Geosci. Model Dev. Discuss., 6, 4009–4051, 2013
www.geosci-model-dev-discuss.net/6/4009/2013/
doi:10.5194/gmdd-6-4009-2013

Geoscientific Model Development Discussions
Open Access

divand-1.0: *n*-dimensional variational data analysis for ocean observations

A. Barth[1], **J.-M. Beckers**[1], **C. Troupin**[2], **A. Alvera-Azcárate**[1], **and L. Vandenbulcke**[3,4]

[1]GHER, University of Liège, Liège, Belgium
[2]IMEDEA, Esporles, Illes Balears, Spain
[3]Seamod.ro/Jailoo srl, Sat Valeni, Com. Salatrucu, Jud. Arges, Romania
[4]CIIMAR, University of Porto, Porto, Portugal

Received: 7 June 2013 – Accepted: 5 July 2013 – Published: 23 July 2013

Correspondence to: A. Barth (a.barth@ulg.ac.be)

Published by Copernicus Publications on behalf of the European Geosciences Union.

Abstract

A tool for multidimensional variational analysis (`divand`) is presented. It allows the interpolation and analysis of observations on curvilinear orthogonal grids in an arbitrary high dimensional space by minimizing a cost function. This cost function penalizes the deviation from the observations, the deviation from a first guess and abruptly varying fields based on a given correlation length (potentially varying in space and time). Additional constraints can be added to this cost function such as an advection constraint which forces the analysed field to align with the ocean current. The method decouples naturally disconnected areas based on topography and topology. This is useful in oceanography where disconnected water masses often have different physical properties. Individual elements of the a priori and a posteriori error covariance matrix can also be computed, in particular expected error variances of the analysis. A multidimensional approach (as opposed to stacking 2-dimensional analysis) has the benefit of providing a smooth analysis in all dimensions, although the computational cost it increased.

Primal (problem solved in the grid space) and dual formulations (problem solved in the observational space) are implemented using either direct solvers (based on Cholesky factorization) or iterative solvers (conjugate gradient method). In most applications the primal formulation with the direct solver is the fastest, especially if an a posteriori error estimate is needed. However, for correlated observation errors the dual formulation with an iterative solver is more efficient.

The method is tested by using pseudo observations from a global model. The distribution of the observations is based on the position of the ARGO floats. The benefit of the 3-dimensional analysis (longitude, latitude and time) compared to 2-dimensional analysis (longitude and latitude) and the role of the advection constraint are highlighted. The tool `divand` is free software, and is distributed under the terms of the GPL license (http://modb.oce.ulg.ac.be/mediawiki/index.php/divand).

GMDD

6, 4009–4051, 2013

1 Introduction

Deriving a complete gridded field based on a set of observations is a common problem in oceanography. In situ observations are generally sparse and inhomogeneously distributed. While satellite observations have typically a better spatial and temporal coverages (but measure only surface data) than in situ data, they present also gaps due to e.g. the presence of clouds (in the case of thermal sea-surface temperature and optical surface properties of the ocean). Since the problem is generally under-determined, if the gridded field is to be derived from the observations alone, a first guess is introduced. The data analysis problem is also closely related to data assimilation where the observations are used in combination with a first guess coming from a model.

Several interpolation methods have been developed and presented in the scientific literature. Direct linear interpolation of the observations is rarely an option for ocean observations which are affected by noise and are not necessarily representative (e.g. a measurement at a specific time might not be representative for a monthly average). Current interpolation methods take therefore, in one way or the other, the uncertainty of the observations into account. Most interpolation methods of uncertain observations can be classified as methods based on optimal interpolation (including Kriging) and variational analysis.

For optimal interpolation methods (Gandin, 1965; Bretherton et al., 1976), the error covariance of the first guess is generally directly specified by analytical functions (Robinson, 1996). When satellite or model data are used, this error covariance can also be specified by its eigenvalues/eigenvectors (Kaplan et al., 1997; Rayner et al., 2003; Beckers et al., 2006) or by an ensemble (Evensen, 2007). Applications to multiple spatial and/or temporal dimensions are common (Høyer and She, 2007; Nardelli et al., 2010) to ensure a continuity of the solution along those dimensions. Analytical functions for the error covariance are based generally on the distance between two given points. However, decoupling water masses separated by land and maintaining at the same time a spatially smooth fields over the ocean is difficult.

GMDD
6, 4009–4051, 2013

In variational analysis, a cost function is formulated where every term corresponds to a desirable property of the field (e.g. close to observations, smooth). The variational approach is equivalent to the optimal interpolation formulation, but instead of specifying directly the error covariance of the first guess, the inverse of this matrix is parametrized: instead of imposing that two adjacent grid cells are correlated to each other, it is required that gradients (and higher-order derivatives) are small (Brasseur et al., 1996; Brankart and Brasseur, 1996). Decoupling water masses separated by land is natural in variational analysis as it can be included using boundary conditions on the spatial derivatives. Variational analysis in 3 or 4 dimensions is common in the context of data assimilation (e.g. Rabier et al., 2000; Dobricic and Pinardi, 2008; Moore et al., 2011b), but most of the data analysis applications to grid observations using variational methods are limited to two dimensions: either two horizontal dimensions (e.g. Troupin et al., 2010) or vertical transects (e.g. Yari et al., 2012). Three or four dimensional (space and time) fields are then obtained by assembling individual analysis. Inhomogeneous data distribution might then lead to spurious abrupt variations along the additional dimensions, which require ad hoc filtering of the assembled field.

The variational approach is also attractive for problems where it is easier to formulate physical properties of the underlying field in terms of constraints than in terms of correlation/covariance. For a two-dimensional surface current analysis for example, one can impose that the horizontal divergence is small (Legler and Navon, 1991; Yaremchuk and Sentchev, 2009) by adding a corresponding term to the cost function. This kind of constraint would be more difficult to implement in interpolation method.

On the other hand, in optimal interpolation one can quite easily derive the error variance of the analyzed fields which is more difficult but feasible for variational methods (Troupin et al., 2012). Optimal interpolation in the local approximation can also be quite efficiently applied to distributed-memory parallel computing architecture.

The aim of this manuscript is to implement and test a variational analysis program that can operate in an arbitrary high dimensional space and with a cost function that can be easily extended with additional constraints. The benefit of this method will be

GMDD
6, 4009–4051, 2013

assessed in comparison to assembled two-dimensional analyzes using an advection constraint forcing the gradients of an analysis to be aligned with a given vector field.

The proposed approach is based on the variational inverse method (Brasseur and Haus, 1991). The method is implemented in the tool DIVA (Data-Interpolating Variational Analysis) which computes the minimum of the cost function in two dimensions using a triangular finite-element mesh (Brasseur et al., 1996; Brankart and Brasseur, 1996, 1998; Troupin et al., 2012). A web interface has also been developed for this tool (Barth et al., 2010). Then we present an extension to *n*-dimension which is called `divand`. To simplify the testing and implementation in an arbitrarily high dimensional space, a regular curvilinear mesh is used for the tool `divand`.

In Sect. 2, the formulation of the method in *n*-dimensional space and derive the analytical kernel functions for an infinitely large domain is introduced. The relationship between the highest derivative needed in the formulation and the dimension of the domain is shown. Section 3 presents the different implemented algorithms. Simple numerical tests are performed in Sect. 4 to show the consistency of the numerical results with the analytical solutions of Sect. 2. Implementation details and capabilities of the tool are given in Sect. 5. The tool is also tested in a realistic configuration to reconstruct global temperature in Sect. 6.

2 Formulation

Variational inverse methods aim to derive a continuous field which is close to the observations and satisfies a series of a priori constraints. In particular, the field should be "smooth". It is therefore important to quantify the "smoothness" of a field. While the interpolated field should be close to observations, it should not necessarily pass through all observations because observations have errors and often do not represent the same

GMDD
6, 4009–4051, 2013

information. A cost function is formulated which includes these both constraints:

$$J[\varphi] = \sum_{j=1}^{N_d} \mu_j [d_j - \varphi(\boldsymbol{x}_j)]^2 + \|\varphi - \varphi_\mathrm{b}\|^2 \tag{1}$$

where d_j are the measurements at the location $\boldsymbol{x}_j$ and μ_j is their weight, φ_b is a background estimate of the field. In order to define the norm $\|\cdots\|$, the length-scale L_i in every domain dimension is introduced. These length-scales form the diagonal elements of the matrix $\mathbf{L}$:

$$\mathbf{L} = \begin{pmatrix} L_1 & 0 & \\ 0 & L_2 & \\ & & \ddots \end{pmatrix} \tag{2}$$

Based on these length-scales, we define the following scaled differential operators for gradient and Laplacian:

$$\tilde{\nabla} = \mathbf{L}\nabla \tag{3}$$

$$\tilde{\nabla}^2 = \nabla \cdot (\mathbf{L}^2 \nabla) \tag{4}$$

A scalar product $\langle f, g \rangle$ of two functions, f and g, is defined using the scaled gradient and Laplacian.

$$\langle f, g \rangle = \frac{1}{c} \int_D \alpha_0 fg + \alpha_1 (\tilde{\nabla} f) \cdot (\tilde{\nabla} g) + \alpha_2 (\tilde{\nabla}^2 f)(\tilde{\nabla}^2 g) + \alpha_3 (\tilde{\nabla}\tilde{\nabla}^2 f) \cdot (\tilde{\nabla}\tilde{\nabla}^2 g) + \ldots \mathrm{d}\boldsymbol{x} \tag{5}$$

We note m the highest derivative in this scalar product. The parameter c is a normalization coefficient that will be chosen later. The coefficients α_i are generally considered positive so that the cost function has certainly a finite minimum.

