



Assimilation of Lagrangian Data in an operational framework

Claire Chauvin, Maëlle Nodet, Arthur Vidard, Pierre-Antoine
Bouttier

**RESEARCH
REPORT**

N° 7840

December 2011

Project-Team Moise



Assimilation of Lagrangian Data in an operational framework

Claire Chauvin, Maëlle Nodet, Arthur Vidard, Pierre-Antoine Bouttier

Project-Team Moise

Research Report n° 7840 — December 2011 — 24 pages

Abstract: In the framework of the Argo program, profiling drifting floats are now routinely launched in the world's oceans. These floats provide (among other information) data about their position, sampled every ten days, representative of their Lagrangian drift. Previous work ([?]) have shown the interest of assimilating this new type of data.

The assimilation of Lagrangian-type data such as position of drifting floats is not straightforward; it involves the careful implementation of a complex, non-linear observation operator but it is of importance for an application in operational oceanography. We propose here to study the addition of such observations in an operational variational system: the ocean model NEMO, coupled with the incremental 4D-Var tool NEMOVAR. In a physical configuration, we compare the impact of several variational assimilation strategies, for a realistic set of observations. The impact of the incremental 4D-Var algorithm is compared to previous 3DFGAT experiments; Lagrangian observations contribute to improve the statistical quality of the analyzed state, but their impact can be limited, since the assimilation of other observations may introduce disorder in the velocity.

Key-words: variational assimilation, NEMO ocean model, NEMOVAR, observation operator, float trajectories, Lagrangian observation, incremental 4D-Var algorithm.

**RESEARCH CENTRE
GRENOBLE – RHÔNE-ALPES**

Inovallée
655 avenue de l'Europe Montbonnot
38334 Saint Ismier Cedex

Assimilation de données lagrangiennes dans un contexte opérationnel

Résumé : Parmi les instruments qui mesurent l'état de l'océan, certains dérivent avec le courant océanique, en transmettant régulièrement leur position et leurs données à un satellite. Dans le cadre du programme Argo, environ 3000 flotteurs collectent des données dans tous les océans, nous donnant ainsi une information précieuse sur leur dérive lagrangienne dans les courants océaniques. L'utilisation de ces trajectoires dans le cadre d'algorithmes d'assimilation variationnelle a révélé un intérêt pour la prévision de l'état de l'océan ([?]). Nous présentons ici les résultats de l'assimilation de ces observations, dites lagrangiennes, dans un cadre opérationnel, en couplant le modèle d'océan NEMO avec le système d'assimilation variationnelle NEMOVAR. L'impact de l'algorithme 4D-Var incremental est comparé à des expériences 3DFGAT. Les observations lagrangiennes permettent d'améliorer la qualité au sens statistique de l'état analysé, mais leur assimilation peut être limitée par l'incrément introduit dans la vitesse analysée par l'assimilation d'autres types d'observations.

Mots-clés : assimilation variationnelle, NEMO, NEMOVAR, opérateur d'observation, trajectoires de flotteurs, observations lagrangiennes, algorithme 4D-Var incrémental.

Introduction

Among all the instruments that measure ocean state, some are drifting with ocean currents. For instance, ALACE, PALACE and SOLO floats measure ocean temperature and salinity profiles from a given depth, to the surface. When they are deployed, they sink to a pre-specified depth (typically 1000-2000 meters). They usually remain at this depth for 7-10 days, and then they rise to the ocean surface to transmit their data and position to orbiting satellites. The float then sinks again, continuing the process. These three types of floats form the backbone of the international Argo program, that collects oceanographic data from 3,000 floats throughout the world's oceans. These profilers can remain in the ocean for 4-5 years, leading to a large amount of information about their trajectories. The idea of this study is to see whether, with a 4D variational data assimilation algorithm, the use of these positions could improve the quality of the analysed state.

The floats, drifting with the ocean current, are by definition Lagrangian particles. The Lagrangian description of the ocean has led to set up different mathematical tools to improve our knowledge (see for example [?, ?]). The question of using these trajectories as observations for an assimilation system has already been addressed in the past; see for example [?, ?, ?, ?, ?], for the reconstruction of Eulerian velocities from Lagrangian data, with application to specific areas of the Mediterranean Sea. These results show the ability, with a few float trajectories, to improve locally the physical behaviour of the current, by introducing in the assimilated state some local effects that are absent in the model. Nevertheless all studies tend to show the need of a lot of floats to get a significant and positive action on the analysed state.

At the same time, direct assimilation of Lagrangian data has been investigated with success. In Extended and Ensemble Kalman Filter framework, the approach developed by Ide, Kuznetsov, Jones and Salman (see [?, ?, ?]) is based on an augmented state vector approach; it does not require the conversion of the positions into velocity data. The variational assimilation of Lagrangian data has been studied by Nodet (see [?]). In this paper, the author explores the different aspects of the method, with twin experiments, showing its great potentiality, as well as its weakness. In a toy model, provided that the number of floats is large enough, the analysed state can be very close to the true state. The depth of parking has an important impact: the floats has to be located at a mid-depth, not too close to the surface, to avoid very strong eddy kinetic energy, but not too deep, where main turbulent behaviours are dissipated. The conclusions show that there are not enough floats in a realistic distribution, but it could in addition to other observations, lead to an improvement of the analysed state. The aim of the present paper is to study whether, under realistic conditions, with an operational system, the use of float trajectories can be relevant.

The operational system we propose to work with is composed by the NEMO model, version 3.0, and the associated variational system NEMOVAR. The NEMO model is a state-of-the-art modelling framework for oceanographic research, operational oceanography seasonal forecast and climate studies. Tangent and adjoint models are now available in the NEMOTAM module of the last release of NEMO. It is widely spread in the oceanography community. Variational assimilation with NEMO is performed in several operational centres in Europe: a 3DFGAT multi-cycle version of NEMOVAR is used on a global configuration ORCA at ECMWF (resolution 1° , with a 10-day assimilation window), and at MetOffice (resolution 0.25° , with a one day assimilation window); INGV uses the 3D-VAR system MFS (Mediterranean ocean Forecasting System, resolution $1/16$ with one day cycle). The development of a multi-incremental version

is the object of a current collaboration between CERFACS, ECMWF and INRIA.

