RESEARCH ARTICLE

Including parameterized error covariance in local ensemble solvers: Experiments in a 1D model with balance constraints

Sergey Frolov | Jeffrey S. Whitaker | Clara Draper

NOAA Physical Sciences Laboratory,
Boulder, Colorado, USA

Correspondence
S. Frolov, NOAA Physical Sciences
Laboratory, 325 Broadway, Boulder, CO
80305-3337, USA.
Email: sergey.frolov@noaa.gov

Abstract

Lack of efficient ways to include parameterized error covariance in ensemble-based local volume solvers (e.g. the local ensemble-transform Kalman filter – the LETKF) remains an outstanding problem in data assimilation. Here, we describe two new algorithms: GETKF-OI and LETKF-OI. These algorithms are similar to the traditional optimal interpolation (OI) algorithm in that they use parameterized error covariance to update each of the local volume solutions. However, unlike the traditional OI that scales poorly as the number of observations increases, the new algorithms achieve linear scalability by using either the observational-space localization strategy of the traditional LETKF algorithm or the modulated ensembles of the gain-form (GETKF) algorithm. In our testing with a simple one-dimensional univariate system, we find that the GETKF-OI algorithm can recover the exact solution within the truncation bounds of the modulated ensemble and the LETKF-OI algorithm achieves a close approximation to the exact solution. We also demonstrate how to extend GETKF-OI algorithm to a toy multivariate system with balance constraints.

KEYWORDS

tools and methods, data assimilation, optimal interpolation, LETKF

1 | INTRODUCTION

Local volume solvers (such as the local ensemble-transform Kalman filter – the LETKF) provide excellent scalability and throughput on modern high-performance computers. However, to achieve the best possible analysis with limited ensemble sizes, a hybrid solution is needed, in which a solution based on the localized ensemble covariance (i.e. the traditional LETKF) is augmented with a solution based on static (parameterized) covariance (Buehner *et al.*, 2010). This is usually achieved by hybridization of the error covariance (Hamill and Snyder, 2000):

$$\begin{cases} (a) : \mathbf{x}^a_{hyb-P} = \mathbf{x}^f + \mathbf{P}^f_{hybrid}\mathbf{H}^T\left[\mathbf{H}\mathbf{P}^f_{hybrid}\mathbf{H}^T + \mathbf{R}\right]^{-1}\left[\mathbf{y} - H\left(\mathbf{x}^f\right)\right] \\ (b) : \mathbf{P}^f_{hybrid} = \alpha_{static}\mathbf{P}^f_{static} + \alpha_{ens}\left(\mathbf{C}_{loc}\circ\mathbf{P}^f_{ens}\right) \end{cases}$$

(1)

where $\mathbf{x}^a_{hyb-P}$ is the analysis based on the hybrid covariance $\mathbf{P}^f_{hybrid}$; $\mathbf{P}^f_{static}$, $\mathbf{C}_{loc}$, and $\mathbf{P}^f_{ens}$ are the hybrid covariance, the parameterized static covariance, the localization

matrix for the flow-dependent ensemble, and the sample ensemble covariance respectively; α_{static} and α_{ens} are the non-negative weights applied to the static and the ensemble covariances.

Alternatively, the hybrid solution can be obtained by taking a weighted average of the ensemble-based and static-covariance-based analysis as in the hybrid Kalman gain algorithm of Penny (2014) and later modified by Bonavita *et al.* (2015):

$$
\begin{cases}
(a): \mathbf{x}_{static}^{inc} = \mathbf{P}_{static}^{f}\mathbf{H}^{T}\left[\mathbf{H}\mathbf{P}_{static}^{f}\mathbf{H}^{T}+\mathbf{R}\right]^{-1}\left[\mathbf{y}-H\left(\mathbf{x}^{f}\right)\right] \\
(b): \mathbf{x}_{ens}^{inc} = \left(\mathbf{C}_{loc}\circ\mathbf{P}_{ens}^{f}\right)\mathbf{H}^{T}\left[\mathbf{H}\left(\mathbf{C}_{loc}\circ\mathbf{P}_{ens}^{f}\right)\mathbf{H}^{T}+\mathbf{R}\right]^{-1}\left[\mathbf{y}-H\left(\mathbf{x}^{f}\right)\right] \\
(c): \mathbf{x}_{hyb-K}^{a} = \mathbf{x}^{f}+\alpha_{static}\mathbf{x}_{static}^{inc}+\alpha_{ens}\mathbf{x}_{ens}^{inc}
\end{cases}
$$
(2)

where $\mathbf{x}_{hyb-K}^{a}$ is the hybrid gain solution.

In current practice at many operational centers, the hybrid solution $\mathbf{x}_{hyb-P}^{a}$ in Equation (1a) and the static solution $\mathbf{x}_{static}^{inc}$ in Equation (2a) are obtained using variational solvers. Unlike the LETKF solution, the variational solver requires substantial inter-processor communications and, depending on the implementation, can scale poorly with either the increase in the number of processing units or with the increase of the analysis grid size. (See Appendix A for considerations on scalability of the LETKF algorithm and the variational solvers).

Several studies attempted to obviate the need for the variational solver in Equations (1a) and (2a) above. Kretschmer *et al.* (2015) used samples from the square root of the climatological error covariance to generate the climatological (static) ensemble for the LETKF solver. However, large ensemble size and localization are needed to develop an effective, high-rank static covariance based on sampling of the climatological ensemble. Recent studies in a simplified low-resolution global circulation model (Kotsuki and Bishop, 2021) showed that, for dense observing networks, climatological ensembles on the order of 400 members or larger are needed to effectively augment a flow-dependent covariance model. However, for such large climatological ensemble sizes, 3DVAR with parameterized covariance is more computationally compelling compared to an LETKF with a large climatological ensemble.

Alternatively, Bishop *et al.* (2015) developed the consistent hybrid ensemble filter (CHEF) algorithm, in which a sequential data assimilation (DA) method is used to solve the local Kalman filter problem with an explicitly hybridized covariance matrix. However, for a realistic observational network with radiance assimilation, this method requires holding impractically large covariances matrixes in memory.

In this work, we start our development by recalling that the traditional optimal interpolation (OI) algorithm (Gandin, 1963) is a local volume solver that utilizes a parameterized covariance matrix $\mathbf{P}_{static}^{f}$. The OI solution for each grid point i is achieved as follows:

$$
\mathbf{x}_{OI}^{inc}(i) = \mathbf{S}_{(i)}\mathbf{P}_{static}^{f}\mathbf{H}_{(i)}^{T}\left[\mathbf{H}_{(i)}\mathbf{P}_{static}^{f}\mathbf{H}_{(i)}^{T}+\mathbf{R}_{(i)}\right]^{-1}\left[\mathbf{y}_{(i)}-H_{(i)}\left(\mathbf{x}^{f}\right)\right] \quad (3)
$$

where $\mathbf{P}_{static}^{f}$ is a parameterized (static) forecast error covariance model used by the OI, and $\mathbf{S}_{(i)}$ is the selection operator that selects the rows of the state vector corresponding to the (i)th grid point. The OI algorithms solves Equation (3) for each grid point (i) independently based on the observations $\mathbf{y}_{(i)}$ closest to each grid point. To denote local observations, we introduced local versions of observations $\mathbf{y}_{(i)}$, observation operator $H_{(i)}\left(\mathbf{x}^{f}\right)$, and observation-error covariance $\mathbf{R}_{(i)}$.

However, we also recall that the traditional OI was largely abandoned in favor of the 3DVAR algorithms around the end of the 20th century. This was motivated by several factors. First, as more radiance observations became available in the early 2000s, the OI algorithm failed to scale as it requires an inverse of the observation-space covariance matrix $\left[\mathbf{H}_{(i)}\mathbf{P}^{f}\mathbf{H}_{(i)}^{T}+\mathbf{R}_{(i)}\right]^{-1}$ in Equation (3), which scales with the cube of the local observation counts. Second, OI implementations often used the computation of the background covariance $\mathbf{H}_{(i)}\mathbf{P}^{f}\mathbf{H}_{(i)}^{T}$ which scaled with the square of the local observation count, and is hard to implement for radiance assimilation due to the use of integrals of the vertical quantities. Third, at the time OI was widely used, the R-localization (Hunt *et al.*, 2007) had not yet been introduced, and the boundaries of the local domains could introduce discontinuities in the analysis. Finally, the migration to the 3DVAR was seen at that time as an incremental step toward implementation of the superior 4DVAR algorithm.

In this paper, we rely on several algorithmic developments that have occurred since the early 2000s. First, the introduction of observation-space localization through R-inflation (Hunt *et al.*, 2007) allows more flexibility in spatial error covariance modeling. Second, introduction of the local analysis methods (Evensen, 2003; Hunt *et al.*, 2007) allows for linear scalability with increasing observation and grid point counts. Also, the development of the hybrid gain algorithm (Penny, 2014) introduced independent computation of the static update and ensemble update without the need to specify a hybrid error covariance, as in Equation (1b). Finally, the introduction of the ensemble modulation product (Bishop *et al.*, 2011) showed how ensemble localization can be achieved by

generating a larger ensemble of "modulated" ensemble members. We combine these algorithmic advances to develop a family of scalable DA algorithms that use parameterized background error covariances in the context of the local volume solvers. We name these algorithms, the LETKF-OI and the GETKF-OI, depending on whether they use observation-space localization (as in the standard LETKF) or model-space localization (as in the standard GETKF).

While the ultimate intention of this work is to develop accurate, efficient, and scalable algorithms for Earth system DA, this paper is focused on the more narrow problem of showing that the new algorithms can replace the 3DVAR solution for the central analysis in Equation (2a), and the Hybrid-3DEnVar solution in Equation (1a). We demonstrate this potential with two synthetic test problems. The first test problem is a univariate assimilation into a one-dimensional model, with prescribed error covariance. The second test problem is the assimilation of two variables that are related through a geostrophic-like balance in the same one-dimensional domain. We test the GETKF-OI in both test problems attempting to replace the 3DVAR solution in Equation (2a) and the Hybrid-3DVAR in Equation (1a). We only tested LETKF-OI for the univariate case and the 3DVAR context Equation (2a) because it was not clear how to include balance constraints in the context of R-localization. We use two sets of experimental protocols. First, we compare LETKF-OI and GETKF-OI solutions against reference solutions obtained using a classic 3DVAR and OI in a simple case of assimilating two observations. Second, to ensure that our conclusions are not biased by assimilating a limited set of two observations, we conduct large randomized studies, where we vary the number and the locations of assimilated observations and the ratio of model versus observation system accuracy.

