RESEARCH ARTICLE

Ensemble Kalman filter for nonconservative moving mesh solvers with a joint physics and mesh location update

Christian Sampson[1] | Alberto Carrassi[2,3] | Ali Aydoğdu[4] | Chris K.R.T Jones[1]

[1]University of North Carolina, Chapel Hill, North Carolina

[2]Department of Meteorology and National Centre for Earth Observations, University of Reading, Reading, UK

[3]Mathematical Institute, University of Utrecht, Utrecht, the Netherlands

[4]Centro Euro-Mediterraneo sui Cambiamenti Climatici, Bologna, Italy

Correspondence
C. Sampson, Department of Mathematics, University of North Carolina at Chapel Hill, Chapel Hill, NC 27599, USA
Email: Christian.Sampson@gmail.com

Present Address
Mathematics Department, University of North Carolina at Chapel Hill, Chapel Hill, NC 27599, USA

Abstract

Numerical solvers using adaptive meshes can focus computational power on important regions of a model domain capturing important or unresolved physics. The adaptation can be informed by the model state or external information, or made to depend on the model physics. In this latter case, one can think of the mesh configuration *as part of the model state*. If observational data are to be assimilated into the model, the question of updating the mesh configuration with the physical values arises. Adaptive meshes present significant challenges when using popular ensemble data assimilation (DA) methods. We develop a novel strategy for ensemble-based DA, for which the adaptive mesh is updated along with the physical values. This involves including the node locations as a part of the model state itself, allowing them to be updated automatically at the analysis step. This poses a number of challenges, which we resolve to produce an effective approach that promises to apply with some generality. We evaluate our strategy with two testbed models in one dimension (1-d), comparing them with a strategy we previously developed that does not update the mesh configuration. We find that updating the mesh improves the fidelity and convergence of the filter. An extensive analysis on the performance of our scheme beyond just the root-mean-squared error (RMSE) is also presented.

KEYWORDS

adaptive meshes, data assimilation, ensemble Kalman filter, Lagrangian solvers

1 | INTRODUCTION

Modern adaptive moving mesh schemes present significant advantages over traditional fixed mesh schemes in many geophysical applications. Adaptive meshes can focus resolution in places of interest, in order to make better use of available computational power (Huang and Russell, 2010), or can be designed to optimise computational cost and accuracy based on external factors, an example being ship and acoustic receiver locations in the prediction of underwater noise pollution from oceanic shipping activity (Trigg *et al.*, 2018). In some applications, one may require the mesh to change as the system evolves to represent the underlying physics better (Weller *et al.*, 2010). Adaptive meshes are typically governed by

Abbreviations: BGM, Burgers' model; EnKF, ensemble Kalman filter; HR, high resolution; HRA, high resolution augmented; KSM, Kuromoto–Shivashinsky model; LR, low resolution; RMSE, root-mean-squared error.

a set of rules suitable for the specific problem being solved.

There are many reasons why solving a geophysical problem in a Lagrangian frame may be appropriate: see, for example, Asch *et al.* (2016) and Jablonowski *et al.* (2004). The associated numerical solver will inevitably be based on a moving mesh, and some of the advantages of a moving mesh described above are delivered a fortiori by the use of such a scheme that advects the nodes with the flow. For instance, this may have the effect of naturally concentrating nodes in locations of increased activity, or resolving coherent structures better. More specifically, if the nodes are advected with the flow, node clusters can provide information about where gradients are large and sinks or eddies exist; likewise, node deserts can indicate where gradients are small. When the nodes are advected by the flow in this way, it is almost inevitable that they will have to change in both number and location in order to maintain solution accuracy for the numerical solver. When using an adaptive mesh that is governed by the model physics like this, the node locations and physical quantities are inexorably coupled. As a consequence, the node locations can be considered part of the model state and the model's solution history.

The key point to note is that, for computational models based on such Lagrangian solvers, the node locations encode the underlying physics and therefore provide information about the overall model state. As such, they are all updated together under the model evolution. In addition, the observational data also reflect the underlying physics and we would therefore expect that the optimal incorporation of such data should update the node locations as well as the values of the physical state variables. We develop here a data assimilation scheme for achieving exactly this impact of data on the mesh itself.

The process of incorporating data into physical models is called data assimilation (DA). A survey of DA methods can be found in Budhiraja *et al.* (2018). Data assimilation has become an integral tool in the geosciences and meteorology, improving numerical weather prediction, and a method for parameter estimation. A review of DA in the geosciences can be found in Carrassi *et al.* (2018). The careful consideration of an adaptive mesh within a data assimilation scheme has been shown to improve forecast skill in variational DA schemes. We will focus here on ensemble methods (Evensen, 2009; Houtekamer and Zhang, 2016) that make use of estimated statistics from an ensemble of model runs at an analysis time step. These methods are attractive when attempting to leverage the information that node locations carry through covariances estimated from the ensemble members. That information is specifically brought in through the cross-covariances between the physical values and the node locations driven by those

values. In the case where the nodes are advected with the flow, these cross-covariances mimic the spatial gradient of the fluid velocities across the model domain. This encodes extra and important physical information into the DA update step. We will use observations of the physical state to update both the physical values and node locations in our approach, which, in this work, will come from a twin model experiment using the models outlined in Section 2.

Adapting existing ensemble methods to adaptive moving mesh models involves tackling some significant challenges. Ensemble DA methods rely on estimated statistics from the ensemble members and for success they must be statistically consistent. The main challenge is the fact that each ensemble member may have nodes in different locations, in different numbers, or both. Previous work along these lines has been carried out in Bonan *et al.* (2017) for an adaptive mesh one-dimensional (1-d) ice sheet model. In that work, the adaptive mesh was conservative, in that each ensemble member has the same number of points. Observations of the ice-sheet edge were also assimilated directly. In this work, we consider updating node locations for a nonconservative adaptive mesh model using Eulerian (*fixed in space*) observations of the physical quantities of the "truth run". A nonconservative mesh means that each ensemble member will have a *different number* of nodes in *different locations*, requiring us to develop methodologies to obtain consistent and meaningful error covariance estimates. Any methodology we develop will necessarily have some disruptive effect on the individual ensemble members themselves in order to achieve a measure of statistical consistency between them. This may come through the addition or removal of nodes or the interpolation of values to specific locations. With this in mind, we define a successful method as one that improves the estimate of the truth over a "control run" with no DA and take special care to study the effects the method has on the ensemble members themselves. We discuss these effects in Section 4 and recommend considerations to minimise any negative effects depending on the application at hand and the desired prediction goal.

Other ensemble approaches aimed at adaptive meshes have been developed. Jain *et al.* (2018) study a tsunami model that uses an adaptively refined mesh, taking the union of all meshes as a reference mesh to which each ensemble member is interpolated to before the update step. In Du *et al.* (2016), a model that uses a 3D unstructured adaptive mesh model for geophysical flows (see also Maddison *et al.*, 2011; Davies *et al.*, 2011) was considered and an EnKF developed, which uses the idea of a reference mesh to carry out the analysis step. The reference mesh is chosen using the idea of supermeshing (see Farrell *et al.*, 2009) and each ensemble member is interpolated to that fixed

reference mesh before the analysis step. These previous studies all concern conservative adaptive meshes.

In Aydoğdu *et al.* (2019), a fixed reference mesh is used in two 1-d models for which the mesh evolves with the flow and undergoes a "remeshing" step, which injects new nodes should two be too far apart or removes nodes should two be too close together. This remeshing means that each ensemble member will likely have different numbers of points in different locations. In that work, two reference meshes are used and these are chosen based on the rules of the remeshing scheme. Ensemble members are mapped to the reference mesh before the update and mapped back to their previous meshes after. Our work goes further, extending the update to the node locations themselves. We use the reference mesh only as a guide to match components of the state vector and augment our state vector with the node locations. A reasonable supposition is that avoiding the mapping scheme will help to lessen disruption of individual ensemble members, providing for better estimates of the error covariances needed for the update step.

This article is structured as follows. In Section 2 we describe the model and adaptive mesh scheme we use in our twin model experiments. In Section 3 we outline the necessary ingredients for an EnKF on nonconservative adaptive mesh models and describe the two implementations of such that we will compare. Numerical results are presented in Section 4, along with the optimised parameters needed for the methods. We follow the results with a discussion in Section 4 on the cross-covariances of the physical variables and node locations, as well as the effect the adapted EnKF schemes have on the ensemble members themselves. We also present considerations on choosing the inflation parameters, depending on the application, and finally in Section 5 we present some concluding remarks and summary.

2 | MODEL AND MESH

2.1 | Adaptive mesh

In this work, we are interested in adaptive meshes that evolve with the flow of a physical system and are non-conservative. We will make use of the same 1-d adaptive mesh scheme developed in Aydoğdu *et al.* (2019) as a prototype of the 2- or 3-d nonconservative adaptive meshes used in some modern numerical models, including the Lagrangian neXt generation Sea Ice Model, neXtSIM (Rampal *et al.*, 2016; Rabatel *et al.*, 2018; Cheng *et al.*, 2020).

The mesh itself is a 1-d mesh defined on the domain $D = [0, L)$ with nodes $\{z_1, z_2, \ldots, z_N\} \in D$. It is assumed that $0 \leq z_i < z_{i+1} < L$ and that the positions of the nodes satisfy criteria that define a valid mesh through two tolerance parameters δ_1, δ_2. A valid mesh is one for which

$$\delta_1 \leq |z_{i+1} - z_i| \leq \delta_2 \quad \forall i \in \mathbb{N} : 1 \leq i \leq N - 1 \quad (1)$$

and

$$\delta_1 \leq |z_1 + L - z_N| \leq \delta_2. \quad (2)$$

These criteria ensure that the mesh is periodic and that no two nodes are closer than δ_1 or further apart than δ_2. Moreover, δ_1 and δ_2 are chosen so that $\delta_2/\delta_1 \geq 2$ and are both divisors of L (see Aydoğdu *et al.*, 2019, for an extensive explanation and details on the assumptions).

The mesh points themselves evolve directly with the velocity **u** as

$$\frac{dz_i}{dt} = \mathbf{u}(t, z_i). \quad (3)$$

Equation 3, together with the physical model updating the velocity (along with any other model state variables), represents a coupled system of equations, which can be solved alternately or simultaneously (Huang and Russell, 2010).

Given that the node locations are a function of time, $z_i = z_i(t)$, it is clear that there will be instances when the criteria for a valid mesh given in Equations 1 and 2 are violated. In such cases, we need a suitable remeshing scheme to enforce our criteria, which is given as follows. For each i, if $|z_{i+1} - z_i| < \delta_1$, z_{i+1} is deleted. Alternately, if $|z_{i+1} - z_i| > \delta_2$, a new point z^* is inserted at the midpoint between z_{i+1} and z_i and the points are re-indexed according to their order from left to right. The most relevant consequence of this is that the number of nodes in the mesh is not constant.

2.2 | Models and observations

In this work, we consider two models for use in our numerical experiments. The first is a diffusive form of Burgers' equation (BGM: Burgers, 1948):

$$\mathbf{BGM} : \quad \frac{\partial u}{\partial t} + u \frac{\partial u}{\partial z} = \nu \frac{\partial^2 u}{\partial z^2}, \quad z \in [0, 1), \quad (4)$$

with viscosity $\nu = 0.08$ and periodic boundary and initial conditions:

$$u(0, t) = u(1, t), \quad (5)$$

$$u(z, 0) = \sin(2\pi z) + \frac{1}{2} \sin(\pi z). \quad (6)$$

The Burgers equation has been used in several DA studies (Cohn, 1993; Verlaan and Heemink, 2001;

Pannekoucke *et al.*, 2018). This model is of particular interest because of the steep gradients near the shock, a motivating reason to use an adaptive mesh.

The second model is a version of the Kuramoto–Sivashinsky (KSM) equation (Papageorgiou and Smyrlis, 1991) given by

$$\text{KSM}: \quad \frac{\partial u}{\partial t} + \nu \frac{\partial u^4}{\partial z^4} + \frac{\partial u^2}{\partial z^2} + u \frac{\partial u}{\partial z} = 0, \quad z \in [0, 2\pi). \tag{7}$$

The periodic boundary and initial condition are defined as

$$BC: u(0, t) = u(2\pi, t), \tag{8}$$

$$IC: u(z, 0) = -\sin(2\pi z). \tag{9}$$

Here the viscosity, $\nu = 0.027$, is chosen so that we see chaotic behaviour in the model. KSM, exhibits spatially extended chaos, a situation different from that of BGM but also one for which adaptive meshes can be beneficial. For this reason, we study this model in addition to the Burgers equation.

Both models are solved using central differences and an Eulerian time-stepping scheme with time steps of 10^{-3} for BGM and 10^{-5} for KSM. The tolerances used in the remeshing scheme outlined in Section 2.1 are $\delta_1 = 0.01$, $\delta_2 = 0.02$ for BGM and $\delta_1 = 0.02\pi$, $\delta_2 = 0.04\pi$ for KSM.

Observations of the physical values are generated from high-resolution "nature" runs for both models. For KSM, there is an initial spin-up to $T = 20$ before observations are taken and the model state at that time is used to initialise the ensemble members in the DA experiments described in Section 4. Mean-zero, Gaussian-distributed white noise is added to the observations for both models and experiments are carried out with differing observation-error standard deviation, σ_o. The observations are Eulerian, that is, they are taken on a fixed-in-time regularly spaced grid on the underlying spatial interval and at regular time intervals. The choice of regular spatial and temporal distributions for the data is made for the sake of simplicity and it can be relaxed without impact on the algorithm setup.

3 | EnKF FOR AN ADAPTIVE MOVING MESH MODEL: AMMEnKF

The ensemble Kalman filter (EnKF) relies on estimates of error statistics using an ensemble of model runs assumed to be Gaussian-distributed. The error estimates themselves are calculated using the state vector formed from each ensemble member. In the case of an Eulerian solver

with a fixed mesh, this calculation is easily carried out, as the number of nodes and their locations are the same for each ensemble member and thus the dimension of the state vector is also the same for each ensemble member. In contrast, for an adaptive moving mesh (AMM), the mesh node locations for each ensemble member will almost certainly be in different locations at an assimilation time. Further, due to the remeshing outlined in Section 2.1, each ensemble member will have different numbers of nodes as well. This makes estimation of the error statistics less direct and gives rise to a need for the development of modified versions of the EnKF suited to models with solvers like those we consider here.