The objective of the present paper is to settle a realistic experiment, with the model NEMO 3.0, and NEMOVAR 2.0, in order to see to what extent the assimilation of float trajectories can improve the analysed state.

The organisation of this paper is as follows: first, we present the 3D-FGAT and 4D-Var incremental formulations. Then, the observation operator for floats is presented, as well as the expression of its tangent. Finally, the experimental set is detailed and a discussion is done about the interest of integrating Lagrangian trajectories in the assimilation system.

1 The assimilation problem

1.1 Formulation of the incremental algorithm

A complete description of data assimilation concepts can be found in [?].

Let x be the state variable, *e.g.* the vector containing the discretization of the dynamics variables: the temperature T , the salinity S , the zonal and meridional components of the velocity, u and v , and the surface elevation η . Starting from an initial state x_0 , the model propagates in time following:

$$\frac{dx}{dt} = \mathcal{M}(t, x; x_0). \quad (1)$$

In variational assimilation, we look for a solution of Equation (1), starting from an initial condition x_0 , that minimizes the cost function:

$$\mathcal{J}(x_0) = \frac{1}{2} \|x_0 - x^b\|_{B^{-1}}^2 + \frac{1}{2} \|\mathcal{G}(x_0) - y\|_{R^{-1}}^2.$$

The vector $\mathcal{G}(x_0) - y$ is composed by the distance between observation y_i at time t_i , and its model counterpart $\mathcal{G}_i(x_0) = \mathcal{H}(\mathcal{M}(t_i, x; x_0))$, where \mathcal{H} is the observation operator, projecting state variables into observation space. The background error covariance matrix is denoted B , and R represents the error covariance matrix. These matrices will be detailed in the next subsection. The background state x^b is a forecast that comes from a previous run of the model, or from climatology; it is the first guess of the algorithm.

In the incremental formulation, we are looking for the analysis (assimilated state) in the form $x^a = x^b + \delta x$ where δx is called the increment. The problem is written as a sequence of convex optimization problems:

1. Initial guess at $k = 0$, $x_0 = x^b$.
2. At iteration k , find $x_k = x_{k-1} + \delta x$ that minimizes the quadratic functional:

$$\mathbf{J}(\delta x) = \frac{1}{2} \|\delta x\|_{B^{-1}}^2 + \frac{1}{2} \|\mathbf{d} - \mathbf{G}_k(\delta x)\|_{R^{-1}}^2, \quad (2)$$

where $\mathbf{d} = y - \mathcal{G}(x_{k-1})$ is called the innovation vector, and \mathbf{G}_k is an linear operator which differs in the different formulations we present in this paper. A Newton-like algorithm is applied for the minimization of the quadratic cost function \mathbf{J} .

3. While $\mathcal{J}(x^k) < \mathcal{J}(x^{k-1})$ go to 2.

1.2 3D-FGAT and 4D-Var incremental formulations

We use two formulations in this paper. In both cases, for the computation of \mathbf{d} , the model counterpart of each observation y_i is propagated to its appropriate time t_i .

1. 3D-FGAT: $\mathbf{G}_k = \mathbf{H} = \frac{d\mathcal{H}}{dx}(\mathbf{x}_{k-1})$, where \mathbf{H} is the tangent linear observation operator at point x_{k-1} . This approximation is the same in the 3D-VAR approximation; the difference is in the construction of the innovation vector \mathbf{d} , which writes $\mathbf{d} = y - \mathcal{H}(\mathcal{M}(x_{k-1}))$ rather than $\mathbf{d} = y - \mathcal{H}(x_{k-1})$. For this reason, the scheme has been named 3D-FGAT for first guess at appropriate time (see [?]).
2. 4D-Var INC: $\mathbf{G}_k = \mathbf{H}_k \mathbf{M}_k = \frac{d\mathcal{H}}{dx}(\mathbf{x}_{k-1}) \frac{d\mathcal{M}}{dx}(\mathbf{x}_{k-1})$, the tangent linear model is applied to the increment δx before applying the tangent linear observation operator.

The 3D-FGAT formulation is the closest formulation to the exact one (4D-Var) that avoids the computation of tangent linear and adjoint operators of \mathcal{M} ; this quality makes it very competitive in terms of accuracy in the result, and execution speed. It will be the reference of our numerical tests. Between 4D-Var and 3D-FGAT formulations, incremental 4D-Var INC formulation propagates the errors from the initial state to the time of each observation by approximating the model with its tangent (its linearization around the first guess), and by transporting information from observations to initial state using the adjoint operator. This implies the implementation of tangent and adjoint models of NEMO, and a longer execution of the assimilation algorithm (several times the execution time of the direct model), but the results of the assimilation scheme are drastically improved, as seen in the experimental section.

1.3 On the tangent linear hypothesis

These two formulations are governed by the tangent linear approximation of the observation operator (for 3D-FGAT and 4D-Var), and of the model (4D-Var). Indeed, formulations of \mathbf{G} are valid when the following expansions:

$$\mathcal{M}(x + \delta x) = \mathcal{M}(x) + \mathbf{M}(\delta x) + O(\|\delta x\|^2), \quad (3)$$

$$\mathcal{H}(x + \delta x) = \mathcal{H}(x) + \mathbf{H}(\delta x) + O(\|\delta x\|^2), \quad (4)$$

hold around the current initial state x . In 3D-FGAT, the tangent linear model is approximated by the identity operator. Equation (4) is valid for the classical observation operators (on the temperature, salinity, sea level altimetry), since observation operators in these cases are linear interpolators of these physical quantities at a given point of the configuration domain. We will see in section 2 the validity of (4) for the Lagrangian observation operator. This operator is very sensitive to its input parameters: with two closed velocity fields, floats can have very different trajectories. The tangent linear hypothesis will be illustrated in a specific NEMO configuration, in section 3.

1.4 Expressions of R and B

The background error covariance matrix B used for the experiments is constructed following the method of [?], and using a multivariate balance operator ([?]). B matrix is modelled as $B = UU^T$, with operator U being the composition of three operators: $U = KDF$. These three operators represent the different interactions between the state variables:

- Operator D^{-1} is a diagonal composed of the estimates of the standard deviations for v^b (the preconditioned background variable). This reflects the confidence on the background state, as well as the variability for each state variable.
- Balance operator K^{-1} produces a set of mutually uncorrelated variables by removing any known dynamical or physical balance relationships between model state variables, and the multivariate component for the background-error covariances in x-space.
- Operator F^{-1} is the inverse of a smoothing operator that acts separately on each of the uncorrelated variables (e.g. a block diagonal operator). FF^T is interpreted as a correlation matrix.

