, and consequently, this method diverges for the largest β value. On the other hand, convergence is always achieved in the RAN-EnKF method; this should be expected based on the theoretical results of Theorem 1. It should be noted that, as the β value increases, the 3D-Var cost function becomes highly non-linear, and as a consequence, more iterations are needed to decrease errors (as in any iterative optimization method). In general, it can be seen that as the number of samples Z increases, the results can be improved regardless of the sub-space dimension U (i.e., for $Z = 10$). However, it is clear that, for highly non-linear observation operators, it is better to have small sub-spaces and a large number of samples.
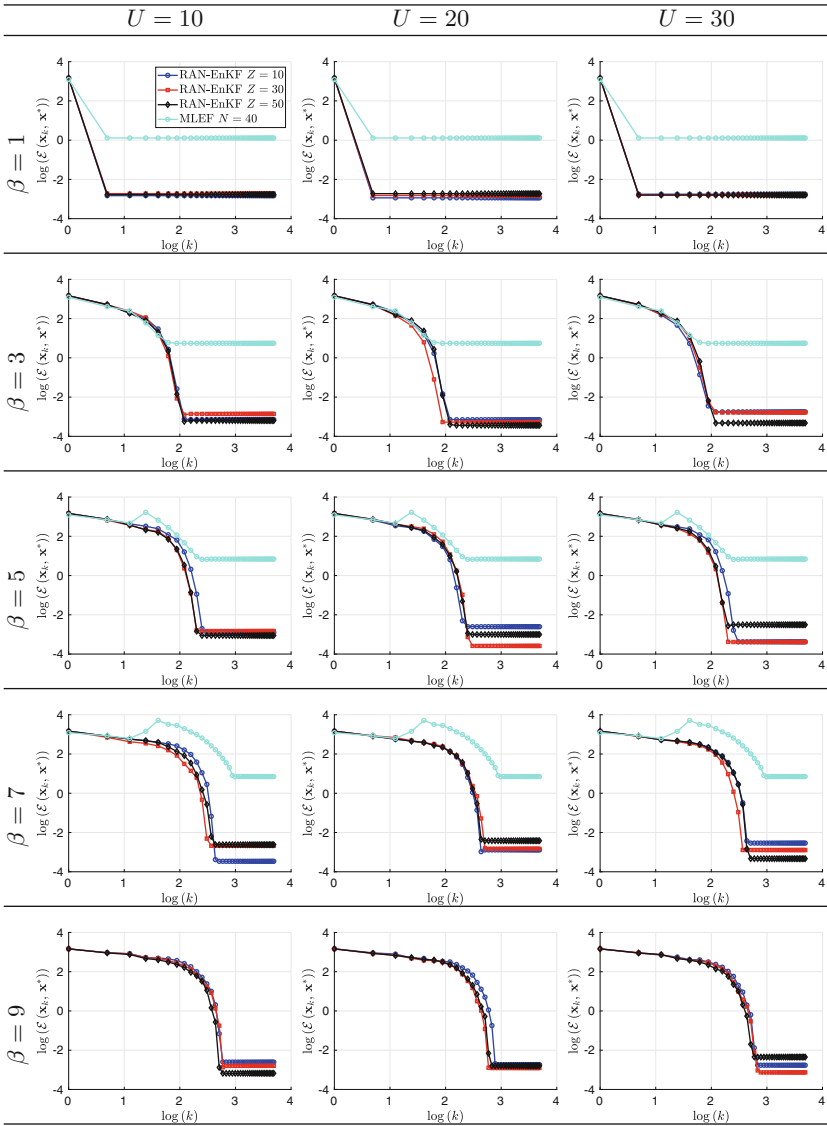
Fig. 1. ℓ_2-norm of errors in the log-scale for the 3D-Var Optimization Problem with different degrees β of the observation operator and dimension of sub-spaces U. For the largest β value, the MLEF diverges and therefore, its results are not reported.
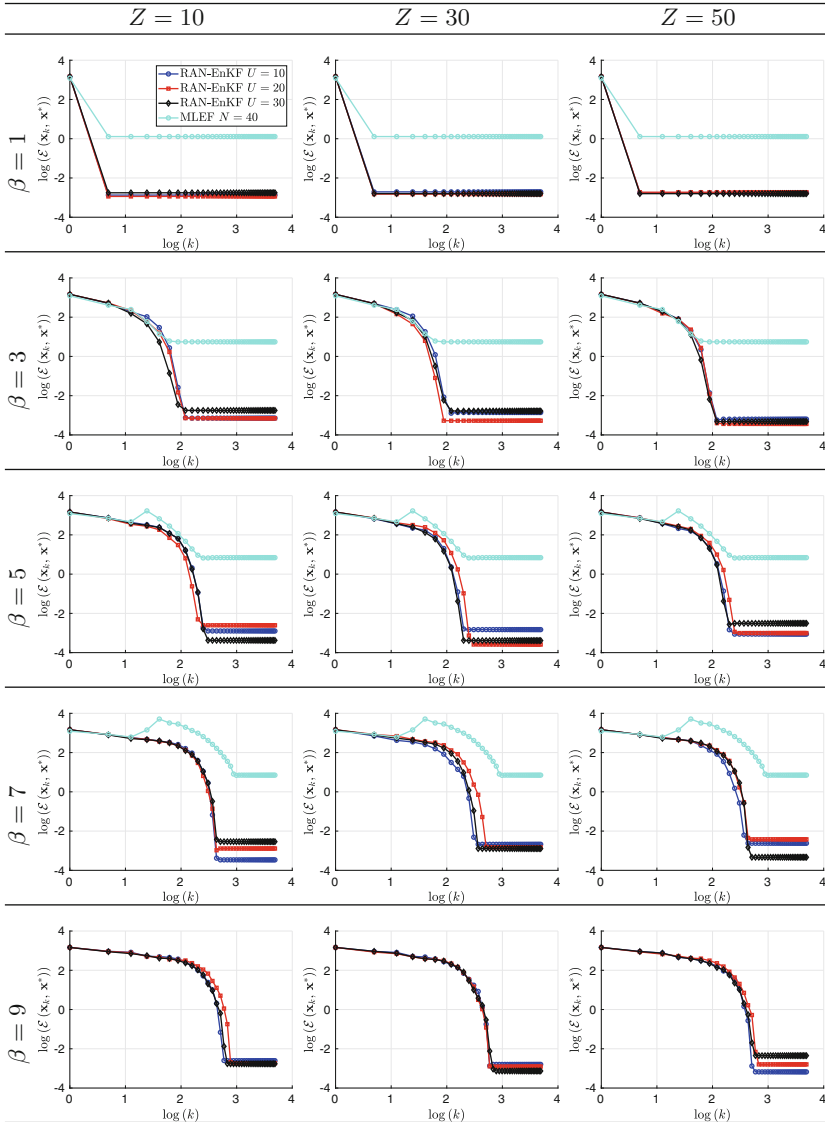
Fig. 2. ℓ_2-norm of errors in the log-scale for the 3D-Var Optimization Problem with different degrees β of the observation operator and number of samples Z. For the largest β value, the MLEF diverges and therefore, its results are not reported.

