Ocean Dynamics (2006) 56: 634–649
DOI 10.1007/s10236-006-0083-0

Lars Nerger · Sergey Danilov · Wolfgang Hiller · Jens Schröter

Using sea-level data to constrain a finite-element primitive-equation ocean model with a local SEIK filter

Received: 12 July 2005 / Accepted: 7 March 2006 / Published online: 1 August 2006

Abstract Inspired by the pioneering work of Christian Le Provost on finite element ocean modeling, a new ocean circulation model was developed over the last few years. It applies a surface triangulation and finite elements for an accurate description of coasts and bathymetry and their steering effect on the ocean circulation. A novel feature is the mesh design, which allows a vertical structure in geopotential (*z*) coordinates without loss of flexibility and avoids pressure gradient errors everywhere except for the lowest layer of abyssal ocean. The model is combined with sea-level measurements and data assimilation, another major research topic of Christian Le Provost. We apply the SEIK filter that was developed in Grenoble while Christian was teaching there. The addition of a local analysis scheme improves the filter performance, first of all, in its variance estimates and also in its mean solution.

Keywords Data assimilation · Finite elements · SEIK filter · Local filter

1 Introduction

Using unstructured meshes and the finite-element method (FEM) in ocean modeling is still uncommon despite the fact that both techniques are well-accepted and frequently used in computational fluid dynamics. Their advantage is the freedom in representing boundaries and, more generally, the ease of refining or coarsening the resolution where appropriate. These features are needed in ocean modeling as many physical processes in the ocean are sensitive to boundary conditions and bottom topography and, thus, require well-resolved representation of coastlines and topography. C. Le Provost was among those who recognized the utility of unstructured meshes for ocean modeling and prompted our interest to the topic. Another side of his activity was using data assimilation as a tool to improve the model prediction skills. This paper combines these two topics by briefly describing the finite-element ocean model (FEOM) and presenting results of sequential data assimilation carried out with FEOM configured for the North Atlantic.

A survey of the use of FEM in oceanography was recently presented in a review article by Pain et al. (2005). Similarly, FEOM has been described in much detail by Danilov et al. (2004, 2005). For the sake of completeness and also to illustrate main differences and similarities between the finite-element and standard finite-difference approaches, we present a brief description of FEOM below.

Sequential data assimilation algorithms based on the Kalman filter are of increasing interest because they promise a good utilization of the available observational information due to the dynamic approach to estimate the error covariance matrix of the estimated state. Further, the sequential character of the algorithms allows for an on-line assimilation procedure that avoids the explicit restart of a numerical model program when new observations are available. To handle the huge computational cost of the Kalman filter when applied to a large-scale model of the ocean or atmosphere, several approximating algorithms have been developed. Most common is the Ensemble Kalman Filter (EnKF, Evensen 1994), which was recently reformulated by its inventor (Evensen 2004). In addition, several variants have been developed which can be classified as square-root filters (Tippett et al. 2003). Besides the EnKF, the SEEK filter (Pham et al. 1998b) has been used in several applications, for example, in the North Atlantic (Brusdal et al. 2003). The SEIK filter (Pham et al. 1998a) is less common. It shares properties of the EnKF and SEEK algorithms. Recently, Nerger et al. (2005a, 2006) showed that assimilation with the SEIK filter bears advantages over both the EnKF and SEEK filters.

Responsible editor: Jacques Verron

L. Nerger (✉)
Global Modeling and Assimilation Office
NASA Goddard Space Flight Center,
Code 610.1 Greenbelt, MD 20771, USA
e-mail: lnerger@gmao.gsfc.nasa.gov

L. Nerger · S. Danilov · W. Hiller · J. Schröter
Alfred Wegener Institute for Polar and Marine Research,
P.O. Box 120161, 27515 Bremerhaven, Germany

With regard to the EnKF, several studies have shown the utility of a localization of the analysis (see e.g., Houtekamer and Mitchell 1998; Hamill et al. 2001) to obtain better state estimates with small ensembles. Similarly, a local analysis in a SEEK filter with stationary error estimates has been used (see, e.g., Penduff et al. 2002). For the local analysis, the state update is only computed on the bases of observations that lie within a specified distance from a grid point of the model domain. This increases the number of degrees of freedom for the analysis and can provide superior estimates of the model state.

We will follow this approach in this paper and introduce a localization method for the SEIK filter, which shows advantages over the global analysis. The resulting local SEIK (LSEIK) filter is then tested in twin experiments with FEOM configured for the North Atlantic. To assess the influence of the localization for the LSEIK algorithm, synthetic observations of the sea surface height are assimilated for different localization radii.

The paper is organized as follows. The basic principles of FEOM are described in Section 2. In Section 3, the SEIK filter is reviewed and the SEIK algorithm with local analysis update (LSEIK) is introduced. The experimental setup for studying the LSEIK filter is described in Section 4. Subsequently, results of the twin experiments are discussed in Section 5, followed by conclusions in Section 6.

2 "FEOM"

2.1 Basic equations

The FEOM uses the primitive equations for the ocean, i.e., it applies under hydrostatic, Boussinesq, and standard approximations

$$\partial_t \mathbf{u} + f\mathbf{k} \times \mathbf{u} + g\nabla_h \eta - \nabla_h A_h \nabla_h \mathbf{u} - \partial_z A_v \partial_z \mathbf{u} = \mathbf{F}, \tag{1}$$

$$\partial_z w + \nabla_h \mathbf{u} = 0, \tag{2}$$

$$\partial_t C + (\mathbf{v}\nabla)C - \nabla_h K_h \nabla_h C - \partial_z K_v \partial_z C = 0. \tag{3}$$

Here $\mathbf{F} = -\nabla_h p_H/\rho_0 - (\mathbf{v}\nabla)\mathbf{u}$, $\mathbf{v} = (\mathbf{u}, w)$ is the full velocity, with $\mathbf{u}$ the horizontal and w vertical velocities, and C is the potential temperature or salinity. The hydrostatic pressure is defined as $p_H = g\int_z^0 \rho dz$, the subscript 'h' stands for 'horizontal', and the standard notation is used otherwise.

Integrating the continuity Eq. 2 in the vertical direction, one obtains the equation on the sea surface height η

$$\partial_t \eta + \nabla \int_{-H}^{0} \mathbf{u} dz = 0. \tag{4}$$

The upper limit of integration is set to zero, implying linear free surface. The set of primitive equations is completed by the fully nonlinear equation of state $\rho = \rho(T, S, p)$ according to the UNESCO standard (for a review, see Gill 1982) and appropriate boundary conditions.

2.2 Finite-element discretization

To shorten our explanation of finite-element discretization, we begin in an unusual way from the tracer equation. The essence of the finite-element approach is that one first projects the governing equation on a set of appropriately chosen test functions $\tilde{\Psi}_i$ to obtain

$$\int \tilde{\Psi}_i(\partial_t C + (\mathbf{v}\nabla)C)d\Omega + \int (K_h \nabla_h \tilde{\Psi}_i \cdot \nabla_h C + K_v(\partial_z \tilde{\Psi}_i)\partial_z C)d\Omega = -\int \tilde{\Psi}_i q_C dS_s. \tag{5}$$

In this equation, the integration by parts is carried out to reduce requirements on differentiability of the field C, q is the surface tracer flux projected on the outer normal (pointing upward), and $d\Omega$ and dS_s denote integration over the entire domain and its surface, respectively. The fluxes through the rigid walls and bottom are assumed to be equal to zero and corresponding integrals drop out. The index i marks the test functions and corresponds to the nodes of the computational mesh in our case.

One of the convenient features of FEM seen at this stage is the implementation of flux boundary conditions. They are taken into account in a natural way as appropriate surface integrals on the right hand side of Eq. 5.

Three further steps are required to obtain the discretized equation. First, one specifies the mesh, then the representation for C, and, finally, the set of the test functions $\tilde{\Psi}_i$. The mesh of FEOM uses partitioning of the computational domain into tetrahedrons explained in Section 2.3. After the mesh is selected, all fields are expanded in series in a system of functions defined at the elements of mesh. FEOM uses linear functions for tracers, implying that

$$C = \sum_j C_j \Psi_j, \tag{6}$$

where Ψ_j is equal to unity at the node j, decreases linearly to zero at neighboring nodes and is zero outside the elements containing the node j. If such an expansion is used, C_j are the nodal values of the field C, while the expansion itself is simply a linear interpolation defining C in a continuous way at any point within the model domain.

Many other expansions are possible, including those that use higher order polynomials on elements, yet they all require the use of additional nodes within elements. The choice in favor of low-order (linear) polynomials in our case is motivated by the complexity of the ocean geometry. One wishes as many elements as possible to resolve the coastlines or continental break in a real-world application. The other extreme is exemplified by the approach based on spectral elements (for a review, see Iskandarani et al. 2003), but then the geometry of the domain is discretized into relatively large elements with less flexibility.

Substituting Eq. 6 into Eq. 5 and choosing $\tilde{\Psi}_i$ to be any of Ψ_j, one obtains the so-called Galerkin approximation of the original Eq. 3. It is written as the matrix problem on coefficients C_j

$$M_{ij}\partial_t C_j + A_{ij}C_j + D_{ij}C_j = R_i,$$

where the matrices involved are

$$M_{ij} = \int \Psi_i \Psi_j d\Omega, \quad A_{ij} = \int \Psi_i \mathbf{v} \cdot \nabla \Psi_j d\Omega,$$

$$D_{ij} = \int (K_h \nabla_h \Psi_i \cdot \nabla_h \Psi_j + K_v (\partial_z \Psi_i) \partial_z \Psi_j) d\Omega,$$

and R_i is the contribution from surface forcing. The presence of the mass matrix M_{ij} in the time derivative term makes the finite-element approach essentially different from the finite differences as time-stepping now involves inverting M_{ij} even if explicit methods are used. Although the mass matrix is sparse (the number of non-zero entries in row i is equal to the number of neighboring nodes of node i), its inverse is a full matrix, which cannot be stored for real-world applications. Thus, the system of equations above should be solved at every time step for $\partial_t C_j$. This implies calling iterative solvers, as matrices are commonly too big to use direct solvers. However, as the solver is to be called anyway, one can also use implicit or semi-implicit methods without essentially changing the CPU load. This is the commonly preferred approach as stability limitations can be an issue on unstructured meshes if explicit methods are employed. Using, for example, the Crank–Nicholson method (which is second-order accurate in time) one obtains

$$\begin{aligned}&(M_{ij}/\Delta t + A_{ij}/2 + D_{ij}/2)C_j^{n+1}\\&= (M_{ij}/\Delta t - A_{ij}/2 - D_{ij}/2)C_j^{n} + R_j.\end{aligned} \tag{7}$$

The full matrix on the left hand side is the stiffness matrix.

