Error Estimation Using Wavelet Analysis for Data Assimilation: EEWADAi*

LELAND JAMESON AND TAKUJI WASEDA

Frontier Research System for Global Change, and International Pacific Research Center,+
University of Hawaii, Honolulu, Hawaii

(Manuscript received 3 May 1999, in final form 16 November 1999)

ABSTRACT

A new method is presented for estimating numerical errors in simulations as a function of space and time. This knowledge of numerical errors can provide critical information for the effective assimilation of external data. The new method utilizes wavelet analysis for the detection of deviation from low-order polynomial structure in the computational data indicating regions of the domain where relatively large numerical errors will occur. This wavelet-based technique has a very low computational cost, and in practice the cost can be considered negligible compared to the computational cost of the simulation. It is proposed here that this be used in the field of data assimilation for fast and efficient assimilation of external data, and a numerical example illustrating that the new method performs better than the existing method of optimal interpolation is given.

1. Introduction

In the field of assimilation the key to success is knowing the errors in either the data or the computed values as a function of time and spatial location. It is rare that one will know such spatial and time information about the errors in the data, but wavelet analysis can give us such information on the errors of the computed values, and hence a mechanism to determine an appropriate weighting between the data and computed values.

We will illustrate the use of wavelet analysis in a new technique to assimilate data into a computational model. The new method is fast and efficient, and we believe that it surpasses other methods of comparable computational cost. The new method is called EEWADAi (Error Estimation using Wavelet Analysis for Data Assimilation). We should explain that the acronym EEWADAi for our new technique has meaning in the Japanese language. It means roughly, ‘‘a good topic of conversation,’’ which we hope our method will become.

The basic idea behind the new method is to use wavelet analysis to detect errors in the computational domain. Based on this fast real-time error analysis, we decide how to weight computed information against observed information. Simply stated, if the computational errors are large in a certain region of the computational domain at certain time, then the observed data will receive a relatively large weight. On the other hand, if the computational errors are small, then the computed information will be considered relatively reliable in this region and will be assigned a small weight.

Conventional sequential data assimilation can be understood as a weighted averaging of the predicted model state and the measurement:

$$\mathbf{x} = \mathbf{x}^{-} + \mathsf{K}[\mathbf{y} - \mathsf{E}\mathbf{x}^{-}], \tag{1}$$

where $\mathbf{x}$ is the model state vector, $\mathbf{y}$ is the observations, E is the observation matrix, and K is the weighting matrix known as the Kalman gain (see, e.g., Wunsch 1996). The art of the sequential data assimilation scheme is the choice of the weighting matrices. A straightforward minimization of the analyzed model error covariance matrix with linear approximation of the model dynamics will result in the so-called Kalman filtering (KF); KF requires a computation of the error evolution equation, the matrix Riccati equation of the Kalman filter. Despite its long time availability of the scheme (Kalman 1960), its full implementation in ocean modeling has been limited for the following reasons: 1) its high demand on the computational resources [storage of $O(n^2)$, integration of $O(2n)$ stepping forward in time], 2) its linear assumption of the model dynamics and the Gaussian statistics, and 3) the lack of precise knowledge of the model error. These limitations have lead modelers to invent a variety of new technology for practical im-

* School of Ocean and Earth Science and Technology Contribution Number 4915 and International Pacific Research Center Contribution Number 23.

+ The International Pacific Research Center is partly sponsored by the Frontier Research System for Global Change.

Corresponding author address: Dr. Takuji Waseda, IPRC–SOEST, University of Hawaii, MSB213, 1000 Pope Road, Honolulu, HI 96822.
E-mail: twaseda@soest.hawaii.edu

plementation: extended Kalman filter (EKF), ensemble Kalman filter, optimal interpolation (OI), and nudging (N). EKF improves the defect of KF by applying linearization at each time step about the most recent solution. However, because the computational requirements are still high, suggestion was made to solve the error prediction equation using the Monte Carlo method, since only a few hundred model states may be sufficient for reasonable statistical convergence (Evensen 1994). A more drastic simplification can be achieved by OI allowing information to influence only the nearby observation points (Ezer and Mellor 1994). Finally, the most heuristic and simple scheme, N, chooses an ad hoc weighting function which decays in time and space (Malanotte-Rizzoli and Holland 1986). In view of applying the assimilation scheme to OGCM, where the size of the state vector $\mathbf{n}$ is of $O(10^8)$ and larger, a practical choice is naturally OI or N. The principal defect of these method is that it does not reflect the evolution or the propagation of model error in space and time. This lead us to think of a diagnostic way of error detection rather than the prognostic way of solving the Riccati equation. EEWADAi thus provides us with a tool to diagnostically detect the error at each time step and use the detected error to determine the weighting matrix $\mathbf{K}$.

The new method is very straightforward. The numerical values of the matrix $\mathbf{K}$ are determined by wavelet analysis and are simply a function of the magnitude of the wavelet coefficients in the same area of the domain where the assimilation occurs. In this manner, the wavelet analysis detects the areas of the domain where large computational errors occur and these areas will have relatively large values of $\mathbf{K}$. By contrast, in areas of the domain where the computational errors are small, the assimilation parameter will be relatively small.

This paper begins by introducing the fundamentals of wavelet analysis and how it is applied in the discrete world of a computer in section 2. Next, a review is given on how numerical methods are fundamentally created in section 3. It is the relationship between how wavelet methods are constructed and how numerical methods are constructed that is fundamental in obtaining an estimate of numerical errors. After our discussion of numerical errors obtained by wavelet analysis, we propose ways to obtain estimates of local error variance in section 4. In section 5 we show our estimate of numerical error variance can be used to construct a Kalman gain matrix for use in Kalman filtering. Finally, in section 6, we test our new method on a Kuroshio model and compare the result to that obtained by a traditional method—optimal interpolation. A conclusion follows.

2. Wavelet analysis

Possibly the most instructive way to think of wavelets is in contrast to traditional analysis techniques such as Fourier analysis. With Fourier analysis we analyze discrete or continuous data using basis functions which are global, smooth, and periodic. This analysis yields a set of coefficients, say, a_k, which gives the amount of energy in the data at frequency k. Wavelet analysis, by contrast, analyzes data with basis functions which are local, slightly smooth, not periodic, and which vary with respect to scale and location. Wavelet analysis thereby produces a set of coefficients $b_{j,k}$ which give the amount of energy in the data at scale j and location k. Wavelet analysis can serve as a good complement to Fourier analysis. In fact, data which are efficiently analyzed with Fourier analysis often are not efficiently analyzed with wavelet analysis and the opposite situation also holds.

For our purposes here we will confine our discussion to the so-called orthogonal wavelets and specifically the Daubechies family of wavelets. The orthogonality property leads to a clear indication when data deviates from a low-order polynomial, the importance of which will become clear when we discuss numerical methods.

a. Theoretical background in the continuous world

To define Daubechies-based wavelets, see Daubechies (1988) and Erlebacher et al. (1996), consider the two functions $\phi(x)$, the scaling function, and $\psi(x)$, the wavelet. The scaling function is the solution of the dilation equation,

$$\phi(x) = \sqrt{2}\sum_{k=0}^{L-1} h_k \phi(2x - k), \tag{2}$$

which carries the name ‘‘dilation equation’’ since the the independent variable x appears alone on the left-hand side but is multiplied by 2, or dilated, on the right-hand side. One also requires the scaling function $\phi(x)$ be normalized: $\int_{-\infty}^{co} \phi(x)\, dx = 1$. The wavelet $\psi(x)$ is defined in terms of the scaling function,

$$\psi(x) = \sqrt{2}\sum_{k=0}^{L-1} g_k \phi(2x - k). \tag{3}$$

One builds an orthonormal basis from $\phi(x)$ and $\psi(x)$ by dilating and translating to get the following functions:

$$\phi_k^j(x) = 2^{-j/2}\phi(2^{-j}x - k) \quad \text{and} \tag{4}$$

$$\psi_k^j(x) = 2^{-j/2}\psi(2^{-j}x - k), \tag{5}$$

where $j, k \in Z$. The dilation parameter is j, and k is the translation parameter.