5   Conclusions

In this paper, we propose an ensemble Kalman filter implementation via line-search optimization; we name it a Random Ensemble Kalman Filter (RAN-EnKF). The proposed method proceeds as follows: an ensemble of model realiza-

tion is employed to estimate background moments, and then, quadratic approximations of the 3D-Var cost function are obtained among iterations via the linearization of the observation operator about current solutions. These approximations serve to estimate descent directions of the 3D-Var cost function, which are perturbed to obtain additional directions onto which analysis increments can be computed. We theoretically prove the global convergence of our optimization method. Experimental tests are performed by using the Lorenz 96 model and the Maximum-Likelihood-Ensemble-Filter formulation. The results reveal that the RAN-EnKF outperforms the MLEF in terms of ℓ_2-norm of errors, and even more, it is able to achieve convergence in cases wherein the MLEF diverges.

References

1. Nino-Ruiz, E.D., Guzman-Reyes, L.G., Beltran-Arrieta, R.: An adjoint-free four-dimensional variational data assimilation method via a modified Cholesky decomposition and an iterative Woodbury matrix formula. Nonlinear Dyn. **99**(3), 2441–2457 (2020)
2. Nino-Ruiz, E.D.: Non-linear data assimilation via trust region optimization. Comput. Appl. Math. **38**(3), 1–26 (2019). https://doi.org/10.1007/s40314-019-0901-x
3. Evensen, G.: The ensemble Kalman filter: theoretical formulation and practical implementation. Ocean Dyn. **53**(4), 343–367 (2003)
4. Stroud, J.R., Katzfuss, M., Wikle, C.K.: A Bayesian adaptive ensemble Kalman filter for sequential state and parameter estimation. Mon. Weather Rev. **146**(1), 373–386 (2018)
5. Greybush, S.J., Kalnay, E., Miyoshi, T., Ide, K., Hunt, B.R.: Balance and ensemble Kalman filter localization techniques. Mon. Weather Rev. **139**(2), 511–522 (2011)
6. Nino-Ruiz, E.D., Sandu, A., Deng, X.: An ensemble Kalman filter implementation based on modified Cholesky decomposition for inverse covariance matrix estimation. SIAM J. Sci. Comput. **40**(2), A867–A886 (2018)
7. Bickel, P.J., Levina, E., et al.: Regularized estimation of large covariance matrices. Ann. Statist. **36**(1), 199–227 (2008)
8. Nino-Ruiz, E.: A matrix-free posterior ensemble Kalman filter implementation based on a modified Cholesky decomposition. Atmosphere **8**(7), 125 (2017)
9. Zupanski, M.: Maximum likelihood ensemble filter: theoretical aspects. Mon. Weather Rev. **133**(6), 1710–1726 (2005)
10. Zupanski, D., Zupanski, M.: Model error estimation employing an ensemble data assimilation approach. Mon. Weather Rev. **134**(5), 1337–1354 (2006)
11. Savard, G., Gauvin, J.: The steepest descent direction for the nonlinear bilevel programming problem. Oper. Res. Lett. **15**(5), 265–272 (1994)
12. Pan, V.Y., Branham, S., Rosholt, R.E., Zheng, A.-L.: Newton's iteration for structured matrices. In: Fast Reliable Algorithms for Matrices with Structure, pp. 189–210. SIAM (1999)
13. Nocedal, J.: Updating Quasi-Newton matrices with limited storage. Math. Comput. **35**(151), 773–782 (1980)

14. Knoll, D.A., Keyes, D.E.: Jacobian-free Newton-Krylov methods: a survey of approaches and applications. J. Comput. Phys. **193**(2), 357–397 (2004)
15. Hosseini, S., Huang, W., Yousefpour, R.: Line search algorithms for locally Lipschitz functions on Riemannian manifolds. SIAM J. Optim. **28**(1), 596–619 (2018)
16. Conn, A.R., Gould, N.I.M., Toint, Ph.L.: Trust region methods, vol. 1. SIAM (2000)
17. Shi, Z.-J.: Convergence of line search methods for unconstrained optimization. Appl. Math. Comput. **157**(2), 393–405 (2004)
18. Zhou, W., Akrotirianakis, I.G., Yektamaram, S., Griffin, J.D.: A matrix-free line-search algorithm for nonconvex optimization. Optim. Methods Softw. **34**, 1–24 (2017)
19. Dunn, J.C.: Newton's method and the Goldstein step-length rule for constrained minimization problems. SIAM J. Control Optim. **18**(6), 659–674 (1980)
20. Dai, Y.-H., Yuan, Y.: A nonlinear conjugate gradient method with a strong global convergence property. SIAM J. Optim. **10**(1), 177–182 (1999)
21. Ravindran, A., Reklaitis, G.V., Ragsdell, K.M.: Engineering Optimization: Methods and Applications. Wiley, Hoboken (2006)
22. Nino-Ruiz, E.D., Yang, X.-S.: Improved Tabu Search and Simulated Annealing methods for nonlinear data assimilation. Appl. Soft Comput. **83**, 105624 (2019)
23. Nino-Ruiz, E.D., Beltran-Arrieta, R., Mancilla Herrera, A.M.: Efficient matrix-free ensemble Kalman filter implementations: accounting for localization. In: Kalman Filters - Theory for Advanced Applications. InTech, February 2018
24. Nino-Ruiz, E.D., Cheng, H., Beltran, R.: A robust non-Gaussian data assimilation method for highly non-linear models. Atmosphere **9**(4), 126 (2018)
25. Gottwald, G.A., Melbourne, I.: Testing for chaos in deterministic systems with noise. Phys. D Nonlinear Phenom. **212**(1–2), 100–110 (2005)
26. Karimi, A., Paul, M.R.: Extensive chaos in the Lorenz-96 model. Chaos Interdiscip. J. Nonlinear Sci. **20**(4), 043105 (2010)
27. Wilks, D.S.: Comparison of ensemble-MOS methods in the Lorenz'96 setting. Meteorol. Appl. **13**(3), 243–256 (2006)
28. Fertig, E.J., Harlim, J., Hunt, B.R.: A comparative study of 4D-VAR and a 4D ensemble Kalman filter: perfect model simulations with Lorenz-96. Tellus A **59**(1), 96–100 (2007)
29. van Leeuwen, P.J., Cheng, Y., Reich, S.: Nonlinear Data Assimilation. FADSRT, vol. 2. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-18347-3