For a nonconservative AMM solver, we see two additional steps to be necessary, each with their own important considerations. We refer to the first key step needed before applying the EnKF as *dimension matching*; this is needed to provide consistent estimations of ensemble statistics. One would need to decide whether to add or remove points from the meshes associated with individual ensemble members to achieve the same number of components among state vectors. In addition, a substep is that of *component pairing*, that is, how to assign which points, possibly in different locations, are to be compared in the state vectors. The second key step comes after applying the update, which we refer to as *dimension return*. This involves deciding whether or not to remove points that had been added or, if points were removed, whether or not to add points back into the ensemble members' meshes. Both of these steps have the potential to disrupt the ensemble statistics and need to be tailored to the model and meshing schemes. Another important consideration is that of mesh coarseness. If a mesh is so coarse that components being compared are not statistically correlated, errors in the EnKF may result and should be considered when designing a suitable EnKF scheme. We will refer to such a suitable scheme as an Adaptive Moving Mesh Ensemble Kalman Filter (AMMEnKF).

One avenue toward an AMMEnKF involves the use of a reference mesh to which each ensemble member can be mapped and on which error statistics can be estimated. This was explored originally in Du *et al.* (2016) and Aydoğdu *et al.* (2019) in the cases of conservative and nonconservative meshes, respectively. In Aydoğdu *et al.* (2019), the use of a reference mesh was explored in 1-d, where the reference mesh itself is chosen based on the properties of the mesh adaptation scheme. In particular, two meshes were explored. The first is a high-resolution (HR) mesh defined by the node proximity tolerance, δ_1, which ensures *at most* one point from each ensemble member can be in any given interval of the partitioned domain. The second is a low-resolution mesh (LR) defined by the node separation tolerance, δ_2, which ensures each ensemble member has

at least one point in any given interval of the partitioned domain. In both cases, each ensemble member is mapped to the reference mesh before error statistics are calculated, and then mapped back to its original mesh after the physical velocity values are updated in the analysis step. The mesh locations were not updated during analysis.

However, the node positions themselves are driven by the physical flow and as such can be considered time-dependent state variables. In this work, we consider updating the node locations by making use of the HR partitioning of the interval domain for the same models considered in Aydoğdu *et al.* (2019). The key difference between previous and current work is that we now augment our state vector with the node locations and update them in the analysis step. We are interested in exploring the use of the augmented state vector to leverage extra statistical information implied by the different meshes among the ensemble members. This is because, in this case, the mesh is connected to the physics and cross-covariances between the physical values and the node locations say something about the system and how statistically correlated component matched values are. Previous work for a conservative moving mesh was carried out by Bonan *et al.* (2017), who also augment their state vector, but with the mesh dimension being conserved to avoid the need for the dimension-matching step.

We will, when needed, describe the methods in Aydoğdu *et al.* (2019), so that the reader may understand the relevant differences. In particular, we focus on the HR method and refer to the augmented state vector as the High Resolution Augmented (HRA) method. In both cases, HR and HRA, the analysis update is preceded and followed by two additional steps: (a) *dimension matching*, when the individual ensemble members (each on its own mesh) are projected on to the uniform, fixed-in-time, reference mesh (HR), or component-paired using the subintervals defined by the reference mesh (HRA), and (b) *dimension return*, when the ensemble members are each given a mesh after their physical values (for HR) or their physical values and node locations (for HRA) have been updated. The full AMMEnKF procedure is detailed in the following subsection for both HR and HRA.

3.1 | Dimension matching

3.1.1 | HR scheme

In order to avoid the statistical consistency issues presented by having ensemble members with differing numbers of nodes at different locations, one can map each ensemble member to a reference mesh. We name the length of the physical domain as L, noting that this is 1 or 2π in our examples. The reference mesh is defined on the physical domain $[0, L)$ into M intervals of equal length $\Delta\gamma$,

$$[0, L) = L_1 \bigcup L_2 \bigcup \ldots \bigcup L_M, \qquad (10)$$

where $L_i = [\gamma_i, \gamma_{i+1})$. In this case, $\gamma_1 = 0$, $\gamma_i = (i-1)\Delta\gamma$ for each i. Further, $\gamma_M = L - \Delta\gamma$, as 0 and L are identified on the periodic domain. The points γ_i form the nodes of the reference grid.

The reference grid is chosen in one of two ways: to ensure that (a) each ensemble member has *at most* one point in each interval, $\Delta\gamma = \delta_1$, or (b) each ensemble member has *at least* one point in each interval, $\Delta\gamma = \delta_2$. The former is referred to as the high-resolution mesh (HR) and the latter as the low-resolution mesh (LR).

Here we focus on the HR mesh, since we partition our physical domain in the same way. The mapping from an ensemble member to the HR mesh will take the jth ensemble member's state vector $\mathbf{x}_j = (u_1 \ldots u_N, z_1 \ldots z_N)_j \in \mathbb{R}^{2N}$ to the vector

$$\mathbf{x}_j = \left(\widetilde{u}, \boldsymbol{\gamma}\right)_j^{\mathrm{T}} = (\widetilde{u}_1 \ldots \widetilde{u}_M, \gamma_1 \ldots \gamma_M)_j^{\mathrm{T}} \in \mathbb{R}^{2M} \text{ with } M \geq N. \qquad (11)$$

Here, $\widetilde{u}_i$ will be the physical value assigned to γ_i through the introduction of a shifted mesh where $L_i \to \widetilde{L}_i = [\gamma_i - \delta_1/2, \gamma_i + \delta_1/2)$ for $i = 2, \ldots M$. The first interval is taken to be $\widetilde{L}_1 = [L - \delta_1/2, L] \bigcup [0, \delta_1/2)$, since we identify 0 and L. If there is a $z_k \in \widetilde{L}_i$, then set $\widetilde{u}_i = u_k$. If there is no such z_k but $z_k \leq \gamma_i$, find k such that $z_k \leq \gamma_i \leq z_{k+1}$ and set

$$\widetilde{u}_i = \frac{u_k + u_{k+1}}{2}, \qquad (12)$$

if there is no such z_k, then set

$$\widetilde{u}_i = \frac{u_1 + u_N}{2}. \qquad (13)$$

This mapping is illustrated in the right branch of Figure 1. Once each ensemble member has been mapped to the fixed reference grid, the standard EnKF can be applied.

3.1.2 | HRA scheme

In the HRA setting, the reference mesh is also used to choose which nodes will be compared, but without changing their locations. We partition the domain $D = [0, L)$ into M subintervals (L_i) each of length δ_1, so that $D = \bigcup_i L_i$. Since δ_1 is the node proximity tolerance, we are guaranteed that each subinterval will have *at most* one point in it. We can then component-match nodes that fall in the same subintervals. If an ensemble member does not have
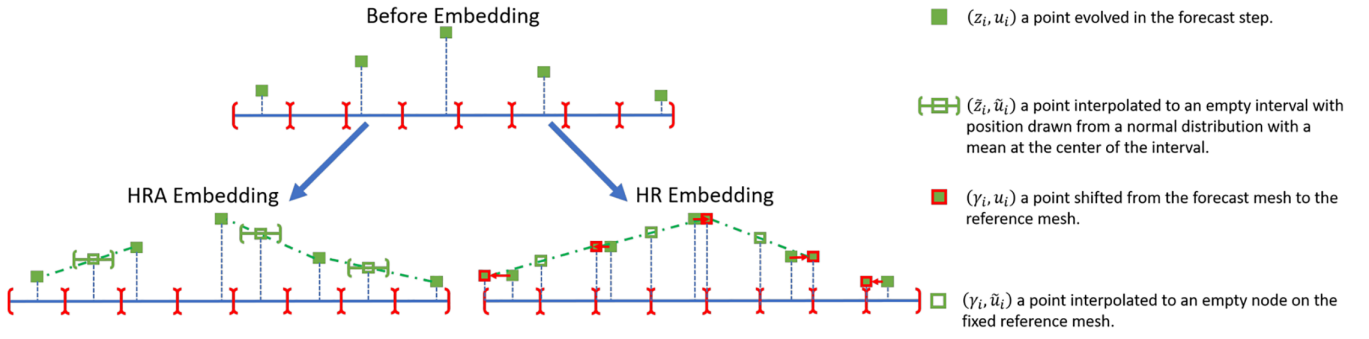
Before Embedding

HRA Embedding   **HR Embedding**

■ (z_i, u_i) a point evolved in the forecast step.

⊞ $(\tilde{z}_i, \tilde{u}_i)$ a point interpolated to an empty interval with position drawn from a normal distribution with a mean at the center of the interval.

■ (γ_i, u_i) a point shifted from the forecast mesh to the reference mesh.

□ $(\gamma_i, \tilde{u}_i)$ a point interpolated to an empty node on the fixed reference mesh.

FIGURE 1   Illustration of the two dimension-matching schemes. In the HR scheme, points are shifted to a fixed reference mesh with empty nodes interpolated to. In the HRA scheme, points are added to empty intervals and interpolated to

a point in a given subinterval, we will insert one, a *ghost point*, based on the nearest neighbours.

In this approach, the state vector of the jth ensemble member on the reference mesh will have the form

$$\mathbf{x}_j = (\mathbf{u}, \mathbf{z})_j^T = (u_1, u_2, \ldots, \tilde{u}_i, \ldots, u_M, z_1,$$
$$z_2, \ldots, \tilde{z}_i, \ldots, z_M)_j^T \in \mathbb{R}^{2M} \quad \text{with} \quad M \geq N, \quad (14)$$

where u_i or $\tilde{u}_i$ would be the value of the velocity in the ith subinterval of the reference mesh. A value with no tilde means that the ensemble member had a point in that interval, while a tilde implies the member did not have a point in the ith interval and one was inserted, with a physical value interpolated to that location. The location of an interpolated point is drawn from the Gaussian distribution, $\mathcal{N}((\gamma_i + \gamma_{i+1})/2, \delta_1/2)$, with a check that the point drawn actually resides in the interval L_i; if not, we draw again until it does. This is illustrated in the left branch of Figure 1. The choice of randomly sampling the node location is done to avoid biasing node locations in intervals that are empty amongst a large proportion of the ensemble members, which can happen, since the nodes are driven by the flow, in areas of divergent flow. It is possible that we end up having an invalid mesh in this process. Nevertheless, we do not enforce validity at this step, as there will be many cases where no location in an empty interval can be chosen, for which there is not a point within a distance of δ_1. This is because the intervals themselves are of size δ_1; we do, however, return to a valid mesh after the update step.

The physical value assigned to a ghost point $\tilde{z}_i$ is calculated by linear interpolation as

$$\tilde{u}_i = \frac{b}{a+b}u_l + \frac{a}{a+b}u_r, \qquad a = \tilde{z}_i - z_l, b = z_r - \tilde{z}_i, \quad (15)$$

where (z_l, u_l) and (z_r, u_r) are the closest nodes to $\tilde{z}_i$ to the left and right respectively and u_l, u_r the corresponding physical values at those nodes. This is done from left to right, which does allow for the possibility that a nearest left neighbour might have been a ghost point. This is, however, excluded in our current experimental case where $\delta_2 = 2\delta_1$, which guarantees that each empty interval will be surrounded by a nonempty interval to its left and right.

3.2 | Observation operator

3.2.1 | HR scheme

For the HR method, the observation operator applied to the jth ensemble member takes the form

$$h(\mathbf{x}_j) = \tilde{u}_i + \frac{z_k^o - \gamma_i}{\gamma_{i+1} - \gamma_i}(\tilde{u}_{i+1} - \tilde{u}_i), \quad (16)$$

where z_k^o is the observation location with $\gamma_i \leq z_k^o \leq \gamma_{i+1}$.

3.2.2 | HRA scheme

In a similar way, we define the observation operator for the HRA method as

$$h(\mathbf{x}_j) = u_i + \frac{z_k^0 - z_i}{z_{i+1} - z_i}(u_{i+1} - u_i), \quad (17)$$

where either z_i, z_{i+1}, u_i or u_{i+1} could have a tilde if they were inserted, due to the ensemble member having no value in the ith interval (see Section 3.1.2).

This form of the observation operator means that we are not considering the location of the observation in the update, just the physical value. This is done since most geophysical measurements will not relate directly to a node position, since the nodes are not physical objects. However, the physics does fundamentally drive node motion and the covariances between physical values and node locations are nonzero in the error covariance matrix. This property is

an integral part of our approach which is described below and illustrated later (in Figure 9).

3.3 | Analysis using the EnKF

Once the dimensions of the state vectors of each ensemble member have been matched, the EnKF can be applied in the usual way. As in Aydoğdu et $al.$ (2019), we will use the stochastic version of the EnKF (Evensen, 2009). This choice does not influence how we design our schemes for AMMs. Our methods will apply to deterministic versions of the EnKFs (Asch et $al.$, 2016) as well.

We define the forecast ensemble matrix $\mathbf{E}^f$ as

$$\mathbf{E}^f = \left[\mathbf{x}_1^f, \ldots, \mathbf{x}_{N^e}^f\right] \in \mathbb{R}^{2M \times N^e}, \qquad (18)$$

where the forecast state vectors $\mathbf{x}_j^f$ take the form in Equation 11 for the fixed reference mesh case and in Equation 14 for the augmented case, where we also update node locations. In Equation 18, M is the number of subintervals, L_i, that partition the domain D into subintervals of size δ_1, and N^e is the number of ensemble members. The vectors $\mathbf{x}_j^f$ are the dimension-matched state vectors and are taken to be the columns of $\mathbf{E}^f$.