It is usual to let the spaces spanned by $\phi_k^j(x)$ and $\psi_k^j(x)$ over the parameter k, with j fixed, be denoted by V_j and W_j respectively,

$$V_j = \sum_{k\in Z}^{\text{span}} \phi_k^j(x), \tag{6}$$

$$W_j = \sum_{k\in Z}^{\text{span}} \psi_k^j(x). \tag{7}$$

The spaces V_j and W_j are related by,

$$\cdots \subset V_1 \subset V_0 \subset V_{-1} \subset \cdots, \quad \text{and} \tag{8}$$

$$V_j = V_{j+1} \bigoplus W_{j+1}, \tag{9}$$

where the notation $V_0 = V_1 \oplus W_1$ indicates that the vectors in V_1 are orthogonal to the vectors in W_1 and the space V_0 is simply decomposed into these two component subspaces.

The coefficients $H = \{h_k\}_{k=0}^{L-1}$ and $G = \{g_k\}_{k=0}^{L-1}$ are related by $g_k = (-1)^k h_{L-k}$ for $k = 0, \ldots, L - 1$. All wavelet properties are specified through the parameters H and G. If one's data are defined on a continuous domain such as $f(x)$, where $x \in R$ is a real number, then one uses $\phi_k^j(x)$ and $\psi_k^j(x)$ to perform the wavelet analysis. If, on the other hand, one's data are defined on a discrete domain such as $f(i)$, where $i \in Z$ is an integer, then the data are analyzed, or filtered, with the coefficients H and G. In either case, the scaling function $\phi(x)$ and its defining coefficients H detect localized low-frequency information, that is, they are low-pass filters, and the wavelet $\psi(x)$ and its defining coefficients G detect localized high-frequency information, that is, they are high-pass filters. Specifically, H and G are chosen so that dilations and translations of the wavelet, $\psi_k^j(x)$, form an orthonormal basis of $L^2(R)$ and so that $\psi(x)$ has M vanishing moments, which determines the accuracy. In other words, $\psi_k^j(x)$ will satisfy

$$\delta_{kl}\delta_{jm} = \int_{-\infty}^{\infty} \psi_k^j(x)\psi_l^m(x)\, dx, \tag{10}$$

where δ_{kl} is the Kronecker delta function, and the accuracy is specified by requiring that $\psi(x) = \psi_0^0(x)$ satisfy

$$\int_{-\infty}^{\infty} \psi(x)x^m\, dx = 0, \tag{11}$$

for $m = 0, \ldots, M - 1$. This statement on accuracy can be explained in an alternative manner. Recall that within the scaling function subspaces V_j one can reconstruct low-order polynomials exactly up to a given order so that $x^0, x^1, \ldots, x^{M-1}$ can be represented exactly by appropriately choosing scaling function coefficients. Accordingly, the subspace W_j (which is orthogonal to V_j) and consequently the basis elements $\psi_j(x)$ of W_j will be orthogonal to all elements contained in V_j, such as these low-order polynomials. So, one can see that by specifying the number of vanishing moments of the wavelets, one has in effect specified the number of polynomials which can be represented exactly and hence the numerical accuracy of the method.

For representing functions in $L^2(R)$ one can see from the above expressions that for any function $f(x) \in L^2(R)$ there exists a set $\{d_{jk}\}$, such that

$$f(x) = \sum_{j\in Z} \sum_{k\in Z} d_k^j \psi_k^j(x), \tag{12}$$

where

$$d_k^j = \int_{-\infty}^{\infty} f(x)\psi_k^j(x)\, dx. \tag{13}$$

The two sets of coefficients H and G are known as quadrature mirror filters. For Daubechies wavelets the number of coefficients in H and G, or the length of the filters H and G, denoted by L, is related to the number of vanishing moments M by $2M = L$. For example, the famous Haar wavelet is found by defining H as $h_0 = h_1 = 1$. For this filter, H, the solution to the dilation Eq. (2), $\phi(x)$, is the box function: $\phi(x) = 1$ for $x \in [0, 1]$ and $\phi(x) = 0$ otherwise. The Haar function is very useful as a learning tool, but because of its low order of approximation accuracy and lack of differentiability, it is of limited use as a basis set. The coefficients H needed to define compactly supported wavelets with a higher degree of regularity can be found in Daubechies (1988). As is expected, the regularity increases with the support of the wavelet. The usual notation to denote a Daubechies-based wavelet defined by coefficients H of length L is D_L.

b. Practical implementation in the discrete world

Naturally, infinite sums and integrals are meaningless when one begins to implement a wavelet expansion on a computer. One must find appropriate ways to implement discrete counterparts to the continuous operations, which were outlined in the previous section. That is, nothing is continuous on a computer, and since the original wavelet analysis was derived in the continuous world of differential and integral mathematics it is necessary to discuss a discrete version of the above continuous theory. Generally, operations such as integration are easily approximated with an appropriate-order quadrature formula, but one would like to use as few quadratures as possible to limit the number of approximations that are made. We will see in this section how we can easily perform all the wavelet decompositions with relatively few approximations.

In a continuous wavelet expansion, functions with arbitrarily small-scale structures can be represented. In practice, however, there is a limit as to how small the smallest structure can be depending on, for example, the numerical grid resolution or the sampling frequency in a signal processing scenario. Hence, on a computer an approximation would be constructed in a finite space such as

$$V_0 = W_1 \oplus W_2 \oplus \cdots \oplus W_J \oplus V_J,$$

with the approximation being

$$P_{V_0} f(x) = \sum_{k\in Z} s_k^J \phi_k^J(x) + \sum_{j=1}^{J} \sum_{k\in Z} d_k^j \psi_k^j(x), \tag{14}$$

with

$$d_k^j = \int_{-\infty}^{\infty} f(x)\psi_k^j(x)\,dx, \qquad s_k^J = \int_{-\infty}^{\infty} f(x)\phi_k^J(x)\,dx$$

utilizing orthogonality. Within this expansion, the scale $j = 0$ is arbitrarily chosen as the finest scale required, and scale J would be the scale at which a kind of local average, $\phi_k^J(x)$, provides sufficient large-scale information, that is, the first term in Eq. (14) provides the local mean around which the function oscillates.

One must also limit the range of the location parameter k. Assuming periodicity of $f(x)$ implies periodicity on all wavelet coefficients, s_k^j and d_k^j, with respect to k. For the nonperiodic case, since k is directly related to the location, a limit is imposed on the values of k when the location being addressed extends beyond the boundaries of the domain.