Construction of matrix U is linked with the preconditioning of the Conjugate Gradient Algorithm that is used in the minimization of (2). Among the minimizers available in NEMOVAR 2.0, we used the so-called *cgmod* algorithm without preconditioning. The algorithm is similar to that of Derber and Rosati ([?]) but has been extended in NEMOVAR to include an option to reorthogonalize the gradients. In our experiments twenty inner loops are used to create the minimisation space.

The observation error covariance is block diagonal, and stands for the confidence on each of the observations. The value of each block is tuned such that the contribution of each kind of observation is balanced in the cost function J .

2 Observation operator for float trajectories

When observations are state variables, such as a distribution of temperature, or salinity, the observation operator simply consists in interpolating the model state variable at the specific observation times and positions. This operator is linear as a function of the state variables.

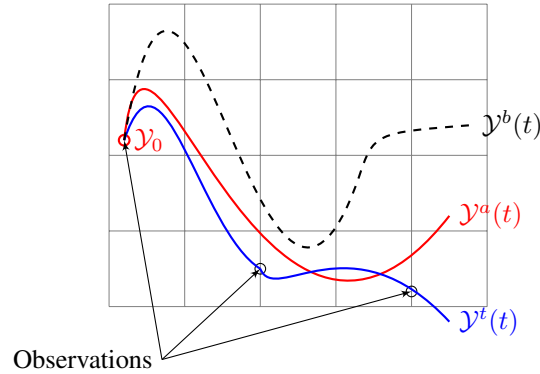
There are several ways to assimilate float positions: pseudo-observation assimilation and direct assimilation.

Eulerian method consists in comparing velocity pseudo-observation (an apparent velocity is deduced from two successive float positions) to the model velocity (see [?, ?]).

In direct assimilation method, float trajectories are constructed following the Lagrangian principle: starting with initial observed float positions, the trajectories are deduced from the assimilated velocity. Positions corresponding to observation times are then extracted, and compared to observations. It has been shown in [?] that this method behaves better than the Eulerian one when the time sampling interval is larger, which is the case for realistic observations. More precisely, the Lagrangian integral time (LIT) is defined for each float as the first time where the float velocity is orthogonal to its initial time velocity. The more turbulent the flow is, the lower the LIT is. Eulerian methods are efficient when float time sampling stays inferior to 20 or 30% of the LIT, whereas Lagrangian method is still good close to LIT (see [?]).

After detailing the construction of our observation operator, we present the first results that validates our observation operator in the NEMO 3.0 – NEMOVAR 2.0 system.

2.1 Main principles



Let us consider a float drifting at a fixed depth z_0 in the fluid domain $\mathcal{D} \subset \mathbf{R}^3$, located on $\mathcal{X}_0 \in \mathbf{R}^2$ at time t_0 . Observations $\mathcal{X}_0, \mathcal{X}_1, \dots, \mathcal{X}_n$ of this float are given at several times t_0, t_1, \dots, t_n inside the assimilation window $[0, T]$. To get the model counterpart of these observations, one has to 1) reconstruct the trajectory in interval $[t_0, t_n]$, using the model velocity $U(t, x) = (u, v)$, from the initial position \mathcal{X}_0 , and 2) extract from this trajectory the points corresponding to the different times of observation t_1, \dots, t_n . The float trajectory is totally determined by the fluid velocity U and initial position \mathcal{X}_0 :

$$\begin{cases} \frac{d\mathcal{X}}{dt} = U(t, (\mathcal{X}(t), z_0)), \\ \mathcal{X}(t_0) = \mathcal{X}_0. \end{cases} \quad (5)$$

Step 1) then consists in discretizing (5) using the model time step Δt , with an adapted discrete scheme. We choose a leap-frog scheme (see [?]): at each time step $t_k = k\Delta t$, one needs to evaluate the velocity field at the position \mathcal{X}_k , which is done by interpolation of the fluid velocity U_k at time t_k . This velocity, called \mathcal{U}_k , defines the float velocity at time t_k . Let $\mathbf{I}(f, \mathcal{X})$ be the operator that interpolates a function f defined in an appropriate space at a point $(\mathcal{X}, z_0) \in \mathcal{D}$. This operator will be detailed in the next subsection 2.2. The discrete time scheme then writes:

- Initialization

$$\begin{cases} \mathcal{X}_0 & \text{given,} \\ \mathcal{U}_0 = \mathbf{I}(U_0, \mathcal{X}_0). \end{cases}$$

- First time step

$$\begin{cases} \mathcal{X}_1 = \mathcal{X}_0 + \Delta t \mathcal{U}_0, \\ \mathcal{U}_1 = \mathbf{I}(U_1, \mathcal{X}_1). \end{cases}$$

- From t_k , $k = 2, \dots, n$:

$$\begin{cases} \mathcal{X}_k = \mathcal{X}_{k-2} + 2\Delta t \mathcal{U}_{k-1}, \\ \mathcal{U}_k = \mathbf{I}(U_k, \mathcal{X}_k). \end{cases}$$

This scheme is very simple, and thus has the advantage to be differentiable (up to the differentiation of \mathbf{I} w.r.t. \mathcal{X}_k). The tangent code associated to the k -th iteration writes:

$$\begin{cases} \delta \mathcal{X}_k = \delta \mathcal{X}_{k-2} + 2\Delta t \delta \mathcal{U}_{k-1}, \\ \delta \mathcal{U}_k = \mathbf{I}(\delta U_k, \mathcal{X}_k) + \delta \mathcal{X}_k \cdot \frac{\partial \mathbf{I}}{\partial \mathcal{X}_k}(U_k, \mathcal{X}_k). \end{cases} \quad (6)$$

Computation of $\delta \mathcal{U}_k$ in (6) raises the problem of the differentiation of \mathbf{I} with respect to the position \mathcal{X}_k . This is discussed in the next section.