Several remarks are noteworthy. First, although the presence of mass matrices dictates using solvers, it simultaneously reduces dispersive properties of the numeric scheme. If the mass matrix were replaced by its lumped form (diagonal approximation with diagonal entries equal to row sums of the original matrix), Eq. 7 would reduce to standard-centered differences on uniform meshes. Weighting the time derivative over the same stencil as the advection term removes the dispersion to a large extent. Second, although the diffusion term is written in Eq. 3 in terms of horizontal and vertical contributions, implementation of other variants in FEOM is similarly straightforward and benefits from the integration by parts employed in Eq. 5. Indeed, the Reddi rotated diffusivity tensor and the Gent–McWilliams parameterization lead then to exactly symmetric and exactly antisymmetric contributions into the diffusion matrix, respectively.

Discretizing the momentum and vertically integrated continuity equations follows the same procedure as described above and uses similar expansions in linear functions for horizontal velocities and η (the latter is defined on surface triangles). The early version of FEOM used backward Euler time-stepping for both momentum and vertically integrated continuity equation and solved for two components of horizontal velocity and η simultaneously by inverting a matrix at every time step that corresponds to combining Eqs. 1 and 4 into a single matrix problem; for a review, see Danilov et al. (2004). Although this approach does not introduce any additional approximations and should be preferred from a mathematical viewpoint, it becomes impractical (numerically inefficient) as the size of the problem increases. The reason is bad-conditioning of the resulting stiffness matrix coming mostly from the inclusion of the vertically integrated continuity equation, which makes it difficult to invert it with available iterative solvers.

A numerically efficient solution requires separation of the barotropic subproblem from the full problem and involves additional approximations. They are caused by the fact that vertically integrated or barotropic velocity cannot belong to the same space of functions as the full horizontal velocity. Currently, two versions of FEOM exist, which differ in the way they solve for η. The first one introduces barotropic velocities and solves first a combined problem for barotropic velocities and η. This involves inversion of a matrix of dimension $3N_{2D}$, where N_{2D} is the number of surface nodes, which is much easier to achieve. The sea surface height found is then substituted into the horizontal momentum equations, which are solved for full horizontal velocities. Although the size of the stiffness matrix in this case is close to that of the early approach, the matrix is much easier to invert. A correction step is done afterward to compensate for the bias in the vertically integrated divergence of the full horizontal velocity.

The other version uses the pressure method similar to that implemented in the MITgcm. It makes a prediction step for the horizontal velocities and then seeks for a velocity correction and new η that make the vertically integrated continuity equation fulfilled. The details pertinent to the realization of both approaches will be reported elsewhere.

Unexpectedly, solving for w and hydrostatic pressure, which is very simple in finite difference approach, is less straightforward with the FEM. Both are first-order problems whose matrices are very inappropriate for inverting with iterative solvers. For this reason, instead of w, FEOM solves for a potential Φ, such that $w = \partial_z \Phi$. This formally transforms the problem to the second-order one. The potential Φ is expanded in linear functions leading to element-wise constant velocities. The integration for hydrostatic pressure is done in the finite difference way. The pressure found is interpreted then as element-wise linear field.

To conclude this brief description, we reiterate that the main difference between the finite-element discretization used by FEOM and the finite difference models is the non-locality of time derivatives and, hence, the need to use solvers at every time step to update prognostic fields. The other significant difference comes from the unstructured character of the mesh, implying that indices of the neighboring nodes follow no simple rule but are retrieved from look-up tables. Both, using solvers and indirect indexing, makes the time step of FEOM normalized by the number of nodes significantly longer than in FD models. Yet it remains affordable for practical applications because it provides resolution only where it is necessary, i.e., for the same quality of solution one needs less computational nodes.

2.3 Mesh

Although there is, in principle, much freedom in choosing the finite-element mesh, the strong vertical stratification of the ocean requires employing vertically aligned discretization. The FEOM choice is simple yet sufficiently flexible. Its 3D mesh is determined by the surface triangular mesh. Figure 1 gives a fragment of the surface mesh for the North Atlantic designed to include the projections of topography lines. The surface mesh defines vertical prisms that are cut by a system of z -levels into smaller prisms, and every small prism is then cut into tetrahedrons (a full prism is divided in three tetrahedrons; the prisms cut by bottom topography can sometimes consist of two or even one element). The bottom surface is allowed to deviate from z - levels. One avoids vertical walls across the continental break by choosing the horizontal mesh in such a way that it coincides with isobaths. The structure of the three-dimensional mesh constructed in this way is illustrated in Fig. 2 by a view at the southwest corner of the North Atlantic mesh employed by us. The mesh follows the topography and is free of pressure gradient errors everywhere except for the bottommost layer over the abyssal part of the bottom. Thus, such an approach combines the advantages of both z - and σ - vertical coordinates. Vertical walls are not prohibited and can be used on coarse meshes. In reality, one decides on where to admit vertical walls and where to leave the bottom topography continuous on the stage of designing the surface mesh.

It should be emphasized that it is only the desire to avoid pressure gradient errors that motivates the choice of z - levels. If some pressure gradient errors are allowed over limited areas, the levels (not necessarily all) can deviate from isopotential surfaces there and follow, for example, the topography. Yet, we admit that the unstructured character of surface mesh makes using high-order interpolations for pressure gradient difficult and is not implemented in the current version of FEOM. The FEOM code is independent of the vertical discretization, and all what is needed is the 3D mesh designed in accordance with one's wishes.

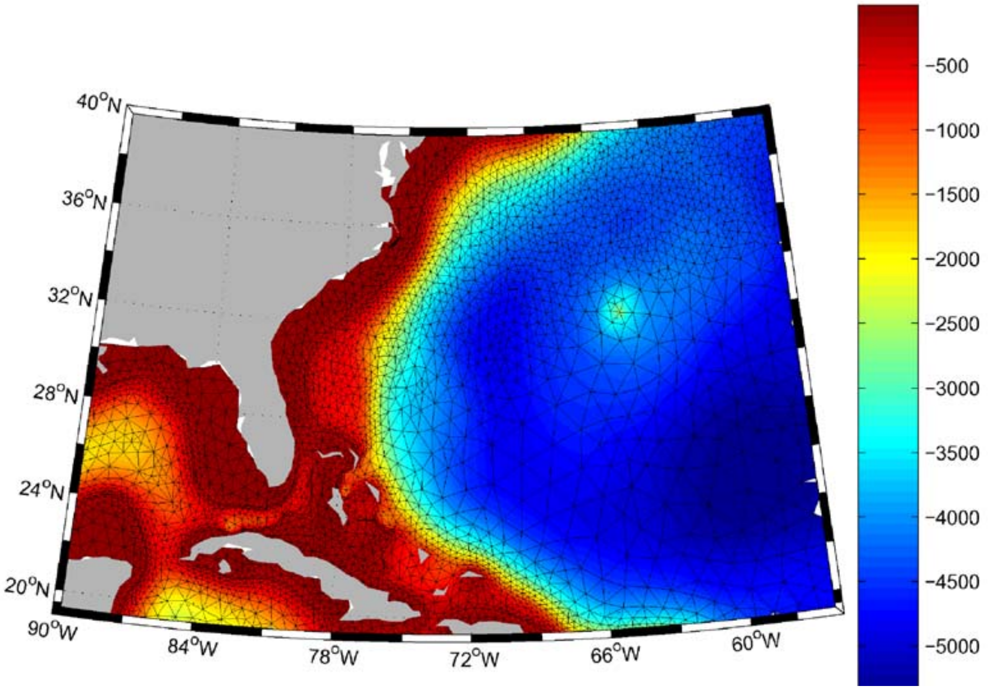

Fig. 1 Fragment of the topography following surface mesh for the North Atlantic. The topography lines included correspond to intersection of z-levels used in the model with the bottom topography

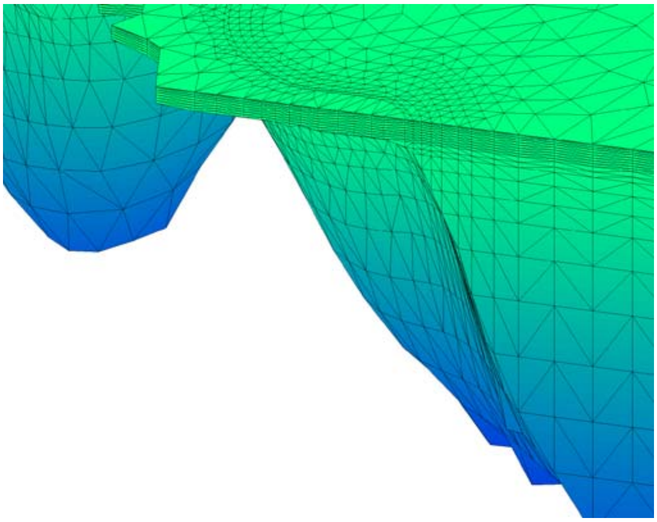

Fig. 2 View at the southwest corner of the North Atlantic mesh. The depth shown is down to −3,000 m, and the fragment is 10 and 5° in longitude and latitude, respectively (from 60° W and 5° N). The horizontal discretization over the continental slope is consistent with the set of z-levels used for vertical discretization. Nodes lying at the abyssal part of the bottom (not shown) are allowed to deviate from the regular set of z-levels

3 SEIK with local analysis

We will now turn to our concept of data assimilation in FEOM. We focus on the SEIK filter (Pham et al. 1998a), which demonstrated advantages over the widely used EnKF and SEEK filter algorithms in recent studies (Nerger et al. 2005a, 2006).

3.1 The SEIK filter

The SEIK filter (Pham et al. 1998a) can be interpreted as an ensemble-based Kalman filter using a preconditioned ensemble and a very efficient scheme to incorporate the observational information during the analysis phase of the filter. The algorithm computes the update of the state estimate in the estimated error sub-space, which is represented by the ensemble of model states. We review the algorithm as an error-subspace algorithm. For a detailed review of the SEIK filter and a comparison with the EnKF and SEEK filters, see Nerger et al. (2005a).

