Variational assimilation of Lagrangian data in oceanography

Maëlle Nodet

Variational assimilation of Lagrangian data in oceanography

Maëlle Nodet

Laboratoire Jean-Alexandre Dieudonné, Université de Nice, Parc Valrose, 06108 Nice cedex 2, France
E-mail: nodet@math.unice.fr

Abstract. We consider the assimilation of Lagrangian data into a primitive equation circulation model of the ocean at basin scale. The Lagrangian data are positions of floats drifting at fixed depth. We aim at reconstructing the four-dimensional space-time circulation of the ocean. This problem is solved using the four-dimensional variational technique and the adjoint method. In this problem the control vector is chosen as being the initial state of the dynamical system. The observed variables, namely the positions of the floats, are expressed as function of the control vector via a nonlinear observation operator. This method has been implemented and has the ability to reconstruct the main patterns of the oceanic circulation. Moreover it is very robust with respect to increase of time-sampling period of observations. We have run many twin experiments in order to analyze the sensitivity of our method to number of floats, time-sampling period and vertical drift level. We compare also the performances of the Lagrangian method to that of the classical Eulerian one.

1. Introduction

The world's oceans play a crucial role in governing the earth's weather and climate. Lack of data has been a serious problem in oceanography for a long time. Since ten years the number of observations has greatly increased, with the availability of satellite altimeter data (ie measurements of the free-surface height of the ocean) from Geosat, Topex/Poseidon, Jason and other satellites. Assimilation of these satellite data has been widely investigated, see for example De Mey (1997) and Luong *et al* (1998), and thus it enables the development of operational oceanographic forecasting tools, like for example the Mercator Assimilation System (SAM). In addition to these remote-sensing data we have in-situ data, from scientific ships, surface mooring buoys or Lagrangian drifting buoys. Among these observations, Lagrangian data, ie positions of drifting floats, play a relevant role for many reasons: firstly their horizontal coverage is very wide (the whole Atlantic Ocean, for example), secondly they give information about currents in depth which are complementary to surface information given by satellite altimeters. For these reasons many national and international programs are organized to deploy drifting floats in the world's oceans.
There are different types of drifting buoys. In the framework of ocean basin scaled localized experiments, oceanographers have datasets from acoustic floats. These floats emit acoustic signals which are recorded by moored listening stations, and the floats positions are calculated every six hours by triangulation. Large datasets are available especially in the Atlantic Ocean (SAMBA, ARCANE-Eurofloat, ACCE experiments).

On a larger scale Argo floats are deployed in the world's oceans (1 617 floats on the 25th January 2005, 3 000 planned) in order to provide vertical temperature and salinity profiles. These floats provide also Lagrangian information, which are their positions every ten days. Indeed they drift freely at a predetermined parking depth (around 1 000 meters), every ten days they descent to begin profiles from greater depth (2 000 meters) then they go back to the surface and they record temperature and salinity profiles during ascent. On the surface they transmit data to satellite and they are located by GPS. Thus many different floats networks and Lagrangian datasets are available.

In parallel, modeling of the ocean system has greatly improved in both quality and realism, and there are many Ocean Global Circulation Models (OGCM), like for example the OPA PArallelized Ocean model (see Madec *et al* 1998). A crucial issue for oceanographers is then to take the best advantage of different types of information included in models to one hand and in various observations to the other hand. Data Assimilation (DA) covers all theoretical and numerical mathematical methods which allow to blend as optimally as possible all sources of information (see the review by Ide *et al* 1997). There are two main categories of DA methods: variational methods based on optimal control theory (Lions 1968) and statistical ones based on optimal statistical estimation. Adjoint method is the prototype of variational methods, introduced in meteorology by Penenko and Obraztsov (1976). Its effective implementation in the framework of atmospheric Data Assimilation, namely four dimensional variational assimilation (4D-Var), has been studied by Le Dimet and Talagrand (1986). Introduction of 4D-Var in oceanography is even more recent (see Thacker and Long 1988, Sheinbaum and Anderson 1990). The prototype of sequential methods is the Kalman filter, introduced in oceanography by Ghil (1989) (see also the review by Ghil and Malanotte-Rizzoli 1991).

Assimilation of Lagrangian data is in the pipeline. Kamachi and O'Brien (1995) used the adjoint method in a Shallow-Water model with upper-layer thickness as control vector. More recently Mead (2004) has implemented a variational method based on the use of Lagrangian coordinates for Shallow-Water equations. Molcard *et al* (2003) and Özgökmen *et al* (2003) implemented optimal interpolation (which is a simple sequential method) in a reduced-gravity quasi-geostrophic model and in a primitive equations model; their method is based on conversion of Lagrangian data into velocity information. These teams have used simulated data in the twin experiments approach: they generate Lagrangian observations from a known "true state", they choose a "first guess" (which is often the result of a previous assimilation process) and they compute an "assimilated state" thanks to information included in both model and data. Beside these studies Filatoff and Assenbaum (2003) assimilate real Argo floats data into a very high resolution model thanks to optimal interpolation.

In this paper we implemented a variational method into a primitive equations model. The aim of variational assimilation methods is to identify the initial state of an evolution problem which minimizes a cost function. This cost function represents the difference between observations and their corresponding model variables. This functional is minimized using a gradient descent algorithm. The gradient is computed by integration of the adjoint model. Thanks to this formulation there is no need to convert Lagrangian data into velocities data: we can use directly the position observations, although they are not variables of the ocean model, but nonlinear functions of the state variables. Moreover this method is a four dimensional one because the temporal dimension of the observations, ie their Lagrangian nature, is

taken into account. But consequently the cost function is not necessarily convex: it involves a so-called observation operator, which links the state variables (here the velocities) and the observed data (here the positions of drifting particles) and this operator is nonlinear. Thus we used an incremental method (see Courtier *et al* 1994) in order to achieve the minimization.
We implement our method using the primitive equations model OPA into an idealized basin. Indeed it seems relevant to use a realistic ocean model like OPA but not to deal with physical difficulties like basin geometry or open boundary conditions. Furthermore OPA capacities enable to evolve toward a more realistic basin. Then we use the twin experiments approach because in this framework we know exactly the system true state and so we are able to quantify the efficiency of our method by comparing assimilated and true states. We choose also to assimilate simulated data for many reasons: firstly Argo floats have not been launched to provide Lagrangian information and the feasibility of exploiting their positions is not ensured. Secondly Lagrangian datasets (from Argo to acoustic floats) are very diversified in terms of number of floats, time-sampling period of the observations and drifting depth. And thirdly using real data is not easy because of the drift problems during ascent and descent for the Argo floats.
The paper is organized as follows: in section 2 we describe the continuous and discrete models and Lagrangian simulated data. In section 3 we present the assimilation method and its implementation. Some numerical experiments are given and commented in section 4. Finally conclusions are given in section 5.