The forecast anomaly matrix $\mathbf{X}^f$ takes the form

$$\mathbf{X}^f = \frac{1}{\sqrt{N^e - 1}} \left[\mathbf{x}_1^f - \overline{\mathbf{x}}^f, \ldots, \mathbf{x}_{N^e}^f - \overline{\mathbf{x}}^f\right], \qquad (19)$$

where $\overline{\mathbf{x}}^f$ is the forecast ensemble mean, defined as

$$\overline{\mathbf{x}}^f = \frac{1}{N^e} \sum_{j=1}^{N^e} \mathbf{x}_j^f. \qquad (20)$$

In the stochastic EnKF, the observations are treated as random variables, so that each ensemble member is compared with a slightly differently perturbed realisation of the observation vector (Burgers et $al.$, 1998). That is, given an observation vector $\mathbf{y}$, we generate N^e observations according to

$$\mathbf{y}_j = \mathbf{y} + \epsilon_j, \qquad 1 \leq j \leq N^e, \qquad \epsilon_j \sim \mathcal{N}(\mathbf{0}, \mathbf{R}), \qquad (21)$$

where $\mathbf{R}$ is the covariance of the assumed zero-mean, white-in-time observation noise ϵ. We can then calculate the normalised anomaly ensemble of observations,

$$\mathbf{Y}_o = \frac{1}{\sqrt{N^e - 1}} \left[\mathbf{y}_1 - \mathbf{y}, \ldots, \mathbf{y}_{N^e} - \mathbf{y}\right] \qquad (22)$$

$$= \frac{1}{\sqrt{N^e - 1}} \left[\epsilon_1, \ldots \epsilon_{N^e}\right], \qquad (23)$$

which in turn defines the ensemble-based observation-error covariance matrix,

$$\mathbf{R}^e = \mathbf{Y}_o(\mathbf{Y}_o)^T. \qquad (24)$$

We then define the observed ensemble-anomaly matrix using our observation operator h as

$$\mathbf{Y} = h(\mathbf{E}^f) - h(\overline{\mathbf{E}}^f), \qquad (25)$$

where the operator h is applied at each column of the matrix $\mathbf{E}^f$. The Kalman gain matrix, $\mathbf{K}$, is then calculated as

$$\mathbf{K} = \mathbf{X}^f \mathbf{Y}^T \left[\frac{1}{N^e - 1} \mathbf{Y}\mathbf{Y}^T + \mathbf{R}^e\right]^{-1}, \qquad (26)$$

which is used, in the stochastic EnKF formulation, to update each ensemble member individually according to

$$\mathbf{x}_i^a = \mathbf{x}_i^f + \mathbf{K}\left[\mathbf{y}_i - h(\mathbf{x}_i^f)\right], \qquad 1 \leq i \leq N^e. \qquad (27)$$

With the HRA method, however, there is the possibility that an ensemble member will have an invalid mesh after the update step. For this reason, the remeshing algorithm is applied to each ensemble member after updating. The remeshing is also tasked with handling points that have moved out of the domain at the analysis update; although not common, it can happen.

We also make use of covariance multiplicative scalar inflation (see Anderson and Anderson, 1999), in which the ensemble forecast anomaly matrix is inflated as

$$\mathbf{X}^f \rightarrow \alpha \mathbf{X}^f, \qquad (28)$$

with $\alpha \geq 1$, before $\mathbf{X}^f$ is used in the analysis update. This parameter is one that can be tuned through numerical experimentation, although approaches exist to make this task automatic and adaptive along the experiments (see, for example, Raanes et $al.$, 2019, and references therein). After updating each ensemble member, the mean of each updated member can be used to obtain an "optimal" estimate of the physical state of the system. A significant difference between the analysis means coming from the HR and HRA methods is the meshes associated with each. The HR method analysis mean is on the equally spaced reference mesh, while the analysis mean mesh for the HRA scheme is calculated by averaging the positions of the nodes in each of the subintervals defined by the reference mesh, which is then passed to the remeshing algorithm to ensure a valid mesh.

3.4 | Dimension return

After the update is complete, each ensemble analysis vector has its dimension returned to its pre-analysis value. For the AMMEnKF-HR scheme, this is a needed (mandatory) step, as the structure of the adaptive mesh is removed during the update step and some kind of map back to the previous mesh state before the next forecast is necessary. For reasons explained below, we include a dimension return step in the AMMEnKF-HRA scheme too, although it is not absolutely necessary in this case. In the AMMEnKF-HRA scheme, the mesh itself is updated and the remeshing scheme is applied to ensure a valid mesh and enforce the periodic boundary conditions if needed. When updating the mesh locations, there is the possibility the boundary conditions are violated, as with any scheme. The remeshing scheme, when present, naturally incorporates the enforcement of the boundary conditions: its application is likely to be sufficient for this purpose.

3.4.1 | HR scheme

Following Aydoğdu *et al.* (2019), in the HR case a backward map is used to return the updated ensemble members to their original meshes before forecasting again. In the forward mapping step, the mapping indices associating the nodes in the adaptive moving mesh with nodes in the reference mesh are stored in an array. These are the indices resulting from the projections on to the HR reference mesh. This allows us to map the updated physical values $\tilde{\mathbf{u}}^a$ back to the mesh that the ensemble member came with into the update step. In other words, the values updated at γ_i are shifted back to their previous node locations. This shift can have the effect of introducing some noise in each ensemble member as physical values determined at one location are moved to another. From there,

the forecast is run until the next assimilation time step. The entire process is illustrated in the right branch of Figure 2.

3.4.2 | HRA scheme

In the HRA case, after the update and remeshing, nodes that are in intervals which were previously unoccupied by a point before the update step are deleted for each ensemble member using the stored indices as in the HR case. This is not specifically necessary to the scheme and performance with and without this step is essentially equivalent. However, we include this step in our analysis, as there may be some applications where keeping the dimension of the ensemble members low is desirable during the forecast step. The process is diagrammed in left branch of Figure 2.

A beneficial by-product of the mapping to and from the reference mesh in the HR scheme is that it induces additional variability among the physical values. This occurs when a value at one location is moved to another in the shift to and from the reference mesh. The net effect is that the ensemble spread stays reasonably large, leading to the healthy functioning of the EnKF. This is not the case in the HRA method, given that physical values and their locations are updated together. As a result, the spread of the ensemble when using the HRA scheme tends to be smaller than in the HR case. This behaviour is shown in Figure 3.

While little spread could also be reflecting the desired analysis convergence to the truth, in practice it is a dangerous situation, as it often induces the filter to underestimate the actual error, leading to filter divergence. We counteract this effect by adding white noise to the physical values, but leave the node locations unaltered. We shall refer to this process as jitter and it can be applied to each ensemble member after the update step. For a given ensemble member analysis vector $\mathbf{x}_j^a$, the jitter is applied to its first M
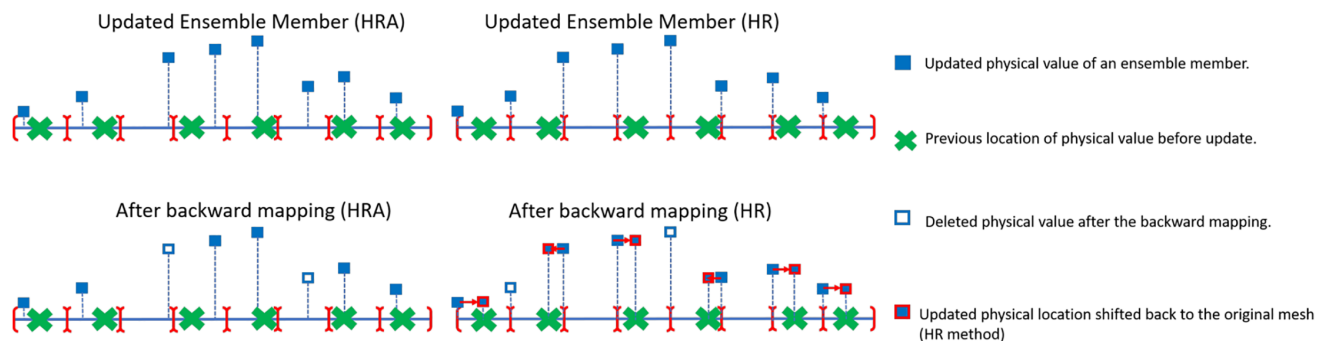

FIGURE 2 The dimension return steps. In the HRA scheme, points that occupy previously empty intervals are simply deleted. In the HR scheme, points that came in from the forecast step are shifted back to their original mesh locations and points inserted at the dimension-matching step are deleted
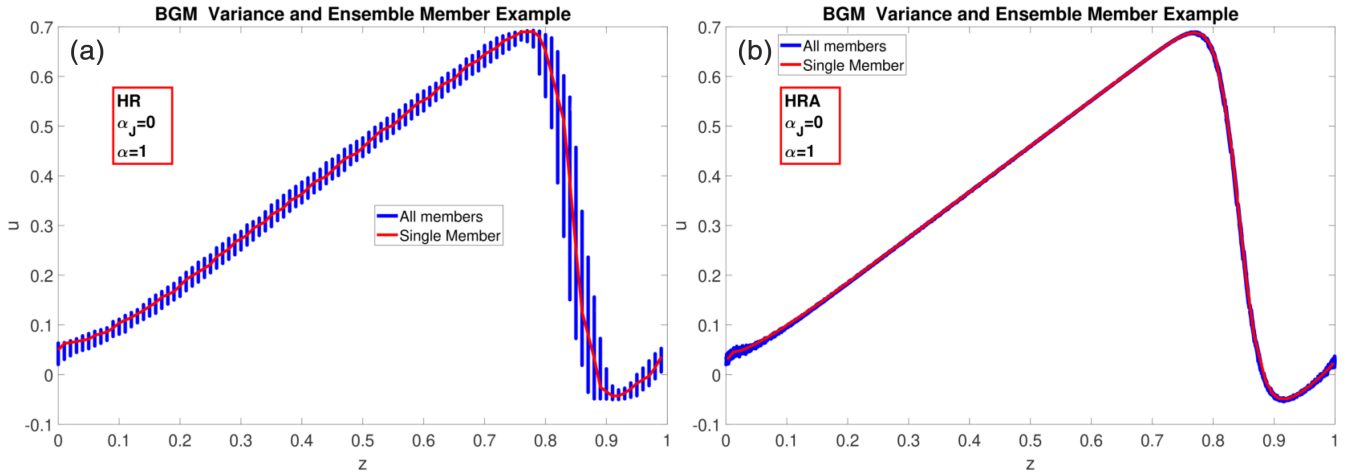
FIGURE 3 Examples of the spread in the forecast ensembles and an example of an forecast ensemble member for (a) HR and (b) HRA schemes. The forward and backward mapping of the HR scheme induces some inherent jitter, increasing the spread. For the HRA case, this does not happen and the ensemble members can collapse quickly. Also notable is the reduced smoothness in the ensemble member shown for the HR case, caused by the mapping procedure. For these experiments, $\alpha_j = 0$, $\alpha = 1$, $\sigma_o = 0.01$, $N^e = 30$, and $I_m = 70$
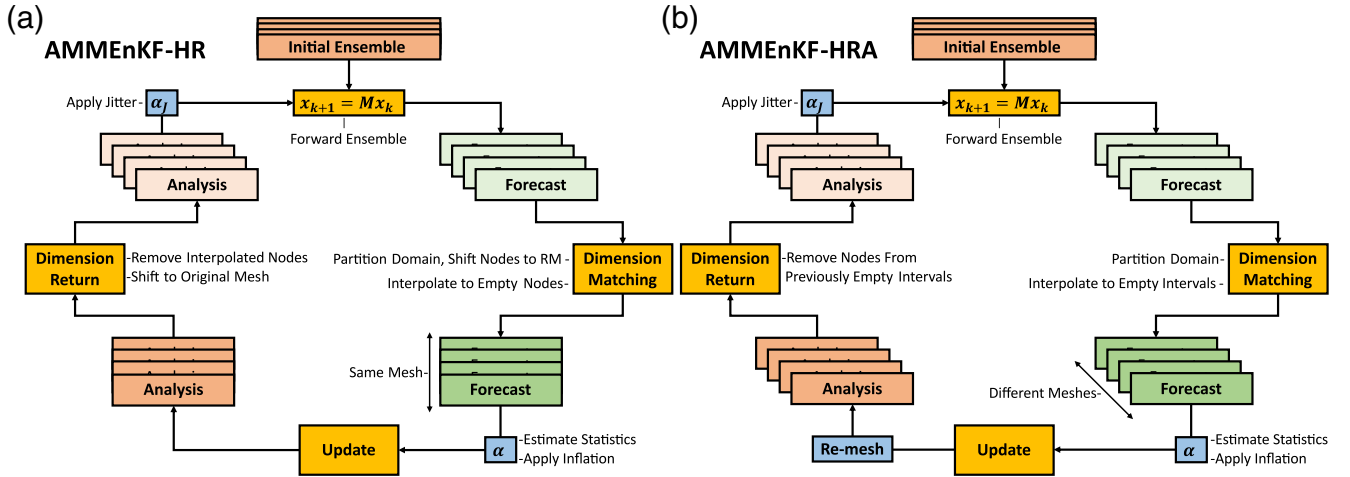
FIGURE 4 Illustration of (a) AMMEnKF-HR and (b) AMMEnKF-HRA schemes. The details of the dimension-matching, update, and dimension return, as well as the addition of α and α_J, can be found in Section 3

components (that is, to the physical values), according to

$$\mathbf{u}_j^a = \mathbf{u}_j^a + \mathcal{N}(\mathbf{0}, \boldsymbol{\sigma}_J)^T \quad \text{with} \quad \boldsymbol{\sigma}_J, \mathbf{0} \in \mathbb{R}^M \quad \text{and}$$

$$\boldsymbol{\sigma}_J = \alpha_J \max_{u_i, u_k \in \mathbf{u}^a} |u_i - u_k|, \tag{29}$$

where the jth ensemble member index is omitted in part of Equation 29 to simplify the notation. The scalar parameter α_J regulates the amount of jitter, and the suffix J in α_J and $\boldsymbol{\sigma}_J$ stands for jitter. We take $0 \leq \alpha_J \leq 1$, so that we add a percentage of the maximum difference between the physical values of the ensemble members. By having α_J dependent on the analysis field, the jitter is adaptive and is similar to an adaptive form of additive inflation (see Anderson and

Anderson, 1999). For consistency across comparisons, we also experimented by applying jitter in the HR method and found improvements in time-averaged root-mean-squared error (RMSE) values for both schemes. The HR and HRA algorithms are diagrammed in Figure 4.

4 | RESULTS AND DISCUSSION

In this section we present the results of numerical experiments designed to measure the performance of the two schemes with different parameter settings. We experiment with the models we named BGM and KSM, which were described in Section 2.2. For BGM, we run for a short

	ν	δ_1, δ_2	N	T	Δt	N_{obs}
BGM	0.008	0.01, 0.02	100	2	0.05	10
KSM	0.027	$0.02\pi, 0.04\pi$	100	5	0.05	20

TABLE 1 The model parameter settings used in each of the DA experiments for the models described in Section 2.2

Note: Here ν is the viscosity, δ_1 and δ_2 the node proximity and distance tolerances, N the size of the reference mesh, T the duration of the experiment, Δt the integration time step, and N_{obs} the number of observations.

time from $t = 0$ to $t = 2$, because of the rapid dissipation in fluid velocity with our chosen viscosity parameter. The time-averaged statistics in the DA experiments are taken for $t > 1$. The ensemble members are initialised by perturbing the initial condition of the nature run. For KSM, an initial spin-up until $T = 20$ is performed and the system's state at $t = T$ is used as the initial condition for the DA experiments that follow. With the initial condition provided by spin-up, the model is then run until $t = T + 5$ and observations are collected for $T < t < T + 5$. For both models, observations are taken at fixed times and on fixed evenly spaced intervals equally dividing the spatial domain. In addition, random white noise with standard deviation σ_o is added to each observation and we vary these values in our experiments described below. The dimension of the reference mesh for both schemes is $M = 100$, however the dimension of the state vector in the AMMEnKF-HRA scheme is twice that of the AMMEnKF-HR scheme, since it has been augmented with the node locations. The parameters used for BGM and KSM in these experiments are summarised in Table 1.