GMDD
6, 4009–4051, 2013

The norm used in the background constraint of Eq. (1) is defined using this scalar product by:

$$\|\varphi\|^2 = \langle \varphi, \varphi \rangle \tag{6}$$

If field φ is discretized on a grid and all elements are grouped into the vector $\boldsymbol{x}$, the cost function can be written as:

$$J(\boldsymbol{x}) = (\boldsymbol{x} - \boldsymbol{x}_\mathrm{b})^\mathrm{T} \mathbf{B}^{-1} (\boldsymbol{x} - \boldsymbol{x}_\mathrm{b}) + \left(\mathbf{H}\boldsymbol{x} - \boldsymbol{y}^\mathrm{o}\right)^\mathrm{T} \mathbf{R}^{-1} \left(\mathbf{H}\boldsymbol{x} - \boldsymbol{y}^\mathrm{o}\right) + J_\mathrm{c}(\boldsymbol{x}) \tag{7}$$

where we also regrouped all observations into vector $\boldsymbol{y}^\mathrm{o}$ and the discretized background field in vector $\boldsymbol{x}_\mathrm{b}$. $\mathbf{H}$ is a discretized local interpolation operator allowing to compare the gridded field with the observations at the data locations.

This cost function is commonly used in optimal interpolation where the matrices $\mathbf{B}$ and $\mathbf{R}$ are the error covariance of the background estimate and of the observations, respectively. The scalar product in Eq. (5) defines the matrix $\mathbf{B}$ and the diagonal matrix $\mathbf{R}$ is composed by the inverse of the data weight $\frac{1}{\mu_j}$. Since this cost function can be extended by additional constraints, we included the additional term $J_\mathrm{c}(\boldsymbol{x})$ to be specified later.

2.1 Kernel

The so-called reproducing kernel $K(\boldsymbol{x}, \boldsymbol{y})$ associated with Eq. (5) is defined by

$$\langle f, K \rangle = f \tag{8}$$

and will be helpful in understanding the covariance structure of $\mathbf{B}$.

If the domain is infinitely large ($D = \mathbb{R}^n$) and the correlation lengths L_i are constant in all dimensions, we can analytically derive the function K. First we assume that the correlation lengths L_i are all equal to one, and later the more general case with arbitrary (but constant) values of L_i will be derived. The derivation follows the Ph.D. thesis of

GMDD
6, 4009–4051, 2013

Brasseur (1994) where the kernel is derived for two-dimensional problems. Substituting the definition of scalar product Eq. (5) in Eq. (8) and by integrating by parts one obtains:

$$\begin{aligned}\langle f, K\rangle &= \frac{1}{c}\int_{\mathbb{R}^n} \alpha_0 f(\boldsymbol{x})K(\boldsymbol{x},\boldsymbol{y}) + \alpha_1\left(\widetilde{\nabla} f(\boldsymbol{x})\right)\cdot\left(\widetilde{\nabla} K(\boldsymbol{x},\boldsymbol{y})\right) \\ &\quad + \alpha_2\left(\widetilde{\nabla}^2 f(\boldsymbol{x})\right)\left(\widetilde{\nabla}^2 K(\boldsymbol{x},\boldsymbol{y})\right) + \ldots \mathrm{d}\boldsymbol{x} \\ &= \frac{1}{c}\int_{\mathbb{R}^n} f(\boldsymbol{x})\left[\alpha_0 K(\boldsymbol{x},\boldsymbol{y}) - \alpha_1\widetilde{\nabla}^2 K(\boldsymbol{x},\boldsymbol{y}) + \alpha_2\widetilde{\nabla}^4 K(\boldsymbol{x},\boldsymbol{y}) + \ldots\right]\mathrm{d}\boldsymbol{x} \\ &= f(\boldsymbol{y})\end{aligned}$$

As this last equation must be true for any function $f(\boldsymbol{x})$, the expression in brackets must be equal to the Dirac function (times c):

$$\alpha_0 K(\boldsymbol{x},\boldsymbol{y}) - \alpha_1\widetilde{\nabla}^2 K(\boldsymbol{x},\boldsymbol{y}) + \alpha_2\widetilde{\nabla}^4 K(\boldsymbol{x},\boldsymbol{y}) + \ldots = c\,\delta(\boldsymbol{x}-\boldsymbol{y})$$

Since the kernel is translation invariant, we can set $\boldsymbol{y} = 0$ without loss of generality. By applying the Fourier transform, we obtain:

$$\widehat{K}(\boldsymbol{k}) = \frac{c}{\alpha_0 + \alpha_1 k^2 + \alpha_2 k^4 + \ldots + \alpha_m k^{2m}}$$

where $\widehat{K}(\boldsymbol{k})$ is the Fourier transform of kernel $K(\boldsymbol{x})$:

$$\widehat{K}(\boldsymbol{k}) = \int_{\mathbb{R}^n} K(\boldsymbol{x}) e^{-i\boldsymbol{x}\cdot\boldsymbol{k}}\,\mathrm{d}\boldsymbol{x}$$

The kernel $K(\boldsymbol{x})$ can thus be found by using the inverse Fourier transform:

$$K(\boldsymbol{x}) = \frac{1}{(2\pi)^n}\int_{\mathbb{R}^n} \widehat{K}(\boldsymbol{k}) e^{i\boldsymbol{x}\cdot\boldsymbol{k}}\,\mathrm{d}\boldsymbol{k}$$

GMDD
6, 4009–4051, 2013
divand-1.0
A. Barth et al.

In particular, the value of the kernel at $\boldsymbol{x} = 0$ corresponds to the integral:

$$K(0) = \frac{2\pi^{\frac{n}{2}}}{\Gamma\left(\frac{n}{2}\right)} \frac{c}{(2\pi)^n} \int_0^\infty \frac{k^{n-1}}{\alpha_0 + \alpha_1 k^2 + \alpha_2 k^4 + \ldots + \alpha_m k^{2m}} \mathrm{d}k$$

where integration variables were transform to n-dimensional polar coordinates and integration was performed over all angles. The kernel is in fact nothing else than the correlation function one would use to create **B** yielding the same result in OI as with the varational approach (Wahba and Wendelberger, 1980). We naturally choose the value of c such that $K(0) = 1$

$$\frac{1}{c} = \frac{2\pi^{\frac{n}{2}}}{\Gamma\left(\frac{n}{2}\right)} \frac{1}{(2\pi)^n} \int_0^\infty \frac{k^{n-1}}{\alpha_0 + \alpha_1 k^2 + \alpha_2 k^4 + \ldots + \alpha_m k^{2m}} \mathrm{d}k$$

Assuming all $\alpha_i \geq 0$, the integral at the right-hand side is defined if $m > \frac{n}{2}$. This condition links thus the number of dimensions n and the order of the highest derivative needed in the formulation. The Fourier transform of the kernel $\widehat{K}$ is a radial function depending on the norm of the wave number $\boldsymbol{k}$. The inverse Fourier transform of a radial function is also a radial function which can be derived with the Hankel transform (Appendix):

$$K(r) = (2\pi)^{-\frac{n}{2}} r^{\frac{2-n}{2}} \int_0^\infty \mathrm{J}_{\frac{n-2}{2}}(kr)\, k^{\frac{n-2}{2}}\, \widehat{K}(k)\, k\, \mathrm{d}k \tag{9}$$

where $\mathrm{J}_\nu(r)$ is the Bessel function of first kind of order ν. To continue the analytical derivations we must make assumptions about the coefficients α_i. We assume that the coefficients α_i are chosen as binomial coefficients.

$$\alpha_i = \frac{m!}{i!(m-i)!} \qquad 1 \leq i \leq m$$

GMDD
6, 4009–4051, 2013

In this case, the Fourier transform of the kernel ($\widehat{K}^{n,m}(k)$) for the highest derivative m and dimension n can be written as,

$$\widehat{K}^{n,m}(k) = \frac{c}{(1+k^2)^m}.$$

Using this expression in Eq. (9), the radial part of the kernel $K^{n,m}(r)$ becomes

$$K^{n,m}(r) = c^{n,m}(2\pi)^{-\frac{n}{2}} r^{\frac{2-n}{2}} \int_0^\infty \mathrm{J}_{\frac{n-2}{2}}(kr) k^{\frac{n-2}{2}} \frac{k}{(1+k^2)^m} \mathrm{d}k. \tag{10}$$

The normalization coefficient is now noted $c^{n,m}$ as it depends on the dimension n and the order of the highest derivative m. By integrating by parts, we can derive a recursion relationship relating the kernels with different values of n and m.

$$\begin{aligned} K^{n,m}(r) &= c^{n,m} \frac{(2\pi)^{-\frac{n}{2}}}{2(1-m)} r^{\frac{2-n}{2}} \\ &\quad \int_0^\infty \mathrm{J}_{\frac{n-2}{2}}(kr) k^{\frac{n-2}{2}} \frac{d}{\mathrm{d}k} \left(\frac{1}{(1+k^2)^{m-1}} \right) \mathrm{d}k \\ &= c^{n,m} \frac{(2\pi)^{-\frac{n}{2}}}{2(m-1)} r^{\frac{4-n}{2}} \int_0^\infty \mathrm{J}_{\frac{n-4}{2}}(kr) k^{\frac{n-4}{2}} \frac{k}{(1+k^2)^{m-1}} \mathrm{d}k \qquad (11) \\ &= \frac{1}{4\pi(m-1)} \frac{c^{n,m}}{c^{n-2,m-1}} K^{n-2,m-1}(r) \qquad (12) \end{aligned}$$

where in step (11), we used the following equation relating Bessel functions of first kind of different order:

$$\frac{d}{\mathrm{d}x}\left(x^p \mathrm{J}_p(x)\right) = x^p \mathrm{J}_{p-1}(x)$$