2 | METHODS

In this section, we will first derive the square-root form of the OI algorithm (GETKF-OI) that represents the parameterized covariance matrix $\mathbf{P}^f_{static}$ using the eigenvectors of the parameterized covariance specified on a local domain. We show similarities of this new GETKF-OI algorithm and the GETKF algorithm of Bishop *et al.* (2017). For the limited case of the univariate problems, we also derive an alternative algorithm (LETKF-OI) based on R-localization. We conclude by illustrating how the new GETKF-OI and LETKF-OI algorithms can be used in hybrid solutions in Equations (1) and (2).

2.1 | Derivation of GETKF-OI

We start derivation of the GETKF-OI algorithm by specifying a parameterized form of the static error covariance following decomposition commonly used in variational solvers (Weaver *et al.*, 2005):

$$\mathbf{P}^f_{static} = \mathbf{KDCD}^T\mathbf{K}^T \qquad (4)$$

where $\mathbf{K}$ is the multivariate balance operator, $\mathbf{D}$ is the diagonal matrix of forecast error standard deviations, and $\mathbf{C}$ is the block diagonal matrix of univariate correlations. Notice that $\mathbf{K}$ is the only multivariate operator in Equation (4).

We then define an approximate square-root decomposition for $\mathbf{P}^f_{static}$ in Equation (4) based on a truncated series of eigenvectors:

$$\begin{cases} \mathbf{P}^f_{static} = \mathbf{E}^{static}\mathbf{\Lambda}^{static}\left(\mathbf{E}^{static}\right)^T \approx \sqrt{\mathbf{P}^{static}_{n_{eigSt}}}\sqrt{\mathbf{P}^{static}_{n_{eigSt}}}^T \\ \sqrt{\mathbf{P}^{static}_{n_{eigSt}}} = \mathbf{E}^{static}_{n_{eigSt}}\sqrt{\mathbf{\Lambda}^{static}_{n_{eigSt}}} = \left[\sqrt{\lambda_1}\mathbf{e}_1, \ldots, \sqrt{\lambda_{n_{eigSt}}}\mathbf{e}_{n_{eigSt}}\right] \end{cases} \qquad (5)$$

where $\mathbf{E}^{static}$ and $\mathbf{\Lambda}^{static}$ are the eigenvectors and the eigenvalues of the $\mathbf{P}^f_{static}$ covariance matrix; $\sqrt{\mathbf{P}^{static}_{n_{eigSt}}}$ is the truncated square root of $\mathbf{P}^f_{static}$ that retains the leading n_{eigSt} eigen modes; and λ_j and $\mathbf{e}_j$ are the *j*th eigenvalue and eigenvector.

In general, the truncated square-root decomposition $\sqrt{\mathbf{P}^{static}_{n_{eigSt}}}$ will be impractical for $\mathbf{P}^f_{static}$ defined on a large global domain (e.g. it will require a large set of global eigenvectors or spherical harmonics to achieve a close approximation). However, in the context of a local volume solver, only a portion of the global covariance that corresponds to the covariance within the scope of the local volume needs factoring (see Appendix B for how an efficient algorithm can be constructed for large domains). To simplify coding for the main part of this paper, we computed $\sqrt{\mathbf{P}^{static}_{n_{eigSt}}}$ based on the eigen decomposition of the global covariance defined on our toy-model domain.

Assuming that the truncated square root of the covariance $\sqrt{\mathbf{P}^{static}_{n_{eigSt}}}$ is available, one can rewrite Equation (3) as follows:

$$\mathbf{x}^{inc}_{GETKF-OI}(i) \approx \mathbf{S}_{(i)}\sqrt{\mathbf{P}^{static}_{n_{eigSt}}}\sqrt{\mathbf{P}^{static}_{n_{eigSt}}}^T$$
$$\mathbf{H}^T_{(i)}\left[\mathbf{H}_{(i)}\sqrt{\mathbf{P}^{static}_{n_{eigSt}}}\sqrt{\mathbf{P}^{static}_{n_{eigSt}}}^T\mathbf{H}^T_{(i)} + \mathbf{R}_{(i)}\right]^{-1}\ldots$$
$$\ldots\left[\mathbf{y}_{(i)} - H_{(i)}\left(\mathbf{x}^f\right)\right] \qquad (6)$$

In most cases, we expect that the number of truncated local eigen modes n_{eigSt} will be smaller than the number of local observations and, hence, it will be more efficient to perform the inverse in the column space of $\sqrt{\mathbf{P}^{static}_{n_{EigSt}}}$ instead of the local observation space as defined in Equation (6). This can be performed in multiple ways. Here, we will use the ensemble-transform formula (Hunt *et al.*, 2007):

$$\mathbf{x}^{inc}_{GETKF-OI}(i) = \mathbf{S}_{(i)} \sqrt{\mathbf{P}^{static}_{n_{eigSt}}}$$

$$\left[\left(\mathbf{H}_{(i)} \sqrt{\mathbf{P}^{static}_{n_{eigSt}}} \right)^T \mathbf{R}^{-1}_{(i)} \left(\mathbf{H}_{(i)} \sqrt{\mathbf{P}^{static}_{n_{eigSt}}} \right) + \mathbf{I} \right]^{-1} \ldots$$

$$\ldots \left[\left(\mathbf{H}_{(i)} \sqrt{\mathbf{P}^{static}_{n_{eigSt}}} \right)^T \mathbf{R}^{-1}_{(i)} \left(\mathbf{y}_{(i)} - H_{(i)} \left(\mathbf{x}^f \right) \right) \right] \quad (7)$$

Notice that the normalization factor $\left(n_{eigSt} - 1 \right)$ is missing in front of the identity matrix $\mathbf{I}$ (like one normally would see in the ETKF formula). This is because $\sqrt{\mathbf{P}^{static}_{n_{eigSt}}}$ has already been normalized by this factor during computation of the $\mathbf{P}^{static}_{n_{eigSt}}$ covariance.

We refer to the solution in Equation (7) as GETKF-OI because of several similarities between (7) and the gain form of the ensemble transform Kalman filter (GETKF) introduced in (Bishop *et al.*, 2017). First, similar to any ETKF filter, we use an ensemble of perturbations, here – the columns of the $\sqrt{\mathbf{P}^{static}_{n_{EigSt}}}$ defined in Equation (5). Second, we can show that $\sqrt{\mathbf{P}^{static}_{n_{EigSt}}}$ can be seen as an operator that modulates a single 1-vector in the GETKF algorithm.

To show the connection between the static covariance modeling and localization in the model space, we can rewrite $\mathbf{P}^{static}_{n_{EigSt}}$ as an operator acting upon the identity matrix $\underline{\underline{\mathbf{1}}}$ under the Schur product:

$$\begin{cases} \mathbf{P}^{static}_{n_{eigSt}} = \mathbf{P}^{static}_{n_{eigSt}} \circ \underline{\underline{\mathbf{1}}} = \left(\sqrt{\mathbf{P}^{static}_{n_{eigSt}}} \sqrt{\mathbf{P}^{static}_{n_{eigSt}}}^T \right) \circ \left(\underline{\mathbf{i}}\,\underline{\mathbf{i}}^T \right) \\ = \left(\sqrt{\mathbf{P}^{static}_{n_{eigSt}}} \Delta \underline{\mathbf{i}} \right) \left(\sqrt{\mathbf{P}^{static}_{n_{eigSt}}} \Delta \underline{\mathbf{i}} \right)^T \\ \sqrt{\mathbf{P}^{static}_{n_{eigSt}}} v \underline{\mathbf{i}} \equiv \left[\sqrt{\mathbf{p}^{static}_{i=1}} \circ \underline{\mathbf{i}}, \ldots, \sqrt{\mathbf{p}^{static}_{i=n_{eigSt}}} \circ \underline{\mathbf{i}} \right] \\ \underline{\underline{\mathbf{1}}} \equiv \underline{\mathbf{i}}\,\underline{\mathbf{i}}^T \end{cases} \quad (8)$$

Recall that the identity $\underline{\underline{\mathbf{1}}}$ under the Schur product is the matrix of ones which is by definition is the outer product of two 1-vectors $\underline{\mathbf{i}} = [1, \ldots, 1]^T$. Because the number "1" does not have upper- and lowercase symbols that we can use to denote the matrix and the vector, we are using the double underscore and the single underscore

notation to make it explicit that $\underline{\underline{\mathbf{1}}}$ is a matrix and $\underline{\mathbf{i}}$ is a vector. Then using the definition of the modulation product (Bishop *et al.*, 2011), $\sqrt{\mathbf{P}^{static}_{n_{eigSt}}} \Delta \underline{\mathbf{i}}$ is the modulation of the 1-vector $\underline{\mathbf{i}}$ by the square root of the static covariance $\sqrt{\mathbf{P}^{static}_{n_{eigSt}}}$.

2.2 | LETKF-OI approximation for univariate, homogeneous covariances

While the GETKF-OI algorithm is appropriate for any covariance that can be modeled using a limited set of eigenvectors on a local domain, the LETKF-OI algorithm offers a straightforward implementation for a set of univariate covariance models of the form:

$$\begin{cases} \mathbf{P}^f = \mathbf{P}^f_{static} = \mathbf{D}\mathbf{C}\mathbf{D}^T \\ \mathbf{C}(i,j) = corr(dist(i,j)) \end{cases} \quad (9)$$

where $\mathbf{C}$ is a univariate correlation function that only depends on the distance between two grid points i and j. In this case, we can approximate the static forecast error covariance model through an action of the observation-space error inflation used in the traditional LETKF filter (Hunt *et al.*, 2007).

Recall that R-localization through observation-error inflation provides an alternative to the model-space localization. The relationship between the two is approximate (Sakov and Bertino, 2011) with, to our knowledge, no known derivation that shows the specific approximations that are taken when model-space localization is transformed into observation-space localization. Since no such formula exists, we will derive the LETKF-OI algorithms using the same logical steps that were taken to derive localization of the ensemble covariance matrix in the traditional LETKF algorithm. To facilitate this comparison, we list the steps in Table 1, with LETKF derivation in the left column and LETKF-OI derivation in the right column. For completeness, we list the final update formula:

$$\mathbf{x}^{inc}_{LETKF-OI}(i) = \mathbf{S}_{(i)}(\mathbf{D}\underline{\mathbf{i}})(\mathbf{D}\underline{\mathbf{i}})^T \mathbf{H}^T_{(i)} \ldots$$

$$\ldots \left[\mathbf{H}_{(i)}(\mathbf{D}\underline{\mathbf{i}})(\mathbf{D}\underline{\mathbf{i}})^T \mathbf{H}^T_{(i)} + \mathbf{L}_{static}\mathbf{R}_{(i)}\mathbf{L}^T_{static} \right]^{-1} \ldots$$

$$\ldots \left[\mathbf{y}_{(i)} - H_{(i)} \left(\mathbf{x}^f \right) \right] \quad (10)$$

where $\mathbf{L}_{static}$ is the diagonal matrix that inflates observational errors based on the distance from the observation to the grid point (i); $\underline{\mathbf{i}}$ is the 1-vector; and the product $\mathbf{D}\mathbf{i}$ is just a vector of forecast error standard deviations.