4.1 | Comparison between AMMEnKF-HR and AMMEnKF-HRA

We compare the performance of the AMMEnKF-HR introduced by Aydoğdu *et al.* (2019) (and recalled in Section 3), with that of the novel augmented formulation AMMEnKF-HRA, also presented in Section 3. We will use two metrics to evaluate the performance. Together with the more standard RMSE of the analysis mean, we consider the time-averaged RMSE for the first spatial derivative of the analysis mean, $\partial_x \tilde{\mathbf{u}}^a$. The gradient of the analysis field allows us to assess how well each of the methodologies preserves derivative information. This is relevant for three reasons: the first is to evaluate if and by how much applying jitter to the analysed ensemble members distorts their curve smoothness. The second is that the mapping scheme in the HR case can create artificially sharp changes in function values. This will happen when mapping to the reference mesh and when the analysis vector is mapped back to the original ensemble member mesh, if the original node location is sufficiently far away

from a reference mesh location. These sharp changes over the domain, due to the jitter, HR mapping, or both, can disrupt local rates of change, with the risk of violating conservation rules, such as incompressibility ($\nabla \cdot \mathbf{u} = 0$), for example. While we make no direct study of conservation laws in this work, we do evaluate the fidelity of the first derivative after the update step for each of these methods as a proxy for the potential violation of conservation laws in more realistic scenarios. The third reason is to gain a sense of how the methods affect the adaptive mesh. We think of a good mesh as being one that distributes available nodes in such a way that the derivatives needed to step the model forward in time are well represented by the mesh. We can see from Figure 5 that the first spatial gradients are comparable between the two methods for BGM, but we see a clear improvement in the spatial gradients from the HRA method with KSM in Figure 6. In these results, the time-averaged RMSEs for the derivatives are obtained using the inflation parameters (α, α_J) that optimise the time-averaged RMSE of the solution analysis mean. Depending on the situation, one may run similar experiments and choose a jitter and inflation that best preserve the first derivative if high fidelity of it is needed. For these models, there is not a great deal of difference in time-averaged RMSEs when using parameters that optimise the RMSE for the first derivative instead of the solution itself.

The comparison is carried out over ranges of the three key experimental parameters: the ensemble size, N^e, the initial mesh size, I_m, and the observation error, σ_o. We study the performance of the methods by running experiments with two of them kept fixed while varying the other. For each parameter setting, the optimal jitter and inflation for each scheme are determined by running tuning experiments that identify the pair of values giving the lowest time-averaged RMSE. In this way, we will compare the best possible configuration of each scheme. The values used in the experiments are given in Table 2. Results are shown in Figures 5 and 6 for BGM and KSM, respectively.

First note that Figures 5 and 6 immediately reveal how the ensemble spread in the HR scheme is typically much larger than that of the HRA scheme, even when performance is comparable, such as in the initial mesh size
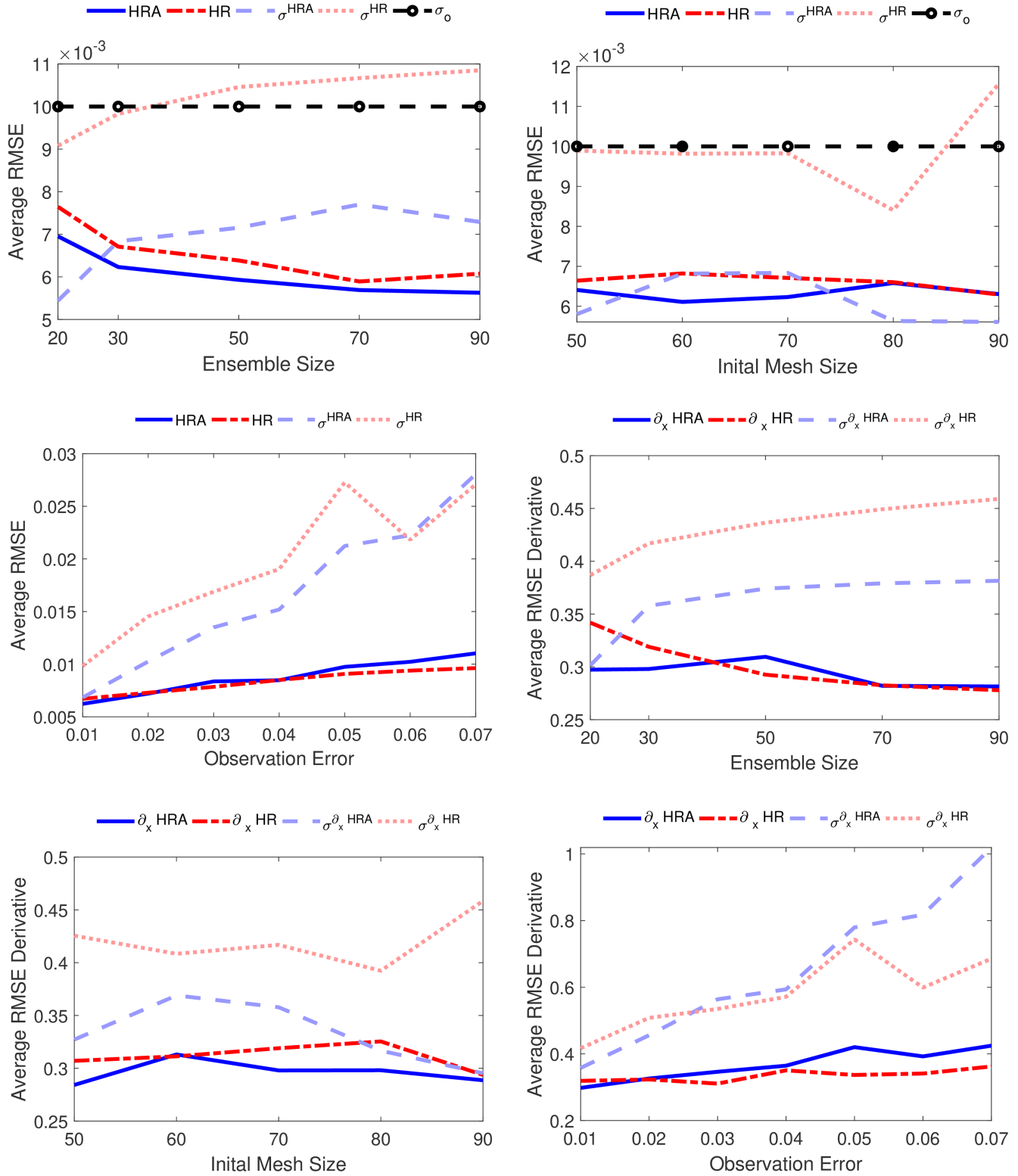
FIGURE 5 Results of the tuning experiments for the HR and HRA schemes with BGM. For each parameter shown along the *x*-axis, the optimal jitter and inflation are used to obtain a time-averaged RMSE for the analysis (top panels). The time-averaged RMSE values for the spatial derivatives (bottom panels) correspond to the parameters that optimise analysis RMSE (see text for details)
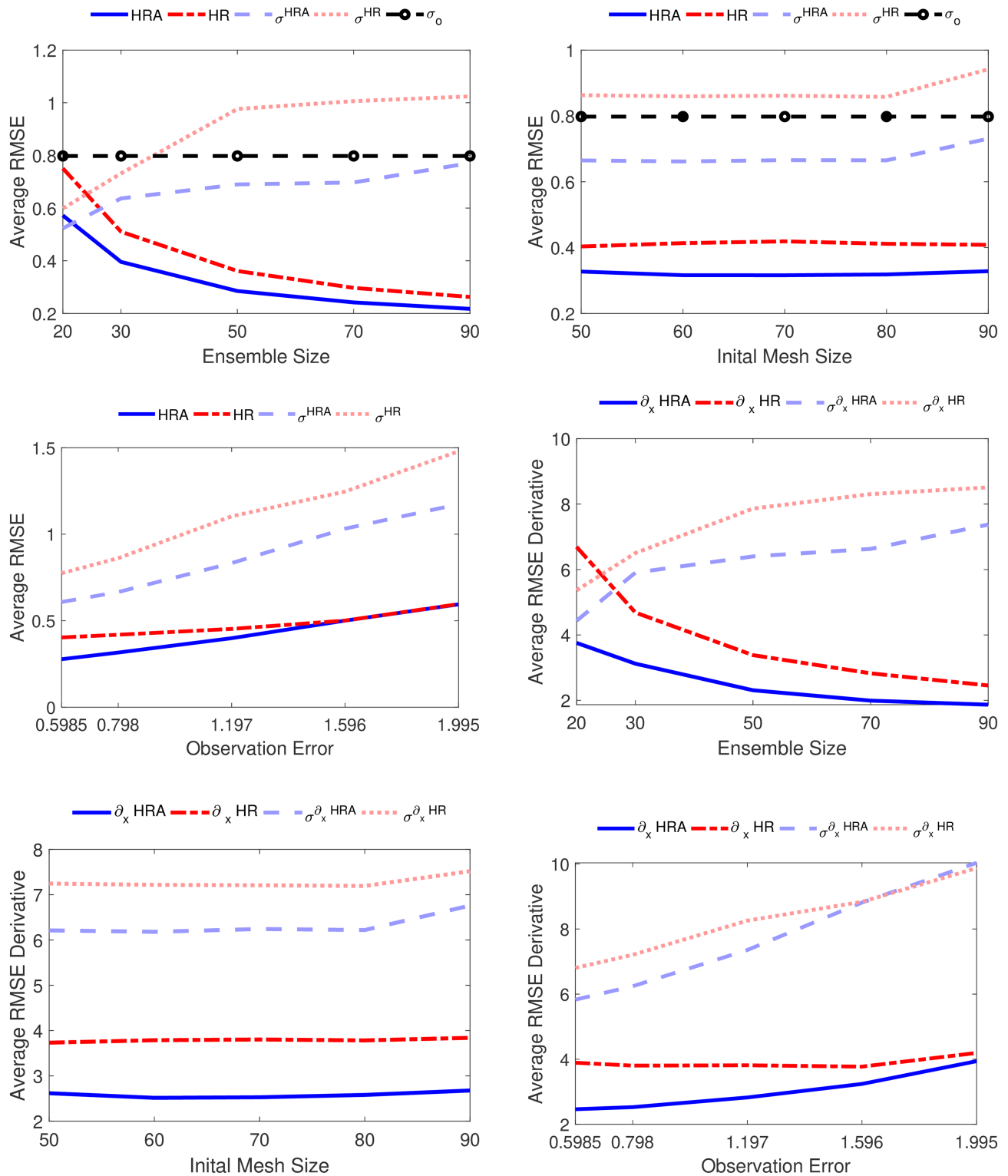
FIGURE 6 Same as Figure 5 but for KSM

TABLE 2   The settings used for the three sensitivity experiments for BGM and KSM

Experiment type	BGM			KSM		
	N^e	I_m	σ_o	N^e	I_m	σ_o
Varying N^e	(20–90)	70	0.01	(20–90)	70	0.798
Varying I_m	30	(50–90)	0.01	40	(50–90)	0.798
Varying σ_0	30	70	(0.01–0.07)	40	70	(0.60–2.0)

Note: For all three experiment types, the ranges of α and α_J optimised over remained the same. The range of α_J differed between the model types, with a range of (0–0.1) for BGM and (0–0.5) for KSM. The range of α was (0–1.6) for both models.

experiments. This is due to the inherent stochasticity of the HR scheme, discussed in Section 3.4.

By looking at the RMSE of the analysis mean, it is evident that the HRA scheme tends to outperform the HR scheme in general, and particularly for smaller ensemble sizes. This is due to the extra information carried in the cross-covariances between the physical values and the node locations. The RMSE of the spatial derivatives is also generally lower in the HRA scheme, except at ensemble size 50 for the BGM case. It is worth reiterating that here we are using parameters optimised for the solution itself and not the first derivatives. We will discuss this behaviour more extensively later in this section, together with other metrics used to understand this particular issue. Figures 5 and 6 also highlight that only marginal improvements in time-averaged RMSE are obtained after an ensemble size of 30 for BGM and 50 for KSM (see panels in the middle column). This kind of behaviour can be observed when the ensemble size is larger than or equal to the dimension of the unstable, neutral subspace of the dynamics (Bocquet and Carrassi, 2017). For both BGM and KSM, we do not see much dependence on the initial mesh size, but HRA performs comparably to or better than the HR scheme in the BGM case and outperforms HR in the KSM case.

We also make a comparison of the performance of each scheme with respect to increasing observation error (last column in Figures 5 and 6). For the BGM case, both schemes perform comparably, but we see better results from the HRA scheme in the KSM case, particularly with regard to the first spatial derivative of the solution. We also remark that the clearer trends in the KSM experiments are likely the result of a longer time average of the RMSE available, as the BGM solution damps quickly, limiting the experimental time window. When the observation error is large enough, both models perform about the same, suggesting that one might choose to accept the extra computational cost of the HRA scheme when the observations are good enough to warrant doing so.

In Figures 7 and 8, for KSM and BGM respectively, we show examples of the time-averaged RMSE surfaces from the experiments described above as a function of inflation and jitter, with the optimal pair of values marked by

a star. The difference in the smoothness of the contour plots arises from the aforementioned fact that we run KSM for a longer time than BGM, which dissipates quickly due to the chosen viscosity term. The longer run provides a larger sample of RMSE values to average over, producing a smoother surface.

The need for some jitter in the HRA method (bottom panels) is highlighted by the fact that the time-averaged RMSE error is higher near the x-axis ($\alpha_J = 0$) for both models, but particularly with BGM. While this is also the case for the HR scheme with BGM, the effect is less pronounced. For the HR method with KSM, there is a region with $\alpha_J = 0$ in which the time-averaged RMSE remains close to the one obtained using optimal jitter and inflation: this is likely achievable due to the chaos in KSM naturally increasing the spread.