In general, Kalman filter algorithms express the problem of estimating the state of a physical system, such as the ocean in terms of the estimated analysis state vector $\mathbf{x}_k^a$ of dimension n at time t_k, and the corresponding covariance matrix $\tilde{\mathbf{P}}_k^a$, which represents the error estimate of the state vector. The SEIK filter, as well as the EnKF, represents these quantities by an ensemble of state vectors

$$\mathbf{X}_k^a = \{\mathbf{x}_k^{a(1)}, \ldots, \mathbf{x}_k^{a(N)}\} \tag{8}$$

of N model state realizations. That is, the state estimate is given by the ensemble mean $\overline{\mathbf{x}_k^a}$, while the ensemble covariance matrix

$$\mathbf{P}_k^a := N^{-1}(\mathbf{X}_k^a - \overline{\mathbf{X}_k^a})(\mathbf{X}_k^a - \overline{\mathbf{X}_k^a})^T \approx \tilde{\mathbf{P}}_k^a\ , \tag{9}$$

with $\overline{\mathbf{X}_k^a} = \{\overline{\mathbf{x}_k^a}, \ldots, \overline{\mathbf{x}_k^a}\}$, is an approximate estimate of the covariance matrix $\tilde{\mathbf{P}}_k^a$. Note that a factor $(N-1)^{-1}$ is usually required to obtain an unbiased estimate for $\tilde{\mathbf{P}}_k^a$. However, in the initialization and re-initialization steps of the SEIK filter (see below), the ensemble $\mathbf{X}_k^a$ is generated in a way that leads to an unbiased estimate of $\tilde{\mathbf{P}}_k^a$ with a factor N^{-1}.

The SEIK filter algorithm can be subdivided into several phases. In the "forecast phase," the state ensemble is integrated by the numerical model to propagate the state and error estimates toward the next time when observations are available. At this time, the "analysis phase" is performed. Here the state and error estimates are updated on the basis of the observations, the ensemble covariance matrix, and the error covariance matrix of the observations. Subsequently, the forecast ensemble is transformed in the "re-initialization phase," such that it represents the error estimate after the state update. Both the analysis and re-initialization phases represent the "update phase" of the filter algorithm. Having completed the update phase, the next forecast phase can start and propagate the estimates to the next observation time. The SEIK filter has to be initialized with an initial state ensemble. It can be generated from estimates of the model state and the state covariance matrix.

The SEIK algorithm is prescribed by the following equations:

Initialization We assume an initial state estimate $\mathbf{x}_0^a$. Furthermore, we suppose that the initial covariance matrix $\mathbf{P}_0^a$ is estimated by a rank-r matrix, which is given in decomposed form as

$$\mathbf{P}_0^a := \mathbf{V}_0\mathbf{U}_0\mathbf{V}_0^T, \tag{10}$$

where $\mathbf{U}_0$ is a $r \times r$ matrix, while $\mathbf{V}_0$ has size $n \times r$. This initialization can be obtained, for example, from a state trajectory from a long model run. The covariance matrix of this trajectory then represents the covariance matrix, while the mean state represents the initial state estimate. The decomposed form can be obtained by computing the singular-value decomposition of the matrix of variations around the trajectory mean state.

Based on these initial estimates, a random ensemble of minimum size $N = r + 1$ can be generated, whose mean and covariance matrix represent $\mathbf{x}_0^a$ and $\mathbf{P}_0^a$ exactly. An example for a procedure to generate such an ensemble is the so-called minimum second-order exact sampling (for a review, see Pham 2001). This procedure corresponds to a transformation of the columns in matrix $\mathbf{V}_0$ by a random matrix with special properties. Let $\mathbf{C}_0$ be a square root of the matrix $\mathbf{U}_0$, i.e., $\mathbf{U}_0 = \mathbf{C}_0^T\mathbf{C}_0$. Then $\mathbf{P}_0^a$ can be written as

$$\mathbf{P}_0^a = \mathbf{V}_0\mathbf{C}_0^T\mathbf{\Omega}_0^T\mathbf{\Omega}_0\mathbf{C}_0\mathbf{V}_0^T, \tag{11}$$

where $\mathbf{\Omega}_0$ is a $N \times r$ random matrix whose columns are orthonormal and orthogonal to the vector $(1, \ldots, 1)^T$. The ensemble of state realizations is then given by

$$\mathbf{X}_0^a = \overline{\mathbf{X}_0^a} + \sqrt{N}\ \mathbf{V}_0\mathbf{C}_0^T\mathbf{\Omega}_0^T, \tag{12}$$

where each column of $\overline{\mathbf{X}_0^a}$ contains the vector $\mathbf{x}_0^a$.

Forecast Let $M_{i,i-1}$ be the nonlinear dynamic model operator that integrates a model state from time t_{i-1} to time t_i. Then each ensemble member $\{\mathbf{x}^{a(\alpha)}, \alpha = 1, \ldots, N\}$ is evolved up to time t_k by iterating the model equation

$$\mathbf{x}_i^{f(\alpha)} = M_{i,i-1}[\mathbf{x}_{i-1}^{a(\alpha)}] + \boldsymbol{\eta}_i^{(\alpha)}. \tag{13}$$

In this equation, the superscript 'f' denotes the forecast, while 'a' denotes the analysis. Each integration is subject to individual Gaussian noise $\boldsymbol{\eta}_i^{(\alpha)}$, which allows to simulate model errors.

Analysis For the update phase, the SEIK filter uses an alternative description of the covariance matrix $\mathbf{P}_k^f$, which allows for a very efficient algorithm. $\mathbf{P}_k^f$ can be computed from the state ensemble according to

$$\mathbf{P}_k^f = N^{-1}\mathbf{X}_k^f\mathbf{T}(\mathbf{T}^T\mathbf{T})^{-1}\mathbf{T}^T(\mathbf{X}_k^f)^T. \tag{14}$$

$\mathbf{T}$ is a $N \times r$ matrix with zero column sums, such as

$$\mathbf{T} = \begin{pmatrix} \mathbf{I}_{r\times r} \\ \mathbf{0}_{1\times \mathbf{r}} \end{pmatrix} - \frac{1}{N}(\mathbf{1}_{N\times r}). \tag{15}$$

Here, $\mathbf{0}$ represents the matrix whose elements are equal to zero. The elements of the matrix $\mathbf{1}$ are equal to one. Matrix $\mathbf{T}$ implicitly subtracts the ensemble mean when computing $\mathbf{P}_k^f$. Now, we can write, in analogy to the covariance matrix in Eq. 10.

$$\mathbf{P}_k^f = \mathbf{L}_k\mathbf{G}\mathbf{L}_k^T \tag{16}$$

with

$$\mathbf{L}_k := \mathbf{X}_k^f\ \mathbf{T}, \quad \mathbf{G} := N^{-1}\left(\mathbf{T}^T\mathbf{T}\right)^{-1}. \tag{17}$$

The analysis step of the SEIK filter computes a new state estimate by updating the forecast estimate that is given by the mean of the forecast ensemble. Using Eqs. 14, 15, 16, 17, the analysis update of the state estimate can be expressed as the weighted average of the columns of $\mathbf{L}_k$:

$$\mathbf{x}_k^a = \overline{\mathbf{x}_k^f} + \mathbf{L}_k\mathbf{a}_k \tag{18}$$

The columns of $\mathbf{L}_k$ span the error subspace represented by the ensemble. The vector a_k of weights can be computed in the error subspace as

$$\mathbf{a}_k = \mathbf{U}_k(\mathbf{H}_k\mathbf{L}_k)^T\mathbf{R}_k{}^{-1}\left(\mathbf{y}_k^o - \mathbf{H}_k\overline{\mathbf{x}_k^f}\right), \tag{19}$$

$$\mathbf{U}_k^{-1} = \rho\mathbf{G}^{-1} + (\mathbf{H}_k\mathbf{L}_k)^T\mathbf{R}_k^{-1}\mathbf{H}_k\mathbf{L}_k. \tag{20}$$

Here, $\mathbf{H}_k$ is the measurement operator that computes what observations would be measured given the state $\mathbf{x}_k$. Further, $\mathbf{R}_k$ is the observation error covariance matrix and $\mathbf{y}_k^o$ denotes the vector of observations. The forgetting factor ρ, $(0 < \rho \leq 1)$ leads to an inflation of the estimated variances of the model state. It can stabilize the filter algorithm and, to some degree, account for model errors and should not be confused with the density of sea water. In the analysis, the covariance matrix is never computed explicitly. However, the analysis covariance matrix is implicitly given by $\mathbf{P}_k^a := \mathbf{L}_k\mathbf{U}_k\mathbf{L}_k^T$.

Re-initialization To proceed with the filter sequence, the ensemble represented by $\mathbf{L}_k$ (Eq. 17) has to be transformed, such that it represents the analysis state $\mathbf{x}_k^a$ and the corresponding covariance matrix $\mathbf{P}_k^a$. Analogously to the generation of the initial ensemble, we can write

$$\mathbf{P}_k^a = \mathbf{L}_k\mathbf{C}_k^T\mathbf{\Omega}_k^T\ \mathbf{\Omega}_k\mathbf{C}_k\mathbf{L}_k^T, \tag{21}$$

where a Cholesky decomposition can be applied directly on the matrix $\mathbf{U}_k^{-1}$ to obtain $\mathbf{C}_k^{-1}(\mathbf{C}^{-1})_k^T = \mathbf{U}_k^{-1}$. The matrix $\mathbf{\Omega}_k$ has the same properties as in the initialization. Usually, it is chosen to be a random matrix, but mathematically it is not required to be so. Accordingly, the re-initialized ensemble members are given by

$$\mathbf{X}_k^a = \overline{\mathbf{X}_k^a} + \sqrt{N}\ \mathbf{L}_k\mathbf{C}_k^T\mathbf{\Omega}_k^T. \tag{22}$$

3.2 The SEIK filter with local update phase

The analysis update of the state estimate is computed from a global weighted average over the full model domain (Eq. 18). All available observations are considered for each spatial location according to their weights computed from the ensemble perturbations in $\mathbf{L}_k$, the observation error covariance matrix $\mathbf{R}_k$, and the residual between the state estimate and the observations, also called innovation, $\mathbf{y}_k^o - \mathbf{H}_k\overline{\mathbf{x}_k^f}$. This global character can be useful, as long-range correlations exist in the ocean. However, it is very unlikely that these correlations can be estimated well by a state ensemble of small size. Typically, the noise in the estimated long-range covariances will be rather large, compared to existing long-range covariances, which are mostly small. Thus, the information contents in the estimated long-range covariances are likely negligible, as has been discussed, for example, by Houtekamer and Mitchell (1998). The noise can even result in spurious correlations that can deteriorate the estimation of the state. Apart from this, the global weighting can lead to a situation in which large local improvements are accompanied by locally less credible estimates in other places that still minimize the global estimation error (Nerger et al. 2005a).

To filter out noisy, and possibly spurious, long-range correlations in the analysis phase of an Kalman-based filter, it is possible to formulate a localized analysis algorithm. It is based on the assumption that observations have negligible influence for the analysis update of a certain grid point if they correspond to a location that has a large distance to that grid point. In this case, only observations within a certain distance from the grid point need to be taken into account for the analysis of the state of this location. The local analysis can also be advantageous due to the fact that the localization increases the degrees of freedom in the update of the state estimate; for a review, see Evensen (2003). In contrast to the global weighted average, each local domain will be updated using a different weight vector $\mathbf{a}_k$. This will eventually lead to a state estimate with smaller estimation errors than a global analysis update.