TABLE 1   Derivation of LETKF and the LETKF- OI)\ algorithms

LETKF derivation	LETKF-OI derivation
Kalman gain for localized ensemble in the model space	
$\mathbf{K}_{KF}(i) = \mathbf{S}_{(i)} \left(\mathbf{C}_{loc} \circ \mathbf{P}_{ens}\right) \mathbf{H}_{(i)}^{T} \left[\mathbf{H}_{(i)} \left(\mathbf{C}_{loc} \circ \mathbf{P}_{ens}\right) \mathbf{H}_{(i)}^{T} + \mathbf{R}_{(i)}\right]^{-1}$	$\mathbf{K}_{OI}^{inc}(i) = \mathbf{S}_{i} \mathbf{D} \left(\mathbf{C}_{static} \circ \underline{\mathbf{1}}\right) \mathbf{D}^{T} \mathbf{H}_{(i)}^{T} \left[\mathbf{H}_{(i)} \mathbf{D} \left(\mathbf{C}_{static} \circ \underline{\mathbf{1}}\right) \mathbf{D}^{T} \mathbf{H}_{(i)}^{T} + \mathbf{R}_{(i)}\right]^{-1}$
Approximating B-localization with R-inflation	
$\mathbf{K}_{KF}(i) \approx \mathbf{K}_{LETKF}(i) =$	$\mathbf{K}_{OI}(i) \approx \mathbf{K}_{LETKF-OI}(i) =$
$= \mathbf{S}_{(i)} \mathbf{X}_{ens} \mathbf{X}_{ens}^{T} \mathbf{H}_{(i)}^{T} \left[\mathbf{H}_{(i)} \mathbf{X}_{ens} \mathbf{X}_{ens}^{T} \mathbf{H}_{(i)}^{T} + \mathbf{L}_{loc} \mathbf{R}_{(i)} \mathbf{L}_{loc}^{T}\right]^{-1}$	$= \mathbf{S}_{(i)} \mathbf{D} \underline{\mathbf{ii}}^{T} \mathbf{D}^{T} \mathbf{H}_{(i)}^{T} \left[\mathbf{H}_{(i)} \mathbf{D} \underline{\mathbf{ii}}^{T} \mathbf{D}^{T} \mathbf{H}_{(i)}^{T} + \mathbf{L}_{static} \mathbf{R}_{(i)} \mathbf{L}_{static}^{T}\right]^{-1}$

Note: Most symbols are defined in the main text. The localization distances in $\mathbf{L}_{loc}$ and $\mathbf{L}_{static}$ are tuned to produce the best results for localization of the ensemble $\mathbf{X}_{ens}$ and for reconstruction of the impact of the static localization.
Abbreviations: LETKF, local ensemble-transform Kalman filter; OI, optimal interpolation.

Equation (10) can be further seen as an ensemble OI update with a single ensemble member $\mathbf{d} = \mathbf{D}\underline{\mathbf{i}}$. With a single ensemble member, it will be always more efficient to use the ETKF formulation that only requires a scalar inverse:

$$\mathbf{x}_{LETKF-OI}^{inc}(i) = \mathbf{S}_{(i)} \mathbf{D}\underline{\mathbf{i}}$$

$$\left[\left(\mathbf{H}_{(i)} \mathbf{D}\underline{\mathbf{i}}\right)^{T} \left(\mathbf{L}_{static} \mathbf{R}_{(i)} \mathbf{L}_{static}^{T}\right)^{-1} \left(\mathbf{H}_{(i)} \mathbf{D}\underline{\mathbf{i}}\right) + \mathbf{I}\right]^{-1} \dots$$

$$\dots \left[\left(\mathbf{H}_{(i)} \mathbf{D}\underline{\mathbf{i}}\right)^{T} \left(\mathbf{L}_{static} \mathbf{R}_{(i)} \mathbf{L}_{static}^{T}\right)^{-1} \left(\mathbf{y}_{(i)} - H_{(i)}\left(\mathbf{x}^{f}\right)\right)\right] \quad (11)$$

With no formal derivation for R-localization used in Hunt *et al.* (2007), and consequently no formal derivation for Equation (10), we notice that it is hard to explain why only a single ensemble member (equal to the standard deviation of the background error) is used in Equation (10). Here, we offer an intuitive interpretation of why LETKF-OI in Equation (10) uses a single ensemble member. Recall that a common way to test the impact of the localization in any ensemble system, is to supply a single 1-vector as an input to the ensemble algorithm, use a single unit innovation, and set observational error equal to one. Then, the analysis increment will depict the localization function with the maximum at the location of the single observation. Equation (10) is the generalization of this idea that also ensures that, in the case of a single observation collocated with a grid point, the maximum increment will be equal to:

$$\max\left(\mathbf{x}_{LETKF-OI}^{inc}(i)\right) = \frac{d_{(i)}^{2}}{d_{(i)}^{2} + \sigma_{ob}^{2}} \quad (12)$$

where $d_{(i)}^{2}$ is the variance of the background error at the grid point (i) collocated with the observation and σ_{ob}^{2} is the variance of the observational error.

2.3 | Hybrid update

2.3.1 | Hybrid gain update

Hybrid gain update is straightforward using either GETKF-OI or LETKF-OI. In this update, two (G)LETKF solvers will run in parallel, one computing analysis for the flow-dependent ensemble using the (G)LETKF algorithm and another for the mean analysis using the (G)LETKF-OI algorithm in Equations (7) or (11). The two analyses are then combined using Equation (2) to obtain a new hybrid analysis mean for the posterior ensemble.

2.3.2 | Hybrid covariance update

Hybrid covariance update in Equation (1a) is possible by forming an augmented modulated ensemble:

$$\mathbf{X}_{hybrid} =$$
$$\left[\left(\frac{\sqrt{\alpha_{ens}}}{n_{ens}-1} \sqrt{\mathbf{C}_{n_{eigLoc}}^{loc}} \Delta\mathbf{X}_{ens}\right), \left(\sqrt{\alpha_{static}} \left[\sqrt{\mathbf{P}_{n_{eigSt}}^{static}} \Delta\underline{\mathbf{i}}\right]\right)\right] \quad (13)$$

where n_{eigLoc} and n_{eigSt} are the number of modes retained in $\sqrt{\mathbf{C}_{n_{eigLoc}}^{loc}}$ and $\sqrt{\mathbf{P}_{n_{eigSt}}^{static}}$ respectively; $\mathbf{X}_{ens}$ are the original ensemble perturbations for the flow-dependent ensemble; and $\sqrt{\alpha_{ens}}$ and $\sqrt{\alpha_{static}}$ are the square roots of the scalar weights in Equation (1b). Following Bishop *et al.* (2017), one might choose to only update the original flow-dependent members $\mathbf{X}_{ens}$ of the complete hybrid ensemble $\mathbf{X}_{hybrid}$ (see equation 10 in Lei *et al.*, 2018).

Notice that different modulation matrixes $\sqrt{\mathbf{C}_{n_{eigLoc}}^{loc}}$ and $\sqrt{\mathbf{P}_{n_{eigSt}}^{static}}$ are needed to specify $\mathbf{X}_{hybrid}$ in Equation (13). Hence, it is not possible to use R-localization with a single

TABLE 2 Data assimilation algorithm

Ref. number	Name	Solution	Loc. (R/M)	Update formula	Figure reference
T.1	3DVAR	Global	N/A	$\mathbf{x}^{inc}_{3DVAR} = \mathbf{P}^f_{static}\,\mathbf{H}^T \left[\mathbf{H}\mathbf{P}^f_{static}\,\mathbf{H}^T + \mathbf{R} \right]^{-1} \mathbf{y}^{inn}$	Figures 4 and 7, Black line.
T.2	Hyb3DEnVAR	Global	M	$\mathbf{x}^{inc}_{hyb-P} = \mathbf{P}^f_{hyb}\,\mathbf{H}^T \left[\mathbf{H}\mathbf{P}^f_{hyb}\,\mathbf{H}^T + \mathbf{R} \right]^{-1} \mathbf{y}^{inn}$	Figures 5 and 7, Blue circle.
T.3	EnVar	Global	M	$\mathbf{x}^{inc}_{envar} = \left(\mathbf{C}_{loc}\circ\mathbf{P}_{ens} \right) \mathbf{H}^T \left[\mathbf{H}\left(\mathbf{C}_{loc}\circ\mathbf{P}_{ens} \right) \mathbf{H}^T + \mathbf{R} \right]^{-1} \mathbf{y}^{inn}$	Used in T.4
T.4	Hybrid gain	Global	M	$\mathbf{x}^{inc}_{hyb-K} = (1-\alpha)\mathbf{x}^{inc}_{3DVAR} + \alpha\mathbf{x}^{inc}_{envar}$	Figures 5 and 7, Red cross.
T.5	OI	Local	N/A	$\mathbf{x}^{inc}_{OI}(i) = \mathbf{S}_{(i)}\mathbf{P}^f_{static}\,\mathbf{H}^T_{(i)} \left[\mathbf{H}_{(i)}\mathbf{P}^f_{static}\,\mathbf{H}^T_{(i)} + \mathbf{R}_{(i)} \right]^{-1} \mathbf{y}^{inn}_{(i)}$	Figures 4 and 7, Blue cross.
T.6	LETKF	Local	R	$\mathbf{x}^{inc}_{LETKF}(i) = \mathbf{S}_{(i)}\mathbf{P}^f_{ens}\,\mathbf{H}^T_{(i)} \left[\mathbf{H}_{(i)}\mathbf{P}^f_{ens}\,\mathbf{H}^T_{(i)} + \mathbf{L}\mathbf{R}_{(i)}\mathbf{L}^T \right]^{-1} \mathbf{y}^{inn}_{(i)}$	Used in T.9
T.7	GETKF-OI	Local	M	$\mathbf{x}^{inc}_{GETKF-OI}(i) = \mathbf{S}_{(i)}\sqrt{\mathbf{P}^{static}_{n_{eigSt}}} \left[\left(\mathbf{H}_{(i)}\sqrt{\mathbf{P}^{static}_{n_{eigSt}}} \right)^T \mathbf{R}^{-1}_{(i)} \left(\mathbf{H}_{(i)}\sqrt{\mathbf{P}^{static}_{n_{eigSt}}} \right) + \mathbf{I} \right]^{-1} \cdots$ $\cdots \left[\left(\mathbf{H}_{(i)}\sqrt{\mathbf{P}^{static}_{n_{eigSt}}} \right)^T \mathbf{R}^{-1}_{(i)}\mathbf{y}^{inn}_{(i)} \right]$	Figures 4 and 7, Green circle.
T.8	LETKF-OI	Local	R	$\mathbf{x}^{inc}_{LETKF-OI}(i) = \mathbf{S}_{(i)}\mathbf{D}\mathbf{ii}^T\mathbf{D}^T\mathbf{H}^T_{(i)} \left[\mathbf{H}_{(i)}\mathbf{D}\mathbf{ii}^T\mathbf{D}^T\mathbf{H}^T_{(i)} + \mathbf{L}_{static}\mathbf{R}_{(i)}\mathbf{L}^T_{static} \right]^{-1} \mathbf{y}^{inn}_{(i)}$	Figure 4, Red line.
T.9	Local-Hyb-gain	Local	M	$\mathbf{x}^{inc}_{loc-hyb-K} = \alpha_{static}\mathbf{x}^{inc}_{GETKF-OI} + \alpha_{ens}\mathbf{x}^{inc}_{LETKF}$	Figure 5, Red line.
T.10	Local-Hyb-P	Local	M	$\mathbf{x}^{inc}_{loc-hyb-P}(i) = \mathbf{S}_{(i)}\mathbf{X}_{hybrid} \left[\left(\mathbf{H}_{(i)}\mathbf{X}_{hybrid} \right)^T \mathbf{R}^{-1}_{(i)} \left(\mathbf{H}_{(i)}\mathbf{X}_{hybrid} \right) + \mathbf{I} \right]^{-1} \cdots$ $\cdots \left[\left(\mathbf{H}_{(i)}\mathbf{X}_{hybrid} \right)^T \mathbf{R}^{-1}_{(i)}\mathbf{y}^{inn}_{(i)} \right]$	Figures 5 and 7, Blue line.