2.2 The interpolation operator \mathbf{I}

In general, interpolations are not linear with the position, implying complex tangent and adjoint operators. If the grid is regular, a cheap solution is to define the interpolation operator using the eight corners Q_i of the mesh that contains the float. We project the horizontal coordinates \mathcal{X} onto the two horizontal meshes that surround the float, using the 2D interpolation operator \mathbf{I} . Coordinates of \mathcal{X} are then the linear combination on the z -coordinate of the two projections.

The operator $\mathbf{I}(f, \mathcal{X})$ is decomposed into three steps: first, find the mesh containing the point \mathcal{X} , then, compute the weights that determine the position of \mathcal{X} inside the mesh:

$$\mathcal{X} = \sum_{i=1}^4 a_i Q_i,$$

and finally, assemble the linear combination of values of f at the corners of the mesh weighted with these coefficients. We can express $\mathbf{I}(f, \mathcal{X})$ by:

$$\mathbf{I}(f, \mathcal{X}) \approx \sum_{i=1}^4 a_i f(Q_i), \quad (7)$$

where a_i are weighting coefficients. The tangent code of (7) writes:

$$\delta \mathbf{I}(f, \mathcal{X}) = \mathbf{I}(\delta f, \mathcal{X}) + \delta \mathcal{X} \cdot \partial_x \mathbf{I}(f, \mathcal{X}). \quad (8)$$

$$= \sum_{i=1}^4 a_i \delta f(Q_i) + \sum_{i=1}^4 \delta a_i f(Q_j) \quad (9)$$

The first term $\mathbf{I}(\delta f, \mathcal{X})$ of (8) is simply the interpolation of the quantity δf at the point \mathcal{X} : this term is classical, and common for all observation operators. The second term, which comes from the Lagrangian formulation, requires the computation of the weight's partial derivatives with respect to \mathcal{X} . In the following, we make the assumption that the perturbation $\delta \mathcal{X}$ around \mathcal{X} remains in the same grid cell. This assumption can be strong since the perturbation, coming from the perturbation of the fluid velocity, can be very chaotic. Nevertheless numerical tests show the good behaviour of this scheme with respect to the velocity field.

Among the methods to evaluate the weights a_i , a few are differentiable. We use the method proposed by Daget, and called general bilinear re-mapping interpolation (see [?]).

Its main ideas are recalled here, in order to write explicitly its tangent and adjoint models. It is based on a mapping from the four points $\{Q_i\}$ onto the unit square cell. The computation of weights a_i is equivalent to the computation of local coordinates $\alpha = (\alpha_1, \alpha_2)$ of \mathcal{X} inside the unit cell. The relation between a_i and α writes:

$$\begin{aligned} a_1 &= (1 - \alpha_1)(1 - \alpha_2) \\ a_2 &= (1 - \alpha_1)\alpha_2 \\ a_3 &= \alpha_1(1 - \alpha_2) \\ a_4 &= \alpha_1\alpha_2. \end{aligned}$$

The associated operator $F : \mathbb{R}^2 \rightarrow \mathbb{R}^4$ defines the transformation:

$$a = F(\alpha).$$

By then, the relation between float position \mathcal{X} and $Q = \{Q_i\}$ writes:

$$\mathcal{X} = {}^tF(\alpha)Q,$$

namely:

$$\begin{aligned} \mathcal{X}^1 &= (1 - \alpha_1)(1 - \alpha_2)Q_1^1 + \alpha_1(1 - \alpha_2)Q_2^1 + \alpha_1\alpha_2Q_3^1 + (1 - \alpha_1)\alpha_2Q_4^1, \\ \mathcal{X}^2 &= (1 - \alpha_1)(1 - \alpha_2)Q_1^2 + \alpha_1(1 - \alpha_2)Q_2^2 + \alpha_1\alpha_2Q_3^2 + (1 - \alpha_1)\alpha_2Q_4^2. \end{aligned}$$

To compute α , Daget used the following algorithm:

- From an initial guess α , inverse the linear system:

$$\delta\mathcal{X} = \widetilde{F}_\alpha(\delta\alpha)Q,$$

where \widetilde{F}_α is the linearization of F around α , and

- update the quantity:

$$\alpha \leftarrow \alpha + \delta\alpha$$

The algorithm stops when the current increment $\delta\alpha$ is small enough. This interpolation operator has been tested in several configurations of NEMO, in order to validate its accuracy for irregular grids. Compared to the other interpolation operator implemented in NEMO, it gives a rather good approximation of a function discretized at grid points, even near the North Pole where the mesh has a strong aspect ratio. Moreover, this elegant interpolation algorithm allows a clean implementation of its tangent operator.

2.3 Tangent and adjoint observation operators

Now that we have written the interpolation \mathbf{I} , the tangent model of \mathcal{H} at time k is given explicitly:

$$\begin{cases} \delta\mathcal{X}_k = \delta\mathcal{X}_{k-2} + 2\Delta t \delta\mathcal{U}_{k-1}, \\ \delta\alpha_k = \widetilde{F}_{\alpha_k}^{-1}(\delta\mathcal{X}_k), \\ \delta a_k = \widetilde{A}_{\alpha_k}(\delta\alpha_k), \\ \delta\mathcal{U}_k = \sum_{i,j=1}^2 a_{ij}\delta U_k(Q_{ij}) + \sum_{i,j=1}^2 \delta a_{k,ij}U_k(Q_{ij}), \end{cases} \quad (10)$$

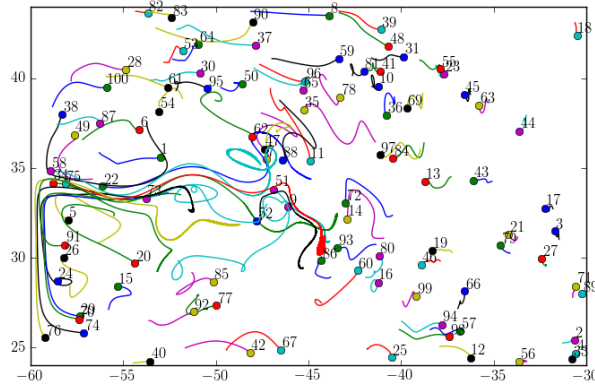


Figure 1: Trajectory of a hundred floats for a two months run, inside the SQB configuration, located between longitudes $-60^\circ W$, $-30^\circ W$ and latitudes $24^\circ N$, $44^\circ N$.

where \tilde{A} is the tangent model of operator A , and $U_k, \delta U_k$ are the direct and tangent model of the velocity field. The adjoint model is easily deduced from the tangent one. Technical details on the implementation of these two operators can be found in [?].