4.2 | Error covariance structures

We study here the ensemble-based forecast-error covariance matrices, $\mathbf{X}^f\mathbf{X}^{f^T}$, with $\mathbf{X}^f$ defined in Equation 18. The structure of the matrices is shown in Figures 9 and 10. The size of $\mathbf{X}^f\mathbf{X}^{f^T}$ for the HRA method in this case is 200×200, while that for the HR case is 100×100. With the HR method, we only have covariances between physical values themselves, in contrast to HRA, where we have covariances between physical values, physical values and node locations, and the node locations themselves. In Figure 9, we show the forecast covariances between the physical values and node locations for the HRA method with both BGM and KSM just before the 10th and 20th assimilation steps, respectively. These error covariance matrices correspond to no jitter or inflation, in an effort to understand the intrinsic differences between the methods. Also shown are the gradient of the corresponding forecast mean and the associated covariance between the physical values u_i and their node location z_i, that is, the diagonal of the matrix. This is done to highlight that the largest covariances occur at sharp gradients and have the same sign. This is natural, since a negative gradient would imply a negative correlation between a physical value and its independent variable,
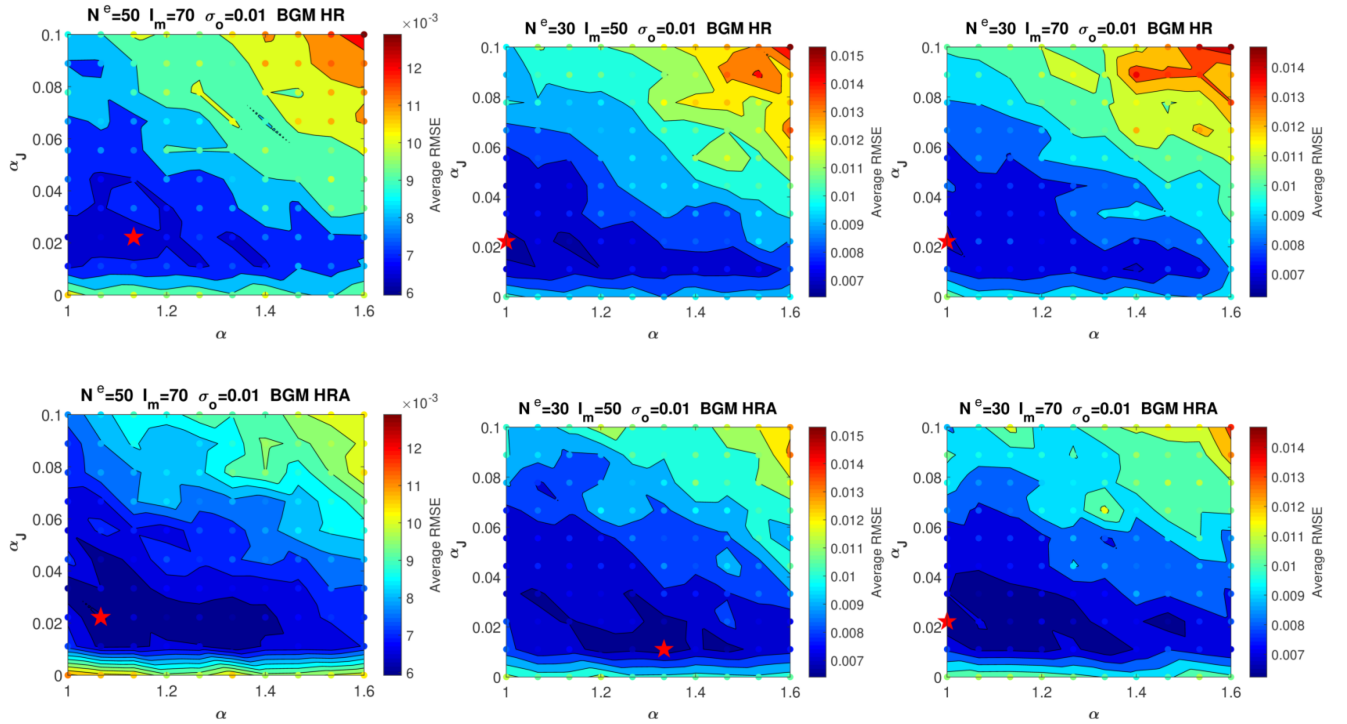
FIGURE 7 Contour plots for the jitter and inflation calibrations for BGM for (top) HR and (bottom) HRA schemes for the experimental parameters. The points in the plot represent the sample points used, while the star represents the jitter and inflation with the lowest time-averaged RMSE. Of particular note is the fact that the HRA scheme has its valley of low RMSE well away from the x-axis, implying that some jitter is beneficial, while this is not as strong a feature with the HR method, due to the inherent stochasticity added during the dimension-matching step

likewise for a positive gradient. In fact, the shape of the diagonal closely matches that of the gradient, demonstrating that including the node locations in the state vector encodes a deeper level of information into the Kalmain gain matrix.

In Figure 10 we show the covariances for the physical values for HR and HRA (i.e., the top left 100×100 block), as well as the HRA covariances of the node locations (i.e., the bottom right 100×100 block), for both BGM and KSM. Typically, the HR covariances are higher in magnitude than that of the HRA scheme, this is because the ensemble members are compared on the same mesh, in conjunction with the effect of the intrinsic stochasticity caused by the mapping to and from the reference mesh. The shock is immediately identifiable in the physical value covariances for BGM as a bright spot near the sharp gradient (cf. Figure 9). The sharp gradients of KSM are also apparent in the physical error covariances. For KSM, there is a strong, albeit regular structure in the matrix for the HR method, resulting from the fixed mesh with some long-distance cross-correlations. Those long-distance correlations are greatly reduced with the HRA method. The correlations between the node locations themselves in the HRA scheme (rightmost panels) are very small, due to the fact that they are not very far from each other, since the

intervals themselves are very small. This means that the extra contribution to the innovation in the HRA scheme comes mainly from the correlations between the physical values and the node locations, as opposed to the node locations themselves. This is indeed desirable, since we need to inject new nodes in the embedding process and would prefer to avoid incidental biases.

4.3 | Ensemble member fidelity

As discussed earlier, the addition of jitter can disrupt the shape of the ensemble members while still improving the analysis mean. This may be problematic if the ensemble members are used to feed information to another model component.

As an example where ensemble member fidelity may be important, we consider the heterogeneous multiscale method (HMM), described for various applications in Weinan (2011). In a general setting, the HMM method connects a macroscale model with parameters dependent on microscale variables to a model of these micro variables in order to simulate a physical process. Typically, one has a macroscale model $F(U, D)$, where U is the physical macroscale variable and D the data needed in order
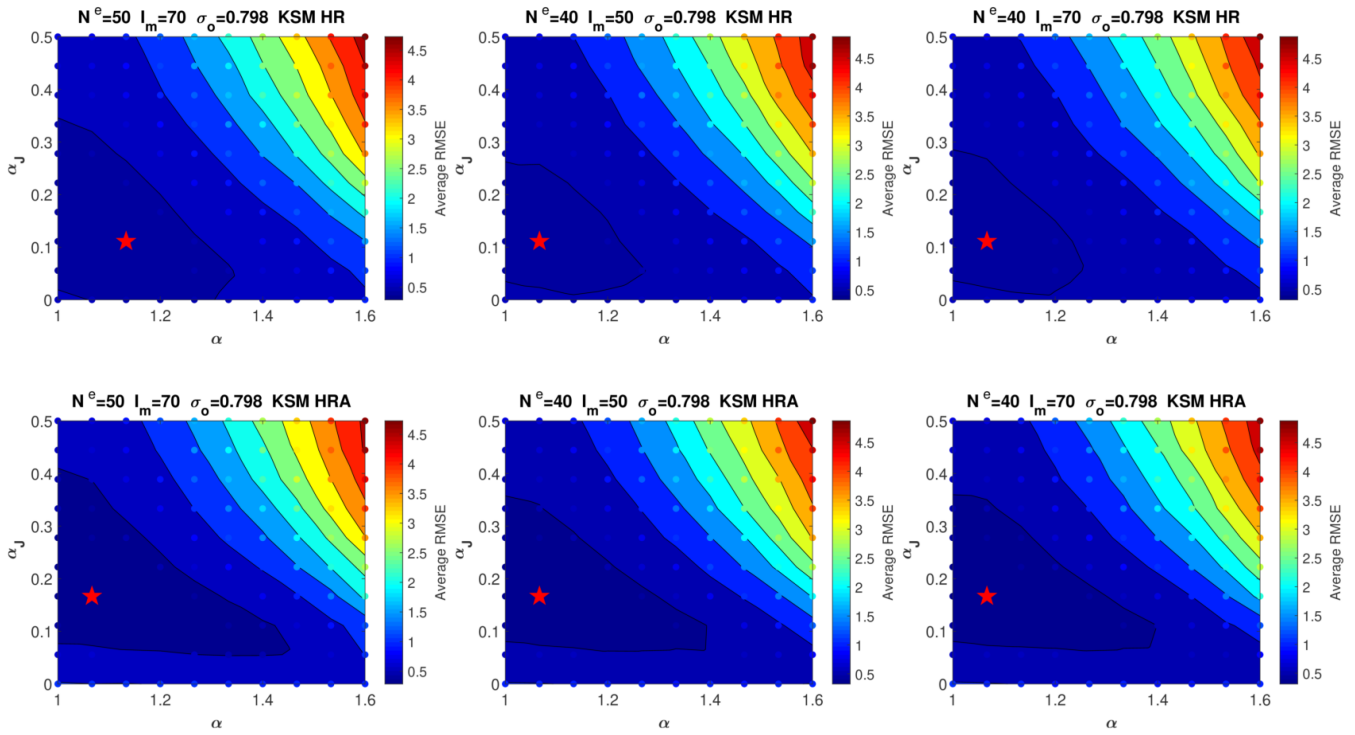
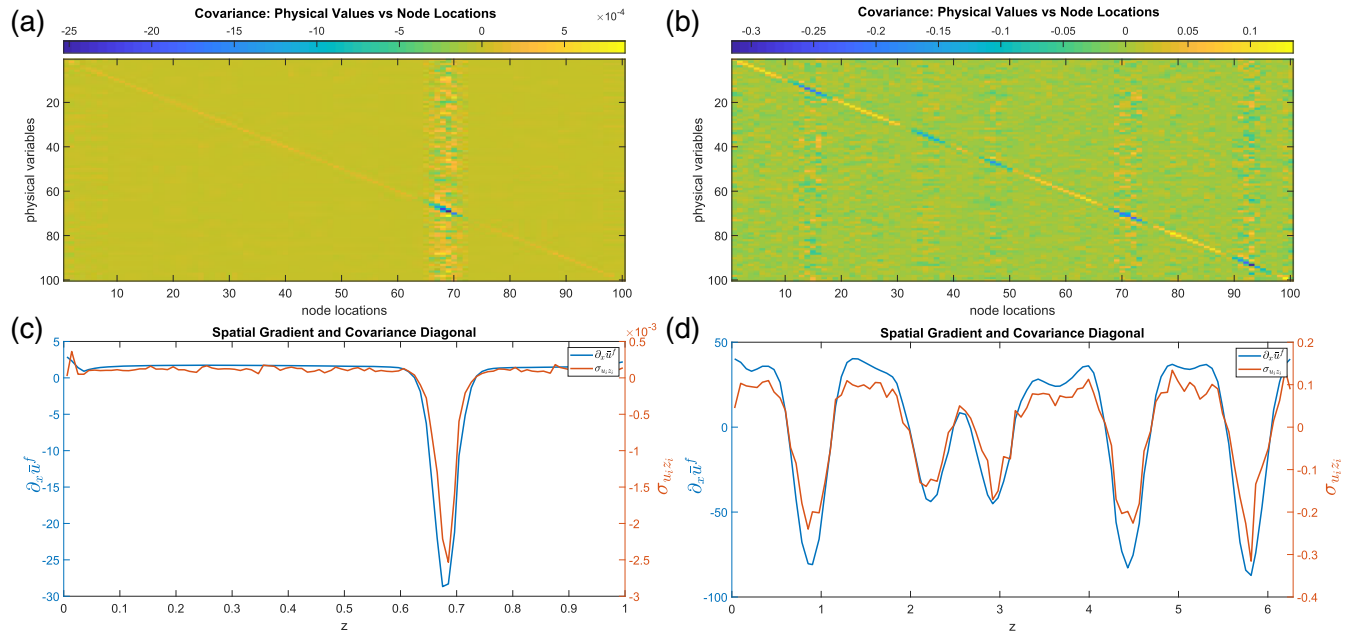**FIGURE 8**   Same as Figure 7 but for KSM