The wavelet decomposition matrix is the matrix embodiment of the dilation equation, Eq. (2), defining the scaling function and the accompanying equation defining the wavelet, Eq. (3). The following two recurrence relations for the coefficients, s_k^j and d_k^j, in Eq. (14) are given as

$$s_k^j = \sum_{n=1}^{L} h_n s_{n+2k-2}^{j-1}, \qquad d_k^j = \sum_{n=1}^{L} g_n s_{n+2k-2}^{j-1},$$

as obtained from Eqs. (2)–(3), and we recall that h_n refers to the chosen filter while we have $g_n = -(-1)^n h_{L-n}$.

Denote the decomposition matrix embodied by these two equations, assuming periodicity, by $\mathsf{P}_N^{j,j+1}$, where the matrix subscript denotes the size of the square matrix while the superscripts indicate that P is decomposing from scaling function coefficients at scale j to scaling function and wavelet function coefficients at scale $j + 1$, that is, $\mathsf{P}_N^{j,j+1}$ maps $\mathbf{s}_j$ onto $\mathbf{s}_{j+1}$ and $\mathbf{d}_{j+1}$:

$$\mathsf{P}_N^{j,j+1}\colon [\mathbf{s}_j] \to \begin{bmatrix} \mathbf{s}_{j+1} \\ \mathbf{d}_{j+1} \end{bmatrix}, \tag{15}$$

where $\mathbf{s}_j$ refers to the vector containing the coefficients at scale j. Note that the vectors at scale $j + 1$ are half as long as the vectors as scale j.

Suppose, for illustration, the wavelet being used is the four-coefficient D_4 wavelet, and that one wants to project from eight scaling function coefficients at scale j to four scaling function coefficients at scale $j + 1$ and four wavelet coefficients at scale $j + 1$. The decomposition matrix for the case of periodic boundary conditions, $\mathsf{P}_8^{j,j+1}$, thus becomes

$$\mathsf{P}_8^{j,j+1} \equiv \begin{bmatrix} h_1 & h_2 & h_3 & h_4 & 0 & 0 & 0 & 0 \\ 0 & 0 & h_1 & h_2 & h_3 & h_4 & 0 & 0 \\ 0 & 0 & 0 & 0 & h_1 & h_2 & h_3 & h_4 \\ h_3 & h_4 & 0 & 0 & 0 & 0 & h_1 & h_2 \\ g_1 & g_2 & g_3 & g_4 & 0 & 0 & 0 & 0 \\ 0 & 0 & g_1 & g_2 & g_3 & g_4 & 0 & 0 \\ 0 & 0 & 0 & 0 & g_1 & g_2 & g_3 & g_4 \\ g_3 & g_4 & 0 & 0 & 0 & 0 & g_1 & g_2 \end{bmatrix}, \tag{16}$$

where the periodicity is reflected in the coefficients being wrapped around. If the boundary conditions are not periodic, then one must extend the data and the scaling function coefficients at every scale in a ‘‘smooth’’ manner. This involves extending the scaling function data with a polynomial one order higher than the number of vanishing moments of the wavelet. But, for the purpose of understanding the method, it is sufficient to study the periodic matrix such as the one that we have created above. It is a matrix such as $\mathsf{P}^{j,j+1}$ that is applied directly to the data and then directly to each level of scaling function coefficients. To be perfectly correct one would first approximate the scaling function coefficients at the finest scale using the raw data; however, in practice it seems to make very little difference if one simply considers the raw data to be the scaling function coefficients. So, for our purposes here we will simply use the raw data as the scaling function coefficients on the finest scale. The repeated application of the matrix $\mathsf{P}^{j,j+1}$ yields the wavelet coefficients at the various scales, and it is these wavelet coefficients that provide a guide to the errors committed during the numerical calculation, as will be further illustrated in the next paragraph and explained precisely in the next section.

Let us consider that the raw data are given and it is assumed to be the scaling function coefficients on the finest scale, $\mathbf{s}_0$. One wavelet decompostion yields the scaling function coefficients and wavelet coefficients at scale $j = 1$, $\mathbf{s}_1$ and $\mathbf{d}_1$. A second application of the wavelet decomposition matrix will yield the vectors $\mathbf{s}_2$ and $\mathbf{d}_2$. It is the vectors $\mathbf{d}_1$, $\mathbf{d}_2$, . . . that yield the critical information on the numerical errors. If, for example, one sees that the values of the $\mathbf{d}_1$ are relatively large in the middle of the vector, then it is clear that within this one-dimensional vector the largest errors will be in the middle of the one-dimensional domain from which this vector was derived. What we care about most is the relative errors being committed, but we also have some interest in the absolute errors, the subject of the next section.

3. Estimating computational errors

As outlined in the previous section, scaling functions are designed to approximate low-order polynomials exactly up to a given order, and wavelets are orthogonal to these same low-order polynomials. Any deviation from low-order polynomial structure in a computational domain can then be detected by wavelet analysis (see Jameson 1998). This measure of deviation from low-order polynomial structure is exactly what is needed to measure computational error. The reason for this is that fundamentally all nonspectral numerical schemes are constructed from low-order algebraic polynomials, and such schemes are exact if the data fall exactly on a low-order polynomial. Actually, spectral methods also follow this same rule, but in the case of Fourier spectral methods, the polynomials are trigonometric and not algebraic and are global instead of local. To be precise, let us review the fundamentals of how numerical methods are constructed and how wavelet analysis can detect errors in a computational scheme.

Numerical schemes for the approximation of partial differential equations on a computer provide a mechanism for taking one set of N numbers to another set of N numbers. To make this transition from one set to the next set, we must take derivatives but the data are a set of points that are not connected. We must, therefore, choose some type of function with which we can connect these points so that we can take a derivative. There are many choices that can be made, but fundamentally we are always working with some type of polynomial be it algebraic, trigonometric, or other.

Simply said, one can generate differencing coefficients by first interpolating a polynomial of any order through a set of data, then differentiating of this polynomial, and evaluating it at a grid point. As long as the number of grid points exceeds the order the polynomial by one, then the interpolation is unique and the differencing coefficients are likewise unique.

Interpolation with algebraic polynomials is probably the most common form of interpolation, and it is from this type of interpolation that common uniform grid finite difference methods can be found. The well-known Lagrange interpolation formula for algebraic interpolation is

$$A_j(x) = \frac{\prod_{k=0,k\neq j}^{n} (x - x_k)}{\prod_{k=0,k\neq j}^{n} (x_j - x_k)}, \tag{17}$$

where $A_j(x_k) = \delta_{jk}$. For given values $w_0, w_1, \ldots, w_n$, the polynomial

$$p_n(x) = \sum_{k=0}^{n} w_k A_k(x) \tag{18}$$

is in p_n (the space of a polynomial of order n) and takes on these values at the points x_i:

$$p_n(x_k) = w_k, \tag{19}$$

for $k = 0, 1, \ldots, n$, and it is the truncation error of this interpolation that determines the order accuracy that one obtains when differentiating.

Recall that the remainder for algebraic polynomial interpolation is

$$f(x) - p_n(x) = \frac{(x - x_0)(x - x_1)\cdots(x - x_n)}{(n + 1)!} f^{(n+1)}(\xi), \tag{20}$$

where ξ lies between the smallest and the largest x_i. Note that when one builds differencing operators after interpolation that the term

$$f^{(n+1)}(\xi) \tag{21}$$

is unchanged, whereas the term involving the products of the point of interest x and the grid points x_i goes from a term of the form

$$\prod_{i=0}^{n} (x - x_i) \tag{22}$$

to a term of the form

$$\prod_{i=0}^{n-1} (x - x_i). \tag{23}$$

In other words, if the computational data are exactly a polynomial of order n or less, then the remainder term is zero and there is no error in the computation.