GMDD
6, 4009–4051, 2013
divand-1.0
A. Barth et al.

or

$$\frac{d}{dk}\left(J_{\frac{n-2}{2}}(kr)k^{\frac{n-2}{2}}\right) = rJ_{\frac{n-4}{2}}(kr)k^{\frac{n-2}{2}}$$

Since $K^{n,m}(\boldsymbol{x})$ is one for $\boldsymbol{x} = 0$, the normalization coefficients are thus related by:

$$c^{n,m} = 4\pi(m-1)c^{n-2,m-1} \tag{13}$$

The recurrence relationship therefore shows that it is sufficient to calculate the kernel ($K^{n,m}$) and the normalization coefficients for $n = 1$ and $n = 2$. For $n = 1$, we find the following solution for the integral in Eq. (10):

$$K^{1,m}(r) = \frac{2}{\Gamma(m-1/2)}\left(\frac{r}{2}\right)^{(m-1/2)}K_{m-1/2}(r)$$

$$c^{1,m} = \frac{2\sqrt{\pi}\,\Gamma(m)}{\Gamma(m-1/2)}$$

where $K_\nu(r)$ is the modified Bessel function of second kind of order ν. For $n = 2$, the solution of Eq. (10) is:

$$K^{2,m}(r) = \frac{2}{\Gamma(m-1)}\left(\frac{r}{2}\right)^{(m-1)}K_{m-1}(r)$$

$$c^{2,m} = 4\pi(m-1)$$

Using the recursion relationship Eqs. (12) and (13) with the solution for $n = 1$ (resp. $n = 2$) one can derive the kernel and $c^{n,m}$ for every odd (resp. even) value of n. After simplifications, it follows that for any n (odd and even) and m, the normalized kernel $K^{n,m}$ and the corresponding normalization factor $c^{n,m}$ can be written as:

$$K^{n,m}(r) = \frac{2}{\Gamma(\nu)}\left(\frac{r}{2}\right)^{\nu}K_\nu(r)$$

$$c^{n,m} = \frac{(4\pi)^{n/2}\Gamma(m)}{\Gamma(\nu)}$$

GMDD
6, 4009–4051, 2013

with $\nu = m - n/2$. Based on these results, we can finally derive the case for a correlation length different from one, using the change of variables $\boldsymbol{x} \rightarrow \mathbf{L}^{-1}\boldsymbol{x}$:

$$K^{n,m}(\boldsymbol{x}) = \frac{2}{\Gamma(\nu)} \left(\frac{|\mathbf{L}^{-1}\boldsymbol{x}|}{2} \right)^{\nu} \mathrm{K}_{\nu}(|\mathbf{L}^{-1}\boldsymbol{x}|)$$

$$c^{n,m} = \frac{(4\pi)^{n/2}\Gamma(m)|\mathbf{L}|}{\Gamma(\nu)}$$

where $|\mathbf{L}|$ is the determinant of the diagonal matrix $\mathbf{L}$:

$$|\mathbf{L}| = \prod_{i=1}^{n} L_i$$

The kernels and normalization coefficients can be expanded further for particular values of n and m leading the lines of Tables 1 and 2. Our results agree with the solution derived by Brasseur et al. (1996) for the case of two dimensions $n = 2$ and $m = 2$.

In the `divand` tool a distinction is made between the actual dimension n and the effective dimension. The effective dimension is the number of dimensions with a non-zero correlation length. Setting a correlation length to zero decouples the different dimensions and is used to emulate the results that one obtains by "stacking" results of two-dimensional analysis as it has been done previously (e.g. Troupin et al., 2010). The normalization coefficient used in this case is based on the effective dimension. This ensures that one obtains exactly the same results of a stacked 2-D interpolation by analyzing data in a 3-D domain with a zero correlation length in the third dimension.

2.2 Additional constraints

In addition to the observation constraint and smoothness constraint, an arbitrary number of other constraints can be included in `divand`. Those constraints are

GMDD
6, 4009–4051, 2013

characterized by the symmetric positive-defined matrix $\mathbf{Q}_i$, the matrix $\mathbf{C}_i$ and the vectors z_i.

$$J_c(\boldsymbol{x}) = \sum_i (\mathbf{C}_i \boldsymbol{x} - z_i)^{\mathrm{T}} \mathbf{Q}_i^{-1} (\mathbf{C}_i \boldsymbol{x} - z_i)$$

Those additional constraints can be re-absorbed in the definition of the operator $\mathbf{H}$, $\mathbf{R}$ and $\boldsymbol{y}^{\mathrm{o}}$:

$$\begin{pmatrix} \boldsymbol{y}^{\mathrm{o}} \\ z_1 \\ z_2 \\ \vdots \end{pmatrix} \rightarrow \boldsymbol{y}^{\mathrm{o}} \qquad \begin{pmatrix} \mathbf{H} \\ \mathbf{C}_1 \\ \mathbf{C}_2 \\ \vdots \end{pmatrix} \rightarrow \mathbf{H} \qquad \begin{pmatrix} \mathbf{R} & & & 0 \\ & \mathbf{Q}_1 & & \\ & & \mathbf{Q}_2 & \\ 0 & & & \ddots \end{pmatrix} \rightarrow \mathbf{R} \tag{14}$$

With these definitions the cost function has again the familiar form:

$$J(\boldsymbol{x}) = \boldsymbol{x}^{\mathrm{T}} \mathbf{B}^{-1} \boldsymbol{x} + (\mathbf{H}\boldsymbol{x} - \boldsymbol{y}^{\mathrm{o}})^{\mathrm{T}} \mathbf{R}^{-1} (\mathbf{H}\boldsymbol{x} - \boldsymbol{y}^{\mathrm{o}}) \tag{15}$$

3 Minimization and algorithms

As the cost function is quadratic, one can obtain its minimum analytically. The derivation of its minimum is well known (e.g. Courtier et al., 1998) and is included here for completeness. If $\boldsymbol{x}^{\mathrm{a}}$ is the minimum of the cost function J, a small variation $\delta\boldsymbol{x}$ of $\boldsymbol{x}^{\mathrm{a}}$, would not change the cost function in the first order of $\delta\boldsymbol{x}$. Noting $^{\mathrm{T}}$ a transposed matrix or vector,

$$\begin{aligned} \delta J &= J(\boldsymbol{x}^{\mathrm{a}} + \delta\boldsymbol{x}) - J(\boldsymbol{x}^{\mathrm{a}}) \\ &= \delta\boldsymbol{x}^{\mathrm{T}} (\mathbf{B}^{-1} + \mathbf{H}^{\mathrm{T}} \mathbf{R}^{-1} \mathbf{H}) \boldsymbol{x}^{\mathrm{a}} - \delta\boldsymbol{x}^{\mathrm{T}} \mathbf{H}^{\mathrm{T}} \mathbf{R}^{-1} \boldsymbol{y}^{\mathrm{o}} = 0 \end{aligned}$$

As $\delta\boldsymbol{x}$ is arbitrary, the expression multiplying $\delta\boldsymbol{x}^{\mathrm{T}}$ must be zero. The optimal state $\boldsymbol{x}^{\mathrm{a}}$ is thus given by:

$$\boldsymbol{x}^{\mathrm{a}} = \mathbf{P}\mathbf{H}^{\mathrm{T}} \mathbf{R}^{-1} \boldsymbol{y}^{\mathrm{o}} \tag{16}$$

GMDD
6, 4009–4051, 2013

where we have introduced the matrix **P**.

$$\mathbf{P}^{-1} = \mathbf{B}^{-1} + \mathbf{H}^{\mathrm{T}}\mathbf{R}^{-1}\mathbf{H} \tag{17}$$

The interpretation of this matrix becomes clear since we can rearrange of the cost function as:

$$J(\boldsymbol{x}) = (\boldsymbol{x} - \boldsymbol{x}^{\mathrm{a}})^{\mathrm{T}}\mathbf{P}^{-1}(\boldsymbol{x} - \boldsymbol{x}^{\mathrm{a}}) + \text{constant} \tag{18}$$

The matrix **P** represents thus the error covariance of the analysis $\boldsymbol{x}^{\mathrm{a}}$ (Rabier and Courtier, 1992; Courtier et al., 1994).

3.1 Primal formulation

The primal formulation of the algorithm follows directly from Eqs. (16) and (17). The matrices **P** and **B** are never formed explicitly and the tool works only with the inverse of these matrices noted $\mathbf{P}_{\mathrm{inv}}$ and $\mathbf{B}_{\mathrm{inv}}$, respectively. In order to give the algorithm in a compact form close to the mathematical equations, we introduce the backslash operator by:

$$\mathbf{X} = \mathbf{A} \setminus \mathbf{B} \tag{19}$$

which is equivalent to solving the system $\mathbf{AX} = \mathbf{B}$ for the matrix (or vector) **X**. With this notation, the primal algorithm reads:

$$\mathbf{P}_{\mathrm{inv}} = \mathbf{B}_{\mathrm{inv}} + \mathbf{H}^{\mathrm{T}}(\mathbf{R} \setminus \mathbf{H}) \tag{20}$$

$$\boldsymbol{x}^{\mathrm{a}} = \mathbf{P}_{\mathrm{inv}} \setminus \left(\mathbf{H}^{\mathrm{T}}(\mathbf{R} \setminus \boldsymbol{y}^{\mathrm{o}})\right) \tag{21}$$

Different approaches have been implemented here to solve the analysis equation in its primal formulation involving either a direct solver, a factorization or the conjugate gradient method.