To perform the localization, Houtekamer and Mitchell (2001) filtered the covariance matrix $\mathbf{P}$ by an element-wise product with a matrix representing correlations of compact support. This technique has also been used by Keppenne and Rienecker (2002), who apply the localization for data assimilation in a parallelized ocean general circulation model. Examining the effect of the introduced smoothing and down-weighting of observations at intermediate distances and the neglect of remote observations showed that for small ensembles the cut-off radius for the observations should be small to obtain a minimal estimation error (Hamill et al. 2001). Typically, an optimal radius that minimizes the estimation error depending on the ensemble size can be determined. On the other hand, Mitchell et al. (2002) showed that the localization causes an imbalance in the analysis state of a primitive equation model, which is due to spurious and non-dynamical modes introduced by the analysis. A rather abstract localization scheme for the EnKF has been developed by Ott et al. (2004).

Below, equations for the local analysis are derived, which do not use an element-wise product to filter and localize the covariances. The formulation neglects observations beyond the cut-off radius, which is equivalent to an element-wise product with a step function. The derivation results in a particularly simple formulation of the local analysis and re-initialization equations. The analysis is performed by a sequence of local updates in disjoint sub-domains. Figure 3 exemplifies the domains for a localized analysis in a structured rectangular grid. The state is updated in the sub-domain $\mathcal{S}$, which is centered at $\mathbf{l}_c$. When we assume anisotropic cut-off radii (l_1, l_2), the influence region of observations for the upper right edge of $\mathcal{S}$ is given by the ellipse $\mathcal{C}$. The full region $\mathcal{D}$, shaded in light gray, is the observation influence region for the whole sub-domain $\mathcal{S}$. This localization differs from that suggested by Ott et al. (2004) who use local domains $\mathcal{S}$ that are not disjoint and coincide with the observation domains $\mathcal{D}$.

In contrast, we assume that the local observation domain $\mathcal{D}$ contains all observations within a certain distance from the grid points in the corresponding domain $\mathcal{S}$, which is disjoint from the other local analysis domains.

The localization of the analysis step can be obtained as a two-step formulation. As all quantities refer to the time index k, we will drop this index here for clarity of notation. In the first step of the localization, we restrict the update to the local analysis domain $\mathcal{S}$, taking into account all available observations. Let $\mathbf{S}_\sigma$ be a linear operator that restricts a global state vector $\mathbf{x}$ of dimension n to its local part $\mathbf{x}_\sigma$ of dimension $n_\sigma < n$ in the sub-domain $\mathcal{S}_\sigma$. The subscript σ denotes the set of parameters, which specifies the sub-domain, for example, the position of its center as well as its extent. Then, the localization of the update equation for the state (Eq. 18) is replaced by

$$\mathbf{S}_\sigma\mathbf{x}^a = \mathbf{S}_\sigma\overline{\mathbf{x}^f} + \mathbf{S}_\sigma\mathbf{L}\mathbf{a}. \tag{23}$$

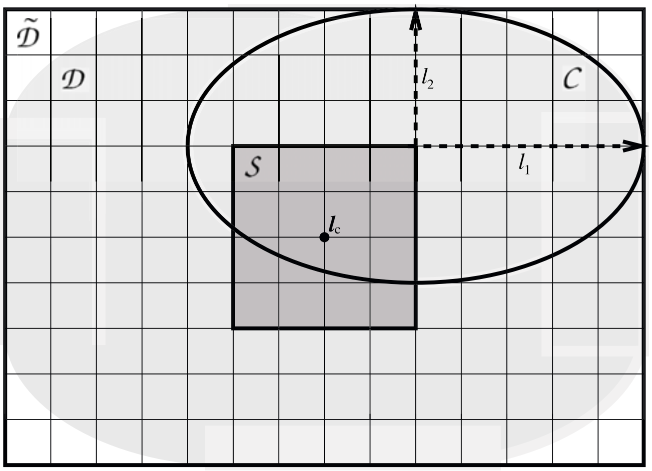

Fig. 3 Domain decomposition for a localized analysis in a structured rectangular grid (following the representation by Keppenne and Rienecker 2002). Region S, centered at the location l_c in the mesh, is the sub-domain in which the state is updated. The ellipse C marks the influence region of observations for the grid point at the upper right edge of region S. C is defined by the cut-off radii l_1 and l_2. The region D shaded in light gray marks the influence region of the observations for the whole region S

Practically, this equation implies that only certain elements of the state vector $\mathbf{x}_k^a$, corresponding to grid points within the local domain $\mathcal{S}_\sigma$, are updated by the ensemble states from the same domain with weights computed from all available observations. Thus, if we use disjoint domains and loop through all of them with Eq. 23, we reestablish the global analysis update by an alternative formulation.

The second step is to localize the observation domain. Let $\mathbf{D}_\delta$ be a linear operator that restricts a global observation vector $\mathbf{y}^o$ of dimension m to its local part $\mathbf{y}_\delta^o$ of dimension m_δ in the sub-domain $\mathcal{D}_\delta$. This amounts to the neglect of observations beyond the sub-domain $\mathcal{D}_\delta$. The subscript δ denotes the current domain, as defined by a set of parameters that specify the sub-domain in the global observation domain analogously to the local state domain. Inserting the restriction operator $\mathbf{D}_\delta$ into Eqs. 19 and 20, we can write the analysis for the local state considering only observations within domain $\mathcal{D}_\delta$ as

$$\mathbf{a}_\delta = \mathbf{U}_\delta(\mathbf{HL})^T\mathbf{D}_\delta{}^T(\mathbf{D}_\delta\mathbf{R}\mathbf{D}_\delta^T)^{-1}\mathbf{D}_\delta\left(\mathbf{y}^o - \mathbf{H}\overline{\mathbf{x}^f}\right), \tag{24}$$

$$\mathbf{U}_\delta^{-1} = \rho_\delta\mathbf{G}^{-1} + (\mathbf{HL})^T\mathbf{D}_\delta{}^T(\mathbf{D}_\delta\mathbf{R}\mathbf{D}_\delta^T)^{-1}\mathbf{D}_\delta\mathbf{HL}\ . \tag{25}$$

The forgetting factor ρ is not required to be the same for each sub-domain $\mathcal{D}_\delta$. Thus, ρ_δ denotes the local forgetting factor.

Now, we define the measurement operator $\mathbf{H}_\delta := \mathbf{D}_\delta\mathbf{H}$. which projects a (global) state vector onto the local observation domain $\mathcal{D}_\delta$. In addition, we define the local observation error covariance matrix in $\mathcal{D}_\delta$ as $\mathbf{R}_\delta := \mathbf{D}_\delta\mathbf{R}\mathbf{D}_\delta^T$. Further, we denote the local state $\mathbf{S}_\sigma\mathbf{x}^a$ as $\mathbf{x}_\sigma^a$ and analogously $\mathbf{S}_\sigma\mathbf{L}$ as $\mathbf{L}_\sigma$. With these definitions we can write the local analysis equations of the SEIK filter analogously to the global Eqs. 18, 19, 20 as

$$\mathbf{x}_\sigma^a = \overline{\mathbf{x}_\sigma^f} + \mathbf{L}_\sigma\mathbf{a}_\delta, \tag{26}$$

$$\mathbf{a}_\delta = \mathbf{U}_\delta(\mathbf{H}_\delta\mathbf{L})^T\mathbf{R}_\delta{}^{-1}\left(\mathbf{y}_\delta^o - \mathbf{H}_\delta\overline{\mathbf{x}^f}\right), \tag{27}$$

$$\mathbf{U}_\delta^{-1} = \rho_\delta\mathbf{G}^{-1} + (\mathbf{H}_\delta\mathbf{L})^T\mathbf{R}_\delta^{-1}\mathbf{H}_\delta\mathbf{L}\ . \tag{28}$$

The localization of the re-initialization phase can be performed analogously to the analysis step. The transformation of the local state ensemble is performed as

$$\mathbf{X}_\sigma^a = \overline{\mathbf{X}_\sigma^a} + \sqrt{N}\ \mathbf{L}_\sigma(\mathbf{C}_\delta)^T\mathbf{\Omega}^T, \tag{29}$$

where $\mathbf{C}_\delta^{-1}(\mathbf{C}_\delta^{-1})^T = \mathbf{U}_\delta^{-1}$. Here, it is important that the same transformation matrix $\mathbf{\Omega}$ is used for each local analysis domain. The rows of the ensemble matrix, which correspond to a single analysis domain, are transformed at once using the information from the matrix $\mathbf{U}_\delta^{-1}$ for the particular domain. This matrix corresponds to local error subspace for the local domain. According to Eq. 28, it is determined by both the local state ensemble and the local observations. To update the full domain, the local analysis and re-initialization can be performed as a sequence of independent local updates on each of the local analysis domains.

Mathematically, the localization amounts to the neglect of long-range correlations in the state covariance matrix during the analysis step; see Appendix. The neglect of long-range correlations increases the rank of the covariance matrix and, hence, leads to a larger dimension of the error-subspace in which the update of the state estimate is computed. As the state covariance matrix is never computed explicitly, this larger dimension is only considered implicitly during the independent analysis updates in the local domains. The rank of covariance matrix represented by the global state ensemble does not increase, as the rank of this matrix depends on the ensemble size N and can be at most $N-1$. In the re-initialization, each local state ensemble is transformed using the same matrix $\mathbf{\Omega}$. This consistent re-initialization results in an ensemble, which represents a covariance matrix with the same rank as the covariance matrix of the forecast ensemble.

The neglect of remote observations together with the independent local analysis updates can lead to imbalance in the ensemble states (for a review, see Mitchell et al. 2002). Further, discontinuities in the ensemble states could be induced. However, discontinuities in the model state between neighboring local analysis domains will be negligible, if the assumption holds that remote observations have a negligible influence to the analysis update. If the local observation domains $\mathcal{D}_\delta$ are sufficiently large, the observation domains of neighboring local analysis domains $\mathcal{S}_\sigma$ are widely overlapping. In this case, almost the same observational information is used for nearby analysis domains $\mathcal{S}_\sigma$. In addition, almost the same local state error covariance matrix is used implicitly. This will result in similar analysis updates for neighboring domains. Due to the imbalance induced by a local analysis, there will be a trade-off between the expected larger error reduction in the analysis step and an accelerated error growth during the forecast phase, compared to the filter algorithm with global analysis, as will be discussed below.

3.3 Implementation of the LSEIK algorithm

Equations 26, 27, 28, 29 can be implemented analogously to the global SEIK filter. The localization operators $\mathbf{S}_\sigma$ and $\mathbf{D}_\delta$ can optimally be implemented as explicit restriction operators. The global measurement operator of the SEIK filter can be applied here, too. The update of the state in the disjoint local domains $\{\mathcal{S}_\sigma\}$ can be performed in a loop over all domains. As the individual updates are independent from each other, the local updates can also be executed in parallel on a parallel computer.