2. Physical model and Lagrangian data

2.1. The Primitive Equations of the ocean

The ocean circulation model used in our study is a Primitive Equations (PE) model. These equations are derived from Navier Stokes equations (mass conservation and momentum conservation, included the Coriolis force) coupled with a state equation for water density and heat equation, under Boussinesq and hydrostatic approximations (for more details see Lions *et al* 1992 and Temam and Ziane 2004).
These equations are written as

$$\begin{cases} \partial_t u - b\Delta u + (U.\nabla_2)u + w\partial_z u - av + \partial_x p = 0 & \text{in } \Omega \times (0,t_f) \\ \partial_t v - b\Delta v + (U.\nabla_2)v + w\partial_z v + au + \partial_y p = 0 & \\ \partial_z p - gT = 0 & \\ \partial_t T - b\Delta T + (U.\nabla_2)T + w\partial_z T + fw = 0 & \text{in } \Omega \times (0,t_f) \\ w(x,y,z) = -\int_0^z \partial_x u(x,y,z') + \partial_y v(x,y,z')\, dz' & \text{in } \Omega \times (0,t_f) \\ U(t=0) = U_0, \qquad T(t=0) = T_0 & \text{in } \Omega \end{cases} \quad (1)$$

where

- $\Omega = \Omega_2 \times (0,1)$ is the circulation basin, where Ω_2 is a regular bounded open subset of $\mathbb{R}^2$, x and y are the horizontal variables and $z \in (0,1)$ is the vertical one, $(0,t_f)$ is the time interval;
- $U = (u,v)$ is the horizontal velocity, w is the vertical velocity, T the temperature and p the pressure;
- $U_0 = (u_0, v_0)$ and T_0 the initial conditions;
- $(\nabla_2.)$ is the horizontal divergence operator and $\Delta = \partial_{xx} + \partial_{yy} + \partial_{zz}$ the 3-D Laplace operator;

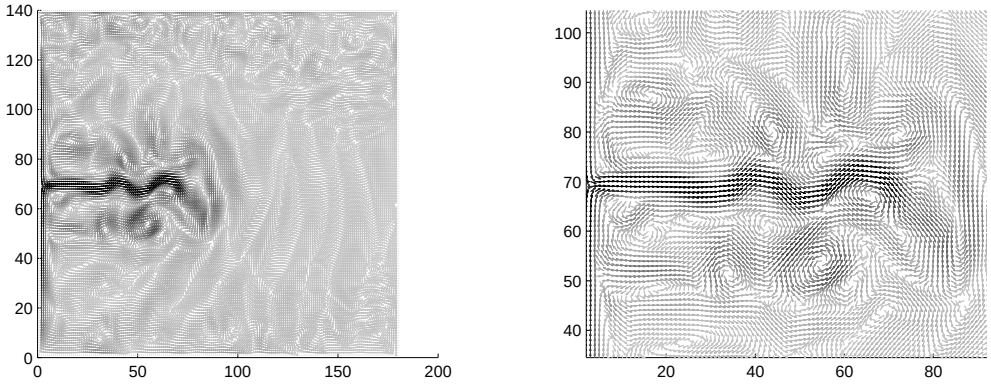

Figure 1. True state : horizontal surface velocity.

- a, b, f, g are physical constants.

The space boundary conditions are

$$\begin{cases} \partial_z u = \tau_u, \quad \partial_z v = \tau_v, \quad T = 0 \quad \text{on } \Gamma_t \\ u = 0, \qquad v = 0, \qquad T = 0 \quad \text{on } \partial\Omega \setminus \Gamma_t \\ \int_{z=0}^{1} \partial_x u + \partial_y v \, dz = 0 \quad \text{in } \Omega_2 \end{cases} \tag{2}$$

where $\tau = (\tau_u, \tau_v)$ is the stationary wind-forcing, $\partial\Omega$ is the boundary of Ω and $\Gamma_t = \Omega_2 \times \{z = 1\}$ is its top boundary.

2.2. Model and configuration

We are using the OPA ocean circulation model developed by LODYC (see Madec *et al* 1998), in its 8.1 version. OPA is a flexible model and can be used either in regional or in global ocean configuration. The prognostic variables are the three-dimensional velocity field (u, v, w) and the thermohaline variables T and S. Discretization is based on finite differences in space and time (leap-frog scheme in time). Various physical choices are available to describe ocean physics.

The characteristics of our configuration are as follows:

- The domain is $\Omega = (0, l) \times (0, L) \times (0, H)$ (longitude, latitude, depth), with $l = 2800$ km, $L = 3600$ km and $H = 5000$ m. It extends from -56^o to -24^o West longitude, and from 22.5^o to 47.5^o North latitude.

- The horizontal resolution is 20 km, there are 11 vertical levels, so that the number of grid points is $180 \times 140 \times 11 = 277200$.

- The time step is 1200 seconds.

The model is integrated for 25 years until a statistically steady-state is reached, which is our "true state" for the twin experiments. Figure 1 shows its horizontal velocity field at the surface, on the whole horizontal grid on the left and on a reduced grid on the right. Dark gray vectors represent large velocities.

2.3. Lagrangian data

Lagrangian data are positions of drifting floats. These floats drift at fixed depth, ie in the horizontal plane $z = z_0$. We denote by $\xi(t) = (\xi_1, \xi_2)(t)$ the position of one float

at time t in the plane $z = z_0$. $\xi(t)$ is the solution of the following differential equation:

$$\begin{cases} \dfrac{d\xi}{dt} &= U(t, \xi(t), z_0) \\ \xi(0) &= \xi_0 \end{cases} \tag{3}$$

where $U = (u, v)$ is the horizontal velocity of the flow and ξ_0 the initial position of the float. It is important to notice that the mapping $U \mapsto \xi$, which links the variables of the model and the Lagrangian observations is non-linear.
In the twin experiments approach we use the true state to simulate the "observed data". We integrate numerically the equation (3) using a leapfrog scheme. This requires the velocity U along the trajectory of the float (ie out of the grid). To achieve this we use the following continuous 2D interpolation 'interp$(U, (x, y))$' of the vector field U at the point (x, y):

$$\begin{aligned} &x_1 = \lfloor x \rfloor, && y_1 = \lfloor y \rfloor, \\ &u_1 = U(x_1, y_1), && u_2 = U(x_1 + 1, y_1), \\ &u_3 = U(x_1, y_1 + 1), && u_4 = U(x_1 + 1, y_1 + 1), \end{aligned}$$

$$\begin{aligned} \text{interp}(U, (x, y)) = {} & u_1 + (u_2 - u_1)(x - x_1) + (u_3 - u_1)(y - y_1) \\ & + (u_1 - u_2 - u_3 + u_4)(x - x_1)(y - y_1) \end{aligned}$$