Consider the simple case of interpolating an algebraic quadratic polynomial $p_2(x)$ to a function $f(x)$ at the grid points $x_0 < x_1 < x_2$: $p_2(x_i) = f(x_i)$, $i = 0, 1, 2$. The remainder term for some ξ, $x_0 \leq \xi \leq x_2$, is

$$f(x) - p_2(x) = (x - x_0)(x - x_1)(x - x_2)\frac{1}{3!} f^{(3)}(\xi). \tag{24}$$

Now, differentiate and evaluate at $x = x_1$ to get

$$\begin{aligned} f'(x_1) - p_2'(x_1) &= (x_1 - x_0)(x_1 - x_2)\frac{1}{3!} f^{(3)}(\xi) \\ &= Ch^2 f^{(3)}(\xi), \end{aligned} \tag{25}$$

where $h = x_1 - x_0 = x_2 - x_1$. If the grid is evenly spaced, then the differences $(x_i - x_j)$ are some integer multiple of the smallest difference, which one can denote by h. If one doubles the number of grid points, then each of the distances $(x_i - x_j)$ becomes half as large and the accuracy for this quadratic example will be 2. To be specific, if the computational data are quadratic polynomial or less, then the remainder is zero and the computation has no error.

In practice, however, computational data will rarely be an exact low-order polynomial; therefore, there is always error. The size of this computational error will depend on the deviation from low-order polynomials and can be readily measured with wavelet analysis. If, on the other hand, the data are exactly a low-order polynomial and the numerical method is ‘‘exact,’’ then wavelets of the corresponding order of accuracy will be orthogonal to the data and all wavelet coefficients will be zero indicating, correctly, that the calculation has no error.

Recall from above that the wavelet coefficients are found as the inner product of the function at hand and the corresponding wavelet,

$$d_k^j = \int_{-\infty}^{\infty} f(x)\psi_k^j(x)\, dx, \tag{26}$$

where the decay of the expansion coefficients depends on the local regularity of $f(x)$ and one can derive the following expression (see Strang and Nguyen 1996):

$$|d_k^j| \leq C2^{-j[(L+1)/2]} \max_{\xi \in [k2^{-j},(k+M-1)2^{-j}]} |f^{(M)}(\xi)|. \quad (27)$$

From Eq. (27) we find that if $f(x)$ behaves like a polynomial of an order less than M inside the small interval, then d_k^j vanishes exactly.

Thus, by considering the magnitude of d_k^j, one obtains a local measure of the deviation of the computational data from low-order polynomials. Indeed, the information given by Eq. (27) is of very local character and and provides an excellent estimate of local calculation error which can be used as an efficient method for data assimilation.

4. Error variance and model variation

Wavelet analysis can detect both the local error in a numerical calculation and the model variation. In a sense, the estimate of local error is a precise concept based on deviation from a local low-order polynomial, whereas model variation is a less precise measure of localized frequency content. Building the error variance from the error estimate is just an averaging process.

a. Wavelet dectection of error variance

As we have seen from the previous sections, wavelet analysis gives us a local measure of deviation from low-order polynomial structure in the data. Given this measure of deviation, then one has an estimate of local numerical error in the model. From these local error estimates we can easily obtain an estimate of local error variance. It is important that this variance estimate remain *local* since the ability to estimate errors locally in both space and time is one of the key strong points of EEWADAi. The first point to note is that the average error will be zero. That is, the local inner product between the wavelet and the data, indicating the local error, will certainly sometimes be positive and sometimes negative and on average will be zero. The next step is to decide which errors to include when estimating the local variance. One can average the squared errors in time or in space. From practical point of view, we will find a local average of the squared errors in space. Therefore, in order to estimate the error variance around, say, grid point (x_I, y_J, z_K) then one sum in the following manner,

$$\sigma^2_{I,J,K} = \frac{1}{(2d+1)^3} \sum_{i=I-d}^{i=I+d} \sum_{j=J-d}^{j=J+d} \sum_{k=K-d}^{k=K+d} [cw(i, j, k)]^2, \quad (28)$$

where $w(i, j, k)$ is the wavelet coefficient at grid point i, j, k. Here, d is a small number used to define a small averaging box around the point of interest, and c is a constant that is needed to scale the wavelet transform to the problem at hand.

In this manner, we can obtain a local and very computationally efficient estimate of local error variance.

b. Wavelet detection of model variation

As explained in previous sections of the paper, wavelet analysis breaks up data into local frequency components. That is, at a given physical space location one can obtain an estimate of the various scales of information present in the vicinity of this physical space location. Using the previously defined notation, we have the ‘‘frequency’’ or ‘‘scale’’ boxes $W_1, W_2, W_3, \ldots$ that contain this local frequency content information. Furthermore, ‘‘variation’’ in a model will appear as a localized oscillation at some scale and this information will appear as a large wavelet coefficient in one of the frequency boxes. That is, there is a one-to-one correspondence between model variation at a given scale and the wavelet energy at that same scale. In fact, variation is exactly what wavelet analysis detects.

c. When variation and error are not *the same*

In a word, where there is *numerical error* there will be variation, but variation does not imply numerical error. Roughly speaking, one can make the argument based on scale information. For example, if one observes wavelet energy in the finest scale box, W_1, then this energy will indicate that the grid point density for the numerical method in a given region of the domain is not sufficient and that numerical errors are commited in this region. Certainly this same W_1 box energy indicates local high-frequency information or local variation. On the other hand, if one observes energy in the coarser scale box W_3, then it will not necessarily indicate an insufficient gridpoint density but, again, it will indicate that variation occurs at scale 3. To be more precise, suppose that in the vicinity of grid point x_k that energy is present in box W_3 but not in box W_1, this will indicate variation but not numerical error. On the other hand, if energy is present in box W_1, then this will indicate both numerical error and variation.

In summary, we note that other techniques such as OI measure only model variation, and this measure of variation is far less precise than the model variation measure that is given by wavelet analysis. In addition, wavelet analysis gives a precise measure of the error committed in the numerical calculation. Both of these estimates, the error estimate and the variation estimate, make wavelet analysis a very powerful and useful tool.

5. Reduced Kalman gain

In this section, we describe the reduction of the Kalman gain for an effective implementation in the large numerical model. The goal is to find a reduced Kalman gain used for correcting the model output in the following formula:

$$\mathbf{x} = \mathbf{x}^- + \mathsf{K}[\mathbf{y} - \mathsf{E}\mathbf{x}^-], \quad (29)$$

where $\mathbf{x}$ is the analyzed model state vector, $\mathbf{x}^-$ is the first-guess model state vector of length M, $\mathbf{K}$ is the Kalman gain, $\mathbf{y}$ is the observation vector of length N, and $\mathbf{E}$ is the observation matrix. The size M will be substantially smaller than the full model state dimension, and the actual reduction we used in this study for the TOPEX/Poseidon altimeter twin experiment will be presented later.