GMDD
6, 4009–4051, 2013

3.1.1 Direct solver

The solver based on SuiteSparse (Davis, 2004a,b) can be used directly if the matrix $\mathbf{R} \setminus \mathbf{H}$ is sparse. This is in particular the case when $\mathbf{R}$ is diagonal. $\mathbf{P}_{\text{inv}}$ will then also be a sparse matrix which can be efficiently stored. This approach is useful when no error field needs to be computed. The direct solver can also be applied for a non-diagonal $\mathbf{R}$ but this approach is prohibitive in terms of computational cost and memory.

3.1.2 Factorization

The inverse of the a posteriori error covariance matrix $\mathbf{P}_{\text{inv}}$ is factorized in the following products:

$$\mathbf{R}_P^T \mathbf{R}_P = \mathbf{Q}_P^T \mathbf{P}_{\text{inv}} \mathbf{Q}_P \tag{22}$$

where $\mathbf{R}_P$ is an upper triangular matrix and $\mathbf{Q}_P$ is a permutation matrix (chosen to preserve the sparse character of $\mathbf{R}_P$). Once the matrix $\mathbf{P}_{\text{inv}}$ is factorized, the product between $\mathbf{P}$ and any vector $\boldsymbol{x}$ can be computed efficiently by:

$$\mathbf{P}\boldsymbol{x} = \mathbf{P}_{\text{inv}} \setminus \boldsymbol{x} = \mathbf{Q}_P \left(\mathbf{R}_P \setminus \left(\mathbf{R}_P^T \setminus \left(\mathbf{Q}_P^T \boldsymbol{x} \right) \right) \right) \tag{23}$$

This approach is useful if the error field is required, since a large number of products between $\mathbf{P}$ and a given vector must be computed. Determining all elements of $\mathbf{P}$ would be prohibitive, but individual elements (such as the diagonal elements) are computed by:

$$\mathbf{P}_{ij} = \boldsymbol{e}_i (\mathbf{P}_{\text{inv}} \setminus \boldsymbol{e}_j) \tag{24}$$

where $\boldsymbol{e}_i$ is the ith basis vector. The tool `divand` returns a matrix-like object allowing to compute any element of $\mathbf{P}$ or the product of $\mathbf{P}$ with a given vector. This latter product is useful if one wants to derive the expected error of an integrated quantity such as a transport along a section or any other weighted sum.

GMDD
6, 4009–4051, 2013

3.1.3 Conjugate gradient method

The conjugate gradient method (Golub and Van Loan, 1996) is commonly used in variational data assimilation (e.g. Moore et al., 2011a). This method provides an iterative solution to linear equations

$$\mathbf{A}\boldsymbol{x}^{\mathrm{a}} = \boldsymbol{b}, \tag{25}$$

where the vector $\boldsymbol{b}$ and the symmetric and positive-defined matrix $\mathbf{A}$ are given here by:

$$\boldsymbol{b} = \mathbf{H}^{\mathrm{T}}(\mathbf{R} \setminus \boldsymbol{y}^{\mathrm{o}}) \tag{26}$$

$$\mathbf{A}\boldsymbol{x}^{\mathrm{a}} = \mathbf{P}_{\mathrm{inv}}\boldsymbol{x}^{\mathrm{a}} + \mathbf{H}^{\mathrm{T}}(\mathbf{R} \setminus (\mathbf{H}\boldsymbol{x}^{\mathrm{a}})) \tag{27}$$

The conjugate gradient algorithm is applied to solve for $\boldsymbol{x}^{\mathrm{a}}$. Lanczos vectors can be saved to compute an approximation of the error of the analysis (Moore et al., 2011b).

3.2 Dual formulation

Using the Sherman–Morrison–Woodbury formula (Golub and Van Loan, 1996), the solution can also be written in the dual formulation (Courtier, 1997):

$$\mathbf{x}^{\mathrm{a}} = \mathbf{B}\mathbf{H}^{\mathrm{T}}\left(\mathbf{H}\mathbf{B}\mathbf{H}^{\mathrm{T}} + \mathbf{R}\right)^{-1}\boldsymbol{y}^{\mathrm{o}} \tag{28}$$

$$\mathbf{P} = \mathbf{B} - \mathbf{B}\mathbf{H}^{\mathrm{T}}\left(\mathbf{H}\mathbf{B}\mathbf{H}^{\mathrm{T}} + \mathbf{R}\right)^{-1}\mathbf{H}\mathbf{B} \tag{29}$$

In this formulation, all implemented methods in `divand` are based on the conjugate gradient method, to solve iteratively the following equation for $\boldsymbol{y}'$:

$$\mathbf{C}\boldsymbol{y}' = \boldsymbol{y}^{\mathrm{o}} \tag{30}$$

where $\mathbf{C}$ represents a symmetric and positively defined matrix specified in operator form:

$$\mathbf{C}\boldsymbol{y}' = \mathbf{H}(\mathbf{P}_{\mathrm{inv}} \setminus (\mathbf{H}^{\mathrm{T}}\boldsymbol{y}')) + \mathbf{R}\boldsymbol{y}' \tag{31}$$

GMDD
6, 4009–4051, 2013

Once the vector $\boldsymbol{y}'$ is known, the analysis $\boldsymbol{x}^{\mathrm{a}}$ is obtained by:

$$\boldsymbol{x}^{\mathrm{a}} = \mathbf{P}_{\mathrm{inv}} \setminus (\mathbf{H}^{\mathrm{T}} \boldsymbol{y}') \tag{32}$$

If $\mathbf{R}$ is a non-diagonal matrix, one can optionally use a preconditioner based on the diagonal elements of $\mathbf{R}$ noted $\mathbf{R}'$.

$$\mathbf{M} = \mathbf{H}\left(\mathbf{P}_{\mathrm{inv}} \setminus \mathbf{H}^{\mathrm{T}}\right) + \mathbf{R}'; \tag{33}$$

The matrix $\mathbf{M}$ can be efficiently factorized in sparse matrices:

$$\mathbf{R}_{\mathrm{M}}^{\mathrm{T}} \mathbf{R}_{\mathrm{M}} = \mathbf{Q}_{\mathrm{M}}^{\mathrm{T}} \mathbf{M}_{\mathrm{inv}} \mathbf{Q}_{\mathrm{M}} \tag{34}$$

where $\mathbf{R}_{\mathrm{M}}$ is an upper triangular matrix and $\mathbf{Q}_{\mathrm{M}}$ is a permutation matrix.

3.2.1 Factorization

If the conjugate algorithm requires a large number of iterations, it is useful to factorize the matrix $\mathbf{P}_{\mathrm{inv}}$ to accelerate the product of $\mathbf{P}$ and a vector, which requires solving a linear system.

$$\mathbf{R}_{\mathrm{P}}^{\mathrm{T}} \mathbf{R}_{\mathrm{P}} = \mathbf{Q}_{\mathrm{P}}^{\mathrm{T}} \mathbf{P}_{\mathrm{inv}} \mathbf{Q}_{\mathrm{P}} \tag{35}$$

where, as before, $\mathbf{R}_{\mathrm{P}}$ is an upper triangular matrix and $\mathbf{Q}_{\mathrm{P}}$ is a permutation matrix. All products of $\mathbf{P}$ times a vector $\boldsymbol{x}$ are computed efficiently by:

$$\mathbf{P}\boldsymbol{x} = \mathbf{P}_{\mathrm{inv}} \setminus \boldsymbol{x} = \mathbf{Q}_{\mathrm{P}}\left(\mathbf{R}_{\mathrm{P}} \setminus \left(\mathbf{R}_{\mathrm{P}}^{\mathrm{T}} \setminus \left(\mathbf{Q}_{\mathrm{P}}^{\mathrm{T}} \boldsymbol{x}\right)\right)\right) \tag{36}$$

4 Numerical tests

4.1 Numerical kernel

It has been verified in numerical tests that the code reproduces well the analytical kernels. The domain is one-dimensional for $\nu = 1/2$, $3/2$ and $5/2$ and two-dimensional

GMDD
6, 4009–4051, 2013

for $\nu = 1$ and 2. Every direction ranges from -10 to 10 and is discretized with 201 grid points. A hypothetical observation located at the center of the domain is analyzed with a signal-to-noise ratio of 1 and with a correlation length-scale of 1. Theoretically the value of the analyzed field should be $1/2$ at the center. The radial solution multiplied by two is compared to the analytical solution of an infinite domain (Fig. 1). In all tests, the scaling and structure of the numerical kernels correspond well to the analytical solutions. For such correspondence, it is necessary that the grid resolution resolves well the correlation length. Qualitative tests have shown that the numerical kernels match well the analytical functions as long as the grid spacing is one forth (or less) of the correlation length.

The kernels differ by the rate at which they decrease to zero. It is important to consider this aspect when comparing the correlation length from analysis using different number of derivatives. Table 3 shows the value of r for which the analytical kernels are $1/2$. They can be used as proportionality coefficients to make the kernels more comparable (Fig. 1).

The kernel for $\nu = 1/2$ has a discontinuous derivative for $r = 0$ which makes this function unfit for practical use. A simple one-dimensional analysis with $m = 1$ (i.e. $\nu = 1/2$) illustrates the problem (Fig. 2). As there is no penalty on the second derivative, the analysis has a discontinuous derivative at every observation location (black dots). Using higher order derivatives resolves this problem. This example shows also that the analyses with higher order kernels ($\nu = 3/2$ and $\nu = 5/2$) are very similar. In these numerical experiments, the correlation length L is the inverse of the values in Table 3. This problem appears in all configurations where $\nu = 1/2$. In particular also in 3-dimensional analyses when the highest derivative in the cost function is a Laplacian. This is surprising because the first derivative is discontinuous despite the fact that the cost function penalizes the second derivative. Per default the cost function in `divand` therefore includes the derivatives up to $1 + n/2$ (rounded upwards) to ensure that the analysis has a continuous derivative.