where $\lfloor . \rfloor$ denotes the floor function, (x_1, y_1), (x_1+1, y_1), (x_1, y_1+1) and (x_1+1, y_1+1) are the grid points which are the nearest neighbors to (x, y). This function is piecewise affine with respect to x and y, continuous with respect to (x, y), linear with respect to u. Thus it is not differentiable in (x, y) everywhere. More precisely it is not differentiable at (x, y) if and only if $x = x_1$ or $y = y_1$. It will be a problem to derive the adjoint code, see paragraph 3.3. However it is accurate enough to approximate the solution of equation (3) and it is very costly to use a differentiable interpolation. Indeed differentiable interpolation at the point (x, y) requires the whole field u (ie the values of u at every grid point), like cubic splines for example. Obviously it cannot be implemented here because of the very high number of grid points.
Let us denote $\xi_k = (\xi_{1,k}, \xi_{2,k})$ the horizontal position of the float at time t_k, U the horizontal velocity of the fluid at time t_k, U_k the velocity at point ξ_k, and h the time step of the ocean model. The algorithm step is schematically

$$\begin{cases} \xi_k &= \xi_{k-2} + 2h\, U_{k-1} \\ U_k &= \text{interp}(U, \xi_k) \end{cases}$$

The dataset is $\{\xi_N, \xi_{2N}, \xi_{3N} ...\}$, where N is an integer. The duration between two data is thus the product of N by the time-step h of the code ($h = 1200$ seconds). We call this the time-sampling period. For example if $N = 72$ we have one data per float and per day and the time-sampling period is thus one day.
Figure 2 represents the data simulated by the algorithm with 2 000 floats for 10 days at level 4.

3. Description and implementation of the variational assimilation method

3.1. Description of the assimilation problem

Without loss of generality we assume that there is only one assimilated data: the dataset is the position $\xi(t_1) = (\xi_1(t_1), \xi_2(t_1))$ in the horizontal plane $z = z_0$ of a single

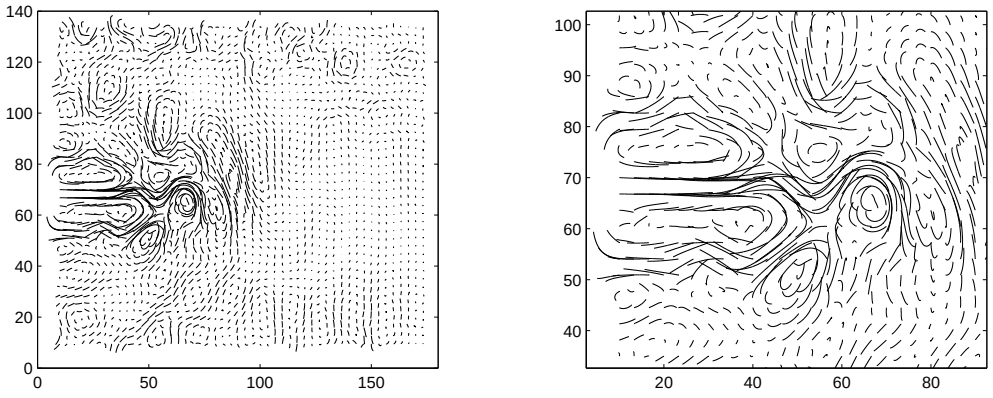

Figure 2. 2 000 trajectories at level 4.

float at a single time t_1. We denote by $\mathbf{y}^o = (\xi_1(t_1), \xi_2(t_1))$ this data. Our problem is to minimize the following cost function with respect to the control vector $\mathbf{x}$:

$$\begin{array}{rcccc} \mathcal{J}(\mathbf{x}) & = & \frac{1}{2}\|\mathcal{G}\mathcal{M}(\mathbf{x}) - \mathbf{y}^o\|^2 & + & \frac{\omega}{2}\|\mathbf{x} - \mathbf{x}^b\|^2_{\mathbf{B}} \\ & = & \mathcal{J}^o(\mathbf{x}) & + & \omega\,\mathcal{J}^b(\mathbf{x}) \end{array} \tag{4}$$

where

- the control vector $\mathbf{x} = (u_0, v_0, T_0)$ is the initial state vector,
- the $\mathbf{B}$-norm is calculated thanks to the background error covariance matrix $\mathbf{B}^{-1}$: $\|\psi\|^2_{\mathbf{B}} = \psi^T\mathbf{B}^{-1}\psi$,
- $\mathbf{x}^b$ is another initial state of the ocean, called the background or first guess, which is required to be close to the minimum $\bar{\mathbf{x}}$,
- $\mathcal{M}$ is the discrete ocean model and $\mathcal{M}(\mathbf{x})$ is the discrete state vector (one value per variable, per grid-point and per time step),
- $\mathcal{G}$ is the discrete non-linear observation operator, which links the state of the fluid (and especially the horizontal velocity U) with the data, $\mathcal{G}\mathcal{M}(\mathbf{x}) = \xi(t)$ where $\xi(t)$ is defined by equation (3),
- $\|.\|$ is the euclidean norm in $\mathbb{R}^2$,
- $\mathbf{y}^o$ is the observations vector.

Then $\mathcal{J}^o$ quantifies the difference between the observations and the state of the system, $\mathcal{J}^b$ is a regularization term. Thanks to this latter term, the inverse problem of finding the minimum $\mathbf{x}^*$ becomes well-posed. The parameter ω represents the relative weight of the regularization term with respect to the observation term and it must be chosen carefully.

3.2. Numerical variational assimilation : incremental 4D-Var

Four dimensional variational assimilation (4D-Var, see Le Dimet and Talagrand 1986) is an iterative numerical method which aims to approximate the solution $\mathbf{x}^*$ of discrete assimilation problems with cost function of type (4).