For simplicity, we assume that the observation of some model variables are available at the exact model grid point. Then the $(N + M)$ by M observation matrix $\mathbf{E}$ can be blocked as the following:

$$\mathbf{E} = [\mathbf{O} \quad \mathbf{I}], \tag{30}$$

where $\mathbf{O}$ is N by M null matrix and $\mathbf{I}$ is an identity matrix of size N by N, satisfying

$$\mathbf{y} = \mathbf{E}\mathbf{x}. \tag{31}$$

The reduced Kalman gain is found, as usual, by minimization of the error covariance,

$$e = \langle(\mathbf{x} - \mathbf{x}_t)(\mathbf{x} - \mathbf{x}_t)^{\mathrm{T}}\rangle, \tag{32}$$

with respect to $\mathbf{K}$, substituting (29) to (32); $\mathbf{x}_t$ is the true state and superscript T denotes transposition. The following formula is readily derived:

$$\mathbf{K}[\mathbf{E}\mathbf{P}\mathbf{E}^{\mathrm{T}} + \mathbf{R}] = \mathbf{P}\mathbf{E}^{\mathrm{T}}, \tag{33}$$

where $\mathbf{P}$ is the $(M + N)$ by $(M + N)$ model error covariance matrix with diagonal element σ_p^2, and $\mathbf{R}$ is the N by N observation error covariance matrix. Typically, $\mathbf{R}$ is assumed to be white noise:

$$\mathbf{R} = \sigma_r^2\mathbf{I}, \tag{34}$$

where $\mathbf{I}$ is N by N identity matrix, and σ_r^2 is the variance of the observation error. Furthermore, $\mathbf{P}$ will be blocked for convenience:

$$\mathbf{P} = \begin{pmatrix} \mathbf{P}_{11} & \mathbf{P}_{12} \\ \mathbf{P}_{21} & \mathbf{P}_{22} \end{pmatrix}, \tag{35}$$

where $\mathbf{P}_{11}$ is M by M matrix, $\mathbf{P}_{12}$ is M by N matrix, $\mathbf{P}_{21}$ is N by M matrix, and $\mathbf{P}_{22}$ is N by N matrix. Here, $\mathbf{P}_{11}$ represents covariance among model data to be corrected, and $\mathbf{P}_{22}$ represents covariance among model data at the observation points. Also, $\mathbf{P}_{12} = \mathbf{P}_{21}^{\mathrm{T}}$, and they represent covariance between data to be corrected and data at the observation points. The resulting Kalman gain can be expressed as

$$\mathbf{K} = \begin{pmatrix} \mathbf{K}_{11} \\ \mathbf{K}_{21} \end{pmatrix} = \begin{pmatrix} \mathbf{P}_{12}(\mathbf{P}_{22} + \mathbf{R})^{-1} \\ \mathbf{P}_{22}(\mathbf{P}_{22} + \mathbf{R})^{-1} \end{pmatrix}, \tag{36}$$

where $\mathbf{K}_{11}$ is M by N and $\mathbf{K}_{21}$ is N by N matrices.

Using (36), we can derive the reduced Kalman gain used in a variety of data assimilation schemes: we will summarize below, the optimal interpolation scheme, the nudging scheme, and EEWADAi. The notable difference of EEWADAi to the other methods is the modeling of the model error covariance matrix, which we show below.

a. *Optimal interpolation (OI) scheme*

When the model error covariance matrix is replaced by the model anomaly covariance matrix, it is called OI. Thus the model error covariance matrix $\mathbf{P}$ will typically take the form

$$p_{ij} = \sigma_m^2 \exp[-(k_x\delta x_{ij})^2 - (k_y\delta y_{ij})^2 - (\omega_t\delta t_{ij})^2], \tag{37}$$

where σ_m^2 represents the variance of the model output anomaly for each grid point; δx_{ij}, δy_{ij}, δt_{ij}, are the distance and time interval between data points; and k_x, k_y, and ω_t represents the spatial and timescale of the covariance. The Kalman gain can be derived from (36), substituting (37),

$$k_{ij} = p_{ik}(p - r)_{kj}^{-1}. \tag{38}$$

The length scales k_x and k_y were set to about 60 to 90 km and ω_t to 5 days. In Figs. 1a,b we show an example of the mean SSH and the variance of the SSH as used in OI.

b. *Nudging scheme (N)*

The model error covariance matrix is the same as in the case of OI. When $\mathbf{P}_{22}$ is neglected in (36), and defining $\alpha = \sigma_m^2/\sigma_r^2$, the Kalman gain can be expressed explicitly as

$$\mathbf{K}_{11} = \alpha\frac{1}{\sigma_m^2}\mathbf{P}_{12}, \qquad \mathbf{K}_{21} = \alpha\frac{1}{\sigma_m^2}\mathbf{P}_{22}. \tag{39}$$

Then, each element of $\mathbf{K}_{11}$, $\mathbf{K}_{21}$ will take the familiar form of nudging:

$$k_{ij} = \alpha \exp[-(k_x\delta x_{ij})^2 - (k_y\delta y_{ij})^2 - (\omega_t\delta t_{ij})^2]. \tag{40}$$

c. *EEWADAi*

Finally, we will present the new assimilation scheme that utilizes the wavelet detected error and the variance to estimate the magnitude of the diagonal element of the error covariance matrix, σ_p^2. We can replace σ_p^2 with the estimate of the error variance as described earlier in (28):

$$\begin{aligned} \sigma_p^2 &= C\sigma_{I,J,K}^2 \\ &= \frac{C}{(2d + 1)^3}\sum_{i=I-d}^{i=I+d}\sum_{j=J-d}^{j=J+d}\sum_{k=K-d}^{k=K+d}[cw(i, j, k)]^2, \end{aligned} \tag{41}$$

where C is a scaling constant to be found. The nondiagonal elements of the error covariance matrix will be provided assuming a Gaussian distribution similar to that used in (37).

An alternative that will make a connection to the more familiar optimal interpolation method is to empirically relate the wavelet detected numerical error, $w(x, y)$ to the model anomaly variance, σ_m^2, both of which can be estimated from the model. For simplicity, we show in Fig. 1c the wavelet detected error taken from the mean SSH field shown in Fig. 1a. Then σ_m^2, shown in Fig.