GMDD
6, 4009–4051, 2013

4.2 Benchmark

The tool `divand` is written such that it can run on MATLAB and GNU Octave. In general, interpreted language tends to be slow on explicit loops. Interpreters like MATLAB can reduce this performance penalty by using Just-In-Time compilation (which is currently not available in Octave). Therefore MATLAB tends to be faster on performance benchmarks (e.g. Chaves et al., 2006; Leros et al., 2010) than GNU Octave. Explicit loops are avoided in the tool `divand` except for the number of dimensions which is a quite short loop (typically 1 to 4 dimensions).

The run-time performance of `divand` for MATLAB (version R2012b) and GNU Octave (version 3.6.4) with two implementations of the BLAS library (either GotoBLAS2 version 1.13 or Intel's Math Kernel Library (MKL) version 10.1) were tested. The benchmark was performed on a Intel Xeon L5420 CPU (using a single core) with 16 GB memory. The domain is a square and the used correlation length is inversely proportional to the number of grid points in one dimension.

Observations come from an analytical function and are located at every 5th grid point (in the two dimensions). In all cases the correctness of the analyzed field was verified by comparing the interpolated field to the original analytical function. The benchmark was repeated 5 times and the median value is shown in Table 4 for the primal algorithm with Cholesky factorization and for the dual algorithm with conjugate gradient minimization (Table 5). For the primal algorithm, all tested versions perform equally well with a slight advantage of Octave for domain sizes of 500×500 and 600×600. Profiling of the code shows that for the primal algorithm most of the time is spent in the Cholesky factorization using the library CHOLMOD included in SuiteSparse in both MATLAB and Octave which explains the similar results.

In general, the dual algorithm is much slower than the primal algorithm with Cholesky factorization. However, it should be reminded that the latter would be unpractical in some cases (in particular with spatially correlated observation errors). The difference between the different interpreters is more pronounced for the dual case. The

GMDD
6, 4009–4051, 2013

benchmarks show that MATLAB is faster for small domain sizes (30×30 and 50×50), while GNU Octave (with GotoBLAS2 and MKL) outperforms MATLAB for larger domain sizes. For small domain sizes, the preparation of the matrices to invert represents a significant fraction of the total run time where GNU Octave tends to be slower due to the lack of a Just-In-Time compiler. For larger domains, the cost is dominated by matrix operations which are faster in Octave for our case. In our benchmark, the Math Kernel Library is slightly faster in Octave than the GotoBLAS2 library.

5 Implementation

The `divand` tool is implemented such that it allows analysis in an arbitrary high dimensional space. Internally the n-dimensional arrays for e.g. the analyzed field are represented as a "flat" vector. To implement the background constraint, the following basic operations are implemented as sparse matrices:

- differentiation along a given dimension
- grid staggering along a given dimension
- trimming the first and last elements of a given dimension

All other differential operators (also represented as sparse matrices) are derived as a product and sum of these basic operations. The grid must be an orthogonal curvilinear grid and some dimensions might be cyclic. The user of the tool can specify additional constraints of the analyzed fields, for example an advection constraint requiring that the gradient of the analysis are aligned to a given vector field.

$$J_a(\phi) = \int_D (\boldsymbol{v} \cdot \nabla\phi)^2 \mathrm{d}D \tag{37}$$

where $\boldsymbol{v}$ is a vector field with n components. Such a constraint is useful in a geophysical context to force a field to be close to a stationary (or time dependent) solution of the

GMDD
6, 4009–4051, 2013

advection equation. The diffusion term is not included as it is generally small for geophysical applications and since the background constraint acts similar to a diffusion.

Consistent error calculations are also possible with the tool `divand` to estimate the error variance of the analyzed field. This error variance reflects among others the distribution of the observations, the correlation length and the background variance error. However, the accuracy of the error estimate of the analysed field depends crucially on the validity of the background and observation error covariance.

The tool introduces new matrix objects which implement several matrix operations (such as multiplication, multiplication by its inverse, extraction of diagonal elements). These new matrix objects include:

- a matrix specified by its inverse and potentially factorized (for $\mathbf{B}$ and $\mathbf{P}$)
- a matrix specified by an analytical function (for $\mathbf{R}$ and $\mathbf{C}_i$)
- a matrix of the form $\mathbf{C}+\mathbf{B}\mathbf{B}^{\mathrm{T}}$, which can be inverted using the Sherman–Morrison–Woodbury formula (for $\mathbf{R}$)
- a matrix composed by block matrices (for additional constraints)
- a matrix whose columns are stored on the disc instead of memory, due to memory limitations (used to save Lanczos vectors).

By adding these new matrix objects one can code the algorithm in a compact way which is close to the original mathematical formulation. For instance, the product of the analysis error covariance matrix $\mathbf{P}$ and vector $\boldsymbol{x}$ can just be coded as `P*x` and the matrix multiplication method of matrix object `P` implements the multiplication using the factorized form of Eq. (23).

GMDD
6, 4009–4051, 2013

6 Realistic test case

The interpolation was tested using pseudo-observations of temperature coming from a global ocean model. The results of the year 2007 from the ORCA2 model (Mathiot et al., 2011) with a spatial varying resolution (generally close to 2 resolution) are used. Making a reconstruction with model data allows to compare the analyzed field to the original model data and to assess the quality of the analysis. The position of the pseudo-observations are the real position of the ARGO observations from year 2007. Observations are extracted from the daily model results and the analysis targets monthly means. This mimics the common setup with real observations where measurements are instantaneous while the analyses represent a mean over a given time period. Only surface data are reconstructed for every month separately (2-D analysis) or all 12 months are considered together (3-D analysis). The analysis is compared to a monthly model climatology for the year 2007 and the RMS difference is calculated.

The central question of this test case is to assess the benefit of a 3-D analysis (longitude, latitude and time) compared to a 2-D analysis (longitude and latitude). Afterward different variants of the analysis are also tested, in particular the advection constraint. Parameters in the analysis (signal-to-noise ratio, spatial and temporal correlation length and strength of the advection constraint) are optimized to ensure the comparison of every approach in its best possible configuration. Analyses are compared to a model climatology obtained by averaging the daily model output.

6.1 2-D analysis

All observations from the same month are considered as data from the same time instance. Signal-to-noise ratio and correlation length are optimized by an exhaustive search of those parameters. The correlation length is chosen here to be identical in both horizontal dimensions. The RMS difference (space and time average) between the analysis and the reference model climatology is minimum for a signal-to-noise ratio of 14 and a correlation length of 1072 km (Fig. 3). The global RMS error of this analysis

GMDD
6, 4009–4051, 2013

is 1.1501 °C (Table 6). This experiment serves as the baseline for other experiments and improvements will be expressed as using the mean square skill score using this experiment as a reference.

Figure 4 (top left panel) shows the RMS error averaged over time for every spatial grid point. This RMS field reflects essentially the data coverage and areas with poor coverage can have RMS errors of 3 °C and more. Due to the sparsity of ARGO data in the coastal area (relative to the near-shore scales of variability), the RMS error is generally the highest near the coast.

As a variant of the 2-D analysis, the stationary advection is added to the cost function.

$$J_a[\varphi] = \int_D (\boldsymbol{v} \cdot \nabla\varphi)^2 \mathrm{d}D = a_s^2 \int_D \left(u\frac{\partial\varphi}{\partial x} + v\frac{\partial\varphi}{\partial y} \right)^2 \mathrm{d}D \tag{38}$$

where $\boldsymbol{v} = (a_s u, a_s v)$. The vector (u, v) represents the monthly-averaged model currents. The coefficient a_s determines how strong the advection constraint should be enforced. It is instructive to visualize the impact of a point observation without and with advection constraint. The correction by a point observation is in fact directly related to the background error covariance. Figure 5 shows the impact of an observation located at 72° W and 36.9° N (white cross). Without advection constraint, the covariance is mostly isotropic. The slight deviation from isotropy is due to the proximity of the coastline. The location with the largest impact is marked by a white circle. One could expect that the location with the largest impact coincides with the location of the observation. This is actually the case for an observation far away from the coastline. However, near the coastline the variance increases (due to the boundary conditions imposed in the smoothness constraint) which explains the relatively high impact near the coast.

With the advection constraint, the covariance is elongated along the path of the Gulf Stream (downstream and upstream). This is a desirable effect since tracers in the ocean tend to be uniform along ocean currents. The variance with advection constraint

GMDD
6, 4009–4051, 2013

is relatively uniform spatially near the location of the observation and thus the location of maximum impact coincides with the position of the observation.

The 2-D analysis with advection constraint has thus in total three parameter: signal-to-noise ratio, spatial correlation and strength of the advection constraint. These parameters are optimized by the Nelder–Mead algorithm (Nelder and Mead, 1965) by minimizing the RMS difference between the analysis and the reference climatology.

6.2 3-D analysis

In a first test, all observations from the same month are again considered coming from the same time instance. Although this is not necessary for a 3-D analysis, it simplifies the comparison with the previous 2-D case where the information of the actual day is not taken into account. Signal-to-noise ratio, spatial correlation and temporal correlation are optimized using the nonlinear Nelder–Mead minimization as before. The RMS is minimum with 0.9822 for signal-to-noise ratio of 27, spatial correlation length of 1373 km and a temporal correlation length of 4.9 months. Extending the analysis from 2-D to 3-D improves the skill (mean square skill score) by 27 %.