In 4D-Var a gradient descent algorithm is used to minimize the cost function, the gradient being obtained by solving the discrete adjoint equations. It is an efficient method but it is very costly when the direct model and the observation operator are

not linear, for at least two reasons. Firstly every iteration of the adjoint method requires one integration of the full non linear direct model and one integration of the adjoint of the linearized model. Secondly the cost function is not necessarily convex and the minimization process may converge to a local minimum, or it may take considerable time to converge, or may not converge at all.
Incremental 4D-Var (see Courtier *et al* 1994) avoids, to some extent, both of these problems. In this approach, the (non linear) model is approximated by a simplified linear model and the (non linear) observation operator is linearized around the background. The cost function becomes quadratical, it has a unique minimum and this minimum is assumed to be close to the one of the full non quadratical cost function. In this approach, the control vector becomes $\delta\mathbf{x} = \mathbf{x} - \mathbf{x}^b$ and the cost function:

$$\mathcal{J}(\delta\mathbf{x}) = \frac{1}{2}\|\mathbf{G}_{\mathcal{M}(\mathbf{x}^b)}\mathbf{M}_{\mathbf{x}^b}\delta\mathbf{x} - \mathbf{d}\|^2 + \frac{\omega}{2}\|\delta\mathbf{x}\|^2_{\mathbf{B}}$$

where $\mathbf{G}_{\mathcal{M}(\mathbf{x}^b)}$ is the observation operator linearized around $\mathcal{M}(\mathbf{x}^b)$, $\mathbf{M}_{\mathbf{x}^b}$ is the simplified model linearized around $\mathbf{x}^b$ and $\mathbf{d} = \mathbf{y}^o - \mathcal{G}\mathcal{M}(\mathbf{x}^b)$ is called the innovation vector. The gradient of $\mathcal{J}$ is obtained via the adjoint model $\mathbf{M}^T_{\mathbf{x}^b}$ and the adjoint observation operator $\mathbf{G}^T_{\mathcal{M}(\mathbf{x}^b)}$:

$$\nabla\mathcal{J}(\delta\mathbf{x}) = \mathbf{M}^T_{\mathbf{x}^b}\mathbf{G}^T_{\mathcal{M}(\mathbf{x}^b)}(\mathbf{G}_{\mathcal{M}(\mathbf{x}^b)}\mathbf{M}_{\mathbf{x}^b}\delta\mathbf{x} - \mathbf{d}) + \omega\mathbf{B}^{-1}\delta\mathbf{x}$$

In that case the minimization process converges quickly. Moreover this approach takes into account weak non-linearities, because the background $\mathbf{x}^b$ is updated once or twice, as we can see in the following algorithm:

Initialization: $\mathbf{x}^r = \mathbf{x}^b$ ($\mathbf{x}^r$ is called the reference state).
BEGINNING OF EXTERNAL LOOP

- Integration of the non-linear model: $\mathbf{x}^r(t_i) = \mathcal{M}(t_0, t_i)(\mathbf{x}^r)$
- Computation of the innovation vector $\mathbf{d}_i$ thanks to the non-linear observation operator: $\mathbf{d}_i = \mathbf{y}^o_i - \mathcal{G}\mathcal{M}(\mathbf{x}^r, t_i)$
  BEGINNING OF INTERNAL LOOP
  - Computation of $\mathcal{J}^o(\delta\mathbf{x})$ using the tangent linear model $\mathbf{M}$ and the linearized observation operator $\mathbf{G}$
  - Computation of the gradient $\nabla\mathcal{J}^o(\delta\mathbf{x})$ thanks to the adjoint model $\mathbf{M}^T$ and operator $\mathbf{G}^T$
  - Minimization via an optimal step gradient descent algorithm

  END OF INTERNAL LOOP
- Update of the assimilated increment $\delta\mathbf{x}^a = \delta\mathbf{x}$
- Update of the reference state $\mathbf{x}^r = \mathbf{x}^r + \delta\mathbf{x}^a$

END OF EXTERNAL LOOP
Computation of the assimilated state $\mathbf{x}^a = \mathbf{x}^r + \delta\mathbf{x}^a$, $\mathbf{x}^a(t_i) = \mathcal{M}(t_0, t_i)(\mathbf{x}^a)$.

3.3. Implementation in OPAVAR

The OPAVAR 8.1 package developed by Weaver *et al* (2003) includes the direct non linear model OPA 8.1 developed by LODYC, the so-called tangent linear model (which is in fact tangent to a simplified direct model), the adjoint model, a minimization module. Weaver has implemented a preconditioning through the $\mathbf{B}$ matrix, via the change of variables $\delta\mathbf{w} = \mathbf{B}^{-1/2}\delta\mathbf{x}$ (following the method introduced by Courtier *et al*

1994). The observation operators of OPAVAR 8.1 are interpolation and projection operators. To assimilate Lagrangian data we have implemented the non linear observation operator (see section 2.3), its linearization around the reference trajectory and the adjoint of the linear observation operator.
To obtain the tangent and adjoint codes of the discrete observation operator we use the recipes for (hand-coding) adjoint code construction of Talagrand (1991) and Giering and Kaminski (1998). The algorithms are schematically:

- Direct code:
$$\begin{cases} \xi_k = \xi_{k-2} + 2h\, U_{k-1} \\ U_k = \mathrm{interp}(U, \xi_k) \end{cases}$$
where 'interp' is the interpolation function of U at point ξ (see section 2.3).
- Linear tangent code:
$$\begin{cases} \delta\xi_k = \delta\xi_{k-2} + 2h\, \delta U_{k-1} \\ \delta U_k = \mathrm{interp}(\delta U, \xi_k) + \delta\xi_k . \partial_{(x,y)} \mathrm{interp}(U, \xi_k) \end{cases}$$
where '$\partial_{(x,y)}$interp' is the derivative of the 'interp' function with respect to (x, y). The function 'interp' is linear with respect of U but it is not derivable in (x, y) at points with integer coordinates. Thus we have chosen the values of that derivative at these points, using finite centered differences.
- We derive finally the adjoint code:
$$\begin{cases} \mathrm{CALL\ adjinterp}(\mathrm{adj}U, \xi_k, \mathrm{adj}U_k) \\ \mathrm{adj}\xi_k = \mathrm{adj}\xi_k + \mathrm{adj}U_k\, \partial_{(x,y)} \mathrm{interp}(U, \xi_k) \\ \mathrm{adj}U_k = 0 \\ \\ \mathrm{adj}\xi_{k-2} = \mathrm{adj}\xi_{k-2} + \mathrm{adj}\xi_k \\ \mathrm{adj}U_{k-1} = \mathrm{adj}U_{k-1} + 2h\, \mathrm{adj}\xi_k \\ \mathrm{adj}\xi_k = 0 \end{cases}$$
where 'adj...' are the adjoint variables and the routine 'adjinterp' is obtained by transposing the function '$\partial_{(x,y)}$interp'.

4. Numerical results

In this section we present the results of our numerical experiments. We begin with a brief description of our choices.

Twin experiments. In these experiments we have assimilated only Lagrangian data and we assume that the true initial temperature and salinity were known. The assimilation window width is 10 days. The background $\mathbf{x}^b$ is the state of the ocean ten days before the true one. We ran the model (without assimilation) with the background as initial state and we obtained a reference state. This reference state (called background in the sequel) will be compared to assimilated state in order to quantify the efficiency of the assimilation process.

The $\mathbf{B}$ *matrix.* The choice of the $\mathbf{B}$ matrix is crucial because of its dual purpose (preconditioning and regularization). Firstly we have tested very simple matrices (identity, energy weights) and the results were quite bad. Then we have used the diffusion filter method (see Weaver and Courtier 2001) which gives good results.