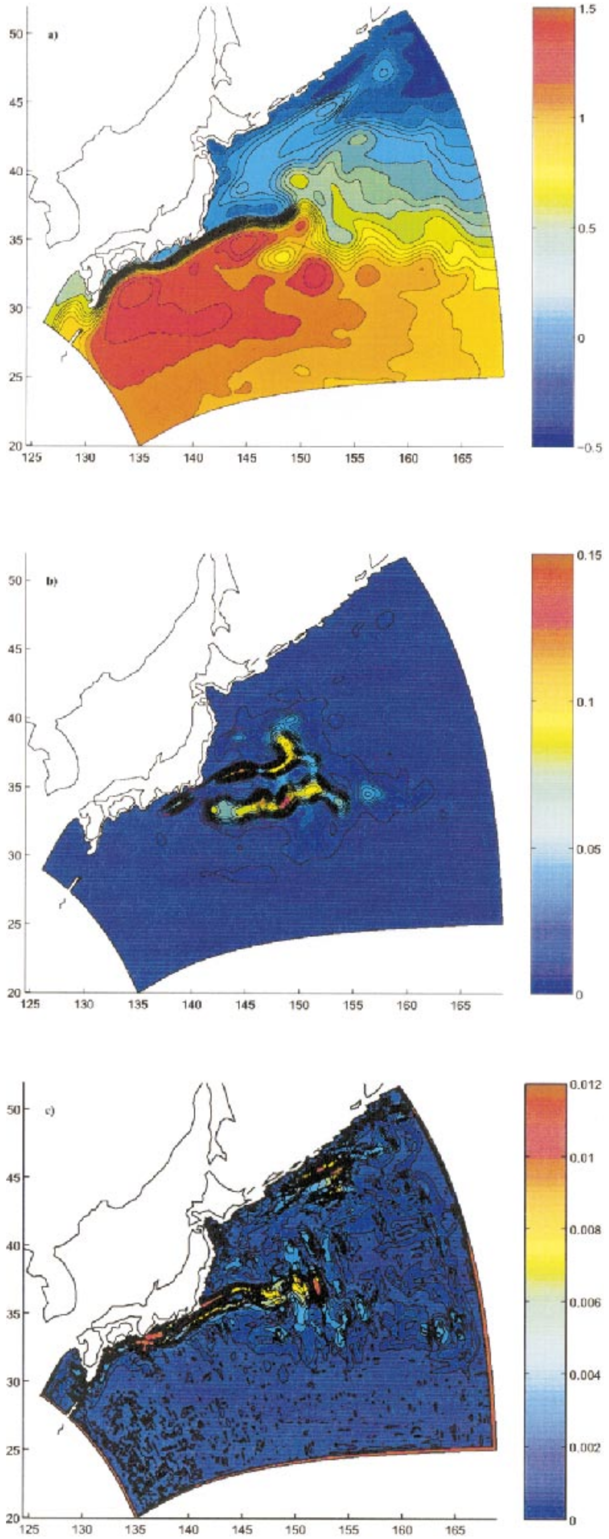

FIG. 1. (a) The mean sea surface height (m). (b) The variance of the sea surface height, σ_m^2 (m^2). (c) The wavelet coefficient w on the finest scale indicating the numerical error in the mean sea surface height data.

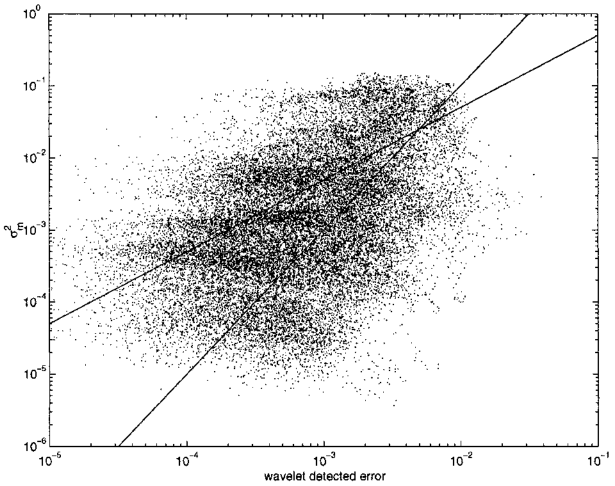

FIG. 2. Model anomaly variance as a function of wavelet detected error. The straight lines are (43) and (44).

1b, can be modeled as a function of the wavelet detected error as shown in Fig. 2:

$$\sigma_m^2 = f[w(x, y)]. \tag{42}$$

Two lines indicated in the figure are relations

$$\sigma_m^2 = c_1 w(x, y) \quad \text{and} \tag{43}$$

$$\sigma_m^2 = c_2 w(x, y)^2, \tag{44}$$

where $c_1 = 5$ and $c_2 = 1000$. We can simply replace σ_p^2 with these empirically determined σ_m^2. Finally, if we further neglect $\mathbf{P}_{22}$ as in the case of nudging, then, the Kalman gain can be expressed explicitly as

$$k_{ij} = c' w(x, y) \exp[-(k_x \delta x_{ij})^2 - (k_y \delta y_{ij})^2 - (\omega_t \delta t_{ij})^2], \tag{45}$$

where $c' = c_1/\sigma_r^2$, for (44). Equation (45) is what we used for our twin experiment. Compared to the gain used in OI, (37) and (38), the gain used in EEWADAi, therefore, will dynamically change with the diagnosed model detected error both in space and in time. In the twin experiments, the constants k_x, k_y, and ω_t were chosen to be identical to the OI case.

Figure 3 shows the actual reduced state we used, similar to that used by Ezer and Mellor (1994). Here, the

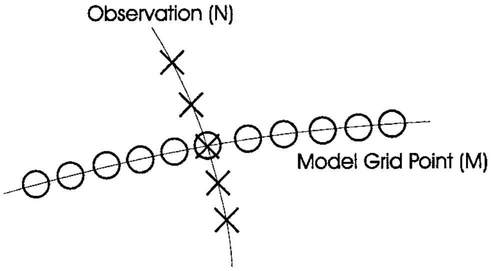

FIG. 3. Reduced model state and observation to obtain localized Kalman gain.

observation data is limited to N data points along the satellite track, indicated by crosses, and those are used to correct M model outputs, indicated by circles, in the neighboring grid points on the grid line intersecting the satellite track. This window is shifted along the track until all the satellite observation data are assimilated. A localized Kalman gain will be derived for this localized state. In this paper, we will compare the performance of OI and EEWADAi for the identical state reduction in the twin experiment. Here, M is chosen to be 11, and N is chosen to be 5.

Below we will give an example of how we actually implemented this method to assimilate the altimeter obtained SSH data into an OGCM. Much of the methodology follows the OI scheme presented by Ezer and Mellor (1994) except that the weighting function now is found by EEWADAi.

In the following equations η is the SSH, T is the temperature, S is the salinity, M represents the model result, O represents the observed data, A denotes the assimilated result also known as the analysis data, and the matrix K is the reduced Kalman gain as described earlier. Also, F is known as the correlation factor, and δT, δS is the deviation from average value of T, S:

$$\eta_{A_i} = \eta_{M_i} + \sum_k K[\eta_{Ok} - \eta_{Mk}], \tag{46}$$

$$T_{A_i} = T_{M_i} + F_T \sum_k K[\eta_{Ok} - \eta_{Mk}], \quad \text{and} \tag{47}$$

$$S_{A_i} = S_{M_i} + F_S \sum_k K[\eta_{Ok} - \eta_{Mk}], \tag{48}$$

and where

$$F_T = \frac{\overline{\delta\eta\delta T}}{\overline{(\delta\eta)^2}}, \tag{49}$$

$$F_S = \frac{\overline{\delta\eta\delta S}}{\overline{(\delta\eta)^2}}. \tag{50}$$

Equations (47) and (48) are used to extend the surface information into the interior of the fluid using a statistical correlation, F_T and F_S. For the OI, K's are found from the OI equation similar to that described by Ezer and Mellor (1994), and for the EEWADAi, K's are determined by (45).

6. Twin experiment

The proposed data assimilation scheme was implemented in the regional Kuroshio circulation model developed by Mitsudera et al. (1997). The model is a sigma coordinate primitive ocean model forced by surface wind stress, heat flux, and salinity flux with a prescribed inflow/outflow transport at the open boundary. The model dimension is 206 × 209 × 32.

The fundamental idea behind the twin experiment is that we begin two numerical simulations with different initial conditions. One of the simulations will be considered the ‘‘control run’’ and the other simulation will be used to test data assimilation schemes. In our case we will test the new wavelet-based method proposed in this paper against the traditional OI method. In the twin experiment one attempts to ‘‘push’’ the simulation run to the control run. In doing so we can compare the performance of the optimal interpolation scheme to the wavelet based method.