2.4 Validation of the Lagrangian observation operator, its tangent and its adjoint

Figure 1 shows the trajectory of a hundred floats, randomly distributed inside the domain, over two months. The model configuration SQB is described in section 3. There are three kinds of areas in figure 1 :

- Far from the jet (around the boundary, and at the east of longitude $40^\circ W$): the floats do not move much from their initial positions. The velocity is not very high, and stays rather regular (keeps almost the same direction and amplitude).
- Inside the jet (region between $30^\circ N$ and $37^\circ N$, until longitude $47^\circ W$): the velocity is high, but keeps the same direction: trajectories are very long, and regular.
- In the middle of the domain, a region localized between $57^\circ W$ and $40^\circ W$, and $30^\circ N$ and $37^\circ N$: this is a very unstable domain, which is reflected by very perturbed trajectories (see floats 47,88 and the end of the red trajectory (float 34), that begins near the left boundary).

Figure 2 shows the mean behaviour of the tangent linear hypothesis with the time window, over the hundred floats of Figure 1. The approximation error is of order 1, *e.g.* it grows linearly with time. The jump at day 8 can be explained as a limit of tangent linear hypothesis validity, although the approximation still remains satisfying after that.

In the SQB configuration, this hypothesis is valid in a 10 day window, which is rather small compared to the validity of the model tangent linear (two months).

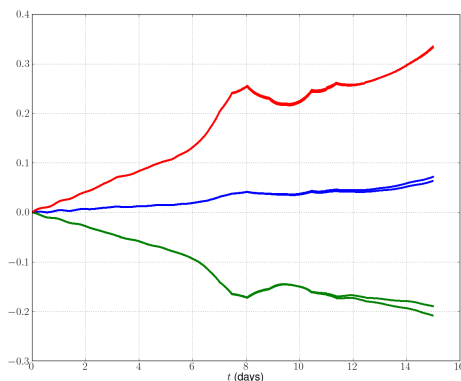


Figure 2: x-axis: number of days, y-axis: mean error $\mathcal{X}(x + \delta x) - \mathcal{X}(x) : \mathbf{X}(\delta \mathbf{x})$ (blue), surrounded by its standard deviation (red and green), for all the floats in Figure 1.

We performed twin experiments, by analysing the impact of the assimilation of 100 float trajectories over ten days. The true trajectories are generated from 100 random initial positions of floats, and they are then transported by the true state inside the assimilation window, at each time step: they are almost 5760 observations per float (except near the boundaries). The configuration of the different components of the assimilation (error covariance matrices, incremental algorithm, minimizer, twin experiments) is described in the next paragraphs.

Previous results (see [?]), in another configuration (with OPA / OPAVAR system), are still valid in our case. For instance, the parking depth of the floats had a serious impact on the quality of the assimilation: too close to the surface, the kinetic energy is too strong, and the velocity field too unstable, to get a relevant information from float trajectories at this depth.

Figure 2.4 shows the RMS error from the true state for all state variables of the SQB model.

Float positions are better assimilated when they are located:

- Several levels under the surface: there is actually a medium depth (1000 meters) where the eddy kinetic energy (eke) has a medium value. Here, the assimilation of Lagrangian data gives its best potentiality.
- The unstable region is a really critical region for assimilating float positions. This picture shows that temperature RMS error is improved of 30% when eliminating floats localized in this region.

It will be important for an operational assimilation of Lagrangian data to make a quality control on their trajectory, by removing those located in chaotic regions.

3 Experimental set up

Ocean configuration, as well as twin experiment set up are detailed in this section. We first present the physics that motivates the use of the so-called SQB configuration, and

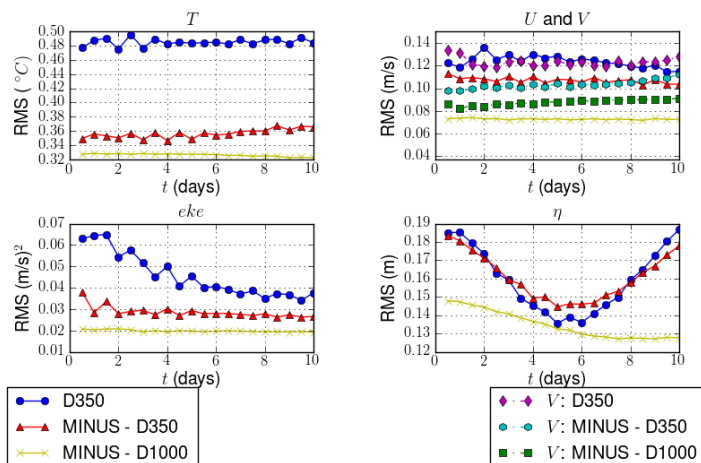


Figure 3: RMS error $x^t - x^a$ in the 10 day assimilation window, for three Lagrangian data assimilation experiments: D350: floats are parked at depth 350 meters, MINUS - D350: same depth as before, but we have deleted some observations located on the unstable region presented before: floats 47, 88, 62, 11, 51, 9 ..., and finally MINUS - D1000: no observations in the unstable region, and floats are all parked at depth 1000 meters. Legend for variables T , u , η is shown on lower left, and for V , on lower right.

then different states and observations used in our experiments.

3.1 Configuration

The ocean circulation model is the NEMO (Nucleus for European Modelling of the Ocean) system, and the configuration used, called SQB, is taken from [?]. SQB is a free-surface model with constant vertical mixing (see figure 4). It represents a mesoscale signal in an idealised basin with a 5000-m deep flat bottom ocean, at mid latitudes (between 25° and $45^\circ N$), with a quarter of a degree resolution. A double-gyre circulation is created by a constant zonal wind forcing that blows westward in the northern and southern parts of the basin and eastward in the middle part of the basin (with a sinusoidal latitude dependence).