Diagnostics. Our diagnostics are based on RMS error between the true velocity and the assimilated one, compared with the RMS error between the true velocity and the background one. The RMS error is plotted as a function of time or of the vertical level or of another parameter. For example, we have the following formula for the time-dependent RMS error:

$$\text{error}(u,t) = \Big(\frac{\sum_{i,j,k} |u_t(i,j,k,t) - u(i,j,k,t)|^2}{\sum_{i,j,k} |u_t(i,j,k,t)|^2} \Big)^{1/2} \tag{5}$$

where u_t is the true state, u the assimilated state (or the background), t is the time and (i,j,k) a grid point, where (i,j) are the horizontal coordinates and k the vertical one.
We made the following experiments: first experiment and diagnostics in section 4.1, sensitivity to the floats network parameters (time sampling of the position measurements, number of floats, drifting level, coupled impact of number and time sampling) in section 4.2, comparison with another variational method in section 4.3.

4.1. First experiment

We present here the results of a typical experiment. There are 3 000 floats drifting at level 4 in the ocean for 10 days. The Lagrangian data are collected once a day. Thus the total amount of data is $2 \times 3\,000 \times 10 = 60\,000$.
Figure 3 (on the left) shows the RMS error of the experiment as function of time, according to formula (5). We have put the error for the background on the same plot. Figure 3 (on the right) shows the total RMS error as a function of the vertical level (where 1 represents the surface and 10 the bottom), according to the formula:

$$\text{error}(u,k) = \Big(\frac{\sum_{i,j,t} |u_t(i,j,k,t) - u(i,j,k,t)|^2}{\sum_{i,j,t} |u_t(i,j,k,t)|^2} \Big)^{1/2}$$

We can see that the error with assimilation is twice lower than without. Moreover the assimilation process improves every vertical level and not only the 4th one. However the RMS error are quite large. It could be explained by the fact that the assimilation window width is only 10 days, which is quite short for this problem. However the computing cost is higher when we enlarge the window and we had to choose between performing a few experiments with a large window or a a lot of experiments with a small one. As our method is new, we have to analyze its sensitivity to various parameters and then we chose to reduce the assimilation window width.
Finally figure 4 shows the horizontal velocity field $U = (u,v)$ at level 1 at the final time. We can notice that the main patterns as the central jet and the bigger eddies are quite similar to the true ones.

4.2. Sensitivity to the floats network parameters

We have performed a lot of experiments to analyze the sensitivity to various parameters of our assimilation process. Indeed in the ocean the network's parameters can widely change, from Argo floats (1000-2000 meters depth, one data per 10 days) to acoustic floats (various depth, time sampling period around 6 hours) or drifters in the upper ocean (near surface, time sampling period very short)...
Here are the parameters that we consider:

- the time sampling period, varying from 6 hours to 10 days,

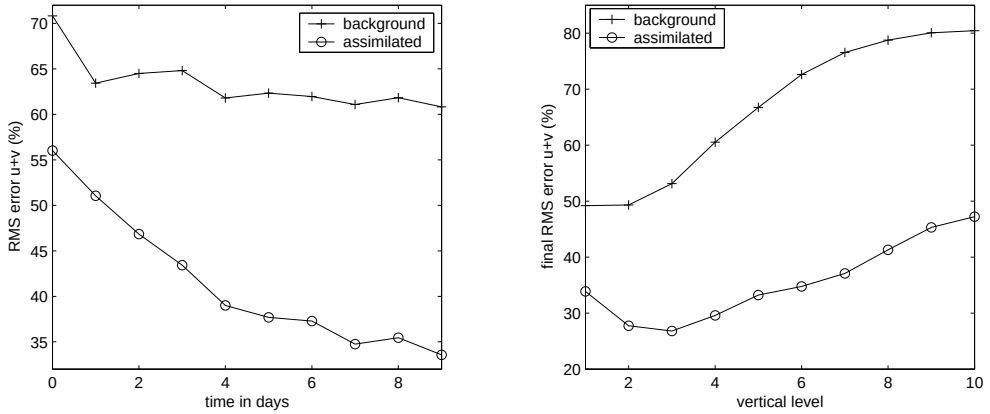

Figure 3. RMS error as a function of time and vertical level.

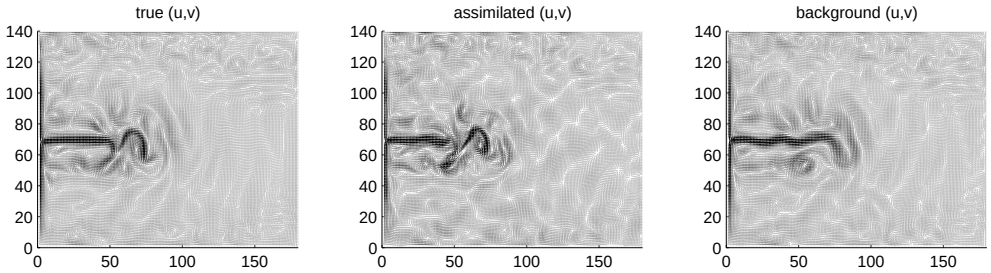

Figure 4. Horizontal velocity field at the final time and level 1

- the number of floats, varying from 300 to 3000,
- the vertical level of drift, varying from 1 (surface) to 10 (bottom),

We analyze also the coupled effect of the number and the time sampling period.

4.2.1. Sensitivity to the time sampling period. The framework of this experiment is the following: we performed seven different experiments with exactly the same initial conditions, namely 3 000 floats at level 4. The only difference in these experiments is the time sampling period, which will be denoted shortly by TSP in the sequel. The experiments are denoted by TSP-xxx where xxx is the time sampling period, in hours (h) or in days (d). Table 1 gives the total RMS error for the experiments, according to the formula:

$$\text{error}(u) = \Big(\frac{\sum_{i,j,k,t} |u_t(i,j,k,t) - u(i,j,k,t)|^2}{\sum_{i,j,k,t} |u_t(i,j,k,t)|^2} \Big)^{1/2} \tag{6}$$

Figure 5 shows the RMS error as a function of time for each TSP experiment, except (for readability) experiments TSP-6h and TSP-12h whose results were very close to those of TSP-1d. It shows also the total RMS error as a function of the time sampling period. We can see that our method is robust with respect to the increase of the time sampling period. This is very encouraging. Indeed every prior study is very sensitive

Table 1. Time sampling period sensitivity analysis.