In Eq. 27, the global forward mean ensemble state is required. If the local observation domains are overlapping, it is efficient to pre-compute the global array $\mathbf{HL}$ and the global vector $\mathbf{H}\overline{\mathbf{x}^f}$ before looping through all local analysis domains. Subsequently, the computation of the analysis update only requires the application of the restriction operator $\mathbf{D}_\delta$ to obtain the corresponding arrays for the local observation domain. The re-initialization step only requires the local arrays $\mathbf{L}_\sigma$ and $\mathbf{U}_\delta^{-1}$, which are computed by the analysis algorithm. This procedure can also be applied for a domain-decomposed model in a multiprocessor parallel environment. In this case, the global quantities $\mathbf{H}\overline{\mathbf{x}^f}$ and $\mathbf{HL}$ should only be computed for the part of the model domain that resides on the current processor. Subsequently, the analysis and the re-initialization steps are performed by each processor for its parallelization sub-domain. Due to the overlapping observation domains, no processor boundaries will be visible with this scheme.

The computing time required for the local filter update will be larger than for the global update. This is, for example, caused by the additional application of the restriction operators, compared to the global algorithms. Furthermore, several operations, such as the computation of the $(N-1)\times(N-1)$ matrix $\mathbf{U}_\delta^{-1}$ or the Cholesky decomposition of this matrix in the re-initialization, need to be repeated for each local analysis domain. However, as the ensemble size N is typically very small (< 100), compared to state dimension, which is of order $10^7 - 10^9$ for today's ocean models, the computing time for the filter update phase will still be negligible in comparison to the time to integrate the state ensemble in the forecast phase.

4 Experimental setup

The model used for data assimilation is configured for the North Atlantic. Its surface triangular mesh covers the area from 7° to 80° N. The horizontal resolution varies between 0.2° to 1.5° degree with mean resolution around 0.5°. The nodes are arranged on 23 vertical levels. In total, the mesh contains approximately 16,000 surface nodes and 220,000 3D nodes and the state dimension amounts to 925,000. Open boundaries of the model domain are replaced by rigid walls with relaxation to WOA94 climatology (Levitus and Boyer 1994 and Levitus et al. 1994) in 5° sponge layer at the southern boundary, north of 60° N and in 3° zone around the Strait of Gibraltar. The broad northern relaxation zone serves to compensate for model deficiencies in representing overflow processes. The model is driven by monthly mean winds from NCEP reanalyses (beginning from 1990). Heat and freshwater fluxes are modeled by relaxation of surface temperature and salinity to seasonal mean values of WOA94. The computations reported below are done with a stabilized version of FEOM using a time step of 2 h. The performance of FEOM in the same configuration is described in Danilov et al. (2005). Here, the model trajectory from Danilov et al. (2005) is used as the ocean "true" state for the twin experiments.

The configuration of the experiments carried out in this paper is analogous to that of Nerger et al. (2006). Using twin experiments starting in December 1992, the filter performance of the LSEIK filter is assessed and compared to that of the global SEIK filter. At the initial time and in monthly intervals, synthetic observations of the full sea surface height are assimilated for 3 months. The monthly interval is larger than the 10-day interval, which is typically used when satellite data is assimilated. However, monthly assimilation will be a good test for the performance of the filter algorithm regarding nonlinearities in the model, because nonlinearities will have a stronger influence over the longer assimilation interval. The observations are generated by adding uncorrelated Gaussian noise to the true model trajectory. For this, a standard deviation of 5 cm is assumed for the observations as well as that the errors are uncorrelated. To initialize the filter algorithms, the initial state estimate has been chosen from a perpetual 1990 seasonal wind model spin-up run. The error covariance matrix is chosen to be represented implicitly by the variability of the 9-year "true" trajectory. This covariance matrix is dominated by a small number of large-scale modes. Accordingly, it can be very well approximated by a matrix of significantly lower rank (for a review, see Nerger et al. 2006).

In the experiments discussed below, ensembles with 8 and 32 members have been used. To stabilize the assimilation process, a forgetting factor of $\rho = 0.8$ was applied in both the SEIK and LSEIK filters. Apart from this, no model error was simulated. Both filters are implemented in the parallel data assimilation framework PDAF (Nerger et al. 2005b). The data assimilation system is configured to integrate eight ensemble members in parallel. Each of the eight model tasks was executed by four processors; thus, 32 processors were used in total on an IBM pSeries 690 computer system.

To study the influence of the localization we choose a configuration in which each local analysis domain $\mathcal{S}_\sigma$ consists of a single water column. For simplicity, the corresponding observation domain $\mathcal{D}_\delta$ is given by a circle defined by the isotropic cut-off radius l_δ, which is centered at the water column. For the unstructured grid of FEOM, a simple search algorithm has been implemented to initialize the observation domain by all grid points lying within the localization-radius l_δ for each single water column.

5 Data assimilation experiments

To assess the influence of the localization of the update phase, we examine the filter performance with regard to the estimation of the sea surface height (SSH) field. The focus lies on general properties rather than physical details.

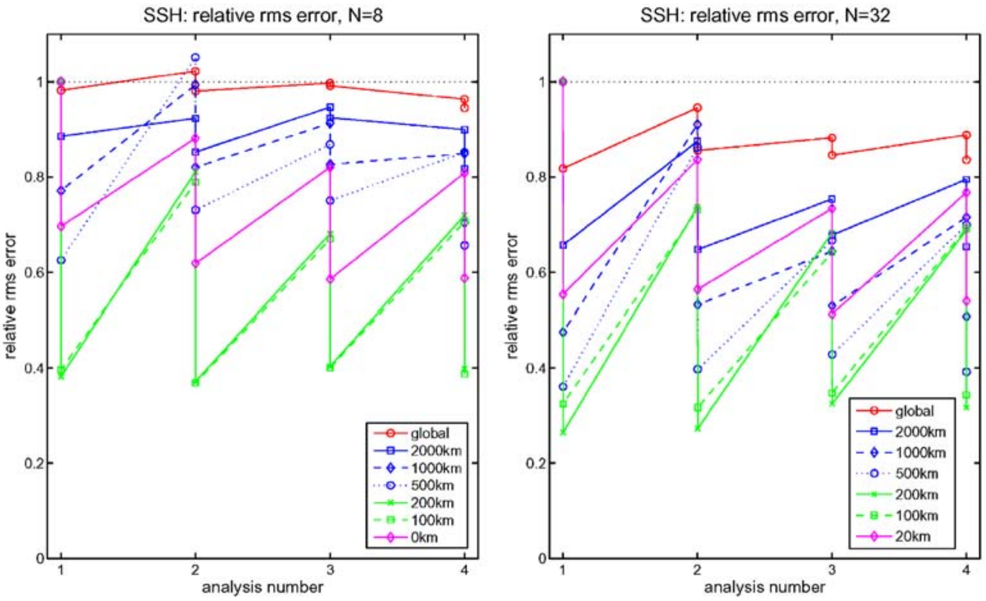

Fig. 4 RMS estimation errors of the SSH from the assimilation with SEIK and LSEIK relative to the errors from a simulation without assimilation. The different lines show results from the global SEIK and the LSEIK algorithm with different localization radii. Shown are results for ensemble sizes of eight members (*left*) and 32 members (*right*), respectively. The errors are strongly reduced by the localization

5.1 Estimation errors of the SSH

Figure 4 shows the area-weighted root mean square (rms) estimation errors from assimilations with the LSEIK and SEIK filters relative to the errors from a model integration of the initial state estimate without assimilation. Shown are results for the global SEIK analysis and the application of the LSEIK filter with different localization radii for the observations. Here, a radius of $l_\delta = 0$ km denotes an assimilation using only the observations available on each single water column. The figure also compares the results for an ensemble size of $N = 8$ (left panel) with those computed with $N = 32$ (right panel). For $N = 32$, the experiment with $l_\delta = 0$ km failed; thus, results for a small radius of 20 km are shown.

Comparing the global SEIK analyses for the two ensemble sizes, the improvement due to increasing the ensemble size to $N = 32$ is clearly visible. First, the relative rms error (RRMSE) is smaller for the larger ensemble. In addition, the error reduction at each analysis update is larger for $N = 32$, compared to $N = 8$. However, the RRMSE never decreases below 0.8. This is partly due to the small ensemble size. However, even with an ensemble of 100 members, the RRMSE after the first analysis update is only decreased to 0.74. The major cause for the small error decrease is due to the fact that the globally estimated covariance matrix differs from the true covariance matrix. This inconsistency finally also results in underestimation of the errors by the filter algorithm (for a review, see Nerger et al. 2006). As the true covariance

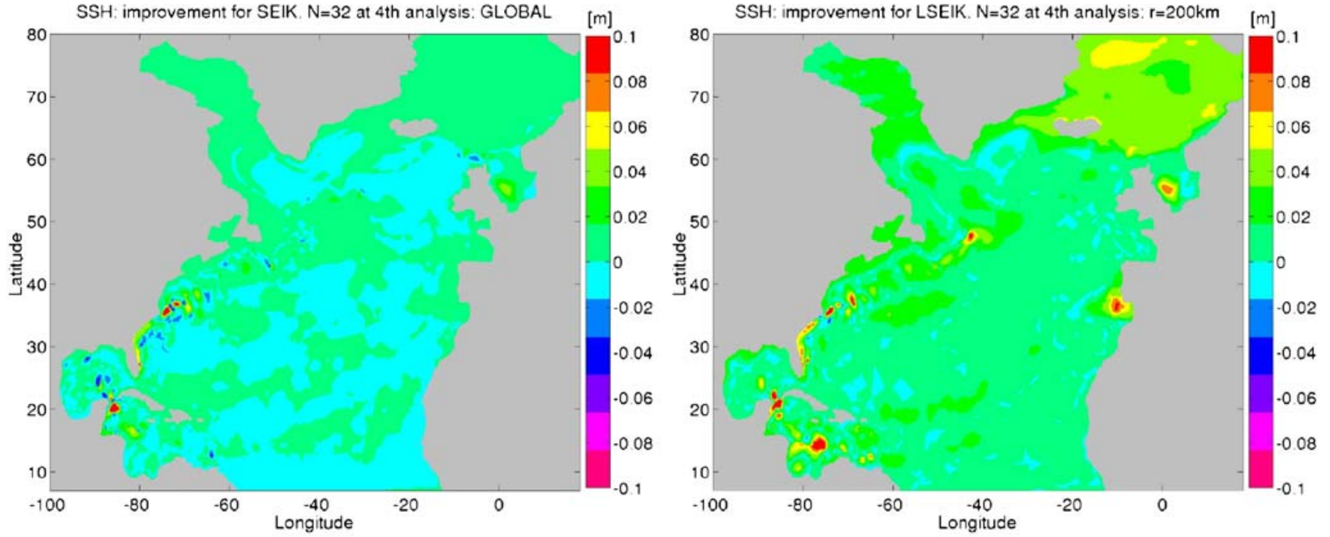
SSH: improvement for SEIK, N=32 at 4th analysis: GLOBAL
SSH: improvement for LSEIK, N=32 at 4th analysis: r=200km
[m]
0.1
0.08
0.06
0.04
0.02
0
-0.02
-0.04
-0.06
-0.08
-0.1
Latitude
80
70
60
50
40
30
20
10
Longitude
-100
-80
-60
-40
-20
0

Fig. 5 Comparison of the improvement or error reduction due to assimilation at the fourth analysis time with the global SEIK analysis (right) and with LSEIK with a localization-radius of 200 km for ensemble sizes of 32 members. The localization results in a significantly larger error reduction of the state estimate

matrix is unknown, this inconsistency can be expected for all practical filter applications.