Abbreviations: GETKF, gain-form ensemble-transform Kalman filter; LETKF, local ensemble-transform Kalman filter; OI, optimal interpolation.

observation localization scale to replace modulation products in Equation (13). As our paper was developed, a third form of localization (*Z*-localization) was introduced in the literature (Kotsuki and Bishop, 2021) that allows for specification of different localization scales for the dynamic and the climatological part of the hybrid ensemble by tapering of the ensemble perturbations similar to Sakov and Bertino (2011). We did not compare our results to *Z*-localization.

3 | TESTING METHODOLOGY

3.1 | Univariate model

We tested the (G)LETKF-OI algorithm using a simple statistical model. In this model, we specify the true forecast error correlation matrix $\mathbf{C}_{static}$ using the Gaspari and Cohn (GC99) correlation function (Gaspari and Cohn, 1999) on a circular domain of length 100 and a horizontal decorrelation scale of 22 grid points (correlation decays from one to zero in 22 points). The true error variance $\text{diag}(\mathbf{DD}^{T})$ varies with a maximum of 1 at the boundaries and a minimum of 0.5 at the grid point 50 (see Figure 1b). The combined true error covariance $\mathbf{P}_{static} = \mathbf{DC}_{static}\mathbf{D}^{T}$ is shown in Figure 1a.

A random draw of the state η of the univariate state was obtained as:

$$\begin{cases} \boldsymbol{\eta} = \mathbf{D}\sqrt{\mathbf{C}_{static}}\mathbf{r}_{\eta} \\ \mathbf{r}_{\eta} : N(\mathbf{0}, \mathbf{I}) \end{cases} \quad (14)$$

3.2 | Multivariate model with a balance constraint

To test the GETKF-OI algorithm in the multivariate problem, we extend the 1D problem in Section 3.1 using a geostrophic-like balance, where the second (velocity-like) variable u is related to the first (height-like) variable η in Equation (14) as follows:

$$\begin{cases} \mathbf{u} = k\frac{\partial\eta}{\partial x} + \beta\mathbf{D}\sqrt{\mathbf{C}_{static}}\mathbf{r}_{u} \\ \mathbf{r}_{u} : N(\mathbf{0}, \mathbf{I}) \end{cases} \quad (15)$$

where k is a unit conversion factor (we use a factor of -10), β is the variance-scale factor that specifies the ratio of the unbalanced $\mathbf{u}$ perturbations compared to the scale of the balanced perturbations (we use $\beta = 1/3$), and $\mathbf{r}_{u}$ is a random state drawn from a normal distribution. The first term on the right in Equation (15) is the balanced part of $\mathbf{u}$ (which encodes a linear relationship between $\mathbf{u}$ and η)

and the second term is the unbalanced term. In geophysical fluid dynamics, the first term might take the form of a geostrophic flow while the second term will correspond to ageostrophic flow.

In matrix notation, a random draw of the multivariate state can be written as:

$$\begin{aligned} \begin{bmatrix} \eta \\ \mathbf{u} \end{bmatrix} &= \mathbf{KD}_{\eta,u}\sqrt{\mathbf{C}_{\eta,u}}\mathbf{r}_{\eta,u} \\ &= \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ k\boldsymbol{\Delta} & \mathbf{I} \end{bmatrix}\begin{bmatrix} \mathbf{D} & \mathbf{0} \\ \mathbf{0} & \sqrt{\beta}\mathbf{D} \end{bmatrix}\begin{bmatrix} \sqrt{\mathbf{C}} & \mathbf{0} \\ \mathbf{0} & \sqrt{\mathbf{C}} \end{bmatrix}\mathbf{r}_{\eta,u} \end{aligned} \quad (16)$$

where $\boldsymbol{\Delta}$ is the differentiation operator implementing $\frac{\partial\eta}{\partial x}$ (we used centered difference). The static error covariance for the multivariate model is shown in Figure 2.

3.3 | Localization parameters and sample ensemble

We specified model-space localization matrix $\mathbf{C}_{loc}$ in Equation 2 using the Gaspari-Cohn localization function with decorrelation scale of 40 grid points. Observation-space localization was tuned for each experiment, with optimal values listed in Section 3.4.2.

For tests in the univariate model, we retained $n_{eigLoc} = 7$ and $n_{eigSt} = 13$ in Equation (13), which retained 99% of the original variance. For the multivariate case, we retained $n_{eigSt} = 25$.

To illustrate the fidelity of the sample covariance and the localization, Figure 3 compares the true covariance (thick black) with sample ensemble covariance $\mathbf{P}_{ens}$ (dotted red), localization function $\mathbf{C}_{loc}$ (dash blue), and the localized ensemble covariance $\left(\mathbf{C}_{loc}\circ\mathbf{P}_{ens}^{f}\right)$ (thin blue) in the univariate case. As expected, Figure 3 shows that the sample ensemble covariance (dotted red) provides a noisy estimate for both the correlations and the variance (true variance of 0.5 and sample estimate of 0.35). While the localization is effective at attenuating the erroneous long-distance correlations, localization is unable to fix a poor estimate of the sample error variance.

The error in the variance estimate is proportional to $1/\sqrt{n_{ens}}$. In the case of sampling from the static error covariance, for example as in (Kretschmer *et al.*, 2015; Kotsuki and Bishop, 2021), one can compensate for the sampling error by rescaling the climatological ensemble to match the prescribed static error variance. However, in the case of the flow-dependent ensemble shown in Figure 3, such a rescaling is not possible as any given sample can either overestimate or underestimate unknown flow-dependent error variance.
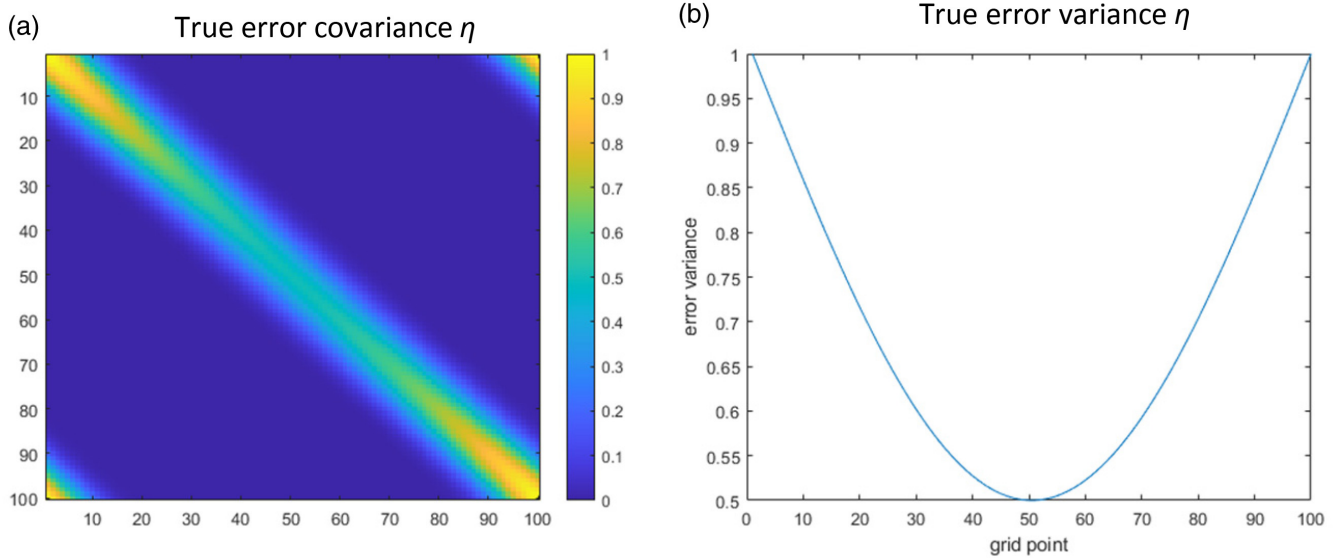
(a)

True error covariance η

(b)

True error variance η

FIGURE 1 Static error covariance (a) and error variance (b) in the univariate model [Colour figure can be viewed at wileyonlinelibrary.com]
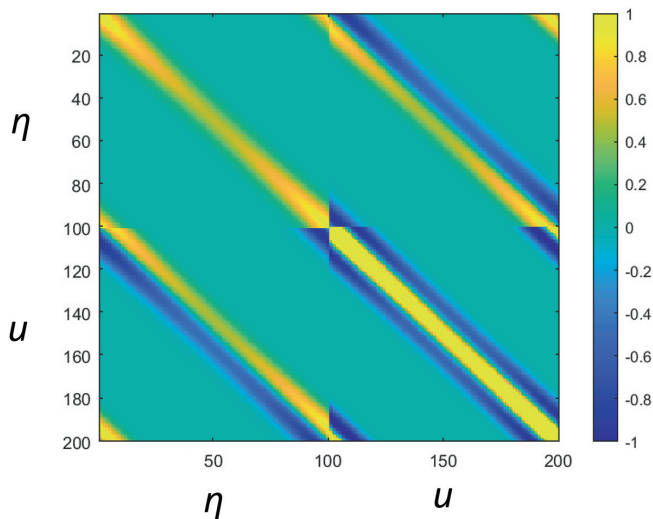

FIGURE 2 True error covariance for the multivariate model [Colour figure can be viewed at wileyonlinelibrary.com]

FIGURE 3 Comparison of cross-covariance between grid point 50 and the rest of the state in the univariate case for: (thick black) true error covariance; (dotted red) sample error covariance generated using 50 ensemble members; (dash blue) localization function; and (solid blue) localized sample covariance [Colour figure can be viewed at wileyonlinelibrary.com]

3.4 | Testing protocol

We conducted two series of tests in this paper. The first series tested the ability of the new algorithms to accurately reproduce assimilation increments from well understood algorithms like 3DVAR, OI, or global hybrid solutions in Equations (1) and (2). To gain intuitive understanding of the results, these tests were conducted by assimilating just two measurements located within a correlation distance from each other (see details in Section 3.4.1).