Experiment	Final Error (%)
TSP-6h	35.2
TSP-12h	35.2
TSP-1d	35.2
TSP-2d	36.7
TSP-3d	36.1
TSP-5d	37.1
TSP-10d	38.6
Background	60.8

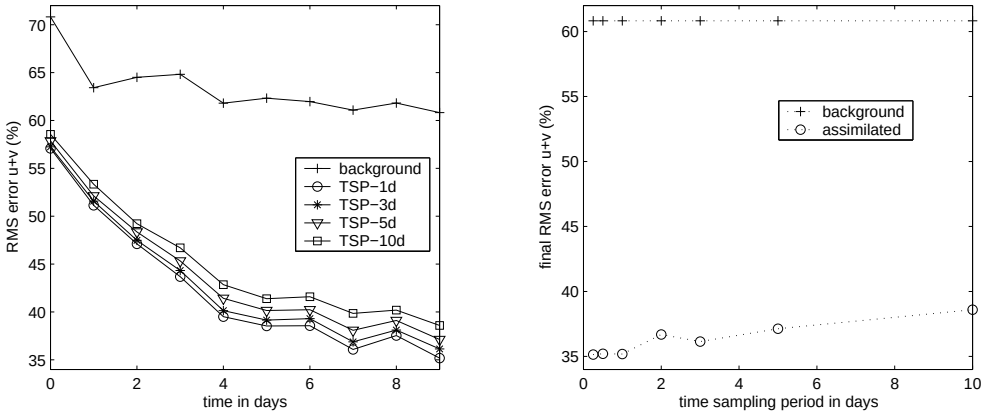

Figure 5. Sensitivity to the time sampling period.

to the TSP and shows quite bad results when the TSP is larger than 2 or 3 days (see Molcard 2003, Mead 2005 and section 4.3). Our method does not show this sensitivity, velocities are quite well reconstructed even when the TSP is large and especially with an Argo-like 10 days period, which is a very positive result.

4.2.2. Sensitivity to the number of floats. We perform five experiments with varying number of floats drifting at the same vertical level (4) and with positions sampled with the same period (6 hours). The experiments are denoted by NUM-xxx, where xxx is the number of floats. Table 2 gives the total RMS error (see formula (6)) for each experiment. Figure 6 shows the RMS error as a function of time and the total RMS error as a function of the number of floats. We can see that the number of floats has great influence on the results. Under a minimal number (1 000) the velocities are badly reconstructed, undoubtedly because there is not enough information to constrain the flow. The results are optimal for a 1 000 floats network and they don't improve with higher numbers. Obviously the information becomes redundant and it is useless to add floats.

4.2.3. Sensitivity to the vertical drift level. Again we perform seven experiments with 3 000 floats drifting at varying vertical level and fixed TSP (6 hours). As usually we

Table 2. Number of floats sensitivity analysis.

Experiment	Final Error (%)
NUM-300	55.7
NUM-500	43.7
NUM-1000	35.2
NUM-2000	36.5
NUM-3000	36.2
Background	63.3

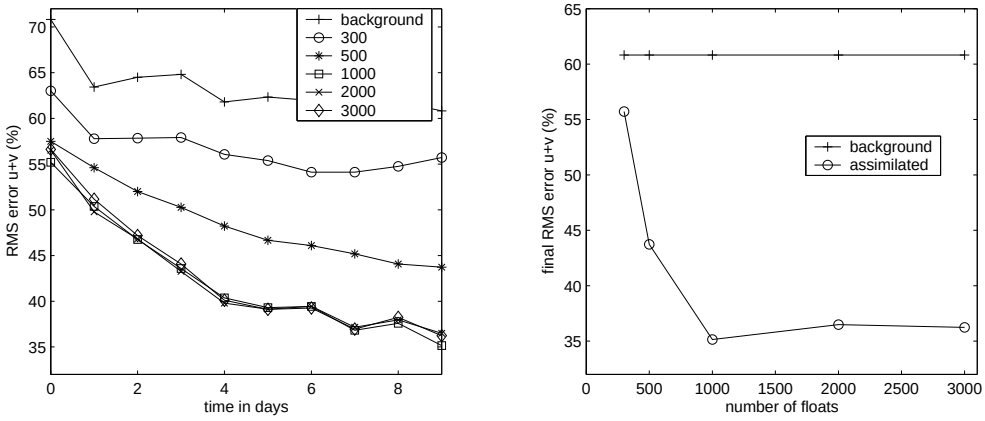

Figure 6. Sensitivity to the number of floats.

denote by LEV-x the experiment involving floats at level x. Table 3 describes the experiments and gives the total RMS errors. Figure 7 shows the RMS error as a function of time for upper levels on the left and lower levels on the right. Again the results are very sensitive to the position of the floats. The three best levels are 3, 4 and 5, ie the intermediate levels. From a physical point of view it is coherent because the information propagates vertically with a finite velocity so that very upper (1, 2) and very lower (7 to 10) levels are penalized. Moreover upper levels (1 to 4) are the most energetic ones (from the kinetic turbulent energy point of view), quasi ten times more than the lower ones (levels 5 to 10), it seems quite natural that the best results are obtained with floats drifting at level 4 which is both intermediate and energetic.

4.2.4. Coupled impact of number of floats and time sampling period. Here we look at the coupled effect of varying number of floats and varying TSP, for example in order to answer the following question: is the total number of data an important variable to measure the efficiency of the assimilation? So we perform nine experiments, denoted by nnn-xxx where nnn is the number of floats and xxx is the TSP. These experiments and their final RMS error are described in Table 4. Figure 8 represent the RMS error as a function of time for the 500-xxx experiments on the left, 1000-xxx in the middle and 2000-xxx on the right with the same scale on the axis of ordinates. The results are complementary to the precedent experiments. Indeed we can see that 1 000 is an optimal number for this configuration whatever the TSP and that our method is

Table 3. Vertical drift level sensitivity analysis.

Experiment	Total Error (%)	Final Error (%)
LEV-1	59.2	55.5
LEV-2	56.8	52.0
LEV-3	49.8	46.7
LEV-4	42.3	35.2
LEV-5	53.6	44.1
LEV-7	59.9	52.5
LEV-10	57.3	47.0
Background	63.3	60.8

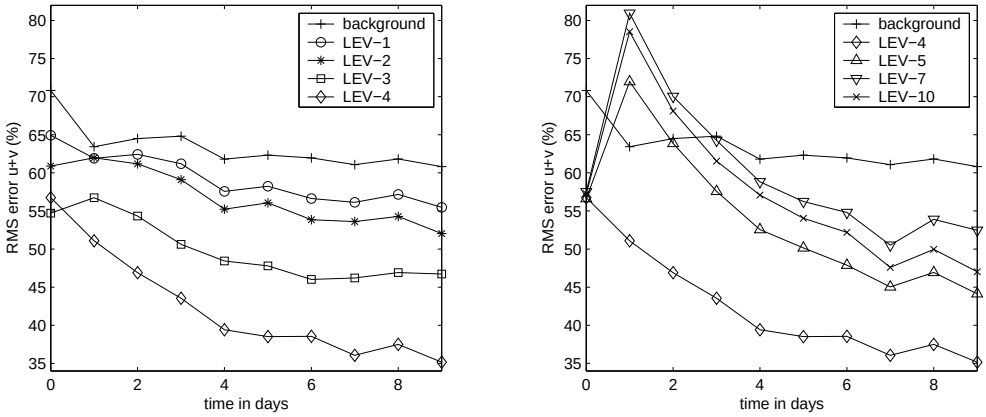

Figure 7. Sensitivity to the vertical drift level

Table 4. Coupled impact of number of floats and TSP.