The localization of the analysis update strongly improves the estimate of the SSH. The smallest values of the RRMSE are obtained for localization-radii of 100 and 200 km with values of about 0.4 and 0.3 for the ensembles of size 8 and 32, respectively. Note that a totally local analysis $l_\delta = 0$ km or small localization radii below 100 km performs worse. Seemingly, the uncorrelated error in the data is not sufficiently damped for small radii. Next to the small relative errors, the error reduction at each analysis update is much larger with a localization radius of 500 km or below, compared to the global SEIK analysis. This shows that the observational information is used more efficiently to reduce the error of the state estimation. Apart from the improved state estimation, the time dependence shows that the error increase during the forecast phase is larger when a local analysis is applied. This is probably caused by an imbalance in the ensemble states, which results from the independent analysis update of the state estimate for distinct grid points (for a review, see Mitchell et al. 2002).

5.2 Effect of localization at fourth analysis phase

To discuss the effects of the localization in detail, we consider the third forecast phase, as well as the fourth analysis update, which takes place at the end of the third cycle of alternating 1-month forecasts and update phases. This update phase was also examined by Nerger et al. (2006) to compare the global SEIK filter with the EnKF.

Figure 5 compares the improvement of the state estimate due to the assimilation with the global SEIK filter (right) and LSEIK with $l_\delta = 200$ km (left) for an ensemble size of 32. The improvement, or error reduction, is given by the difference of the absolute values of the true estimation error for the SSH from the estimated state before and after the analysis. The figures show that the areas in which the global analysis performed best are improved in a similar manner when the local analysis is applied. However, with the local analysis, there are many more improvements on the local scale of several Rossby radii like in the Caribbean or in the Gulf stream region. In addition, the estimate in the Nordic seas is improved, while with the global analysis only a marginal error reduction is obtained here. For an ensemble of only eight members, the improvement is similar to the case of $N = 32$. Hence, increasing the ensemble size to 32 members only enhances marginally the estimates.

As the full SSH field is observed, the error reduction can be directly related to the estimated and true variances of the SSH. Figures 6 and 7 show the field of true and estimated standard deviations of the SSH for the case of $N = 32$ for the global SEIK and the LSEIK filter with $l_\delta = 200$ km, respectively. In general, the figures show that the filter algorithms underestimate the true errors. The underestimation is more severe for the global SEIK filter than for LSEIK. We attribute this fact to the inconsistency between the estimated and true errors at the initialization (for a review, see Nerger et al. 2006). The inconsistency is not a peculiarity of our experiments but a general issue, because the true errors and error correlations are unknown in applications with real observations. The inconsistency is larger for the global error fields than for a local patch of 200-km radius. Accordingly, better error estimates are obtained with the local analysis, which result in a larger improvement of the estimate of the SSH field as visible in Fig. 5.

Comparing the errors after the third analysis (top) with the errors before the fourth analysis (middle), the change of the true and estimated errors due to the nonlinear ensemble forecast is visible. For both filter variants, the true errors, given by the absolute value of the difference between ensemble mean and the true state, grow in the Gulf of Mexico, as well as in the Caribbean and the Gulf stream region, which are the regions with the strongest nonlinearities. Further, the true error in the Nordic seas increases, which is caused by the relaxation performed in this region. The error in the North Sea and near Gibraltar has grown, too. This is particularly visible for LSEIK. The differences in the error increase for both filter variants are due to the different ensembles of both filters.

The estimated errors are represented by the spread of the state ensembles. In the experiments, the integrations of the ensemble states results in a strong increase of the estimated errors in the Caribbean, the Gulf of Mexico, and the Gulf stream region. The estimated error in the North Sea increases. The increases of the estimated errors are similar for both filter methods. A successful filter will not only produce a solution that is reasonably close to the true ocean state; it will also provide realistic error estimates, as is common in data assimilation. There are slight differences, like a small increase of the error in the Nordic seas is estimated from the ensemble of LSEIK, while this is absent in the global SEIK filter. However, the ensemble integration in both filter methods is able to reproduce the error increase in the regions with strong nonlinearity. The increase of the true errors in the Nordic seas, as well as near Gibraltar, is less well-estimated. Due to the relaxation performed here, it should be considered as a model bias, which cannot be represented by the ensemble spread.

The influence of the localization of the analysis update is visible when the errors before the fourth analysis (middle) are compared with those after the fourth analysis (bottom). For the global analysis update by SEIK, shown in Fig. 6, the true errors are only reduced in local areas in the Gulf Stream region, the Gulf of Mexico, and in the Caribbean as is also visible from the improvement shown in the left panel of Fig. 5. Overall, the errors are only reduced by a very small amount and the error pattern remains unchanged. The improvements correspond to areas with the largest estimated errors. The reduction of the error estimate generally corresponds well with the true error reduction. However, the error estimates and their reductions are much smaller than the true errors and their reductions. There are regions with large error estimates that generally correspond to regions with significant true estimation errors. At these

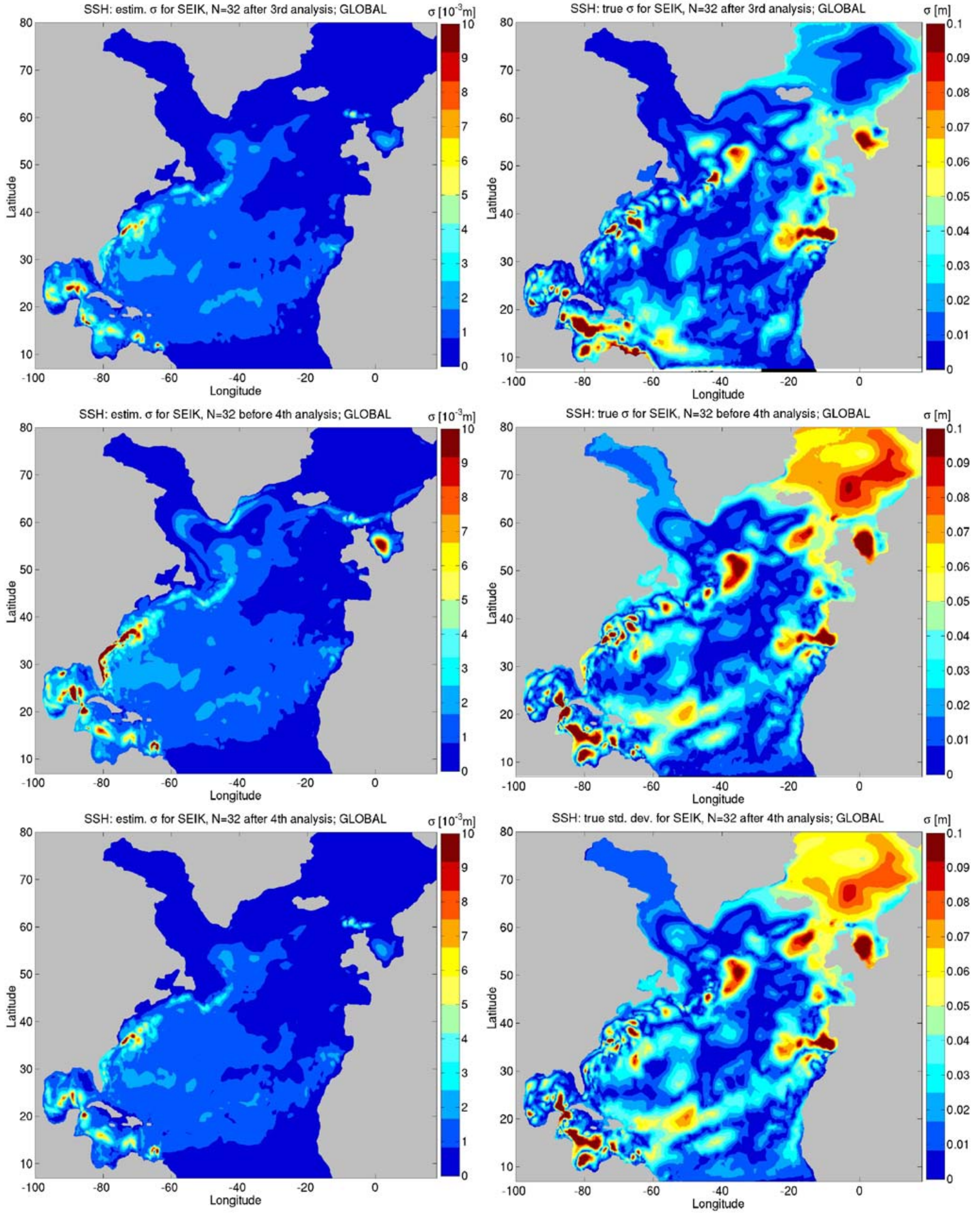
SSH: estim. σ for SEIK, N=32 after 3rd analysis; GLOBAL
σ [10^{-3}m]
SSH: true σ for SEIK, N=32 after 3rd analysis; GLOBAL
σ [m]
SSH: estim. σ for SEIK, N=32 before 4th analysis; GLOBAL
σ [10^{-3}m]
SSH: true σ for SEIK, N=32 before 4th analysis; GLOBAL
σ [m]
SSH: estim. σ for SEIK, N=32 after 4th analysis; GLOBAL
σ [10^{-3}m]
SSH: true std. dev. for SEIK, N=32 after 4th analysis; GLOBAL
σ [m]
Latitude
Longitude
80
70
60
50
40
30
20
10
-100
-80
-60
-40
-20
0
10
9
8
7
6
5
4
3
2
1
0
0.1
0.09
0.08
0.07
0.06
0.05
0.04
0.03
0.02
0.01
0

Fig. 6 Standard deviation σ of the SSH estimate after the third analysis (*top*), before the fourth analysis (*middle*), and after the fourth analysis (*bottom*) for the SEIK filter with global analysis. The *left column* shows the filter-estimated errors, while the *right column* shows true errors. Errors are significantly underestimated by the SEIK filter