The second family of tests randomized the experiment configuration to eliminate the possibility that our conclusions are biased by the simplicity of the double

observation experiment. See details of randomized tests in Section 3.4.2.

3.4.1 | Two observation test case

To test the equivalence between the two methods, we compared the increments by assimilating a pair of observations. In the univariate case, the η variable was

observed at grid points 35 and 55. In the multivariate case, we only assimilated an η observation at point 35 and a u observation at point 55. The magnitude of the innovation (y-$H[x^f]$) was one. The observational error was equal to the prior error variance. We chose to use two observations located within the correlation influence of each other to verify that our implementation correctly represents interaction of two observations.

To quantify the difference between the control increment x_{cntrl}^{inc} and the increment from a new experiment $x_{\exp}^{inc}$, we used the normalized root mean square error (NRMSE):

$$NRMSE = 100 * \frac{\left\| x_{cntrl}^{inc} - x_{\exp}^{inc} \right\|_2}{\left\| x_{cntrl}^{inc} \right\|_2} \% \qquad (17)$$

3.4.2 | Randomized tests

To test how the approximation errors in the (G)LETKF-OI algorithms change as a function of observation density and the ratio of the observation to forecast error, we conducted experiments with 1000 trials each. In each trial a random number of observations were selected and placed in a random number of locations in the domain. Each trial generated innovations by observing a randomly drawn state from the true covariance model. Randomly drawn observation errors, consistent with the R covariance, were added to the innovation vector. We tested three levels of ratios between observation and forecast error variances (R2P ratio) equal to 1/5, 1, and 5. For each value of the R2P ratio, we tuned the R-inflation parameter in the LETKF-OI algorithm, with the optimal values of Gaspari-Cohn localization distance of 9.2, 13.2, and 18.0 respectfully. The approximation errors were computed using Equation (17) where we used the true "twin" increment as x_{cntrl}^{inc}.

4 | RESULTS

4.1 | Univariate case

4.1.1 | Comparison of algorithms with the static covariance update

Figure 4 shows that the global 3DVAR solution and the local OI solutions can recover almost identical analysis increments (compare thick black line and blue crosses, NRMSE of 0.01%). When we used GETKF-OI algorithm in Equation (7), we produce the analysis increment that was indistinguishable from either 3DVAR or the OI algorithms (NRMSE = 0.7%). This is expected as the only approximation of the GETKF-OI algorithms compared to the classical
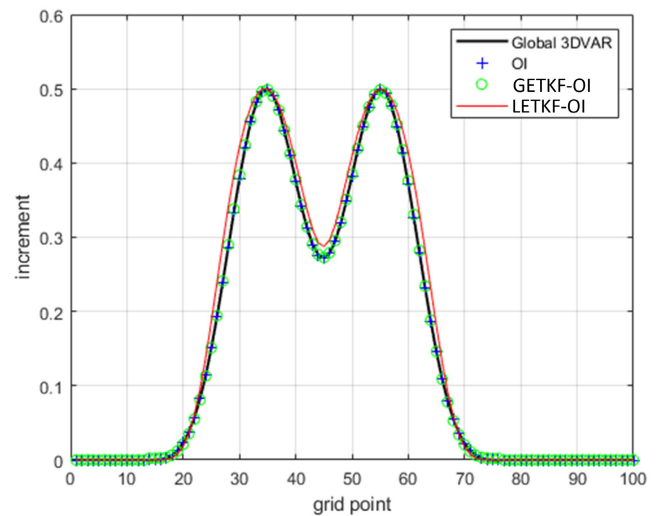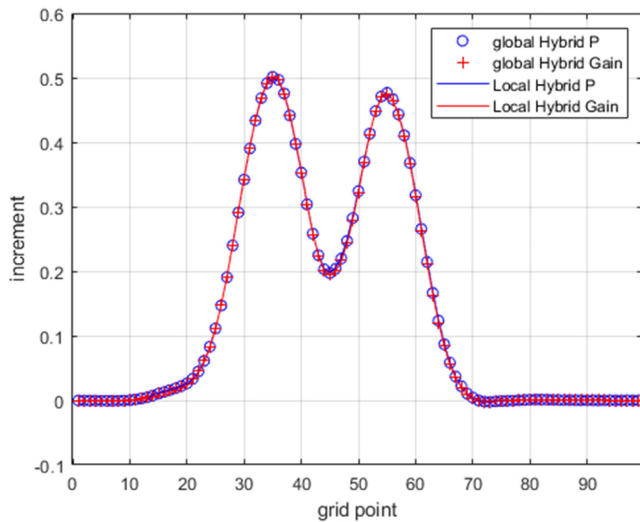

FIGURE 4 Comparison of global 3DVAR increment for the univariate case with three local solutions: (thick black) global 3DVAR solution; (blue cross) traditional OI; (green circle) GETKF-OI using Equation T.7; and (red thin line) LETKF-OI using Equation T.8. Table 2 also lists specific formulas used to compute analysis increments in this figure. [Colour figure can be viewed at wileyonlinelibrary.com]

OI algorithm is that we truncated the eigen spectrum of the static correlation matrix $\mathbf{P}_{static}$ to retain 99% of the variance. By comparison, the LETKF-OI algorithm that relies on the R-inflation for horizontal localization produces a very close (but not perfect) approximation to the 3DVAR analysis increment (compare solid black and red lines, NRMSE = 8%). This difference is expected as the R-inflation is an approximation to model-space localization, and the two are not equivalent.

4.1.2 | Comparison of algorithms with the hybrid covariance update

Figure 5 shows that all four hybrid updates produce practically indistinguishable increments (less than 1% NRMSE difference from the global hybrid-P in Equation T.2). The four algorithms considered include global (circle and cross) and local (solid lines) updates with hybrid error covariance formula (blue) and hybrid gain (red). The equivalence between local and global hybrid gain follows from a close correspondence of the 3DVAR and GETKF-OI solutions found in Figure 4. The equivalence between the global and local hybrid covariance update follows from our ability to express the global hybrid covariance $\mathbf{P}_{hyb}$ as an augmented ensemble of modulated static and flow-dependent ensemble members (see Equation 13).
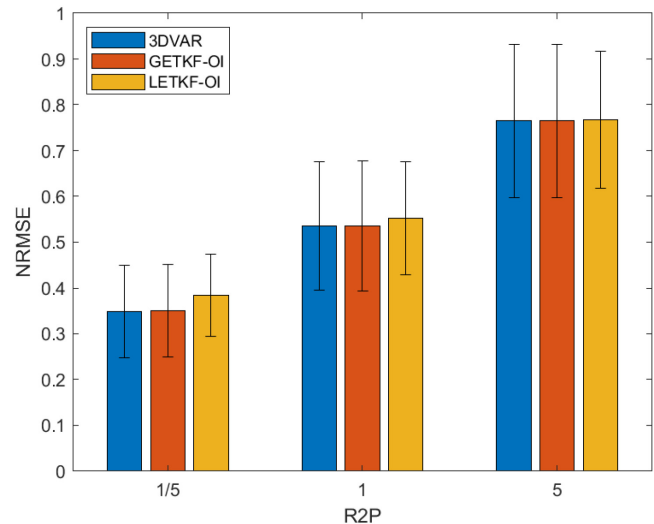
FIGURE 5 Comparison of global and local hybrid increments. (blue circle) Global hybrid P solution (Equation T.2); (blue solid) local hybrid P solution using augmented ensemble (Equation T.10); (red cross) global hybrid gain solution (Equation T.4); and (red solid and Equation T.9) local hybrid gain solution. Table 2 lists specific formulas used to compute analysis increments in this figure [Colour figure can be viewed at wileyonlinelibrary.com]
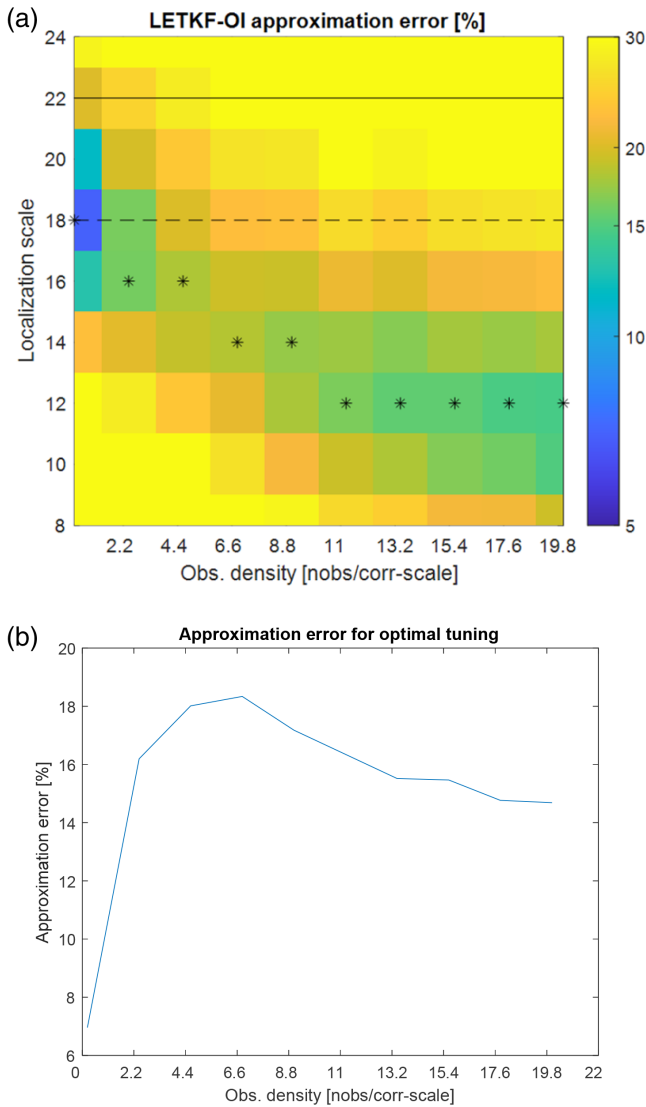

FIGURE 6 NRMSE for global 3DVAR and local LETKF-OI and GETKF-OI solutions in randomized trials as a function of R2P ratio. Mean error is shown with solid bars and the standard deviation of error is shown with vertical bars. Abbreviations: GETKF, gain-form ensemble-transform Kalman filter; LETKF, local ensemble-transform Kalman filter; NRMSE, normalized root mean square error; OI, optimal interpolation [Colour figure can be viewed at wileyonlinelibrary.com]

4.1.3 | Approximation errors

Figure 6 shows that in large randomized studies, all three methods (LETKF-OI, GETKF-OI, and global 3DVAR) produce similarly accurate reconstructions of a random true state. Figure 6 shows that GETKF-OI and 3DVAR produce almost numerically indistinguishable results despite truncation of the covariance spectra in GETKF-OI. Interestingly, the LETKF-OI algorithm has very similar average NRMSE errors. This is despite the fact that there were about 10% differences between the LETKF-OI and the 3DVAR increment. We attribute this to the fact that the ability of the DA algorithm to reproduce a true "twin" state was in large part determined by the imperfections of the observing network due to sampling density and added observational noise.