Experiment	Total number of data	Final Error (%)
500-5D	1 000	46.6
500-3D	1 500	44.0
500-1D	5 000	44.0
2000-5D	4 000	37.1
2000-3D	6 000	36.8
2000-1D	20 000	35.9
1000-5D	2 000	35.1
1000-3D	3 000	35.1
1000-1D	10 000	34.8
Background	-	60.8

stable with respect to large TSP whatever the number of floats. Thus we can conclude that, in our configuration, it seems optimal to launch around 1 000 floats and that the TSP can be chosen quite large, which is very encouraging in view of the Argo floats network parameters.

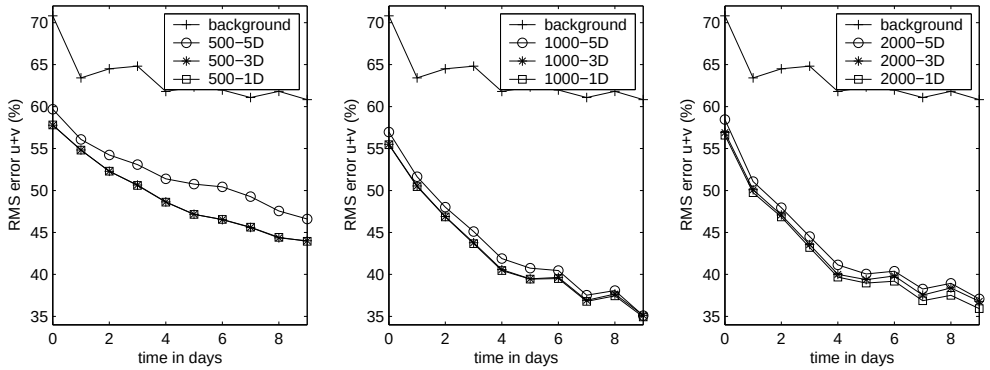

Figure 8. Coupled impact of number and time sampling period.

4.3. Comparison with the "Eulerian" method

A classical method in oceanography is to assimilate the velocity observations deduced from the Lagrangian data according to the following finite differences formula:

$$\begin{array}{rcl} \dfrac{\xi_1(t_{k+1}) - \xi_1(t_k)}{t_{k+1} - t_k} & \approx & u(\xi_1(t_k), \xi_2(t_k), z_0, t_k) \\ \dfrac{\xi_2(t_{k+1}) - \xi_2(t_k)}{t_{k+1} - t_k} & \approx & v(\xi_1(t_k), \xi_2(t_k), z_0, t_k) \end{array} \tag{7}$$

Then the velocity data are treated as Eulerian data (measured at non-fixed points). We implement this method in the 4D-Var framework. The observation operator is very easier to write (and to differentiate and transpose) because it is an interpolation at the points of the true (fixed) floats trajectories. We compare the results for this method said "Eulerian" and for our "Lagrangian" one. Experiments have the same characteristics (3 000 floats and varying TSP), their names are LAG-xxx or EUL-xxx with xxx the TSP, as described in Table 5. Figures 9 and 10 represent the RMS error as a function of time and vertical level. We can see that the "Eulerian" approach is slightly better than the "Lagrangian" one when the TSP is small (one day), moreover its computation time is 10% lower. Indeed the TSP is small enough so that the formula (7) is a very good approximation: the displacement vector between two successive positions is quasi tangent to the trajectory (ie quasi collinear with the velocity vector). However we can see that the error for the "Lagrangian" method is more homogeneous as a function of the vertical level. For larger TSP (3 days or more) the "Lagrangian" method is obviously better than the "Eulerian" one: the approximation formula (7) is not valid any more. We can see that our method is able to extract information from the positions data even if the TSP is large.

5. Conclusion

This paper shows that the problem of assimilating Lagrangian data can be solved by a variational adjoint method into a realistic primitive equations ocean model. We have implemented a Lagrangian method which takes into account the four dimensional (space and time) nature of the observations. Qualitatively the main patterns of the fluid flow are well identified at each vertical level, although the floats drift at a single

Table 5. Comparison between the "Eulerian" and "Lagrangian" methods.

Experiment	Total Error	Final Error (%)
EUL-1D	40.1	33.8
LAG-1D	41.4	33.6
EUL-3D	50.9	44.4
LAG-3D	43.0	36.1
EUL-5D	54.4	48.9
LAG-5D	43.9	37.1
EUL-10D	59.3	55.0
LAG-10D	45.2	38.6
Background	63.3	60.8

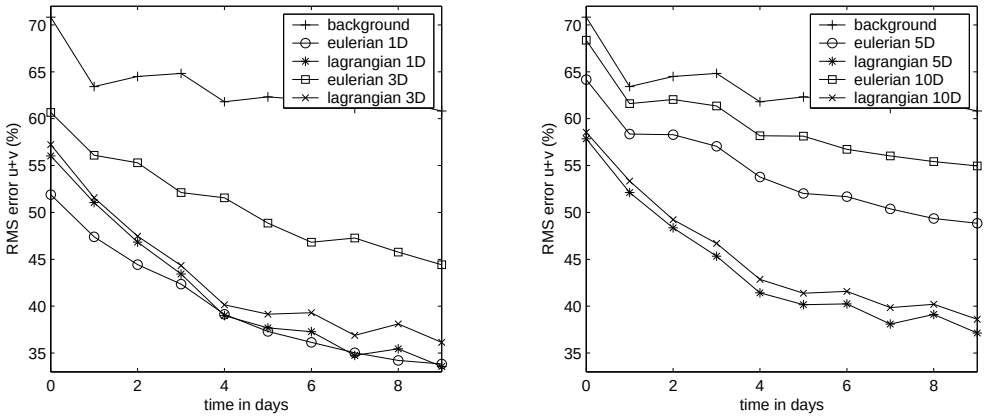

Figure 9. "Eulerian" vs "Lagrangian", error as a function of time.