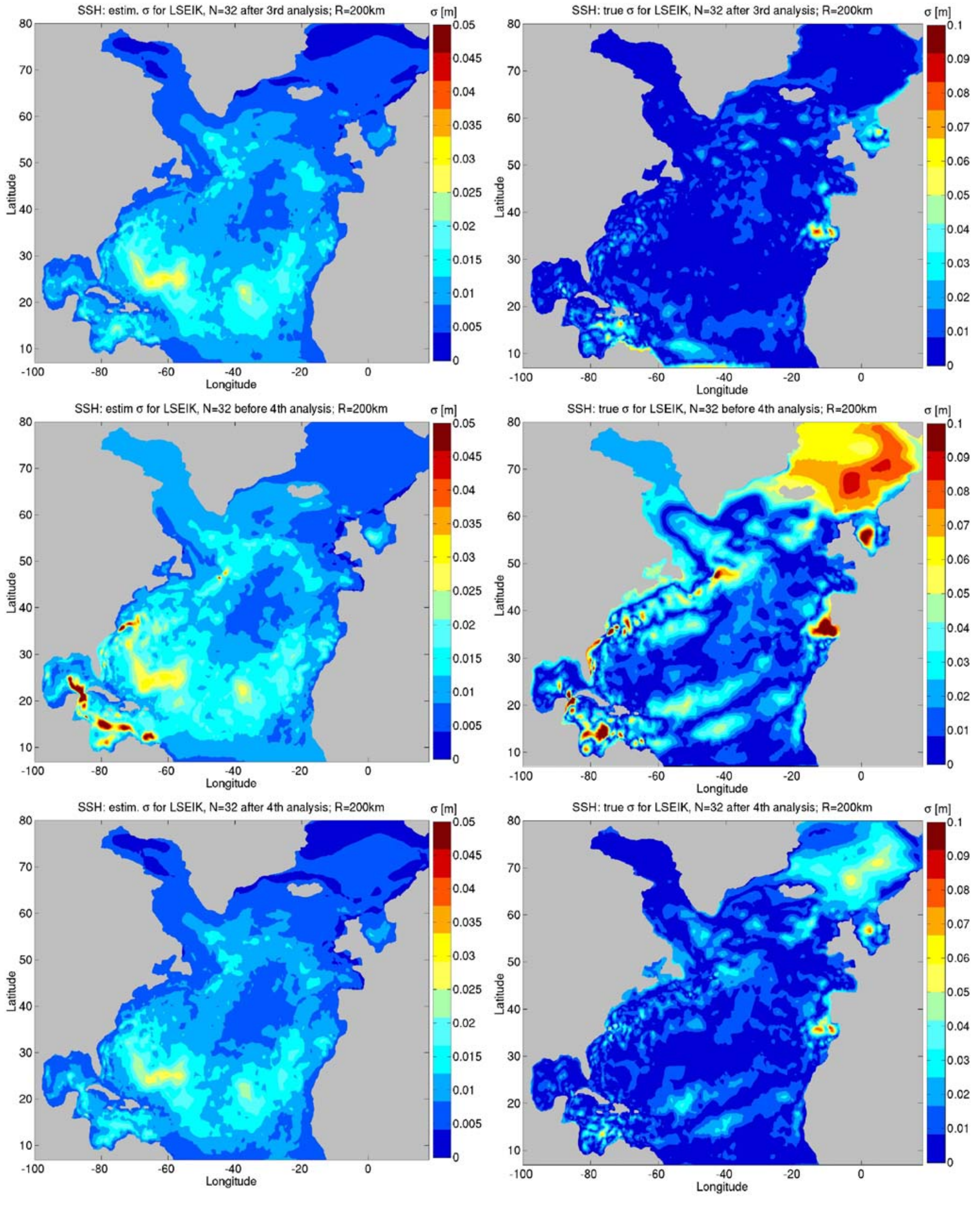
SSH: estim. σ for LSEIK, N=32 after 3rd analysis; R=200km
SSH: true σ for LSEIK, N=32 after 3rd analysis; R=200km
SSH: estim σ for LSEIK, N=32 before 4th analysis; R=200km
SSH: true σ for LSEIK, N=32 before 4th analysis; R=200km
SSH: estim. σ for LSEIK, N=32 after 4th analysis; R=200km
SSH: true σ for LSEIK, N=32 after 4th analysis; R=200km
Latitude
Longitude
σ [m]
80
70
60
50
40
30
20
10
-100
-80
-60
-40
-20
0
0.05
0.045
0.04
0.035
0.03
0.025
0.02
0.015
0.01
0.005
0
0.1
0.09
0.08
0.07
0.06
0.05
0.04
0.03
0.02
0.01
0

Fig. 7 Standard deviation σ of the SSH estimate after the third analysis (*top*), before the fourth analysis (*middle*), and after the fourth analysis (*bottom*) for the LSEIK filter with a localization radius of $l_\delta = 200$ km. The *left column* shows the filter-estimated errors, while the *right column* shows true errors. Errors are underestimated, but to a lesser degree than with the global SEIK filter

locations, no improvement of the state was possible with the global analysis update due to its global averaging character.

Figure 7 shows that the reduction of both the true and estimated errors is much larger when the local analysis is applied. In the true error field, almost all of the strong errors in the Gulf stream region, the Gulf of Mexico, and in the Caribbean have vanished due to the assimilation. In addition, the errors west of Gibraltar and in the Nordic seas are reduced, despite the fact that the estimated error is unrealistically small here. The improvement in these regions is governed by the large residual between the state estimate and the observations, which yields a large weight toward the observations. The comparison with the global analysis shows that these corrections are only possible with the local analysis. For the global analysis, the weight for the analysis update is dominated by other locations where large residuals and large error estimates of the state occur.

Comparing the true error fields after the third analysis (top) with those after the fourth analysis (bottom), it is obvious that the local analysis corrects practically all errors that have been generated during the forecast phase. An exception is the Nordic seas, where the bias effect of the relaxation to climatology cannot be corrected by the filter algorithm without a special estimate of the bias (see, e.g., Keppenne et al. 2005). Considering the estimated errors, is it also visible that all errors are corrected, which correspond to realistically estimated errors. This observation underlines the necessity of obtaining good variance estimates, in addition to a good mean solution in data assimilation. The estimated error at about 25° N, 58-70° W is only slightly reduced. However, in this region, the residual between state estimate and observations is very small; thus, no large improvement can be obtained. The larger error estimate in this region is due to the inconsistency between the true and estimated covariance matrices in the initialization (for a review, see Nerger et al. 2006).

5.3 Influence of the localization on non-observed fields

To examine the effect of the data assimilation on non-observed fields, we discuss the velocity field at 100 m depth. As only the SSH is observed, the velocities are updated only via the estimated cross-correlations between this field and the observed SSH. Figure 8 shows the area-weighted rms estimation errors relative to the errors from a model integration of the initial state estimate without assimilation. Analogous to Fig. 4, results for the global SEIK analysis and the LSEIK filter with different localization radii for the observations are shown.

For the global SEIK filter, it is clearly visible that the increase of the ensemble size improves the estimate of the velocity field. Due to the larger ensemble size, the covariances between the SSH and the velocity field can be better represented by the state ensemble. The RRMSE for the global SEIK filter with ensemble size 32 is even slightly smaller for the velocities than for the SSH. This is caused by the combined effect of state improvement by assimilation and the error decrease during the second forecast phase.

For the local SEIK filter, the RRMSE is significantly reduced, compared to the global SEIK filter. While the error reduction is less obvious for large localization radii of 2,000 and 1,000 km, the lowest RRMSE is obtained for smaller localization radii of 100 and 200 km. For $N = 8$ and $N = 32$ minimum RRMSE values of about 0.5 and 0.35 are obtained, respectively. These errors are larger than those obtained for the SSH (0.37 for $N = 8$ and 0.27 for $N = 32$). This is expected, because the velocity field is updated via the estimated covariances between this field and the SSH. As the estimates of the covariances are inferior, compared to the variance estimates of the SSH field, smaller improvements in the estimation of the velocity field are obtained.

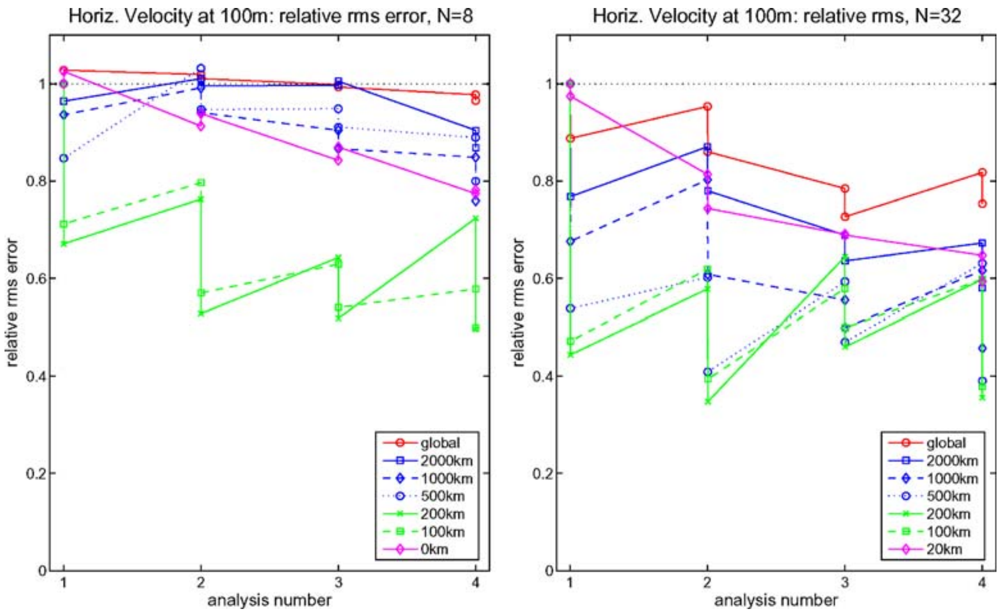

Fig. 8 RMS estimation errors of the horizontal velocity at 100 m depth from the assimilation with SEIK and LSEIK relative to the errors from a simulation without assimilation. The different lines show results from the global SEIK and the LSEIK algorithm with different localization radii. Shown are results for ensemble sizes of eight members (*left*) and 32 members (*right*), respectively. Also for this non-observed field, the errors are strongly reduced by the localization

The assimilation experiment for $N = 8$ with a total localization (localization radius $l_\delta = 0$ km) exhibits a special behavior that is, to some extent, also visible for the experiment with $N = 32$ and $l_\delta = 20$ km. The analysis updates are rather small and increase the RRMSE at all times. In contrast, the forecast phase results in decreasing RRMSE values. Here, we observe the phenomenon of overfitting, i.e., the model is adjusted not only to the data but also to the noise.