For every spatial grid point the time-averaged RMS error is computed (Fig. 4). To facilitate the comparison with the 2-D case, the difference of these RMS values is also shown (Fig. 4, bottom panel). Red shows areas where the 3-D analysis is better and in blue areas the 2-D analysis is more accurate. The RMS error is generally reduced by the 3-D analysis in coastal areas where few observations are present. However also a small degradation is observed in the open ocean. This is attributed to the fact that signal-to-noise ratio and correlation length are optimized globally. It is probable that space-dependent parameters (distinguishing for example between open ocean and near shore conditions) will improve the analysis even more.

In a second test, the actual day of the pseudo-measurements are used in the analysis (noted fractional time in Table 6) which improves slightly the analysis. The small increase of the signal-to-noise ratio is consistent with the fact that by providing the exact date (instead of the month), the observations are more coherent. The relative small

GMDD
6, 4009–4051, 2013

improvement is related to the fact that the optimal temporal correlation is 4.9 months and larger than one month (the time resolution of the original 3-D experiment).

The analysis is also performed using the advection constraint based on model currents. Since the time dimension is included, it is possible to use the non-stationary advection constraint:

$$J_a[\varphi] = \int_D (\boldsymbol{v} \cdot \nabla\varphi)^2 \mathrm{d}D = a_s^2 \int_D \left(u\frac{\partial\varphi}{\partial x} + v\frac{\partial\varphi}{\partial y} + \frac{\partial\varphi}{\partial t} \right)^2 \mathrm{d}D \tag{39}$$

where $\boldsymbol{v} = (a_s u, a_s v, a_s)$ and a_s determines the strength of the advection constraint. Figure 6 shows the impact of a point observation in the 3-D case. Without advection constraint the covariance is essentially uniform with a small modulation due to the proximity of the coastline. With a time-dependent advection constraint a distinction between upstream and downstream is made if two different time steps are considered. The location of the observation is more strongly connected to the upstream area of previous time instances and more strongly related to the downstream area of the following time instances. The time-dependent covariances with the advection constraint can thus relate different time instances taking the advection into account. As in the 2-D case, the advection constraint is introduced with a proportionality coefficient a_s allowing to tune the strength of this effect. The calibration of this parameter is related to, among others, the overall significance of advection compared to other processes and the accuracy of the current field.

The optimal values of the analysis parameters are shown in Table 6. To compare the different variants, a skill-score relative to the 2-D case (without advection) has been computed:

$$\text{skill score} = 1 - \frac{\text{RMS}^2}{\text{RMS}^2_{\text{2-D}}} \tag{40}$$

It follows that inclusion of the advection constraint in the 2-D analysis improves the skill by 29 %. It is surprising to see that this improvement is of similar amplitude as the

GMDD
6, 4009–4051, 2013

improvement obtained by including the time dimension as the later requires the solution of a 12 times larger system (the number of months). Including the exact date of a measurement instead of its month leads to only a small improvement. The best analysis is obtained when using 3-D domain in combination with the advection constraint leading to an improvement of 44 %. Including the advection constraint has again the beneficial effect of increasing the optimal signal-to-noise ratio as the observations are more coherent along flow lines.

7 Conclusions

A variational analysis tool has been developed and tested with a realistic data distribution from ARGO, but with pseudo-observations extracted from a model. This allows to compare the analysis to model climatology data and to quantitatively compare different analyses. Parameters are optimized by minimizing the difference between the analysis and the model climatology. However in practice, a cross-validation data set is needed for such optimization which ideally should be homogeneously distributed. An improvement with 3-D (longitude, latitude and time) versus 2-D analysis (horizontal only) was shown. A relatively larger reduction of the RMS error was also observed by including the advection constraint (stationary in the 2-D case and time-dependent in the 3-D case). However, it should be noted that the current fields used here are dynamically coherent with the tracer fields as they come from the same model. In a realistic setup with real observations, an improvement similar to the one reported here will require quite accurate current fields.

The source code of the tool `divand` is available from http://modb.oce.ulg.ac.be/mediawiki/index.php/divand and distributed under the terms of the General Public License. The toolbox will also be made available through to the Octave extensions repository octave-forge.

GMDD
6, 4009–4051, 2013

Appendix A

Hankel transform

We note $\widehat{f}(\boldsymbol{k})$ the Fourier transform of a function $f(\boldsymbol{x})$:

$$\widehat{f}(\boldsymbol{k}) = \int_{\mathbb{R}^n} e^{-i\boldsymbol{k}\boldsymbol{x}} f(\boldsymbol{x})\, \mathrm{d}^n\boldsymbol{x} \tag{A1}$$

If the function $f(\boldsymbol{x})$ is a radial function $f(\boldsymbol{x}) = F(r)$, then its Fourier transform also depends only on the module of the wave-number vector $\widehat{f}(\boldsymbol{k}) = \widehat{F}(k)$ where $r = |\boldsymbol{x}|$ and $k = |\boldsymbol{k}|$.

The Fourier transform of a radial function in $\mathbb{R}^n$ is given in terms of the Hankel transform by (Arfken, 1985):

$$k^{\frac{n-2}{2}} \widehat{F}(k) = (2\pi)^{\frac{n}{2}} \int_0^\infty \mathrm{J}_{\frac{n-2}{2}}(kr) r^{\frac{n-2}{2}} F(r) r\, \mathrm{d}r \tag{A2}$$

where $\mathrm{J}_\nu(r)$ is the Bessel function of first kind of order ν. The inverse Fourier transform is given by a similar relationship:

$$r^{\frac{n-2}{2}} F(r) = (2\pi)^{-\frac{n}{2}} \int_0^\infty \mathrm{J}_{\frac{n-2}{2}}(kr) k^{\frac{n-2}{2}} \widehat{F}(k) k\, \mathrm{d}k \tag{A3}$$

Acknowledgements. This work was funded by the SeaDataNet FP7 EU project (grant 283607), by the EU project EMODnet Chemistry, by the project PREDANTAR (SD/CA/04A) from the federal Belgian Science policy and the National Fund for Scientific Research, Belgium (F.R.S.-FNRS).

GMDD
6, 4009–4051, 2013
divand-1.0
A. Barth et al.

References

Arfken, G.: Mathematical Methods for Physicists, 3rd Edn., Academic Press, Orlando, FL, 795 pp., 1985. 4035

Barth, A., Alvera-Azcárate, A., Troupin, C., Ouberdous, M., and Beckers, J.-M.: A web interface for griding arbitrarily distributed in situ data based on Data-Interpolating Variational Analysis (DIVA), Adv. Geosci., 28, 29–37, doi:10.5194/adgeo-28-29-2010, 2010. 4013

Beckers, J.-M., Barth, A., and Alvera-Azcárate, A.: DINEOF reconstruction of clouded images including error maps – application to the Sea-Surface Temperature around Corsican Island, Ocean Sci., 2, 183–199, doi:10.5194/os-2-183-2006, 2006. 4011

Brankart, J.-M. and Brasseur, P.: Optimal analysis of in situ data in the Western Mediterranean using statistics and cross-validation, J. Atmos. Ocean. Tech., 13, 477–491, doi:10.1175/1520-0426(1996)013<0477:OAOISD>2.0.CO;2, 1996. 4012, 4013

Brankart, J.-M. and Brasseur, P.: The general circulation in the Mediterranean Sea: a climatological approach, J. Marine Syst., 18, 41–70, doi:10.1016/S0924-7963(98)00005-0, 1998. 4013

Brasseur, P.: Reconstitution de champs d'observations océanographiques par le Modèle Variationnel Inverse: Méthodologie et Applications, Ph.D. thesis, Université de Liège, Collection des publications, Sciences appliquées, Liège, Belgium, 1994. 4016

Brasseur, P. and Haus, J.: Application of a 3-D variational inverse model to the analysis of ecohydrodynamic data in the Northern Bering and Southern Chuckchi Seas, J. Marine Syst., 1, 383–401, doi:10.1016/0924-7963(91)90006-G, 1991. 4013

Brasseur, P., Beckers, J.-M., Brankart, J.-M., and Schoenauen, R.: Seasonal temperature and salinity fields in the Mediterranean Sea: climatological analyses of an historical data set, Deep-Sea Res., 43, 159–192, doi:10.1016/0967-0637(96)00012-X, 1996. 4012, 4013, 4020

Bretherton, F. P., Davis, R. E., and Fandry, C. B.: A technique for objective analysis and design of oceanographic experiment applied to MODE-73, Deep-Sea Res., 23, 559–582, doi:10.1016/0011-7471(76)90001-2, 1976. 4011

Chaves, J. C., Nehrbass, J., Guilfoos, B., Gardiner, J., Ahalt, S., Krishnamurthy, A., Unpingco, J., Chalker, A., Warnock, A., and Samsi, S.: Octave and Python: high-level scripting languages productivity and performance evaluation, in: HPCMP Users Group Conference, 2006, IEEE, 429–434, 2006. 4027