The jet created by this forcing is unstable so that the flow is dominated by chaotic mesoscale dynamics, with largest eddies that are ~ 100 km wide, velocities of ~ 1 m/s and dynamic height differences of ~ 1 m. All this is very similar in shape and magnitude to what is observed in the Gulf Stream (North Atlantic) or in the Kuroshio (North Pacific)

3.2 Identical twin experiments

Twin experiments consist in generating synthetic observations from a known set of model state variables (true state). Twin experiments are said to be identical if true state generation and assimilation are performed with the same model (same physics, same resolution). The advantage is that we can then easily compare analysed states and synthetic true states, to quantify the performance of our algorithm. In this set up,

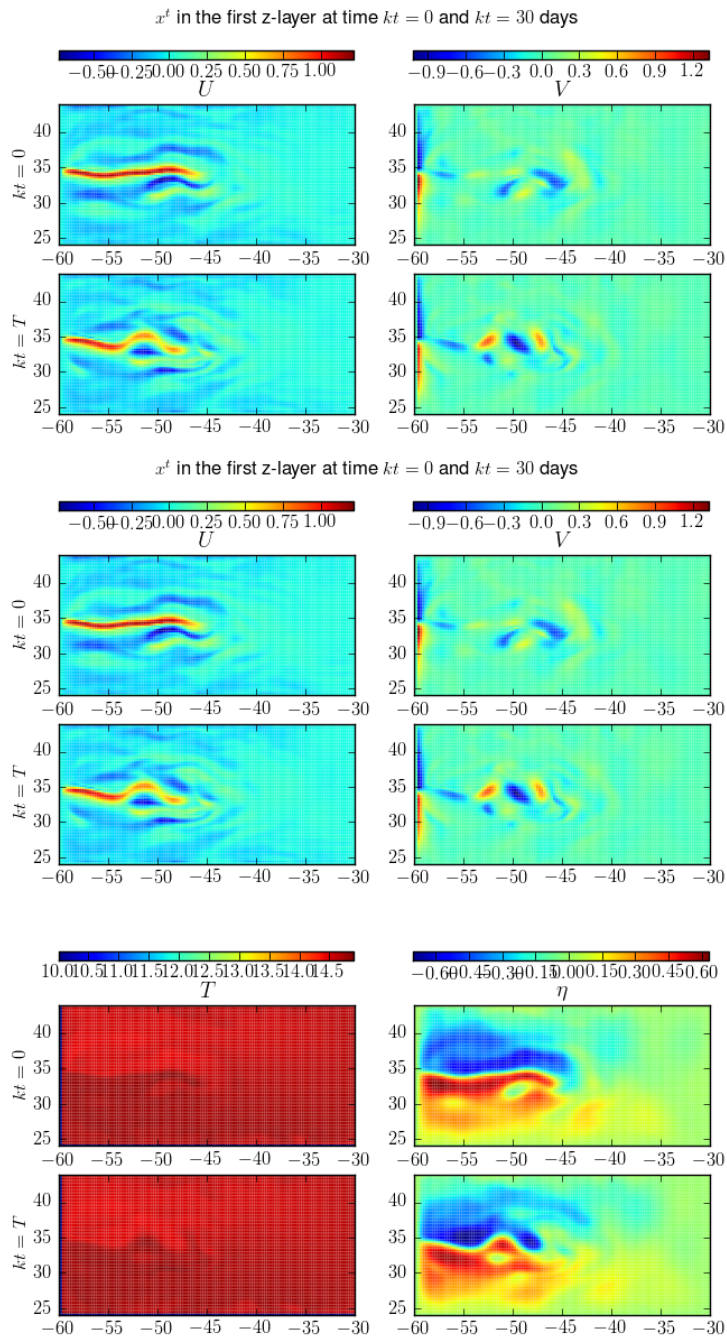


Figure 4: Map of the state variables (surface layer) of the true state for the SQB configuration, at initial time (first line) and after 30 days (second line).

Comparison between $\mathcal{M}(t, X^t)$ and $\mathcal{M}(t, X^b)$

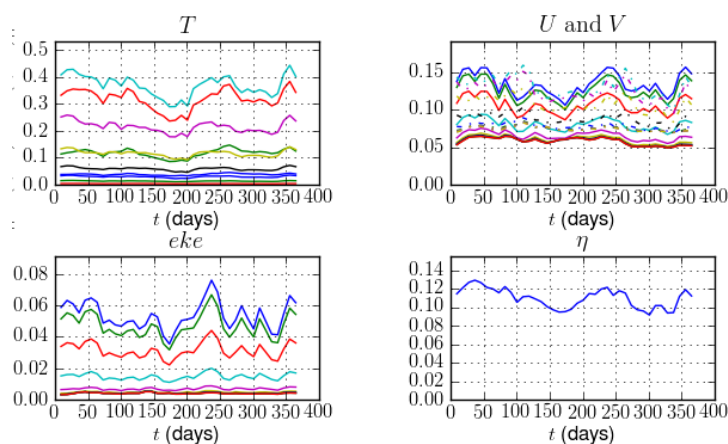


Figure 5: RMS error between background state and true state, in the first year of simulation, and for each z-layer. Units are: T ($^{\circ}C$), U and V (m/s , eke ($(m/s)^2$), and η (m).

the true state is obtained after a spin up of 106 years, and the background state is the model state one month before the true state.

Figure 5 shows the RMS error between the true state and the background state for one year, for the eleven vertical layers. This error remains more or less constant during this period, revealing the variability of SQB.

3.3 Observations

There are 49 profiler stations inside the geographical domain of SQB, for January, 2010 (see figure 6). They cover 148 vertical levels, from 200 to 2000 meters. Most of them derive at the depth of 1000 meters. All these stations supply 12453 observations during these period.

In thirty days, they are few observations of positions per stations and actually only 46 stations transmit two or three times their coordinates. The interest of this study is to see whether the assimilation of these trajectories added to PRF and SLA observations leads to a better convergence of the analysed state to the true state, and if it adds relevant physical effects on the velocity.

SLA observations are created mimicing the satellite ENVISAT, that covers periodically the domain. The simulated period is January 2010, where there were 12819 observations of Sea Level Anomaly.

We take these two realistic distributions of PRF and SLA observations, and interpolate values of the true state (temperature and sea level anomaly) at each realistic observation point. Lagrangian observations are generated from initial positions of the PRF stations (see figure 7).