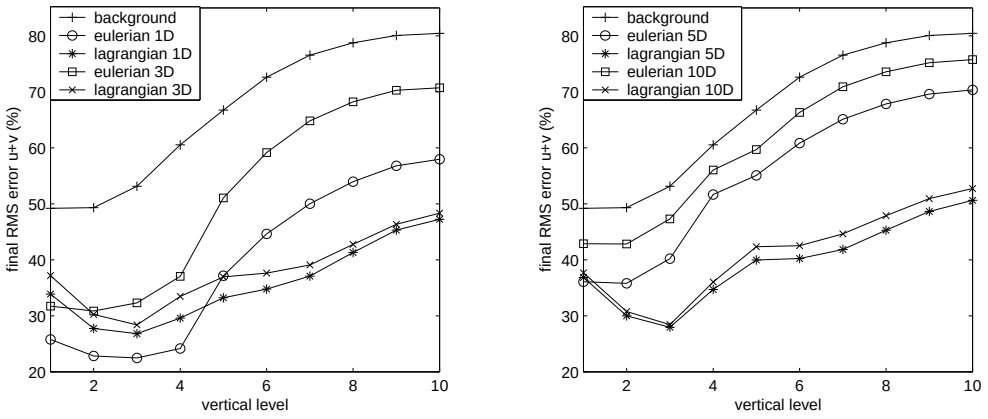

Figure 10. "Eulerian" vs "Lagrangian", error as a function of the vertical level .

determined level. Quantitatively RMS errors are divided by two and it seems rather difficult to reduce this error without increasing the size of the assimilation window.
This method performs very better than the Eulerian one, which consists in interpreting Lagrangian data as velocity information, especially when the time-sampling period of the observations is larger than two or three days. When this period is one day or less the Eulerian method performs slightly better but the transfer of information to lower levels is better achieved by the Lagrangian one.
We have tested the sensitivity of our method to the characteristics of the dataset. It is very sensitive to the vertical drift level, and the best results are obtained for intermediates ones, especially level 4 (around 1000 meters depth). It is also very sensitive to the number of floats, but the more is not the better, it seems useless to launch more than 1 000 floats in our configuration. It is very robust with respect to the increase of the time-sampling period. Thus it enables us to assimilate with the same efficiency data from acoustic floats (six hours between two samples) and Argo floats (ten days).
Also the performances of this method have been assessed with a ten-days assimilation window. It would be interesting to increase this up to one month. It would be also interesting to assimilate both Lagrangian and complementary data, such as altimeter information or Argo temperature and salinity profiles.

Acknowledgments

The author thanks Anthony Weaver and LODYC for the source code of OPAVAR 8.1. Numerical computations were performed on the NEC SX5 vector computer at IDRIS. This work is supported by French project Mercator.

References

Courtier P, Thépaut J-N and Hollingsworth A 1994 A strategy for operational implementation of 4D-Var, using an incremental approach *Quart. J. Roy. Meteor. Soc* **120** 1367–87

De Mey P 1997 Data assimilation at the oceanic mesoscale: a review *Journal of the Meteorological society of Japan* **75** 1B 415–27

Evans LC 1998 *Partial differential equations* Graduate Studies in Mathematics vol 19 (American Mathematical Society: Providence, Rhode Island)

Filatoff N and Assenbaum M 2003 Adjoint assimilation of ARGO float displacement data into an eddy-resolving ocean model *EGS AGU EUG joint assembly (Nice: France)*

Ghil M 1989 Meteorological data assimilation for oceanographers Part I: description and theoretical framework *Dyn. Atmos. Oceans* **13** 171–218

Ghil M and Manalotte-Rizzoli P 1991 Data assimilation in meteorology and oceanography *Adv. Geophys.* **23** 141–265

Giering R and Kaminski T 1998 Recipes for Adjoint Code Construction *ACM Trans. On Math. Software* **24** 4 437–74

Ide K, Courtier P, Ghil M and Lorenc AC 1997 Unified notation for data assimilation: operational, sequential and variational *J. Meteor. Soc. Japan* **75** 1B 181–9

Kamachi M and O'Brien J 1995 Continuous data assimilation of drifting buoy trajectory into an equatorial Pacific Ocean model *Journal of Marine Systems* **6** 159–78

Le Dimet FX and Talagrand O 1986 Variational algorithms for analysis and assimilation of meteorological observations: theoretical aspects *Tellus* A **38** 97

Lions J L 1968 *Contrôle optimal de systèmes gouvernés par des équations aux dérivées partielles* (Paris: Dunod Gauthier-Villars)

Lions J-L, Temam R and Wang S 1992 On the equations of the large-scale ocean *Nonlinearity* **5** 5 1007–53

Luong B, Blum J and Verron J 1998 A variational method for the resolution of a data assimilation problem in oceanography *Inverse Problems* **14** 979–97

Madec G, Delecluse P, Imbard M and Levy C 1998 OPA8.1 ocean general circulation model reference manual *Notes du pole de Modelisation de l'IPSL* **11**

Mead J 2005 Assimilation of simulated float data in Lagrangian coordinates *Ocean Modeling* **8** 369–94

Molcard A, Piterbarg L, Griffa A, Özgökmen T and Mariano A 2003 Assimilation of drifter observations for the reconstruction of the Eulerian circulation field *J. Geophys. Res. Oceans* **108** C03 3056 1–21

Özgökmen T, Molcard A, Chin T, Piterbarg L and Griffa A 2003 Assimilation of drifter observation in primitive equation models of mid-latitude ocean circulation *J. Geophys. Res. Oceans* **108** C07 3238 31 1–31 17

Penenko V V and Obraztsov N N 1976 A variational initialization method for the fields of the meteorological elements *Sov. Meteorol. Hydrol.* **11** 1–11

Sheinbaum J and Anderson D L T 1990 Variational assimilation of XBT data Part I *J. Phys. Oceanogr.* **20** 672–88

Talagrand O 1991 The use of adjoint equations in numerical modeling of the atmospheric circulation *Automatic differentiation of algorithms, Proc. 1st SIAM Workshop, Beckenridge/ CO (USA)* 169–180

Talagrand O and Courtier P 1987 Variational assimilation of meteorological observations with the adjoint vorticity equation. I: Theory *Quarterly Journal of the Royal Meteorological Society* **113** 478 1311–28

Temam R 1983 *Navier-Stokes equations* (North-Holland Pub. Company)

Temam R and Ziane M 2004 Some mathematical problems in geophysical fluid dynamics *Handbook of Mathematical Fluid Dynamics 3* (Friedlander S and Serre D Editors Elsevier)

Thacker W C and Long R B 1988 Fitting dynamics to data *J. Geophys. Res.* **93** 1227–40

Weaver A T and Courtier P 2001 Correlation modeling on the sphere using a generalized diffusion equation *Q. J. R. Meteorol. Soc.* **127** 1815–46

Weaver A T, Vialard J and Anderson D L T 2003 Three- and Four-dimensional variational assimilation with an ocean general circulation model of the tropical Pacific Ocean Part I: formulation, internal diagnostics and consistency checks *Mon. Wea. Rev.* **131** 1360–78