GMDD
6, 4009–4051, 2013

Courtier, P.: Dual formulation of four-dimensional variational assimilation, Q. J. Roy. Meteorol. Soc., 123, 2449–2461, doi:10.1002/qj.49712354414, 1997. 4024
Courtier, P., Thépaut, J.-N., and Hollingsworth, A.: A strategy for operational implementation of 4D-Var, using an incremental approach, Q. J. Roy. Meteorol. Soc., 120, 1367–1387, doi:10.1002/qj.49712051912, 1994. 4022
Courtier, P., Andersson, E., Heckley, W., Pailleux, J., Vasiljevic, D., Hamrud, M., Hollingsworth, A., Rabier, F., and Fisher, M.: The ECMWF implementation of three-dimensional variational assimilation (3D-Var), Part 1: Formulation, Q. J. Roy. Meteorol. Soc., 124, 1783–1808, doi:10.1002/qj.49712455002, 1998. 4021
Davis, T. A.: A column pre-ordering strategy for the unsymmetric-pattern multifrontal method, ACM T. Math. Software, 30, 165–195, doi:10.1145/992200.992205, 2004a.
Davis, T. A.: Algorithm 832: UMFPACK, an unsymmetric-pattern multifrontal method, ACM T. Math. Software, 30, 196–199, doi:10.1145/992200.992206, 2004b.
Dobricic, S. and Pinardi, N.: An oceanographic three-dimensional variational data assimilation scheme, Ocean Model., 22, 89–105, doi:10.1016/j.ocemod.2008.01.004, 2008. 4012
Evensen, G.: Data Assimilation: the Ensemble Kalman Filter, Springer, 279 pp., 2007. 4011
Gandin, L. S.: Objective Analysis of Meteorological Fields, Israel Program for Scientific Translation, Jerusalem, 242 pp., 1965. 4011
Golub, G. H. and Van Loan, C. F.: Matrix Computations, 3rd edn., Johns Hopkins University Press, Baltimore, 1996. 4024
Høyer, J. L. and She, J.: Optimal interpolation of sea surface temperature for the North Sea and Baltic Sea, J. Marine Syst., 65, 176–189, doi:10.1016/j.jmarsys.2005.03.008, 2007. 4011
Kaplan, A., Kushnir, Y., Cane, M., and Blumenthal, M.: Reduced space optimal analysis for historical datasets: 136 years of Atlantic Sea surface temperatures, J. Geophys. Res., 102, 27835–27860, doi:10.1029/97JC01734, 1997. 4011
Legler, D. M. and Navon, I. M.: VARIATM-A FORTRAN program for objective analysis of pseudostress wind fields using large-scale conjugate-gradient minimization, Comput. Geosci., 17, 1–21, doi:10.1016/0098-3004(91)90077-Q, 1991. 4012
Leros, A., Andreatos, A., and Zagorianos, A.: Matlab-Octave science and engineering benchmarking and comparison, in: Proceedings of the 14th WSEAS international conference on Computers: part of the 14th WSEAS CSCC multiconference, 746–754, 2010. 4027

GMDD
6, 4009–4051, 2013

Mathiot, P., Goosse, H., Fichefet, T., Barnier, B., and Gallée, H.: Modelling the seasonal variability of the Antarctic Slope Current, Ocean Sci., 7, 455–470, doi:10.5194/os-7-455-2011, 2011. 4030

Moore, A. M., Arango, H. G., Broquet, G., Edwards, C., Veneziani, M., Powell, B., Foley, D., Doyle, J. D., Costa, D., and Robinson, P.: The Regional Ocean Modeling System (ROMS) 4-dimensional variational data assimilation systems: Part 2 – Performance and application to the California Current System, Prog. Oceanogr., 91, 50–73, doi:10.1016/j.pocean.2011.05.003, 2011a. 4024

Moore, A. M., Arango, H. G., Broquet, G., Powell, B. S., Weaver, A. T., and Zavala-Garay, J.: The Regional Ocean Modeling System (ROMS) 4-dimensional variational data assimilation systems: Part 1 – System overview and formulation, Prog. Oceanogr., 91, 34–49, doi:10.1016/j.pocean.2011.05.004, 2011b. 4012, 4024

Nardelli, B. B., Colella, S., Santoleri, R., Guarracino, M., and Kholod, A.: A re-analysis of Black Sea surface temperature, J. Marine Syst., 79, 50–64, doi:10.1016/j.jmarsys.2009.07.001, 2010. 4011

Nelder, J. A. and Mead, R.: A simplex method for function minimization, Comput. J., 7, 308–313, doi:10.1093/comjnl/7.4.308, 1965. 4032

Rabier, F. and Courtier, P.: Four-dimensional assimilation in the presence of baroclinic instability, Q. J. Roy. Meteorol. Soc., 118, 649–672, doi:10.1002/qj.49711850604, 1992. 4022

Rabier, F., Järvinen, H., Klinker, E., Mahfouf, J.-F., and Simmons, A.: The ECMWF operational implementation of four-dimensional variational assimilation, Part 1: Experimental results with simplified physics, Q. J. Roy. Meteorol. Soc., 126, 1143–1170, doi:10.1002/qj.49712656415, 2000. 4012

Rayner, N. A., Parker, D. E., Horton, E. B., Folland, C. K., Alexander, L. V., Rowell, D. P., Kent, E. C., and Kaplan, A.: Global analyses of sea surface temperature, sea ice, and night marine air temperature since the late nineteenth century, J. Geophys. Res.-Atmos., 108, 4407, doi:10.1029/2002JD002670, 2003. 4011

Robinson, A. R.: Physical processes, field estimation and an approach to interdisciplinary ocean modeling, Earth-Sci. Rev., 40, 3–54, doi:10.1016/0012-8252(95)00030-5, 1996. 4011

Troupin, C., Machin, F., Ouberdous, M., Sirjacobs, D., Barth, A., and Beckers, J.-M.: High-resolution climatology of the North-East Atlantic using Data-Interpolating Variational Analysis (Diva), J. Geophys. Res., 115, C08005, doi:10.1029/2009JC005512, 2010. 4012, 4020

Troupin, C., Barth, A., Sirjacobs, D., Ouberdous, M., Brankart, J.-M., Brasseur, P., Rixen, M., Alvera-Azcárate, A., Belounis, M., Capet, A., Lenartz, F., Toussaint, M.-E., and Beckers, J.-M.: Generation of analysis and consistent error fields using the Data Interpolating Variational Analysis (DIVA), Ocean Model., 52–53, 90–101, doi:10.1016/j.ocemod.2012.05.002, 2012. 4012, 4013

Wahba, G. and Wendelberger, J.: Some new mathematical methods for variational objective analysis using splines and cross validation, Mon. Weather Rev., 108, 1122–1143, 1980. 4017

Yaremchuk, M. and Sentchev, A.: Mapping radar-derived sea surface currents with a variational method, Cont. Shelf Res., 29, 1711–1722, doi:10.1016/j.csr.2009.05.016, 2009. 4012

Yari, S., Kovačević, V., Cardin, V., Gačić, M., and Bryden, H. L.: Direct estimate of water, heat, and salt transport through the Strait of Otranto, J. Geophys. Res.-Oceans, 117, C09009, doi:10.1029/2012JC007936, 2012. 4012

GMDD
6, 4009–4051, 2013

Table 1. Kernel as a function of non-dimensional radius $\rho = |\mathbf{L}^{-1}\boldsymbol{x}| \geq 0$ for different values of the dimension n and the highest derivative m.

	$m = 1$	$m = 2$	$m = 3$
$n = 1$	$e^{-\rho}$	$(1+\rho)e^{-\rho}$	$(1+\rho(1+\rho/3))e^{-\rho}$
$n = 2$	–	$\rho \mathrm{K}_1(\rho)$	$\frac{\rho^2}{2}\mathrm{K}_2(\rho)$
$n = 3$	–	$e^{-\rho}$	$(1+\rho)e^{-\rho}$
$n = 4$	–	–	$\rho \mathrm{K}_1(\rho)$
$n = 5$	–	–	$e^{-\rho}$

GMDD
6, 4009–4051, 2013

Table 2. Normalization coefficient $c^{n,m}$ for different values of the dimension n and the highest derivative m.

	$m=1$	$m=2$	$m=3$
$n=1$	2	4	$\frac{16}{3}$
$n=2$	–	4π	8π
$n=3$	–	8π	32π
$n=4$	–	–	$32\pi^2$
$n=5$	–	–	$64\pi^2$

GMDD
6, 4009–4051, 2013

Table 3. Radial distance where the kernel is 1/2 for different values of ν.

ν	r_h
1/2	0.69315
1	1.25715
3/2	1.67835
2	2.02700
5/2	2.33026

GMDD
6, 4009–4051, 2013

Table 4. Run time in seconds for different domain sizes for MATLAB (R2012b) and octave version 3.6.4 (using GotoBLAS2 or MKL) for the primal algorithm with Cholesky factorization.

	MATLAB-R2012b	Octave-GotoBLAS2	Octave-MKL
100	0.304	0.415	0.451
200	1.694	1.808	1.692
300	5.235	5.155	5.088
400	7.527	8.104	8.388
500	15.501	13.905	14.115
600	31.457	24.906	25.156

GMDD
6, 4009–4051, 2013

Table 5. Run time in seconds for different domain sizes for MATLAB (R2012b) and octave version 3.6.4 (using GotoBLAS2 or MKL) for the dual algorithm.

	MATLAB-R2012b	Octave-GotoBLAS2	Octave-MKL
30	0.039	0.111	0.145
50	0.122	0.161	0.203
100	3.021	2.194	1.879
200	60.685	42.194	35.367
300	443.830	275.641	230.988

GMDD
6, 4009–4051, 2013

Table 6. Summary of all experiments with the optimal parameter values.

Experiment	λ	L_X (km)	L_T (month)	advection	RMS (°C)	skill-score (%)
2-D	14.0	1072	–	–	1.1501	0
2-D, advection	71.2	1171	–	7.22	0.9696	29
3-D	27.0	1397	4.9	–	0.9822	27
3-D, fractional time	29.5	1373	4.7	–	0.9820	27
3-D, fractional time, advection	53.8	1477	4.6	1.19	0.8589	44

GMDD
6, 4009–4051, 2013

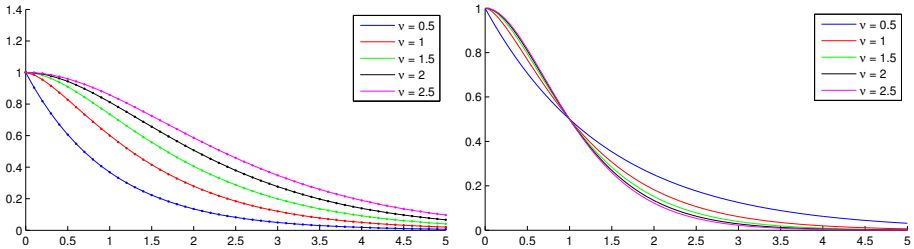

Fig. 1. Solid lines show the analytical kernels for different values of ν and the dots show the numerical kernel (left) and analytical kernels with scaled r_h (right).