4.1.4 | Limitations of the LETKF-OI algorithm

Figure 7 shows that LETKF-OI's performance depends on the observation density (number of observations with a correlation scale). The errors are minimal when observations are sparse (RMSE = 7% for obs. density < 1) and increase as observations get closer (maximum RMSE of 18% when 6.6 observations were present within a single correlation scale). Furthermore, Figure 7 shows that

optimal observation localization length scale is a function of observation density (18 grid points for isolated observations and 12 grid points for dense observations). Both optimal length scales are shorter than the "true" correlation scale (shown with the solid line) that was used to generate the "twin" solution.
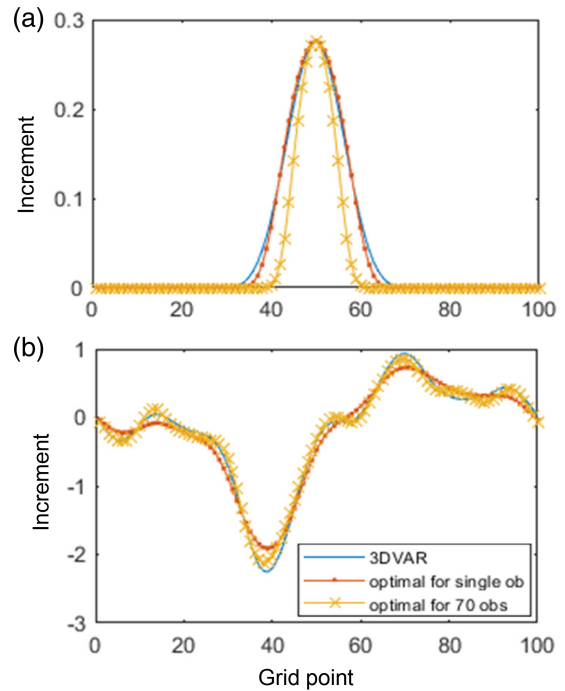
In a realistic scenario, observation densities are likely to change in space and a single localization scale might not be optimal. Figure 8 illustrates this scenario by comparing the increment for the 3DVAR (blue line), the LETKF-OI increment optimized for the sparse observing network (red line), and the LETKF-OI increment for the dense observing network (yellow line). When a single observation is assimilated, it is possible to tune LETKF-OI to reproduce the reference 3DVAR solution closely (compare blue and red lines in Figure 8a). However, if we use LETKF-OI tuned for high observation densities to assimilate sparse observations, we produce an increment that is markedly different from the optimal LETKF-OI or the reference 3DVAR increment (compare the yellow line to the red and blue lines in Figure 8a). Similar situation occurs when we use LETKF-OI tuned for a single observation to assimilate dense observations (the red line in Figure 8b is too smooth compared to the blue and the yellow lines).

To further understand the reason behind the degradation of the LETKF-OI approximation with dense observing networks, we plotted (Figure 9) the eigen spectra for the local matrix ($\mathbf{H}_{(i)}\mathbf{PH}_{(i)}{}^{\mathrm{T}} + \mathbf{R}_{(i)}$) where (i) is for the

FIGURE 8 Increments produced by assimilating a single observation (a) and 50 randomly distributed observations (b) [Colour figure can be viewed at wileyonlinelibrary.com]
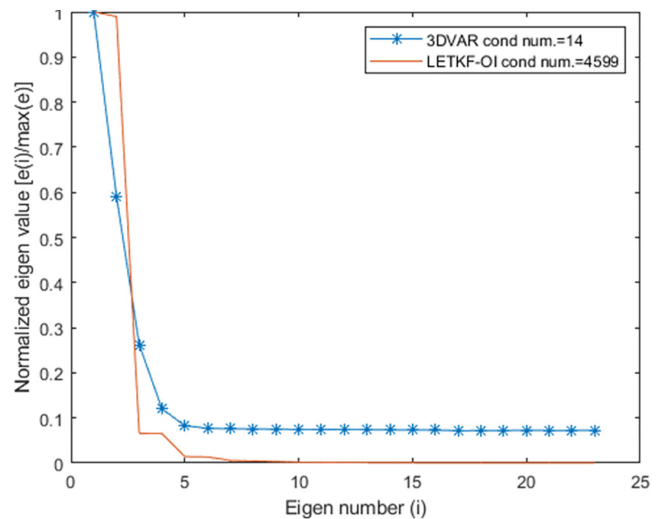
FIGURE 7 LETKF-OI approximation error. (a) Approximation error (shown in color) as a function of observation density (x-axis) and observation localization scale (y-axis). Lines in (a) indicate: (solid line) correlation scale for the covariance used in the 3DVAR increment; (dashed line) observation localization tuned using a single observation; (stars) observation localization scale optimized for specific observation density. (b) Approximation error for the optimal tuning of localization as a function of observation density (e.g. approximation errors at the location of * in a). Errors were computed compared to the 3DVAR increment using 30 randomly drawn innovations. Abbreviations: LETKF, local ensemble-transform Kalman filter; OI, optimal interpolation [Colour figure can be viewed at wileyonlinelibrary.com]

FIGURE 9 Eigen spectra of the local $(\mathbf{H}_{(j)}\mathbf{P}\mathbf{H}_{(j)}^{T} + \mathbf{R}_{(j)})$ with every grid point observed in the local volume. Abbreviations: LETKF, local ensemble-transform Kalman filter; OI, optimal interpolation [Colour figure can be viewed at wileyonlinelibrary.com]

50th grid point. Figure 9 shows that unlike the 3DVAR, the LETKF-OI expression is poorly conditioned (condition number of 4,599 vs 14). The LETKF-OI spectrum also has just a few eigen clusters while the 3DVAR spectrum decays gradually. We interpret these results as the LETKF-OI having fewer degrees of freedom to model

the interaction of the dense observations located within a single correlations scale. Specifically, LETKF-OI does not account for correlations between observations within a volume while GETKF-OI, 3DVAR, and OI all allow for such observation–observation correlations. This is
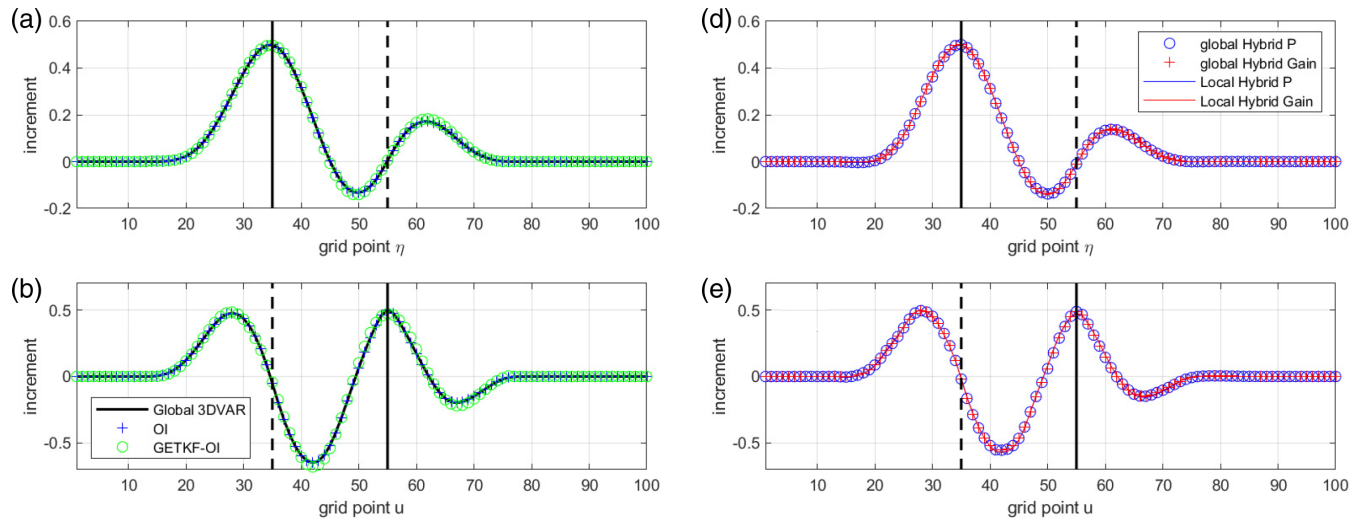
FIGURE 10 Comparison of increments for the multivariate test problem. The top panel shows increment to η and bottom to u variable. The left column shows increments from deterministic algorithms (3DVAR, OI, and GETKF-OI) and the right column shows increments for hybrid algorithms (global and local version of the hybrid covariance and hybrid gain). Vertical lines show location of the η observation (solid black line) and of the u observation (dotted black line). Table 2 lists specific formulas used to compute analysis increments in this figure. Abbreviations: GETKF, gain-form ensemble-transform Kalman filter; OI, optimal interpolation [Colour figure can be viewed at wileyonlinelibrary.com]

consistent with the finding of Wang *et al.* (2021) that observation-space localization results in a lower rank ensemble than modulated ensemble in the GETKF solver.

4.2 | Multivariate case

Figure 10 shows that when we assimilate two observations (η located at point 35 and u located at point 55), the GETKF-OI algorithm (green circle in panels a and b) can effectively reproduce reference solutions (3DVAR and OI, solid black line and blue crosses in panels a and b). Figure 10 also shows that the local and global hybrid solutions are equivalent, which is expected given a close approximation of the GETKF-OI solution to the 3DVAR in Figure 10a,b.

5 | SUMMARY

In this paper, we derived a new class of local data assimilation algorithms that can be used for DA using hybrid and parameterized static error covariances. Using a simple statistical model for a one-dimensional problem, we demonstrated that it is possible to achieve numerical equivalency between the 3DVAR-like global algorithm with a parameterized covariance and the new GETKF-OI update algorithm. The modulated ensemble in the GETKF-OI is achieved through modulation product of the truncated square root of the parameterized covariance and a single

1-vector. A similar result was achieved for a hybrid error covariance matrix. We note that for large model domains, storing of the truncated square roots of the parameterized covariance required by GETKF-OI is impractical. Instead we point out (see Appendix B for details) that in the context of the local volume solver it is sufficient to store the leading eigenvectors of the covariance function defined on the footprint of the local volume.