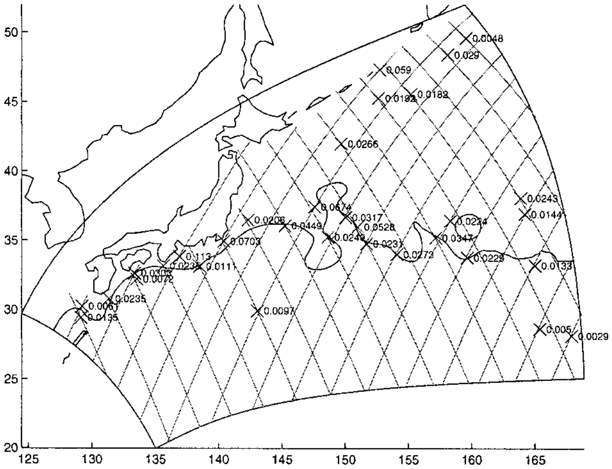

FIG. 4. Satellite paths and one possible Kuroshio meander. The wavelet coefficient with the largest maximum value along the satellite path is shown.

a. Details of experiment

The model was run for six years with annual mean forcing following 90 days of diagnostic run with Levitus (1982) temperature and salinity climatology. For the twin experiments, the output from day 1800 to day 1980 (5 to 5½ yr) was used as a control run and the output from day 1980 to 2160 (5½ to 6 yr) was used as a simulation run (model run).

Data mimicking the TOPEX/Poseidon altimeter observation were sampled from the SSH output of the control run on the model grid point near the satellite track and therefore have a lower spatial resolution (7∼15 km) than the real data (7 km). The temporal sampling rate is the exact TOPEX/Poseidon track sampling rate. For the region of interest, 32 tracks passed through at 9.91 days repeat cycle (see Fig. 4).

The necessary statistical quantities such as the correlation factors were obtained from the model run from day 1800 to 2160 (fifth year). The method used to relate the surface data into the interior of the fluid is similar to Ezer and Mellor (1994) and is described in detail in their report.

The assimilation was conducted in a sequential manner, that is, initiated whenever the observation is available and continued for a short relaxation period following the observation. The relaxation was introduced in order to smooth the shock introduced by the assimilation. The relaxation time scale, ω_t, was chosen to be

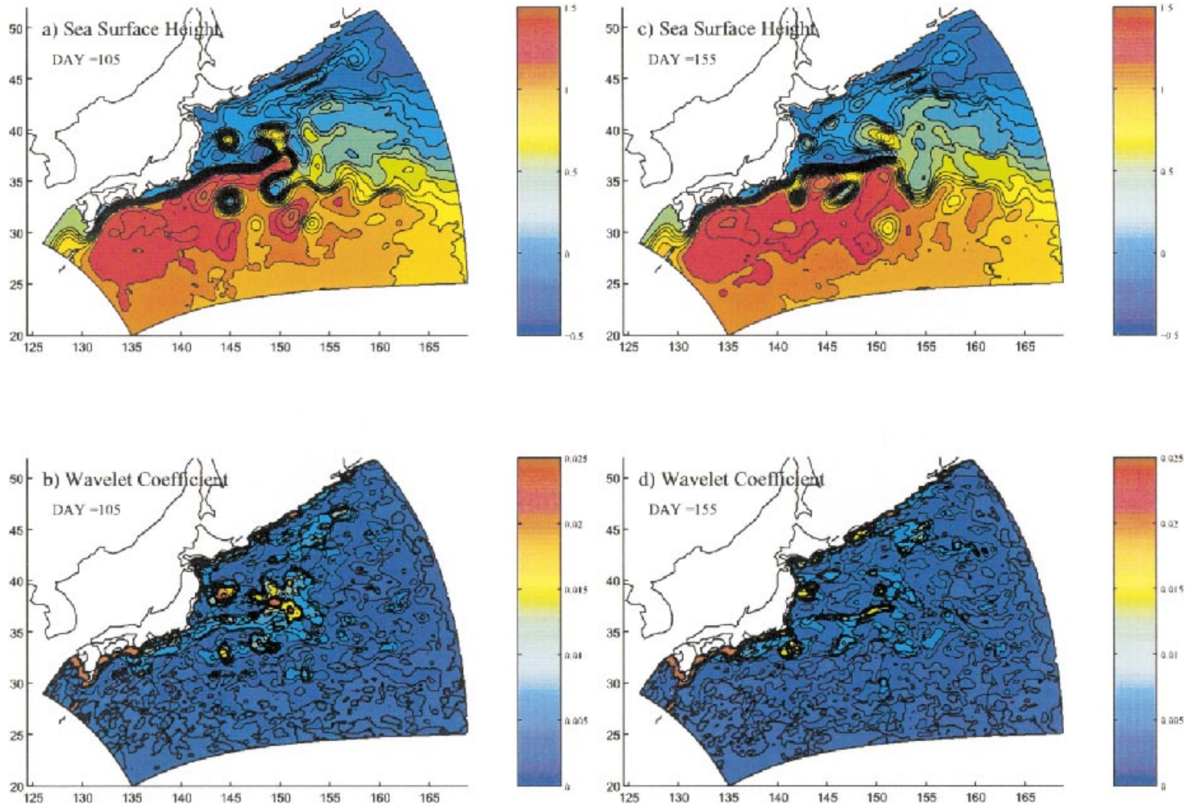

FIG. 5. (a) The Kuroshio snapshot at day 105 and (b) the corresponding wavelet-detected error. (c) The Kuroshio snapshot at day 155 and (d) the corresponding wavelet-detected error.

around 0.3 days, shorter than the track sampling interval.

b. EEWADAi, weighting, and the twin experiment

First, let us understand the nature of the error detection method of EEWADAi. Here we take two snapshots from the Kuroshio model and apply wavelet analysis to each of the two snapshots. The magnitude of the wavelet coefficients will indicate the magnitude of the computational error committed in the region of the wavelet analysis. In the regions where wavelet analysis has detected a large computational error, our assimilation method ''eewadai'' will heavily weight the data from an external source while giving a very small weight to the computed result. In other words, if we desire to assimilate satellite data into our computational model, eewadai tells us where our computed result is reliable and where it cannot be trusted.

The two snapshots from our computational model are given in Figs. 5a,c, and the associated wavelet detected computational error is given in Figs. 5b,d. Note from the two eewadai error plots that the majority of the domain is a relatively solid blue color and that in a small percentage of the domain one can observe red, which indicates that a large computational error is committed. In this case, we would choose to trust the computed result in the blue regions of the domain and we would trust, say, the satellite data in the red regions of the domain. Given this information on errors one can choose a variety of ways to execute an effective assimilation scheme, such as simply pushing the scheme or implementing a reduced Kalman gain technique.