FIGURE 9   Top row: examples showing the forecast covariances between the physical values and the node locations in the HRA method for BGM and KSM, right before the (a) 10th and (b) 20th assimilation steps, respectively. Bottom row: the spatial gradient of the forecast mean and covariance between u_i and z_i corresponding to the covariance matrices above; this highlights the extra information encoded into the Kalman gain when using the HRA scheme

for the macroscale model to be complete, a stress tensor for example. Paired with the macroscale model, we have a microscale model $f(u, d) = 0$ and $d = d(U)$, where d represents the data needed to set up the microscale model

and is dependent on the macroscale state. Typically the HMM process proceeds as follows. (a) Given the current state of the macro variables, initialise the micro variables using the needed micro model data $d = d(U)$. (b) Evolve
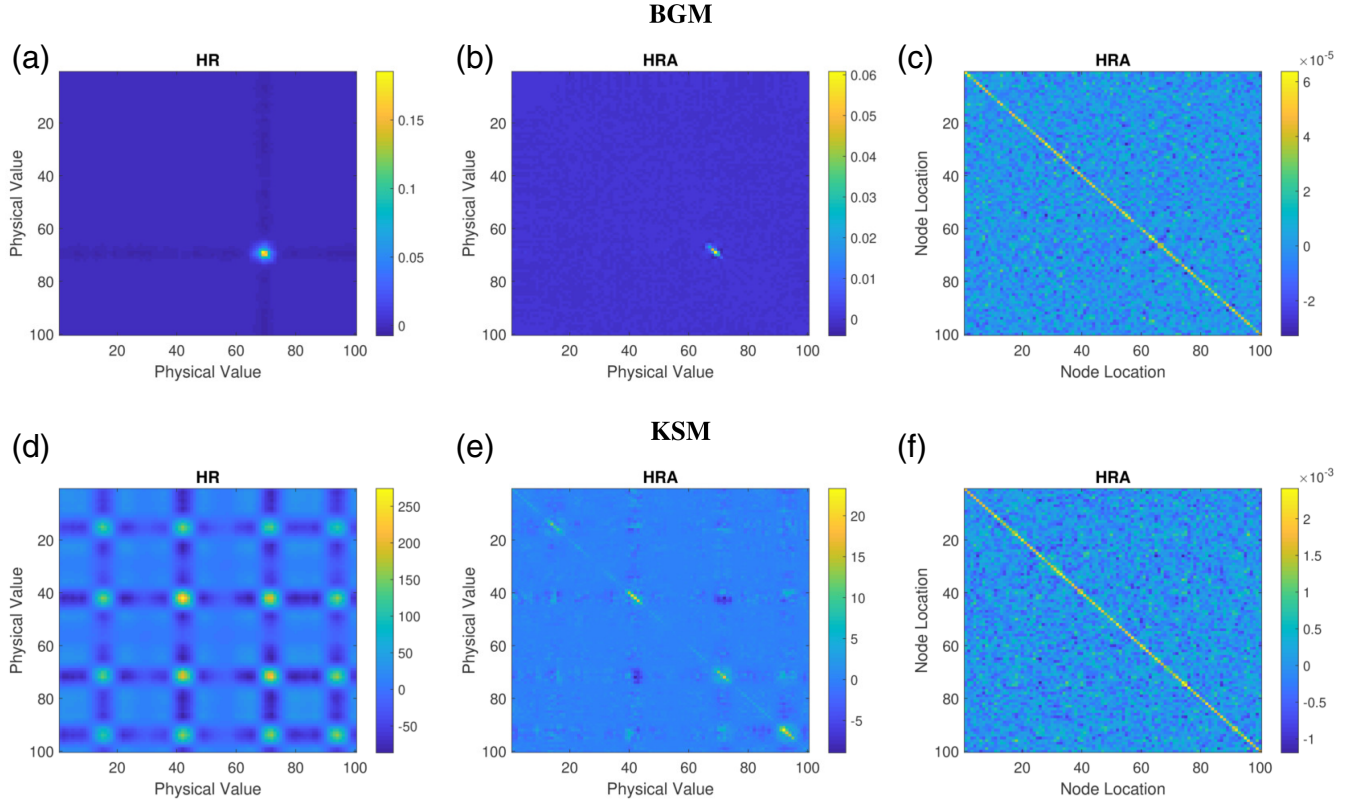
BGM

KSM

FIGURE 10 Examples of the error covariance matrix before the update step for the HR and HRA methods. (a–c) corresponds to BGM, while (d–f) corresponds to KSM. Each row shows $\sigma_{u_i u_j}$ (left and middle columns) for the HR and HRA schemes, respectively, while the right column shows $\sigma_{z_i z_j}$, which only exists for the HRA scheme

the micro variables for some micro model time steps. (c) Through the appropriate method, calculate D, needed for the macro model. (d) Evolve the macro variables using the macro solver. If an ensemble member is a solution of either the macro or micro models, disruption in their fidelity will naturally cause a problem with steps (a) and (c) through an inaccurate calculation of d or D and would then likely propagate such errors in the evolution steps. An example of a system like this can be found in Cloud-Resolving Convection Parameterization (CRCP), see Grabowski (2001). There, a macro model solving inviscid moist equations is coupled with a micro model representing subgrid-scale cloud physics.

We saw in Figures 7 and 8 that there are regions of low analysis mean RMSE for relatively high values of α_J, suggesting that the mean is smoothing out the added noise from jitter. In Figure 11 we show examples of the analysis mean, truth, and a typical ensemble member of BGM for a fixed inflation and three values of α_J. The inflation chosen corresponds to the optimal value found for an ensemble size of 50. We show $\alpha_J = 0$, the optimal α_J (in terms of lowest time-averaged RMSE), and a larger α_J for which the ensemble mean still has low time-averaged RMSE. The figure clearly shows that applying jitter to

the ensemble members has the potential to disrupt them (see the waving profile of the displayed arbitrarily chosen ensemble member), especially if the scheme requires one to act on each ensemble member, as we do here with dimension-matching and return. Depending on the application, such as a model using the HMM framework, it may be better to sacrifice a small amount of analysis accuracy to preserve the fidelity of each ensemble member in terms of representing a valid solution to the underlying PDE. In other applications, that may not matter quite so much.

To quantify better the effect that adding jitter and inflation may have on the PDE fidelity of the ensemble members, we look at three different metrics: the average of the time-averaged variance of the difference between the ensemble members and the truth at each node (σ_{ens}), the kurtosis of the same difference (k_{ens}), and the average of the time-averaged RMSE errors of the ensemble members (RMSE$_{\text{ens}}$). If $\mathbf{d}_i^{\tau_j} = \left(\mathbf{u}_i^{\tau_j} - \mathbf{u}_T^{\tau_j} \right) \in \mathbb{R}^M$ is the difference between the ith ensemble member and the truth at assimilation time τ_j, then we can define these quantities as

$$\sigma_{\text{ens}} = \frac{1}{N^{\text{e}}} \sum_{i=0}^{N^{\text{e}}} \frac{1}{N_{\text{an}}} \sum_{\tau_1}^{\tau_n} \frac{1}{M} \sum_{k=0}^{M} \left(d_{i_k}^{\tau_j} - E\left[d_{i_k}^{\tau_j} \right] \right)^2, \qquad (30)$$
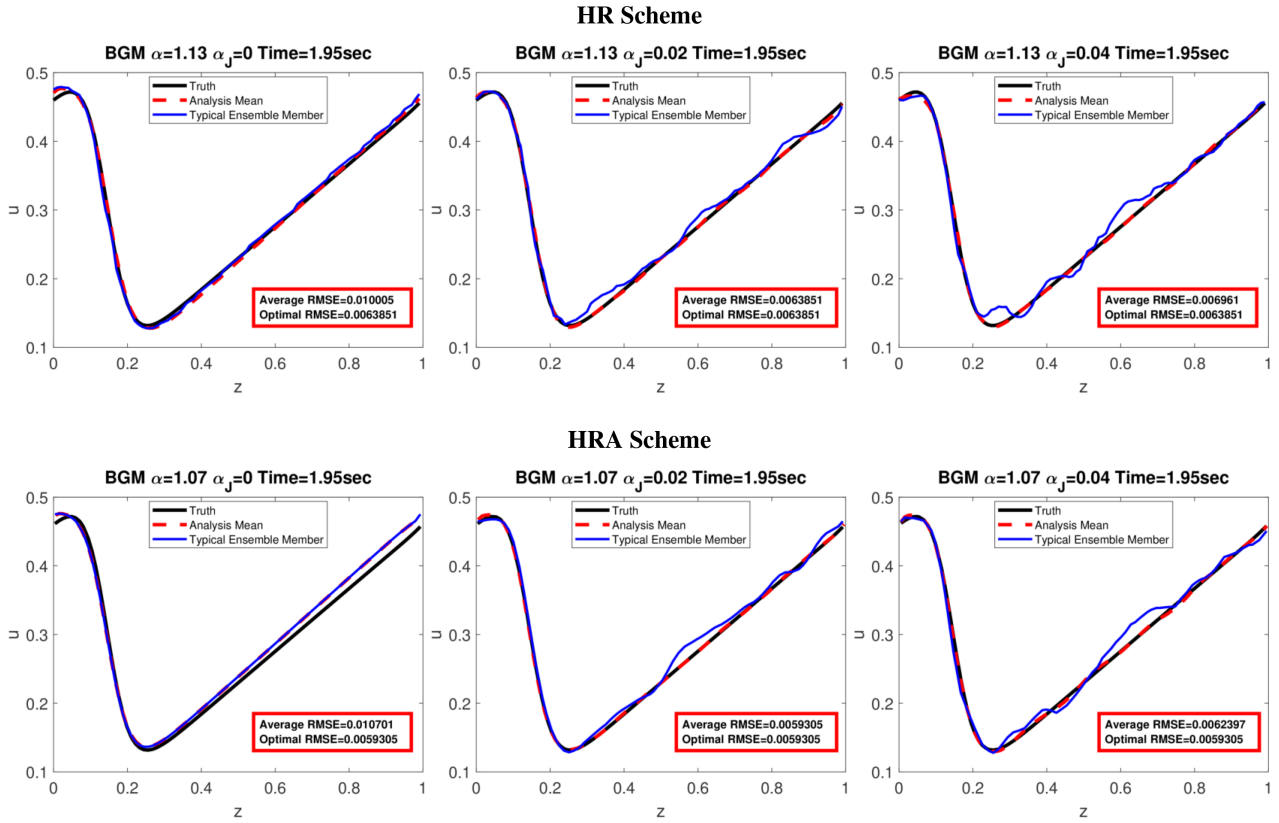
HR Scheme

HRA Scheme

F I G U R E 11 Examples of BGM analysis and an ensemble member for different inflation and jitter choices. Top HR and bottom HRA. The RMSE of the analysis mean can be low even when the ensemble members themselves represent unrealistic solutions to the PDE

$$k_{\text{ens}} = \frac{1}{N^{\text{e}}} \sum_{i=0}^{N^{\text{e}}} \frac{1}{N_{\text{an}}} \sum_{\tau_1}^{\tau_n} \frac{\frac{1}{M} \sum_{k=0}^{M} \left(d_{i_k}^{\tau_j} - E\left[d_{i_k}^{\tau_j} \right] \right)^4}{\left(\frac{1}{M} \sum_{k=0}^{M} \left(d_{i_k}^{\tau_j} - E\left[d_{i_k}^{\tau_j} \right] \right)^2 \right)^2}, \quad (31)$$

$$RMSE_{\text{ens}} = \frac{1}{N^{\text{e}}} \sum_{i=0}^{N^{\text{e}}} \frac{1}{N_{\text{an}}} \sum_{\tau_1}^{\tau_n} \frac{1}{M} \sqrt{\sum_{k=0}^{M} \left(d_{i_k}^{\tau_j} \right)^2}, \quad (32)$$

where N_{an} is the number of assimilation steps completed. If the ensemble members have a low σ_{ens}, this would suggest they do not deviate from the mean error along the domain axis, which may suggest that the shape of the curve is consistent with the true solution to the PDE and had not been overly distorted by inflation or jitter.

The kurtosis can give us a measure of how much the errors are concentrated around the mean error. A low value for the kurtosis suggests a more uniform distribution, with the normal distribution having a kurtosis of 3. Kurtosis above 3 would suggest either that the probability mass is concentrated around the mean and values far from the mean are rare, or that the probability mass is concentrated in the tails. In this particular case, a high

k_{ens} likely implies the ensemble members are not overly distorted by jitter and inflation, with large errors occurring infrequently along the domain. A low k_{ens} signals that the ensemble members are distorted by jitter and inflation, with larger deviations from the mean occurring more uniformly. However, one may have a large k_{ens} with ensemble members that have high PDE fidelity but are far apart from each other and/or far from the truth. Nevertheless, in this analysis we are looking at a *long time average* and expect the ensemble to converge around the true solution in time. Examples of ensemble members with low and high kurtosis at a specific time are shown in Figure 12. Ideally, one would hope for each ensemble member to have a low variance, low RMSE, and high kurtosis calculated from $\mathbf{d}_i^{\tau_j}$. In Figures 13–16 we show σ_{ens}, k_{ens}, and $RMSE_{\text{ens}}$ as a function of α and α_J for ensemble members before and after the update step. The lowest values for σ_{ens} and $RMSE_{\text{ens}}$ are denoted by a star, while the largest value of k_{ens} is denoted by a star. We calculate these metrics for both the forecast ensemble, right before the update step, and the analysis ensemble afterwards.

When comparing the forecast and analysis metric surfaces for BGM, there is a detectable change in structure for σ_{ens} and $RMSE_{\text{ens}}$ before and after the update. Before the update, both metrics tend to increase with increasing

Kurtosis Examples

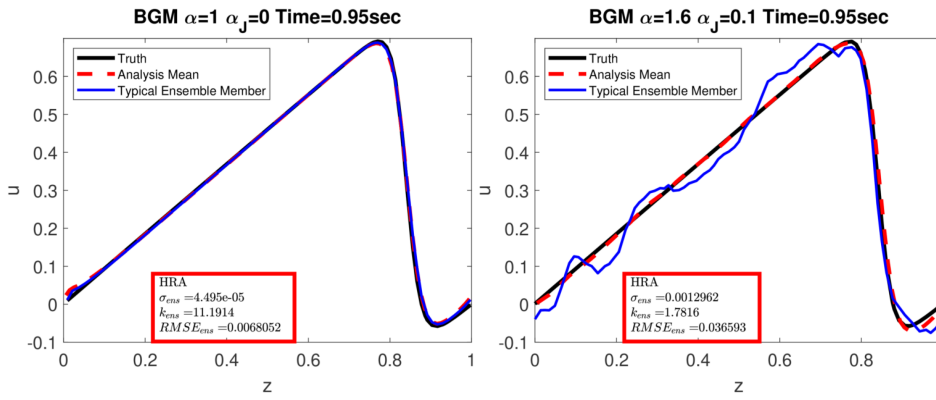
FIGURE 12 Examples of the three statistical measures we use to measure curve distortion
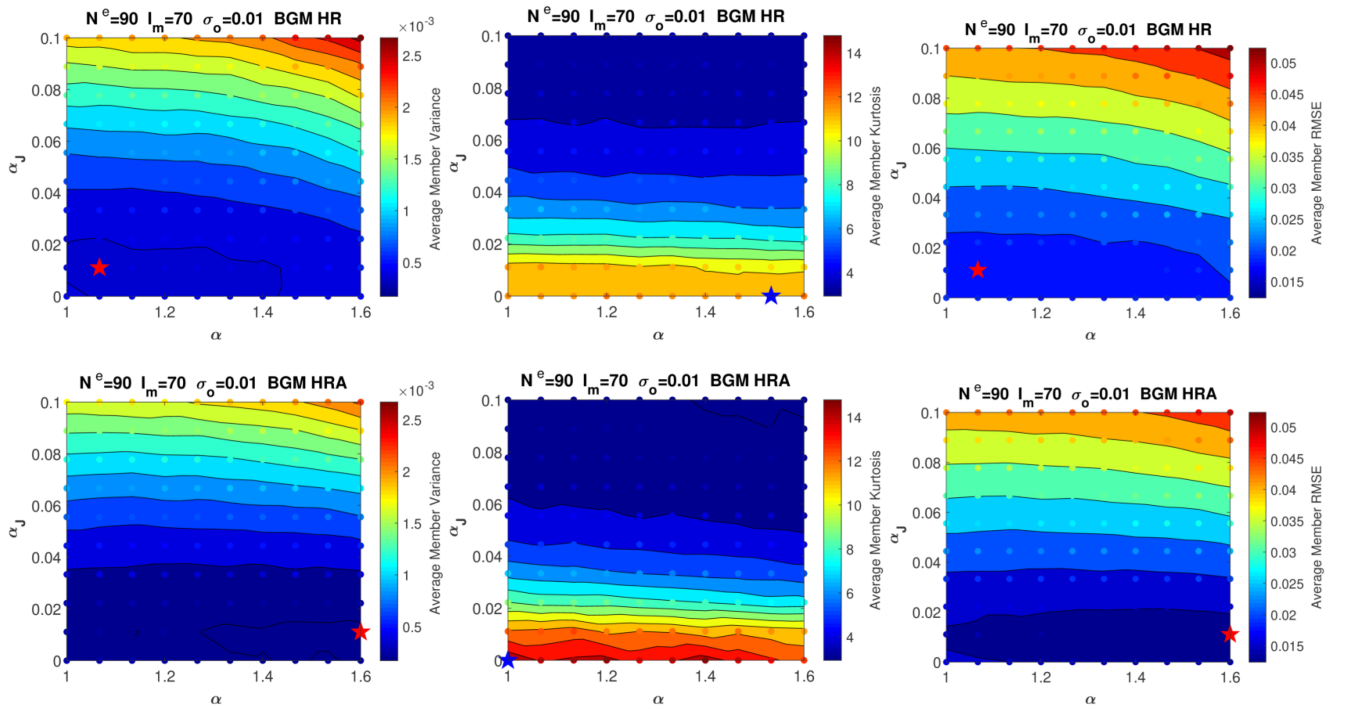
Extra Metrics Forecast Members (BGM)