We also showed that for a univariate case, with a potential gain in the computational speed and a straightforward implementation, it is possible to achieve close (but not perfect) correspondence between the 3DVAR solution and the LETKF-OI algorithm. This single ensemble member used by the LETKF-OI is set to be equal to the standard deviation of the forecast error. Close examination of the tuning required for LETKF-OI to represent the reference 3DVAR solution illuminated limitations of the R-localization that are often obscured when it is used to localize ensemble-based covariance matrixes. Specifically (see Figure 8 for details), we find that: (a) the tuned observation-space localization length scale is always shorter than the length scale of the corresponding model-space covariance; (b) the value of the tuned observation-space's localization depends on the density of the observing network; and (c) LETKF-OI does not account for correlations between observations within a local volume.

Finally, using large randomized trials, we found that the global 3DVAR, GETKF-OI, and LETKF-OI could all reconstruct the randomly-generated true solution with

equal accuracy. This was despite the fact that there was about a 10% difference between the 3DVAR and LETKF-OI increment. We attribute this similarity between the three systems to the fact that in practice, as well as in our randomized studies, the accuracy of the DA reconstruction is determined by the density and the accuracy of the observation network.

6 | DISCUSSION

The findings of this paper are encouraging and provide DA algorithm designers with a new opportunity to use parameterized covariance modeling in highly-scalable LETKF-like DA algorithms. We speculate that addition of the high-rank static covariance model in the GETKF algorithm can improve the quality of the flow-dependent ensemble and possibly allow for cycling systems with smaller numbers of flow-dependent ensembles without degradation in the analysis and forecast skill scores. Extensive testing in real-world applications will be essential to understand the benefits and the limitations of the proposed algorithms. Our initial application of the LETKF-OI algorithm to the assimilation of snow-fall data shows promise and will be presented in a separate publication.

We expect that further progress in cycling systems with multivariate balances (like the ocean and the atmosphere) will require further development of the GETKF-OI approach described here. This will involve experimentation with realistic oceanic and atmospheric observations and backgrounds. Comparisons with reference 3DVAR solutions and evaluations of the impact of hybrid update on the ensemble members will be essential. We suspect that it would be possible to model the static covariance as a separable process, with a separate set of eigenvectors for the vertical and horizontal correlation functions. Such implementation can be similar to: (a) Lei *et al.* (2018), where the GETKF algorithm is used to assimilate satellite radiances that represent vertical integrals of atmospheric quantities; and (b) Wang *et al.* (2021) who used eigenvectors to characterize horizontal multi-scale correlation functions. The Joint Effort for Data Assimilation Integration (JEDI) framework provides a unique opportunity for developing GETKF-OI algorithms because it provides the reference 3DVAR solutions, vertical balance operators, and variable transforms needed to capture horizontal balances.

AUTHOR CONTRIBUTIONS
Sergey Frolov: Conceptualization; formal analysis; funding acquisition; investigation; methodology; software; validation; visualization; writing – original draft; writing – review and editing. **Jeffrey S Whitaker:** Funding acquisition; investigation; supervision; writing – review and editing. **Clara Draper:** Investigation; writing – review and editing.

ORCID
Sergey Frolov https://orcid.org/0000-0002-9081-1979

REFERENCES
Bauer, P., Quintino, T, Wedi, N, Bonanni, A, Chrust, M, Deconinck, W, Diamantakis, M, Düben, P, English, S, Flemming, J, Gillies, P, Hadade, I, Hawkes, J, Hawkins, M, Iffrig, O, Kühnlein, C, Lange, M, Lean, P, Marsden, O, Müller, A, Saarinen, S, Sarmany, D, Sleigh, M, Smart, S, Smolarkiewicz, P, Thiemert, D, Tumolo, G, Weihrauch, C, Zanna, C, Maciel, P. (2020) The ECMWF scalability Programme: Progress and plans. Reading, UK. https://doi.org/10.21957/gdit22ulm.

Bishop, C.H., Hodyss, D., Steinle, P., Sims, H., Clayton, A.M., Lorenc, A.C., Barker, D.M. and Buehner, M. (2011) Efficient Ensemble Covariance Localization in Variational Data Assimilation. *Monthly Weather Review*, 139(2), 573–580. https://doi.org/10.1175/2010MWR3405.1.

Bishop, C.H., Huang, B. and Wang, X. (2015) A non-Variational consistent hybrid ensemble filter. *Monthly Weather Review*, 143, 5073–5090. https://doi.org/10.1175/MWR-D-14-00391.1.

Bishop, C.H., Whitaker, J.S. and Lei, L. (2017) Gain form of the ensemble-transform Kalman filter and its relevance to satellite data assimilation with model space ensemble covariance localization. *Monthly Weather Review*, 145(11), 4575–4592. https://doi.org/10.1175/MWR-D-17-0102.1.

Bonavita, M., Hamrud, M. and Isaksen, L. (2015) EnKF and hybrid gain ensemble data assimilation. Part II: EnKF and hybrid gain results. *Monthly Weather Review*, 143(12), 4865–4882. https://doi.org/10.1175/MWR-D-15-0071.1.

Buehner, M., Houtekamer, P.L., Charette, C., Mitchell, H.L. and He, B. (2010) Intercomparison of Variational data assimilation and the ensemble Kalman filter for global deterministic NWP. Part I: description and single-observation experiments. *Monthly Weather Review*, 138(5), 1550–1566. https://doi.org/10.1175/2009MWR3157.1.

Chatterjee, A.G., Verma, M.K., Kumar, A., Samtaney, R., Hadri, B. and Khurram, R. (2018) Scaling of a fast Fourier transform and a pseudo-spectral fluid solver up to 196608 cores. *Journal of Parallel and Distributed Computing*, 113, 77–91. https://doi.org/10.1016/j.jpdc.2017.10.014.

Cummings, J.A. and Smedstad, O.M. (2013) Variational data assimilation for the Global Ocean. In: Park, S. and Xu, L. (eds) *Data Assimilation for Atmospheric, Oceanic and Hydrologic Applications*, Vol. II, Springer, Berlin, Heidelberg. pp. 303–343. https://doi.org/10.1007/978-3-642-35088-7_13.

Evensen, G. (2003) The ensemble Kalman filter: theoretical formulation and practical implementation. *Ocean Dynamics*, 53(4), 343–367.

Fisher, M. (2003) Background error covariance modelling. In: *Seminar on Recent Developments in Data Assimilation for Atmosphere and Ocean*. ECMWF: Reading, pp. 45–63.

Gandin, L. (1963) *Objective Analysis of Meteorological Fields*. Leningrad: Hydrometeoizdat.

Gaspari, G. and Cohn, S.E. (1999) Construction of correlation functions in two and three dimensions. *Quarterly Journal of the Royal Meteorological Society*, 125, 723–757.

Hamill, T.M. and Snyder, C. (2000) A hybrid ensemble Kalman filter–3D Variational analysis scheme. *Monthly Weather Review*, 128(8), 2905–2919. https://doi.org/10.1175/1520-0493(2000)128<2905:AHEKFV>2.0.CO;2.

Hamrud, M., Bonavita, M. and Isaksen, L. (2015) EnKF and hybrid gain ensemble data assimilation. Part I: EnKF implementation. *Monthly Weather Review*, 143(12), 4847–4864. https://doi.org/10.1175/MWR-D-14-00333.1.

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

Kotsuki, S. and Bishop, C.H. (2021) Implementing hybrid background error covariance into the LETKF with attenuation-based localization: experiments with a simplified AGCM. *Monthly Weather Review*, 1(aop), 283–302. https://doi.org/10.1175/MWR-D-21-0174.1.

Kretschmer, M., Hunt, B.R. and Ott, E. (2015) Data assimilation using a climatologically augmented local ensemble-transform Kalman filter. *Tellus A: Dynamic Meteorology and Oceanography*, 67(1), 26617. https://doi.org/10.3402/tellusa.v67.26617.

Lei, L., Whitaker, J.S. and Bishop, C. (2018) Improving assimilation of radiance observations by implementing model space localization in an ensemble Kalman filter. *Journal of Advances in Modeling Earth Systems*, 10(12), 3221–3232. https://doi.org/10.1029/2018MS001468.

Ménétrier, B. (2020) *Normalized Interpolated Convolution from an Adaptive Subgrid documentation*. Available at: https://github.com/benjaminmenetrier/nicas_doc.

Ménétrier, B. and Auligné, T. (2015) Optimized localization and hybridization to filter ensemble-based Covariances. *Monthly Weather Review*, 143(October), 3931–3947. https://doi.org/10.1175/MWR-D-15-0057.1.

Penny, S.G. (2014) The hybrid local ensemble transform Kalman filter. *Monthly Weather Review*, 142(6), 2139–2149. https://doi.org/10.1175/MWR-D-13-00131.1.

Purser, R.J., Wu, W.-S., Parrish, D.F. and Roberts, N.M. (2003) Numerical aspects of the application of recursive filters to variational statistical analysis. Part I: spatially homogeneous and isotropic Gaussian covariances. *Monthly Weather Review*, 131, 1524–1535. https://doi.org/10.1175//1520-0493(2003)131<1524:NAOTAO>2.0.CO;2.

Rosmond, T. and Xu, L. (2006) Development of NAVDAS-AR: non-linear formulation and outer loop tests. *Tellus A*, 56(1), 45–58.

Sakov, P. and Bertino, L. (2011) Relation between two common localisation methods for the EnKF. *Computational Geosciences*, 15(2), 225–237. https://doi.org/10.1007/s10596-010-9202-6.

Wang, X., Chipilski, H.G., Bishop, C.H., Satterfield, E., Baker, N. and Whitaker, J.S. (2021) A multiscale local gain form ensemble transform Kalman filter (MLGETKF). *Monthly Weather Review*, 149(3), 605–622. https://doi.org/10.1175/MWR-D-20-0290.1.

Weaver, A. and Courtier, P. (2001) Correlation modelling on the sphere using a generalized diffusion equation. *Quarterly Journal of the Royal Meteorological Society*, 127(575), 1815–1846. https://doi.org/10.1256/smsqj.57517.

Weaver, A.T., Deltel, C., Machu, E., Ricci, S. and Daget, N. (2005) A multivariate balance operator for variational ocean data assimilation. *Quarterly Journal of the Royal Meteorological Society*, 131(613), 3605–3625. https://doi.org/10.1256/QJ.05.119.