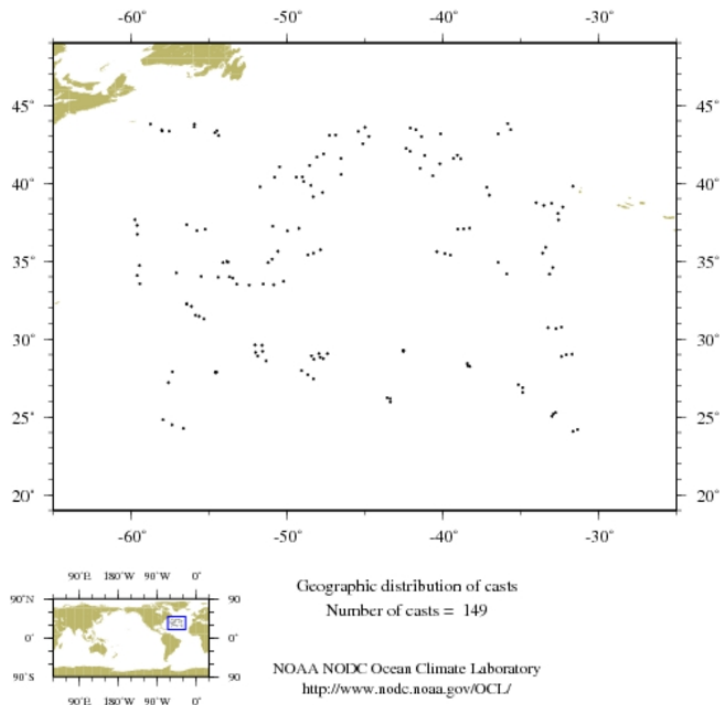


Figure 6: Distribution of profile data (PRF) in the assimilation time window (January, 2010), inside the SQB domain.

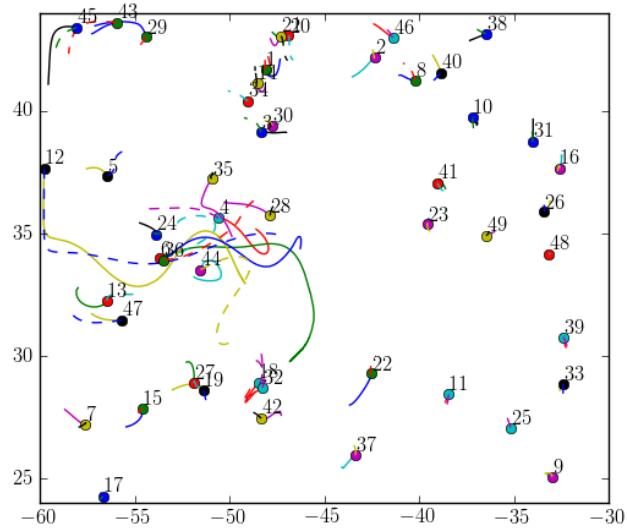


Figure 7: Float trajectories generated from the true state (line), and the background state (dashed), during one month. Initial position for each float corresponds to the first observation of PRF data inside the one month assimilation window (Figure 6).

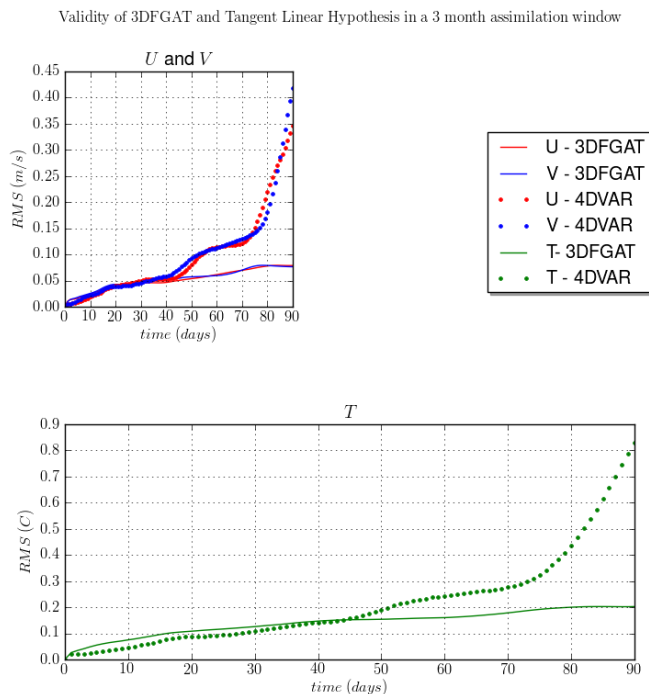


Figure 8: Comparison of the RMS error for the validation of equation (3): 3DVAR hypothesis (line), and 4DVAR (dot), for the three variables U and V (top figure) and T (bottom figure). The time window (x axis) equals 90 days.

3.4 Tangent Linear Hypothesis in this configuration

The tangent linear hypothesis (Equation (3)) does not hold for any time window. Indeed, since \mathcal{M} is linearized around a model state, this approximation is invalidated when the perturbation is too large. To compare the time validity of the tangent linear hypothesis, with the time validity of the 3DVAR approximation, we compare the L2-norm for different time steps, on a 3 month window of the two following quantities:

$$\begin{aligned} (3DVAR) \quad \mathcal{M}(x + \delta x) &= \mathcal{M}(x) + \delta x, \\ (4DVAR) \quad \mathcal{M}(x + \delta x) &= \mathcal{M}(x) + \mathbf{M}(\delta x). \end{aligned}$$

The increment δx used for both of these calculations comes from the first outer loop of a 4DVAR-inc assimilation using PRF+SLA observations.

Figure 8 shows that in the SQB configuration, the 3DVAR hypothesis is rather good, since RMS error (11) remains stable inside all the time window. The Tangent linear hypothesis is yet not so good as expected: it is better than the 3DVAR approximation during the first forty days, but then the tangent linear model is not stable anymore.

4 Numerical results on the assimilation of Lagrangian data

In all experiments, we used the same parameters: four inner loops of 20 iterations each. The cost function has almost reached its minimum after the first outer loop, but smaller scale features are improved by adding several outer loops. For assimilation of sea level altimetry data (SLA) and profiles (PRF), best results are obtained for a one month assimilation window. This window will be reduced in the case of float trajectory assimilation, to fulfil the tangent linear hypothesis of the Lagrangian observation operator. A 15 day window is used in this case, and compared with assimilation of SLA and PRF+SLA in a 15 day window.