6 Conclusions and outlook

Inspired by the vision of Christian Le Provost, we have developed a finite-element ocean general circulation model and advanced filtering techniques. The ocean model is very flexible in its representation of coastlines and bottom topography, which have a dominating effect in steering the circulation. The choice of triangulation also allows for a bottom following description in the context of a geopotential (z-coordinate) model and avoids the problem of erroneous pressure gradient calculations.

Our North Atlantic version of FEOM was used as test bed for a localized SEIK filter, an ensemble-based variant of Kalman filtering. The artificial measurement error was chosen to be high. Model biases and nonlinearities show up. The best data assimilation results are obtained with a local analysis scheme (LSEIK) of moderate influence radius of 100 to 200 km. This scheme is far more efficient with a small number of ensemble members than a global analysis scheme with four times as many members. The good performance can be attributed to a much better estimate of the variance of the solution, which allows appropriate error reduction by the observations.

In future, the artificial boundaries of the North Atlantic model will be removed by embedding the area in a global model setup, with focus on the North Atlantic in a seamless way. Real radar altimetry will be assimilated in this model to make circulation estimates more realistic.

Acknowledgements We are grateful to the two anonymous reviewers for their comments, which helped to improve the text. The computations have been performed on the IBM pSeries 690 Computer of the HLRN network under the project hbk00015.

Appendix

1.1 Relation of the local SEIK analysis to an explicitly localized state covariance matrix

The local SEIK analysis derived in Section 3.2 amounts to implicitly neglecting long-range correlations in the state covariance matrix. Here, we will show this explicitly by rewriting the analysis Eqs. 26, 27, 28.

Combining Eqs. 26 and 27 and writing explicitly the localization operator x_σ, it holds:

$$\mathbf{S}_\sigma \mathbf{x}^a = \mathbf{S}_\sigma \overline{\mathbf{x}^f} + \mathbf{S}_\sigma \mathbf{L}\mathbf{U}_\delta (\mathbf{H}_\delta \mathbf{L})^T \mathbf{R}_\delta^{-1} \left(\mathbf{y}_\delta^o - \mathbf{H}_\delta \overline{\mathbf{x}^f} \right) \tag{30}$$

Now, define the operator $\tilde{\mathbf{S}}_{\sigma\delta}$ as the linear operator that restricts a vector on the observation domain $\mathcal{D}_\delta$ to the local analysis domain $\mathcal{S}_\sigma$. Thus, $\mathbf{S}_\sigma = \tilde{\mathbf{S}}_{\sigma\delta}\mathbf{D}_\delta$ and

$$\tilde{\mathbf{S}}_{\sigma\delta} \mathbf{x}_\delta^a = \tilde{\mathbf{S}}_{\sigma\delta} \left[\overline{\mathbf{x}_\delta^f} + \mathbf{L}_\delta \mathbf{U}_\delta (\mathbf{H}_\delta \mathbf{L})^T \mathbf{R}_\delta^{-1} \left(\mathbf{y}_\delta^o - \mathbf{H}_\delta \overline{\mathbf{x}^f} \right) \right] \tag{31}$$

Now, we consider the case that the observation operator $\mathbf{H}$ involves only local operations. If single water columns are used as local analysis domains, this is fulfilled if $\mathbf{H}$ operates only on each single water column. Thus, possible observations are, e.g., the sea surface height or surface temperature. However, also vertical integration would be allowed. In this case, the action of the observation operator on some state vector $\mathbf{x}$ followed by the localization operator to the observation domain $\mathbf{D}_\delta$ can be written as

$$\mathbf{D}_\delta \mathbf{H} \mathbf{x} = \tilde{\mathbf{H}}_\delta \mathbf{D}_\delta \mathbf{x} = \tilde{\mathbf{H}}_\delta \mathbf{x}_\delta, \tag{32}$$

where $\tilde{\mathbf{H}}_\delta$ denotes the observation operator, which is local to the domain $\mathcal{D}_\delta$. Using Eq. 32 with Eq. 31, we finally obtain

$$\tilde{\mathbf{S}}_{\sigma\delta} \mathbf{x}_\delta^a = \tilde{\mathbf{S}}_{\sigma\delta} \left[\overline{\mathbf{x}_\delta^f} + \mathbf{L}_\delta \mathbf{U}_\delta \mathbf{L}_\delta^T \tilde{\mathbf{H}}_\delta^T \mathbf{R}_\delta^{-1} \left(\mathbf{y}_\delta^o - \tilde{\mathbf{H}}_\delta \overline{\mathbf{x}_\delta^f} \right) \right] \tag{33}$$

with

$$\mathbf{U}_\delta^{-1} = \rho_\delta \mathbf{G}^{-1} + (\tilde{\mathbf{H}}_\delta \mathbf{L}_\delta)^T \mathbf{R}_\delta^{-1} \tilde{\mathbf{H}}_\delta \mathbf{L}_\delta \ . \tag{34}$$

Equations 33 and 34 show that, for local observation operators, the local analysis can be formulated as a full SEIK analysis performed solely using the ensemble, the state estimate, and observations on the local observation domain $\mathcal{D}_\delta$. The application of the localization operator $\tilde{\mathbf{S}}_{\sigma\delta}$ amounts to the choice of a sub-domain of $\mathcal{D}_\delta$, like the central water column of the local observation domain. If the observation operator is non-local, e.g., computing finite-difference derivatives, state information from the outside of the observation domain is required. However, this usually only involves nearest-neighbor-grid points, which do not change the general situation of the local analysis.

The local analysis covariance matrix is given as $\mathbf{P}^a_\delta = \mathbf{L}_\delta \mathbf{U}_\delta \mathbf{L}^T_\delta$ and is computed only by fields on the local domain $\mathcal{D}_\delta$. Thus, any covariances of a length scale larger than the local observation domain are neglected, while the covariances in $\mathbf{P}^a_\delta$ are the same as the corresponding covariances in the global matrix $\mathbf{P}^a$ represented by the ensemble of size N. In general, it is possible to obtain an analysis with the global SEIK filter that provides the same analysis state and covariance matrix. For this, however, an ensemble of a much larger size would be required to represent the larger rank of the covariance matrix and to provide the same estimates for the covariances, which are considered in the local analysis.

References

Brusdal K, Brankart JM, Halberstadt G, Evensen G, Brasseur P, van Leeuwen PJ, Dombrowsky E, Verron J (2003) A demonstration of ensemble based assimilation methods with a layered OGCM from the perspective of operational ocean forecasting systems. J Mar Syst 40–41:253–289

Danilov S, Kivman G, Schröter J (2004) A finite-element ocean model: principles and evaluation. Ocean Model 6:125–150

Danilov S, Kivman G, Schröter J (2005) Evaluation of an eddy-permitting finite-element ocean model in the North Atlantic. Ocean Model 10:35–49

Evensen G (1994) Sequential data assimilation with a nonlinear quasi-geostrophic model using Monte Carlo methods to forecast error statistics. J Geophys Res 99(C5):10143–10162

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

Evensen G (2004) Sampling strategies and square root analysis schemes for the EnKF. Ocean Dynamics 54:539–560

Gill AE (1982) Atmosphere–ocean Dynamics. Academic, New York

Hamill TM, Whitaker JS, Snyder C (2001) Distance-dependent filtering of background error covariance estimates in an ensemble Kalman filter. Mon Weather Rev 129:1790–2776

Houtekamer PL, Mitchell HL (1998) Data assimilation using an ensemble Kalman filter technique. Mon Weather Rev 126:796–811

Houtekamer PL, Mitchell HL (2001) A sequential ensemble Kalman filter for atmospheric data assimilation. Mon Weather Rev 129:123–137

Iskandarani M, Haidvogel DB, Levin JC (2003) A three-dimensional spectral element method for the solution of the hydrostatic primitive equations. J Comput Phys 186:397–425

Keppenne CL, Rienecker MM (2002) Initial testing of a massively parallel ensemble Kalman filter with the poseidon isopycnal ocean circulation model. Mon Weather Rev 130:2951–2965

Keppenne CL, Rienecker MM, Kurkowski NP, Adamec DA (2005) Ensemble Kalman filter assimilation of temperature and altimeter data with bias correction and application to seasonal prediction. Nonlinear Process Geophys 12:491–503

Levitus S, Boyer T (1994) World ocean atlas. In: Temperature, vol 4. NOAA, Washington, District of Columbia

Levitus S, Burgett R, Boyer T (1994) World ocean atlas. In: Salinity, vol 3. NOAA, Washington, District of Columbia

Mitchell HL, Houtekamer PL, Pellerin G (2002) Ensemble size, balance, and model-error representation in an ensemble Kalman filter. Mon Weather Rev 130:2791–2808

Nerger L, Hiller W, Schröter J (2005a) A comparison of error subspace Kalman filters. Tellus 57A:715–735

Nerger L, Hiller W, Schröter J (2005b) PDAF—the parallel data assimilation framework: experiences with Kalman filtering. In: Zwieflhofer W, Mozdzynski G (eds) Use of high performance computing in meteorology. Proceedings of the 11th ECMWF Workshop, World Scientific, pp 63–83

Nerger L, Danilov S, Kivman G, Hiller W, Schröter J (2006) Data assimilation with the ensemble Kalman filter and the SEIK filter applied to a finite element model of the North Atlantic. J Mar Syst (in press)

Ott E, Hunt B, Szunyogh I, Zimin AV, Kostelich EJ, Corazza M, Kalnay E, Patil DJ, Yorke JA (2004) A local ensemble Kalman filter for atmospheric data asimilation. Tellus 56A:415–428

Pain CC, Piggot MD, Goddard AJH, Fang F, Gorman GJ, Marshall DP, Eaton MD, Power PW, Oliveira CRED (2005) Three-dimensional unstructured mesh ocean modelling. Ocean Model 10:5–33

Penduff T, Brasseur P, Testut CE, Barnier B, Verron J (2002) A four-year eddy-permitting assimilation of sea-surface temperature and altimetric data in the South Atlantic Ocean. J Mar Res 60:805–833

Pham DT (2001) Stochastic methods for sequential data assimilation in strongly nonlinear systems. Mon Weather Rev 129:1194–1207

Pham DT, Verron J, Gourdeau L (1998a) Singular evolutive Kalman filters for data assimilation in oceanography. C R Acad Sci Ser II 326(4):255–260

Pham DT, Verron J, Roubaud MC (1998b) A singular evolutive extended Kalman filter for data assimilation in oceanography. J Mar Syst 16:323–340

Tippett MK, Anderson JL, Bishop CH, Hamill TM, Whitaker JS (2003) Ensemble square root filters. Mon Wea Rev 131:1485–1490