FIGURE 13 Surfaces for σ_{ens}, K_{ens}, and $RMSE_{ens}$ as a function of jitter and inflation for BGM using the forecast ensemble members, right before update, for HR (top row) and HRA (bottom row). It is notable that both σ_{ens} and $RMSE_{ens}$ increase primarily as a function of jitter (α_J) with less dependence on multiplicative inflation (α), while the same is true for decreasing kurtosis evidenced by horizontal contours. This identifies the jitter as the primary source of ensemble member distortion. The stars represent the lowest values and stars the highest values

α_J somewhat independently of α, and after the update the metrics are significantly reduced for lower values of α. This is evidenced by the relatively horizontal contours in the forecast surfaces, which change after the update. However, the general structure of the kurtosis remains similar before and after the update, implying that the update is reducing the size of the error, but that the distortions remain among the ensemble members for larger values of α_J. The analysis mean is smoothed through averaging still providing low $RMSE_{ens}$. Comparing the kurtosis surfaces between the HR and HRA schemes also again shows that some jitter

is desirable in the HRA scheme, as the larger k_{ens} values are away from the x-axis. This may be caused by the ensemble members collapsing around a solution and then deviating from the truth in time, due to model instability and over-confidence in the model solutions. It is notable that the lowest values for σ_{ens} and $RMSE_{ens}$ occur for almost the same value of α_J at which we have our maximum k_{ens} for the analysis metric surfaces in the HRA case.

When making similar considerations between the forecast and analysis metric surfaces for KSM, we see a similar change in structure between the σ_{ens} and $RMSE_{ens}$
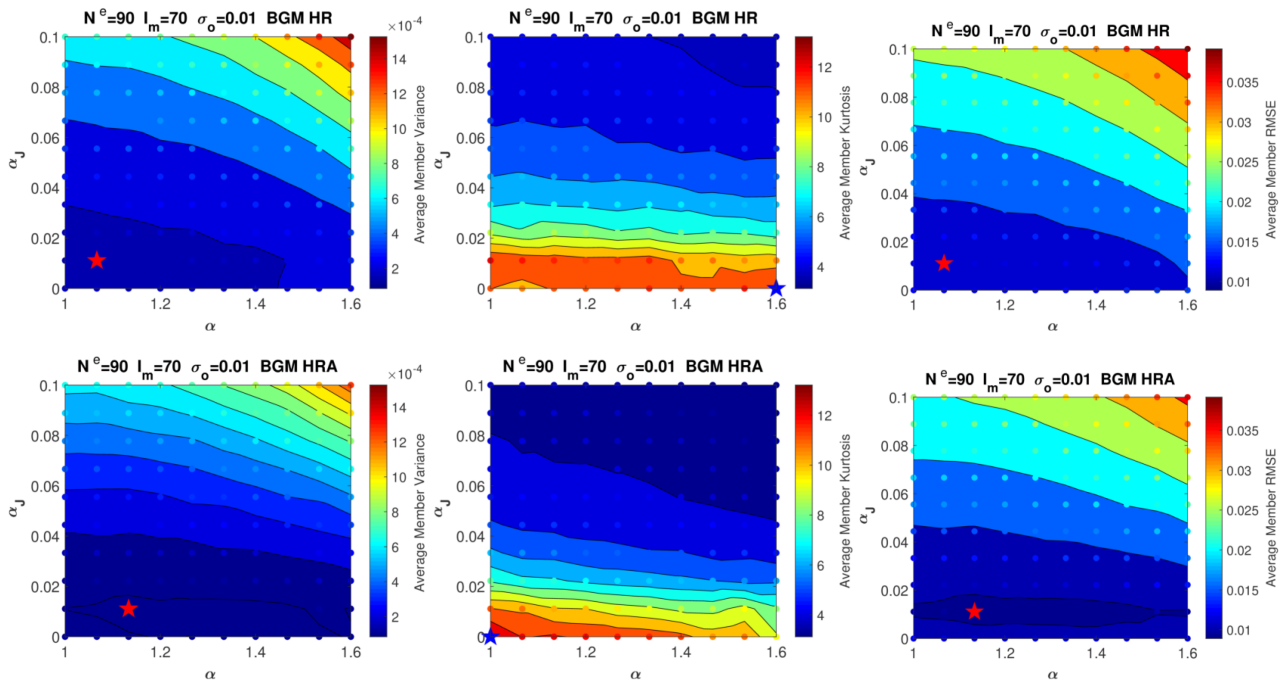
Extra Metrics Analysis Members (BGM)

FIGURE 14 Surfaces for σ_{ens}, K_{ens}, and RMSE$_{ens}$ as a function of jitter and inflation for BGM using the analysis ensemble members for HR (top row) and HRA (bottom row). It is notable that horizontal contours shown in the same analysis for the forecast members are now slanted, with improvements in σ_{ens} and RMSE$_{ens}$ for lower values of inflation (α). This is due to the update step, however the kurtosis remains relatively unchanged, implying that the members are still distorted for larger values of α_J and only the scale of the errors has been reduced. The stars represent the lowest values and stars the highest values

surfaces to that for the BGM case. However, the contours on the KSM forecast surfaces are less horizontal, implying that multiplicative inflation alone can increase the average errors. This does not necessariliy mean that the solutions are of low fidelity, given that the chaos exhibited by KSM can simply produce ensemble members that are further from the truth but still viable solutions of the underlying PDE. In fact, they do seem to be corrected at the analysis step. Interestingly, the kurtosis surface structure changes more for the HR scheme than for HRA between forecast and analysis. It is important to note that there is a significant difference in the range of k_{ens} between BGM (1–15) and KSM (3–5). This is likely due to the presence of chaos in KSM naturally increasing the spread of the ensemble members, causing some to have little distortion, and thus a closer to normal distribution of errors. The small range in k_{ens} likely makes this a less informative measure for KSM.

How one would choose α and α_J depends on the problem at hand: minimizing only the time-averaged RMSE may be the desired outcome, but if ensemble member fidelity is important, considering other metrics, such as those presented for the forecast ensemble, may also be important. For example, with BGM there is a drop in k_{ens} from 10 to 8 when going from $\alpha_J = 0.01$ to

$\alpha_J = 0.02$, the optimal value suggested when using 50 ensemble members (cf. Figure 7), for the forecast ensemble with almost no trade-off in the time-averaged RMSE between the two values. Depending on how sensitive another model component may be to the fidelity of the ensemble members, a more careful choice of inflationary parameters may be warranted.

5 | CONCLUSIONS

Adaptive mesh solvers have the potential to improve model skill and predictions greatly, but present difficulties for traditional data assimilation methods such as the EnKF. We consider here the case of a nonconservative, adaptive moving mesh, for which each member of an ensemble will potentially have different numbers of nodes in different locations. The key steps in an EnKF scheme for models of this sort are *dimension matching*, often involving interpolation with a substep of pairing state vector components, should they be in different locations, and *dimension return*. Dimension return involves removing points that were added in the matching step, or potentially adding back in points that were removed.
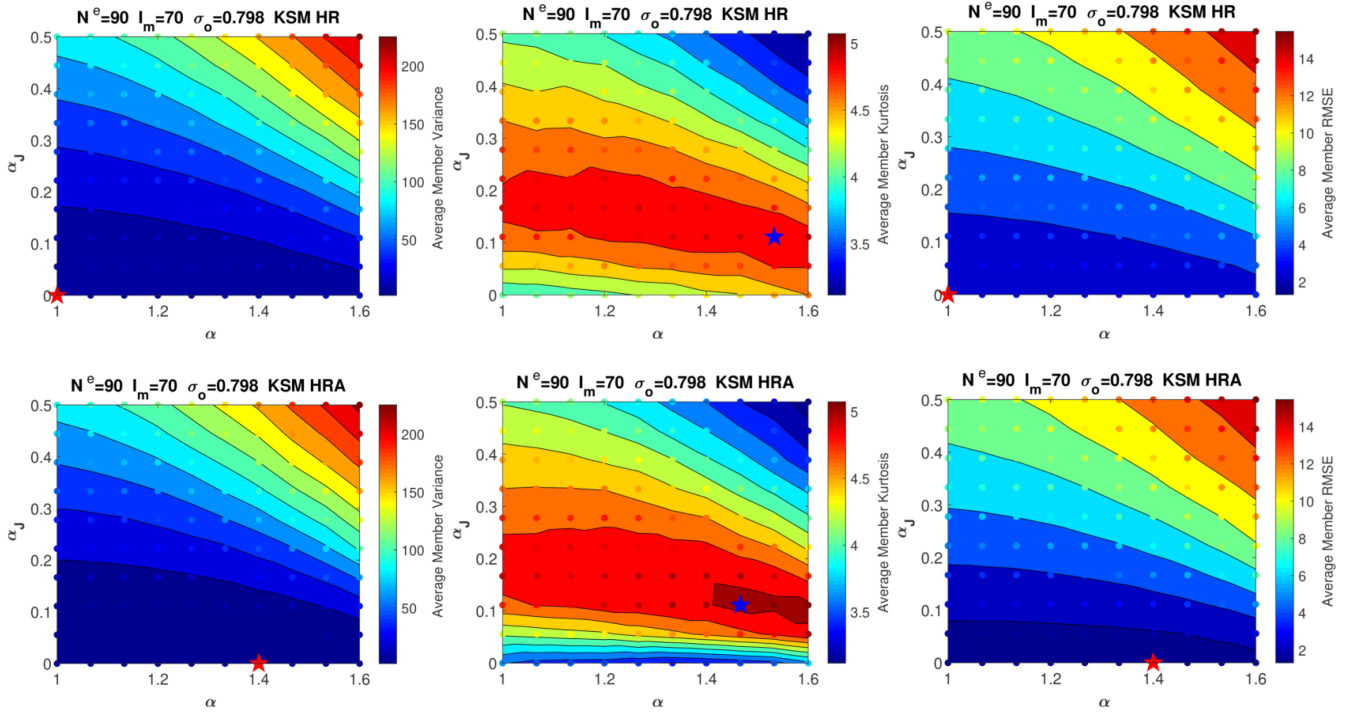
Extra Metrics Forecast Members (KSM)

FIGURE 15 Surfaces for σ_{ens}, K_{ens}, and $RMSE_{ens}$ as a function of jitter and inflation for KSM using the forecast ensemble members, right before update, for HR (top row) and HRA (bottom row). It is notable that both σ_{ens} and $RMSE_{ens}$ increase primarily as a function of jitter (α_J) with less dependence on multiplicative inflation (α), while the same is true for the decreasing kurtosis evidenced by horizontal contours. This identifies the jitter as the primary source of ensemble member distortion. The stars represent the lowest values and stars the highest values

Building on the work presented in Aydoğdu *et al.* (2019), we develop an EnKF scheme for a nonconservative, adaptive moving mesh solver in 1-d using an augmented state vector that includes the locations of the nodes, locations that are also updated in the analysis step. Dimension matching is done using the properties of the adaptive mesh scheme itself, via a partition of the domain with intervals of the same size as the proximity tolerance δ_1, thus guaranteeing that each interval will contain at most one node. In the HR scheme developed in Aydoğdu *et al.* (2019), component pairing is done by shifting the nodes in each interval to their nearest interval boundaries and then interpolating new points to any empty interval boundaries. In this way, the HR method compares ensemble members on the same mesh, updating only the physical values of the nodes. Dimension return is then performed by deleting interpolated points and shifting the updated physical values back to the previous mesh. In contrast, the HRA method leaves ensemble member points where they are and interpolates new points to empty intervals with the location drawn from a normal distribution with variance $\delta_1/2$ and a check that the location resides within that interval. Component pairing is then performed using the corresponding intervals. Next, the state vector is formed with

the node locations appended, and both physical values and nodes are updated. After the update, the remeshing scheme is applied to enforce a valid mesh, and points in previously empty intervals are deleted.