In practice, we would be concerned not with the numerical errors in the entire domain at one time, but in the numerical errors committed on and in the vicinity of the source of external data. For example, if the source of external data is a satellite, then one would want to know the errors on or near the satellite path (see Fig. 4). In this diagram, we show the location and value of the largest absolute value of the wavelet coefficients along the satellite path. Such information would be used, say, to indicate that the location where the largest wavelet coefficient occurs is the region along the satellite path where the numerical data can be trusted the least and the satellite data would be trusted the most thus leading to a straightforward and computationally efficient data assimilation scheme. Now let us see how EEWADAi performs compared to the traditional OI technique.

c. Results of the twin experiment

We present the L_2 difference of two of the flow fields, sea surface height, and temperature at a given level

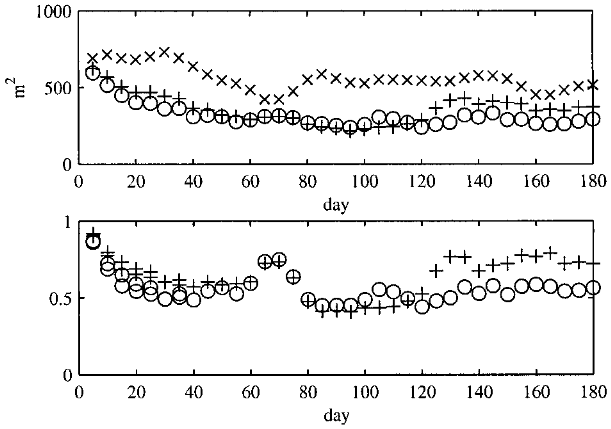

FIG. 6. The three curves in the upper plot represent the L_2 difference between the control run and the assimilation for sea surface elevation for the cases for no assimilation (×), for the wavelet-based method (○), and for the method of optimal interpolation (+). The two curves in the bottom plot are the normalized version of the wavelet method (○) and the optimal interpolation method (+) from the upper plot. These have been normalized by result where the assimlation has been turned off.

between the control run and the test run at 5-day intervals during the experiment. We can see from the two plots presented (see Figs. 6 and 7) that the two solutions are not noticeably different in quality up to day 120. However, after about day 120 one can see that the wavelet based method clearly gives a better solution than the OI method for both the temperature field and the sea surface height field. In fact, we consider the results from the later portion of the simulation to more relevant than the early part since we believe the methods have reached a kind of steady state.

We should note the wavelet method presented here has yet to be "optimized" and that even without any optimization at all it has outperformed an existing and established method. We attribute this early success to the fact the wavelet based method is actually giving a direct measure of error as a function of space and time, whereas optimal interpolation, in our view, has no "direct" relationship with error. In future versions of our method we certainly will improve the performance by analytically deriving an optimal relationship between the wavelet coefficients and the weighting matrix or by simple trial and error. In the next section, we summarize our results.

7. Conclusions

In the field of data assimilation one can summarize the weakness of current methods as saying there is insufficient knowledge of errors, either from the computational side or from the external source of data. Without knowledge of errors, one cannot hope to assimilate external data efficiently. From the simple example of incorporating real-time sea surface height data into a model run, one must know roughly which is more accurate; the height given by the numerical calculation or the height given by the external data source, say, data from a satellite. Generally, we will have some knowlegde of the satellite errors but not as a function of space and time. For example, we might know that the satellite data error is some kind of skewed Gaussian with a certain standard deviation. But, we will generally have no knowledge of how this error changes with spacial location over the ocean or with time. On the other hand, if one has some knowledge of the errors in the computational scheme, then one will have an estimate of the relative errors between the satellite data and computed data. In this paper we have proposed a method which takes a large step toward solving the problem of lack of error knowledge. Our wavelet-based method can give information on the errors committed by the computational scheme and this information coupled with a slight knowledge of the satellite errors, even if they are constant, can be used to effectively decide which information can be trusted more in which portion of the domain. The wavelet-based methods works by detecting where the numercial scheme cannot give a correct answer by estimating the deviation from a local low-order polynomial. Furthermore, the numerical cost of the wavelet error estimator is neglible. That is, one only needs to perform the wavelet analysis in the region of the domain where the new data are available and the cost of this analysis is very cheap, only a small constant times the number of grid points analyzed. This is in strong contrast to some existing methods where the cost of assimilation can dominate the calculation. Furthermore, in our numerical example where we compared the new wavelet-based method with the existing optimal

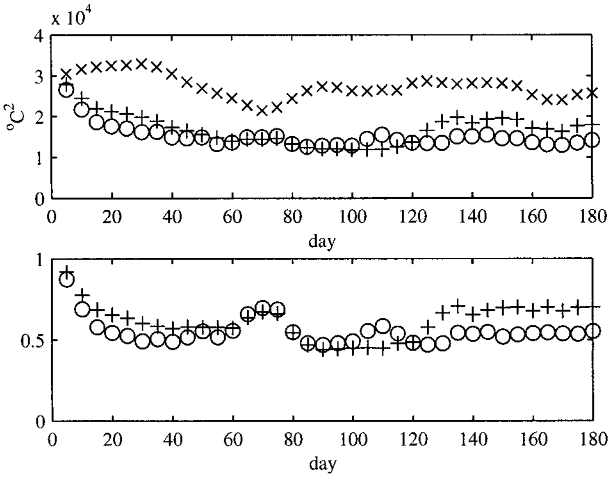

FIG. 7. The three curves in the upper plot represent the L_2 difference between the control run and the assimilation for the temperature field at a given level for the cases for no assimilation (×), for the wavelet based method (○), and for the method of optimal interpolation (+). The two curves in the bottom plot are the normalized version of the wavelet method (○) and the optimal interpolation method (+) from the upper plot. These have been normalized by result where the assimlation has been turned off.

interpolation method, we have shown that even at this early state of its existence the wavelet-based method without any optimization has outperformed the existing and certainly more mature method.

To conclude, the wavelet-based method gives a direct measure of computational errors as a function of space and time, and this information is critical to have an effective assimilation scheme.

Acknowledgments. We would like to express our thanks to Dr. Mitsudera and Dr. Yaremchuk of the IPRC for their valuable comments and ecouragement. We also would like to express our thanks to Dr. Yoshikawa and Mr. Taguchi for providing us with the Kuroshio regional circulation model for the twin experiment. This research was supported by Frontier Research System for Global Change.

REFERENCES

Daubechies, I., 1988: Orthonormal basis of compactly supported wavelets. *Commun. Pure Appl. Math.,* **41,** 909–996.

Erlebacher, G., M. Y. Hussaini, and L. Jameson, Eds., 1996: *Wavelets: Theory and Applications.* Oxford University Press, 528 pp.

Evensen, G., 1994: Sequential data assimilation with a nonlinear quasi-geostrophic model using Monte Carlo methods to forecast error statistics. *J. Geophys. Res.,* **99,** 10 143–10 162.

Ezer, T., and G. L. Mellor, 1994: Continuous assimilation of Geosat altimeter data into a three-dimensional primitive equation Gulf Stream model. *J. Phys. Oceanogr.,* **24,** 832–847.

Jameson, L., 1998: A wavelet-optimized, very high order adaptive grid and order numerical method. *SIAM J. Sci. Comput.,* **19,** 1980–2013.

Kalman, R. E., 1960: A new approach to linear filtering and prediction problems. *J. Basic. Eng.,* **82D,** 35–45.

Levitus, S., 1982: *Climatological Atlas of the World Ocean.* National Oceanic and Atmospheric Administration Prof. Paper 13, 173 pp.

Malanotte-Rizzoli, P., and W. R. Holland, 1986: Data constraints applied to models of the ocean general circulation. Part I: The steady case. *J. Phys. Oceanogr.,* **16,** 1665–1682.

Mitsudera, H., Y. Yoshikawa, B. Taguchi, and H. Nakamura, 1997: High-resolution Kuroshio/Oyashio System Model: Preliminary results (in Japanese with English abstract). Japan Marine Science and Technology Center Rep. 36, 147–155.

Strang, G., and T. Nguyen, 1996: *Wavelets and Filter Banks.* Wellesley-Cambridge Press, 490 pp.

Wunsch, C., 1996: *The Ocean Circulation Inverse Problem.* Cambridge University Press, 442 pp.