APPENDIX A: CONSIDERATIONS ON THE SCALABILITY OF THE 3DVAR-LIKE AND LETKF-LIKE ALGORITHMS

While the scalability of the LETKF and GETKF are well established (Bishop *et al.*, 2017; Kotsuki and Bishop, 2021), scalability of the 3DVAR-like algorithms with a parametric covariance function have strong dependence on the implementation of the matrix–vector product of the state vector and the parameterized covariance:

$$\mathbf{a} = \mathbf{P}^{\mathbf{static}}\mathbf{x} \qquad (A1)$$

Several implementations of the Equation (A1) exist, including spherical harmonics and wavelet basis (Fisher, 2003), diffusion operator (Weaver and Courtier, 2001), recursive filters (Purser *et al.*, 2003), explicit convolution on a sparse grid (Ménétrier and Auligné, 2015), and direct computations (Rosmond and Xu, 2006; Cummings and Smedstad, 2013). All of the above methods have different computational cost for a given problem size and different scaling performance with the increase in the number of processing units (strong scaling) or the problem size (weak scaling). However, all of the above methods are characterized by the need for global communication that often limits practical scalability of the Equation (A1).

Of the above-mentioned methods, we can further comment on the practical scalability of the spherical harmonics and explicit convolution—possibly the two

fastest implementations for Equation (A1) for existing global model grids. Following (Chatterjee *et al.*, 2018; Bauer *et al.*, 2020), the computational cost and the scalability of the spectral transforms is limited by the global fast Fourier transform algorithms that are communication bound due to use of all-to-all communications in the global matrix transpose and matrix–matrix multiplications. The explicit convolution methods (Normalized Interpolated Convolution from an Adaptive Subgrid—NICAS) of (Ménétrier and Auligné, 2015) provide one of the fastest implementation of the Equation (A1). In part, because NICAS utilizes sparse grids that are determined by the length scales of the correlation functions and not by the resolution of the analysis grid. Ménétrier (2020) provides a formal analysis of the communication and computational costs for NICAS. The computational costs on a global grid scale with the sixth power of the sparse grid resolution and the communication costs scale with the square of the number of the processing elements. It is important to note that there is an upper limit to the communication scalability of the NICAS software that depends on the correlation scales of the grid. This limit is on the order of ~200–800 processing-elements for correlation scales typical in the atmosphere (2,000–1,000 km) and is closer to 3,000 processing-elements for correlation scales typical for the ocean (500 km).

The attractive feature of the LETKF algorithm is that local analyses are computed independently. This leads to linear strong scaling of LETKF (linear speed up with an increase in processing elements given a fixed size of the problem). The weak scaling of the LETKF-like algorithms (scaling with the increase of the computational problem) has three components: (a) scaling with the number of grid points, (b) scaling with an increase in the ensemble size, and (c) scaling with increase in the observational density. Given fixed ensemble size and observational density, LETKF/GETKF algorithms scales linearly with increase in number of grid points since each local analysis is computed independently. For example, doubling of the resolution on the 2D global grid will lead to 4-fold increase in the number of local analyses that needs to be computed. This is compared to a factor of 64 increase in the number of computations required by the NICAS software for doubling of the horizontal resolution of the sparse grid.

Central to the local analysis update in Equation (7) and T.7 is the computation of the following matrix inverse (size of the number of ensemble members):

$$\left[(\mathbf{HZ})_{(i)}^T \mathbf{R}_{(i)}^{-1} (\mathbf{HZ})_{(i)} + \mathbf{I} \right]^{-1} \quad \text{(A2)}$$

where $(\mathbf{HZ})_{(i)}$ is the matrix of the ensemble perturbations in the observational space local to the grid point

(i) (size of the number of observations by the number of ensemble perturbations); $\mathbf{R}_{(i)}^{-1}$ is the inverse of the diagonal matrix (size of the number of observations); and $\mathbf{I}$ is the identity matrix (size of the number of ensemble members). The computation cost of A2 consists of the cost of the matrix–matrix product (equal to $n_{ens}^2 n_{local.obs.}$), and the cost of the matrix inverse (scales as n_{ens}^3 or better). In most cases, the number of local observations exceeds the number of ensemble perturbations (Hamrud *et al.*, 2015), hence, the dominant cost of A2 is the matrix–matrix multiplication followed by the cost of the matrix inverse. Cost estimate of A2 also suggests that LETKF-like algorithms scale as a cube of the number of ensemble members and have linear scalability with an increase in the number of local observations.

APPENDIX B: FAST ALGORITHM FOR ENSEMBLE MODULATION IN A LOCAL VOLUME

In this paper, as well as in the multiscale localization paper of Wang *et al.* (2021), one faces a problem of efficient computations of the observations for the modulated ensemble:

$$\mathbf{H}(\sqrt{\mathbf{C}}\Delta \mathbf{X}) = \left[\mathbf{H}\left(\sqrt{\mathbf{c}}_1 \circ \mathbf{x}_1 \right), \dots, \mathbf{H}\left(\sqrt{\mathbf{c}}_i \circ \mathbf{x}_j \right) \right] \quad \text{(B1)}$$

where $\mathbf{H}\left(\sqrt{\mathbf{c}}_i \circ \mathbf{x}_j \right)$ is the observation of the $\mathbf{x}_j$ ensemble member modulated by the $\sqrt{\mathbf{c}}_i$ column of the modulation matrix. In the context of this paper, Equation (B1) is further simplified as:

$$\mathbf{H}\left(\sqrt{\mathbf{P}_{n_{eigSt}}^{static}} \Delta \mathbf{i} \right) = \mathbf{H}\sqrt{\mathbf{P}_{n_{eigSt}}^{static}} \quad \text{(B2)}$$

where $\sqrt{\mathbf{P}_{n_{eigSt}}^{static}}$ is the truncated square root of the static covariance.
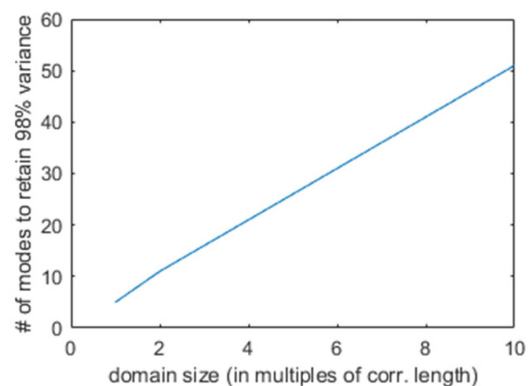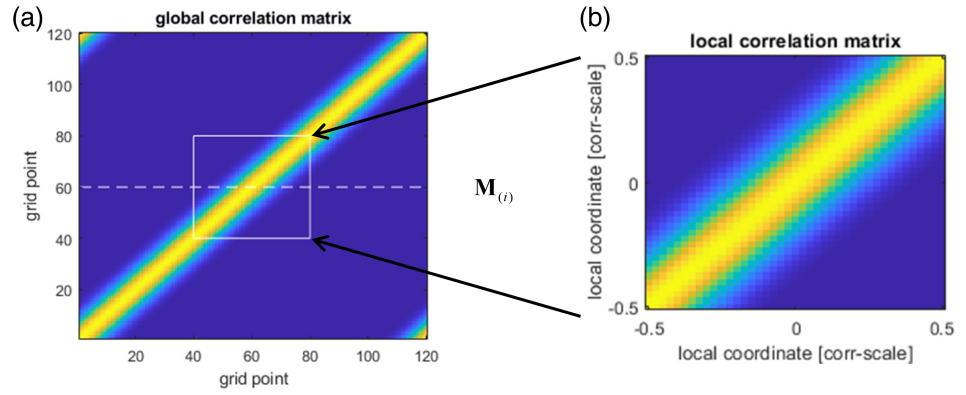

FIGURE B1 Number of EOFs required to represent 98% of the global correlation matrix as the size of the global 1D domain increases [Colour figure can be viewed at wileyonlinelibrary.com]

For a small domain considered in this paper, $\sqrt{\mathbf{P}^{static}_{n_{eigSt}}}$ can be approximated by taking a square root of the global covariance matrix. However, as a size of the domain increases in comparison to the correlation length, one needs to retain an increasing number of eigenvectors (Figure B1)—linear in 1D but potentially cubic increase for 3D.

Here, instead of storing the square root of the global covariance matrix $\sqrt{\mathbf{P}^{static}_{n_{eigSt}}}$, we instead propose to store the local square root $\sqrt{\mathbf{P}^{static}_{n_{eigSt}\,(local)}}$ defined in normalized coordinates $[-1:1]$ (with the center of the volume at 0 and correlations decaying to zero for normalized coordinates outside of this $[-1:1]$ range). The local square root $\sqrt{\mathbf{P}^{static}_{n_{eigSt}\,(local)}}$ can then be related to the global square root $\sqrt{\mathbf{P}^{static}_{n_{eigSt}}}$ by developing a linear mapping operator $\mathbf{M}_{(i)}$ (see Figure B2). Then Equation (B2) can be re-written as:

$$\mathbf{H}_{(i)}\sqrt{\mathbf{P}^{static}_{n_{eigSt}}} = \mathbf{H}_{(i)}\mathbf{M}_{(i)}\sqrt{\mathbf{P}^{static}_{n_{eigSt}\,(local)}} = \mathbf{H}_{(local)}\sqrt{\mathbf{P}^{static}_{n_{eigSt}\,(local)}} \tag{B3}$$

where $\mathbf{H}_{(local)} = \mathbf{H}_{(i)}\mathbf{M}_{(i)}$ is the observation operator that can operate in the local coordinates. In Figure B2 we demonstrate a case where the global correlation matrix

(panel a) is defined on the domain that is three times the size of the correlation length scales. Figure B2 illustrates how operator $\mathbf{M}_{(i)}$ maps correlation matrix defined in the normalized local coordinates (panel b) onto the global domain in panel a.

Equation (B3) provides great computational savings for large global domains by reducing the storage for $\sqrt{\mathbf{P}^{static}_{n_{eigSt}}}$. However, in the context of the local volume solver, Equation (B3) also requires re-observing the same observation locations as they repeat in overlapping observational domains. In case global observation operator $\mathbf{H}$ is used to observe local covariance imbedded in a global domain, the observation operation can become expensive if observation operator requires substantial MPI communication. Hence, we advocate that a practical implementation of the GETKF-OI algorithm in large domains will benefit from development of linearized observational operator $\mathbf{H}_{(local)}$ that can operate in the local coordinate system. In a special case of point observations and the $\sqrt{\mathbf{P}^{static}_{n_{eigSt}\,(local)}}$ considered in this paper, the implementation of the $\mathbf{H}_{(local)}$ is reduced to the interpolation of eigen columns of $\sqrt{\mathbf{P}^{static}_{n_{eigSt}\,(local)}}$ to the observation locations in the reference frame of the local volume. That is, we suggest that observation of the modulated ensemble can be replaced by linear interpolation of eigenvectors.