We find that, when updating the node locations, ensemble collapse becomes a problem and some additive or multiplicative inflation can become necessary. This is less of an issue for the HR method, due to some inherent stochasticity arising from the mapping procedures, although jitter and multiplicative inflation can improve RMSE values in that case as well. Given an initial mesh size, ensemble size, and observation error, the jitter and inflation are optimised with twin model experiments. When this is done, we find that the HRA method typically provides better performance in terms of the time-average RMSE for both the analysis field and its gradient. When using additive inflation, such as jitter as we have defined it, there is the potential to distort ensemble members while still obtaining a good analysis mean. This could be problematic in some frameworks, such as the HMM framework discussed in Section 4. To quantify the severity of the distortions, we calculate several metrics defined in Equations 30–32. From this analysis, we can see that the addition of jitter is primarily responsible for distorting the
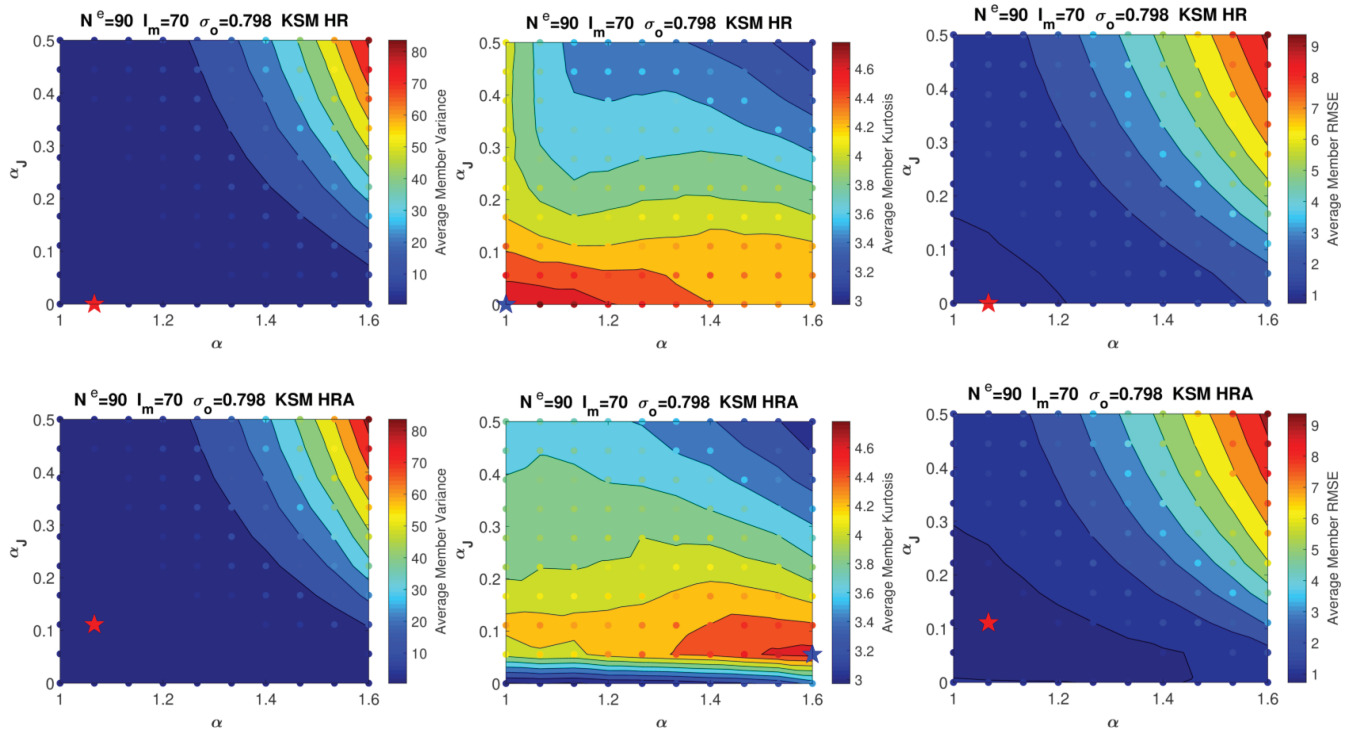
Extra Metrics Analysis Members (KSM)

FIGURE 16 Surfaces for σ_{ens}, K_{ens}, and $RMSE_{ens}$ as a function of jitter and inflation for KSM using the analysis ensemble members for HR (top row) and HRA (bottom row). It is notable that horizontal contours shown in the same analysis for the forecast members are now slanted, with improvements in σ_{ens} and $RMSE_{ens}$ for lower values of inflation (α). The stars represent the lowest values and stars the highest values

ensemble members, while multiplicative inflation has less of an effect. Low RMSE of the analysis mean would need to be weighed against the preservation of ensemble member fidelity in each particular application.

There is a natural difference in computational efficiency between these two approaches. When using the augmented state vector of the HRA scheme, the size of the error covariance matrix is doubled compared with that of the HR scheme. For very high-dimensional models this may be problematic, yet, as can be seen in Figure 6, when updating the node locations the first spatial derivative RMSE is much improved and, if that information is important, the extra computational cost may be worth it. It remains to be seen how a scheme like this plays out in models set on 2-d or 3-d adaptive meshes. We speculate that the utility in the inclusion of the cross-covariances between physical values and node locations may be far more significant in these higher dimensional cases.

The complexity and range of types of pattern that can form in 2-d or 3-d is far greater than is possible in 1-d. This implies that significantly more information may be carried in the cross-covariances between physical values and node locations. Further, if the motivation for the use of an AMM scheme is to focus computational power in

regions of strong gradients, updating those node locations in accordance with where observations of those gradients are large may be very advantageous. If the remeshing rules for the AMM model are based on strict considerations of node distances and mesh geometries, a 2-d or 3-d analogue of the HR reference mesh should be attainable, enabling the application of both the HR and HRA schemes presented here. An example of such a model is the novel Lagrangian sea-ice model neXtSIM (Rampal *et al.*, 2016; Cheng *et al.*, 2020). neXtSIM uses a finite-element method based on a triangular nonconservative adaptive mesh, with strict rules on the distance between nodes and angles between edges, and was the motivation behind our exploration in 1-d presented in this work. The authors are currently working on implementing the precursor of the current method (Aydoğdu *et al.*, 2019, i.e., without node updates) in neXtSIM. Implementing the HR scheme in 2-d is more straightforward than for the HRA scheme, as in HR all ensemble members are mapped to exactly the same grid and only the physical values are updated. Significant new challenges arise for HRA in 2-d. The key to implementing HRA is the configuration of an augmented vector that includes variables characterising the underlying numerical mesh. In 1-d there is little ambiguity, as

the natural approach is to just add the grid points of the underlying mesh. However, in 2-d the characterisation of the "grid" in terms of such vectors is not as straightforward when, for instance, using a finite-element method. In that case, physical values may be stored at the centres of the elements, whereas the elements themselves are determined by their vertices. Consideration of the mesh type—triangular, hexagonal, square, etc.—also adds complications. The development of 2-d HRA schemes and their viability is the subject of ongoing work.

ACKNOWLEDGEMENTS

CONFLICT OF INTEREST

The authors attest to no conflicts of interest regarding this work.

ORCID

Christian Sampson https://orcid.org/0000-0002-4275-4877

Ali Aydoğdu https://orcid.org/0000-0002-1557-7502

REFERENCES

Anderson, J.L. and Anderson, S.L. (1999) A Monte Carlo implementation of the nonlinear filtering problem to produce ensemble assimilations and forecasts. *Monthly Weather Review*, 127(12), 2741–2758.

Asch, M., Bocquet, M. and Nodet, M. (2016) Data assimilation: methods, algorithms, and applications, *Fundamentals of Algorithms*. Philadelphia, PA: SIAM. ISBN: 978-1-611974-53-9.

Aydoğdu, A., Carrassi, A., Guider, C.T., Jones, C.K.R.T. and Rampal, P. (2019) Data assimilation using adaptive, nonconservative, moving mesh models. *Nonlinear Processes in Geophysics*, 26(3), 175–193.

Bocquet, M. and Carrassi, A. (2017) Four-dimensional ensemble variational data assimilation and the unstable subspace. *Tellus A*, 69(1), 1304504.

Bonan, B., Nichols, N.K., Baines, M.J. and Partridge, D. (2017) Data assimilation for moving mesh methods with an application to ice sheet modelling. *Nonlinear Processes in Geophysics*, 24(3), 515–534. https://doi.org/10.5194/npg-24-515-2017.

Budhiraja, A., Friedlander, E., Guider, C., Jones, C. and Maclean, J. (2018) Assimilating data into models. In A.G.M. Fuentes and R. Smith (Eds.), *Handbook of Environmental and Ecological Statistics*. Oxfordshire: Taylor and Francis.

Burgers, G., Van Leeuwen, P.J. and Evensen, G. (1998) Analysis scheme in the ensemble Kalman Filter. *Monthly Weather Review*, 126(6), 1719–1724. https://doi.org/10.1175/1520-0493(1998)126<1719:ASITEK>2.0.CO;2.

Burgers, J.M. (1948). A mathematical model illustrating the theory of turbulence. In R. Von Mises and T. Von Kárán (Eds.), *Advances in Applied Mechanics, Vol. 1*, pp. 171–199. SAN DIEGO, CA: Elsevier, https://doi.org/10.1016/S0065-2156(08)70100-5.

Carrassi, A., Bocquet, M., Bertino, L. and Evensen, G. (2018) Data assimilation in the geosciences: an overview of methods, issues, and perspectives. *Wiley Interdisciplinary Reviews: Climate Change*, 9(5), e535. https://doi.org/10.1002/wcc.535.

Cheng, S., Aydoğdu, A., Rampal, P., Carrassi, A. and Bertino, L. (2020) Probabilistic forecasts of sea ice trajectories in the Arctic: impact of uncertainties in surface wind and ice cohesion. *Oceans*, 1(4), 326–342.

Cohn, S.E. (1993) Dynamics of short-term univariate forecast-error covariances. *Monthly Weather Review*, 121(11), 3123–3149.

Davies, D.R., Wilson, C.R. and Kramer, S.C. (2011) Fluidity: a fully unstructured anisotropic adaptive mesh computational modeling framework for geodynamics. *Geochemistry Geophysics Geosystems*, 12(6). https://doi.org/10.1029/2011GC003551.

Du, J., Zhu, J., Fang, F., Pain, C.C. and Navon, I.M. (2016) Ensemble data assimilation applied to an adaptive mesh ocean model. *International Journal for Numerical Methods in Fluids*, 82(12), 997–1009. https://doi.org/10.1002/fld.4247.

Weinan, E. (2011) *Principles of Multiscale Modeling*. Cambridge: Cambridge University Press. https://www.worldcat.org/search?q=bn%3A9781107096547&qt=advanced&dblist=638. ISBN 9781107096547.

Evensen, G. (2009) *Data Assimilation: The Ensemble Kalman Filter* (2nd edition). Heildelberg: Springer-Verlag.

Farrell, P.E., Piggott, M.D., Pain, C.C., Gorman, G.J. and Wilson, C.R. (2009) Conservative interpolation between unstructured meshes via supermesh construction. *Computer Methods in Applied Mechanics and Engineering*, 198(33), 2632–2642. https://doi.org/10.1016/j.cma.2009.03.004.

Grabowski, W.W. (2001) Coupling cloud processes with the large-scale dynamics using the Cloud-Resolving Convection Parameterization (CRCP). *Journal of the Atmospheric Sciences*, 58(9), 978–997. https://doi.org/10.1175/1520-0469(2001)058<0978:CCPWTL>2.0.CO;2. ISSN 0022-4928.

Houtekamer, P.L. and Zhang, F. (2016) Review of the ensemble Kalman Filter for atmospheric data assimilation. *Monthly Weather Review*, 144(12), 4489–4532. https://doi.org/10.1175/MWR-D-15-0440.1.

Huang, W. and Russell, R.D. (2010) *Adaptive Moving Mesh Methods*, Vol. 174. New York, NY: Springer Science and Business Media.

Jablonowski, M.C., Herzog, J.E., Penner, R.C.O., Stout, Q.F. and van Leer, B. (2004) Adaptive Grids for Weather and Climate Models. Seminar on Recent developments in numerical methods for atmospheric and ocean modelling, 6-10 September 2004. Shinfield Park, Reading: ECMWF. https://www.ecmwf.int/node/10138.

Jain, P.K., Mandli, K., Hoteit, I., Knio, O. and Dawson, C. (2018) Dynamically adaptive data-driven simulation of extreme hydrological flows. *Ocean Modelling*, 122, 85–103. https://doi.org/10.1016/j.ocemod.2017.12.004. ISSN 1463-5003.

Maddison, J.R., Marshall, D.P., Pain, C.C. and Piggott, M.D. (2011) Accurate representation of geostrophic and hydrostatic balance in unstructured mesh finite element ocean modelling. *Ocean Modelling*, 39(3), 248–261. https://doi.org/10.1016/j.ocemod.2011.04.009. ISSN 1463-5003.

Pannekoucke, O., Bocquet, M. and Ménard, R. (2018) Parametric covariance dynamics for the nonlinear diffusive Burgers equation. *Nonlinear Processes in Geophysics*, 25(3), 481–495. https://doi.org/10.5194/npg-25-481-2018.

Papageorgiou, D.T. and Smyrlis, Y.S. (1991) The route to chaos for the Kuramoto–Sivashinsky equation. *Theoretical and Computational Fluid Dynamics*, 3(1), 15–42. https://doi.org/10.1007/BF00271514.

Raanes, P.N., Bocquet, M. and Carrassi, A. (2019) Adaptive covariance inflation in the ensemble Kalman Filter by Gaussian scale mixtures. *Quarterly Journal of the Royal Meteorological Society*, 145(718), 53–75.

Rabatel, M., Rampal, P., Carrassi, A., Bertino, L. and Jones, C.K.R.T. (2018) Impact of rheology on probabilistic forecasts of sea ice trajectories: application for search and rescue operations in the Arctic. *Cryosphere*, 12(3), 935–953. https://doi.org/10.5194/tc-12-935-2018.

Rampal, P., Bouillon, S., Ólason, E. and Morlighem, M. (2016) neXtSIM: a new Lagrangian sea-ice model. *Cryosphere*, 10(3), 1055–1073. https://doi.org/10.5194/tc-10-1055-2016.

Trigg, L.E., Chen, F., Shapiro, G.I., Ingram, S.N. and Embling, C.B. (2018) An adaptive grid to improve the efficiency and accuracy of modelling underwater noise from shipping. *Marine Pollution Bulletin*, 131, 589–601. https://doi.org/10.1016/j.marpolbul.2018.04.034.

Verlaan, M. and Heemink, A.W. (2001) Nonlinearity in data assimilation applications: a practical method for analysis. *Monthly Weather Review*, 129(6), 1578–1589. https://doi.org/10.1175/1520-0493(2001)129<1578:NIDAAA>2.0.CO;2.

Weller, H., Ringler, T., Piggott, M. and Wood, N. (2010) Challenges facing adaptive mesh modeling of the atmosphere and ocean. *Bulletin of the American Meteorological Society*, 91(1), 105–108. https://doi.org/10.1175/2009BAMS2907.1.