GMDD
6, 4009–4051, 2013

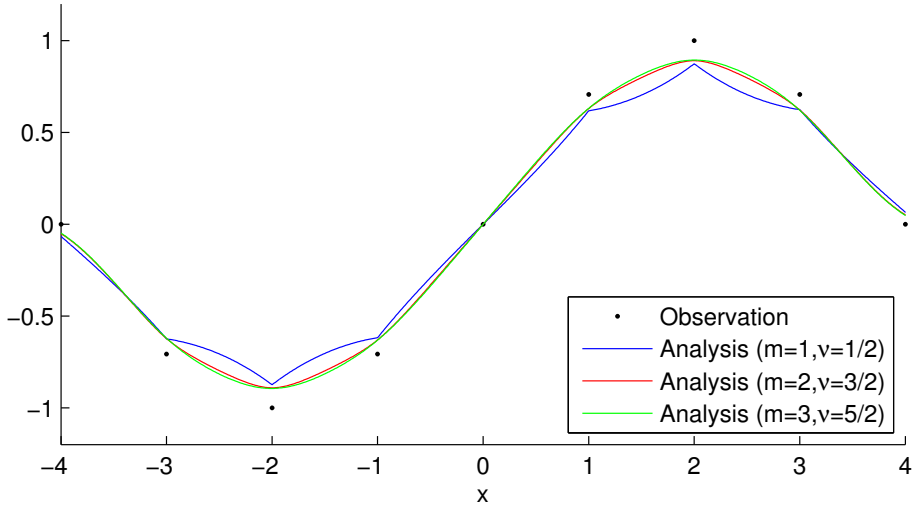

Fig. 2. Impact of higher-order derivatives.

GMDD
6, 4009–4051, 2013

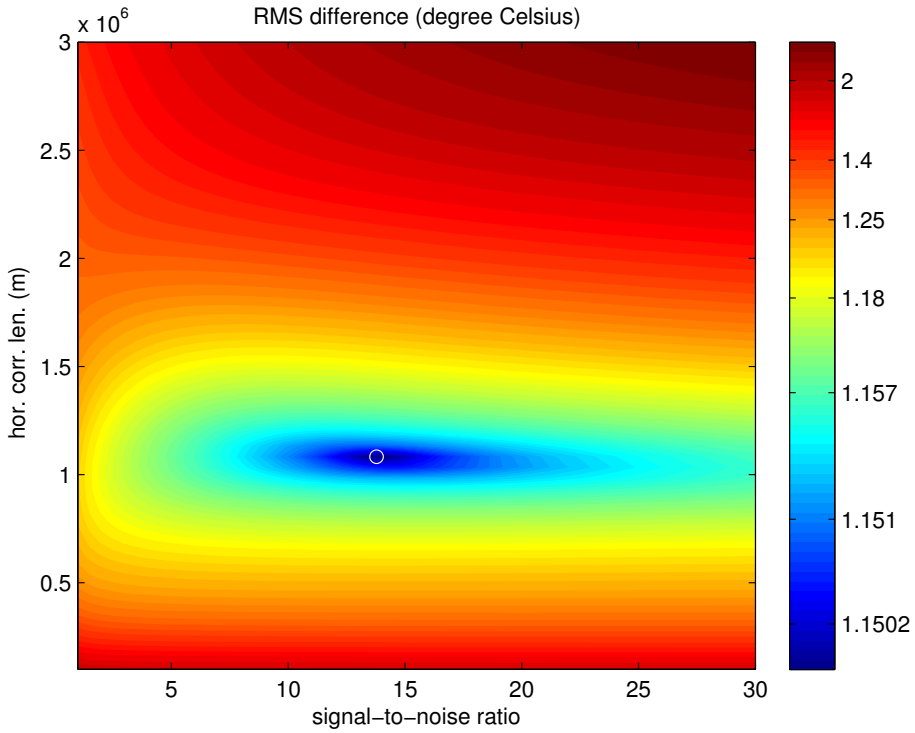

Fig. 3. RMS difference between the reference climatology and the analysis for different values of signal-to-noise ratio and correlation length. A non-linear color-map is used en enhance detail near the minimum.

GMDD
6, 4009–4051, 2013

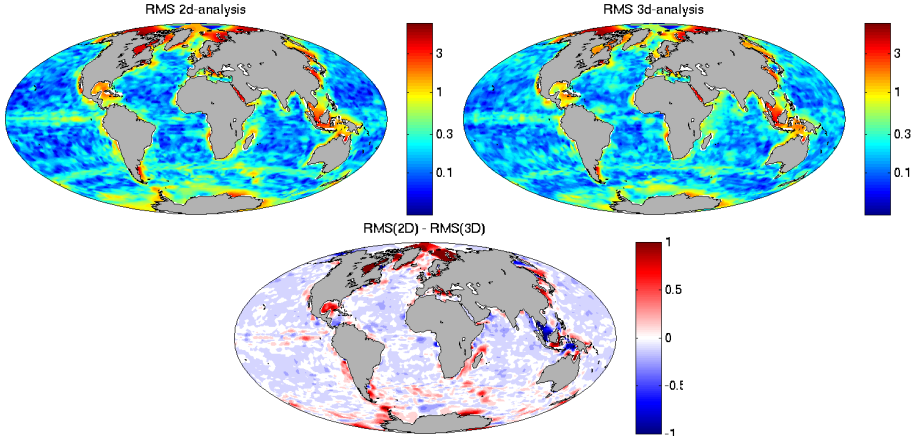

Fig. 4. Top left: RMS difference (averaged over time) between 2-D analysis and the model reference climatology. Top right: idem for 3-D. Bottom: difference of RMS error of the 2-D analysis minus the RMS error of the 3-D analysis.

GMDD
6, 4009–4051, 2013

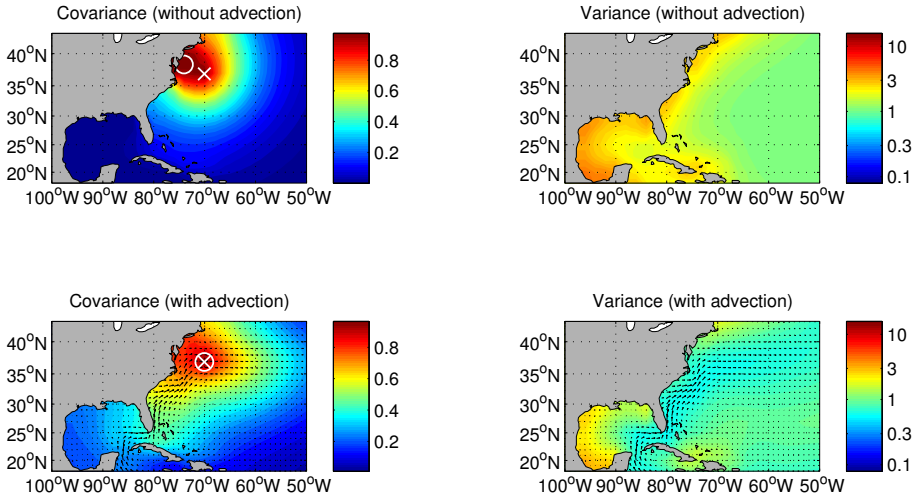

Fig. 5. 2-D case: background error covariance (left panels) relative to the location marked by a cross, and surrounding grid points and background variance (right panels). The upper (lower) panels correspond to the case without (with) advection constraint. The circle indicates the grid point with the highest covariance.

GMDD
6, 4009–4051, 2013

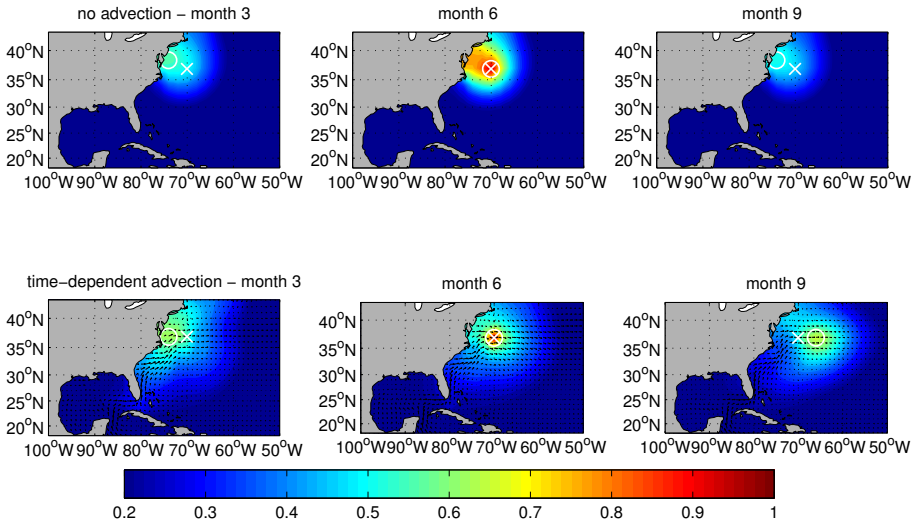

Fig. 6. 3-D case: background error covariance without (upper row) and with advection constraint (lower row) for a data point located at the cross and at month 6.

GMDD
6, 4009–4051, 2013