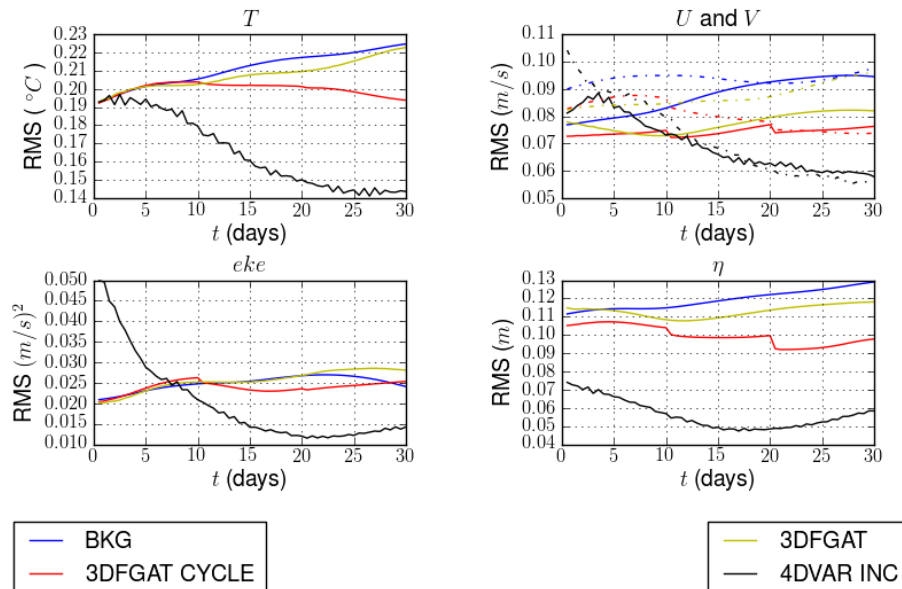


Figure 9: Comparison of RMS error with the true state for several assimilation strategies using SLA observations, for the 5 variables T , U , V , eke and η . Three assimilations are compared to the RMS error with background (blue): a 3DFGAT assimilation in a 30 days assimilation window (yellow), a cycle of 3 10 days 3DFGAT assimilations (red), and 4D-Var INC assimilation over the 30 day assimilation window (black). These quantities are shown using '-' for V variable (top right figure).

4.1 Assimilation of SLA observations

We first compare the RMS error $x^t - x^a$ using different assimilation strategies:

- 3D-FGAT: a 30 day cycle using 3D-FGAT algorithm.
- 3D-FGAT CYCLE: a 30 day assimilation using 3 cycles of 10 days using 3D-FGAT algorithm.

- 4D-Var INC: incremental 4D-Var assimilation in a 30 day window, using five outer loops. The use of multiple outer loops does not improve significantly the distance to the true state; the cost function does not decrease a lot after the first outer iteration. Yet, the quality of the increment is better: small scale features appear.

Figure 9 compares the three algorithms 3D-FGAT, 3D-FGAT CYCLE and 4D-Var INC. While 3D-FGAT and 3D-FGAT CYCLE slightly decreases the RMS error (at least at the beginning of the assimilation window), 4D-Var INC algorithm leads to a rather significant improvement: RMS error is smaller for all the variables.

The propagation of corrections from η to the other variables by the mean of the adjoint model has a significant impact on velocities.

Next paragraph shows the result of an assimilation using only PRF observations (temperature). This will illustrate again the particular role of variable T inside this configuration, and using these minimization parameters.

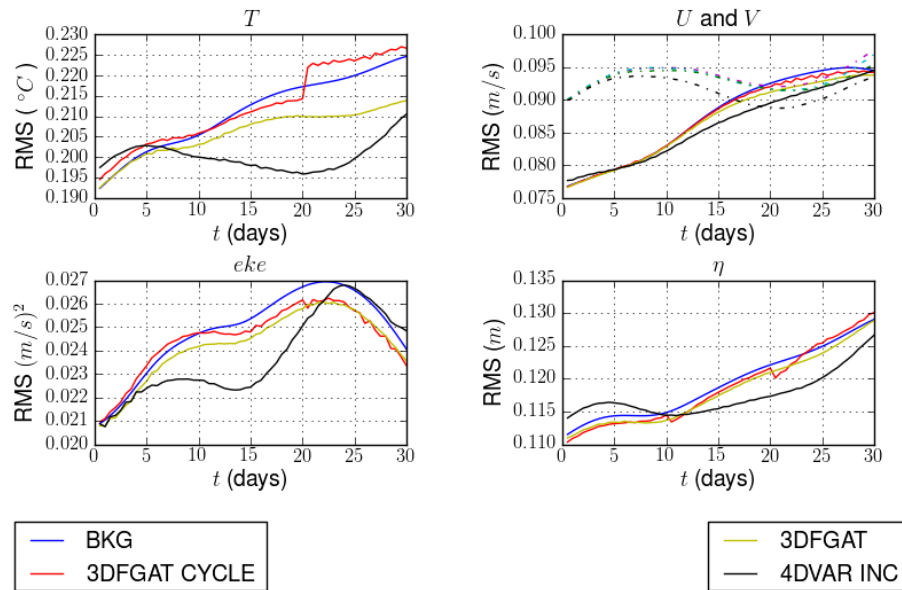


Figure 10: Comparison of RMS error with the true state for several assimilation strategies using PRF observations, for the 5 variables T , U , V , eke and η . Three kinds of assimilation are compared to the RMS error with background (blue): a cycle of 3 3DFGAT with 10 day assimilation window (red), a 3DFGAT assimilation inside the 30 day assimilation window (yellow) and the 4D-Var INC algorithm inside the 30 day assimilation window (black).

4.2 Assimilation of PRF observations

Figure 10 represents RMS error after temperature profile assimilation. For variables u , v and η , the effect of any kind of assimilation is small. 3D-FGAT assimilation does not improve significantly the results; 4D-Var INC yet improves the state variable T :

the propagation of corrections using the adjoint model appears to be very useful in this case. Again, from these results, the link between variable T and the other ones seem to be not very well modelled. The modelling of B may not be valid for the SQB configuration.

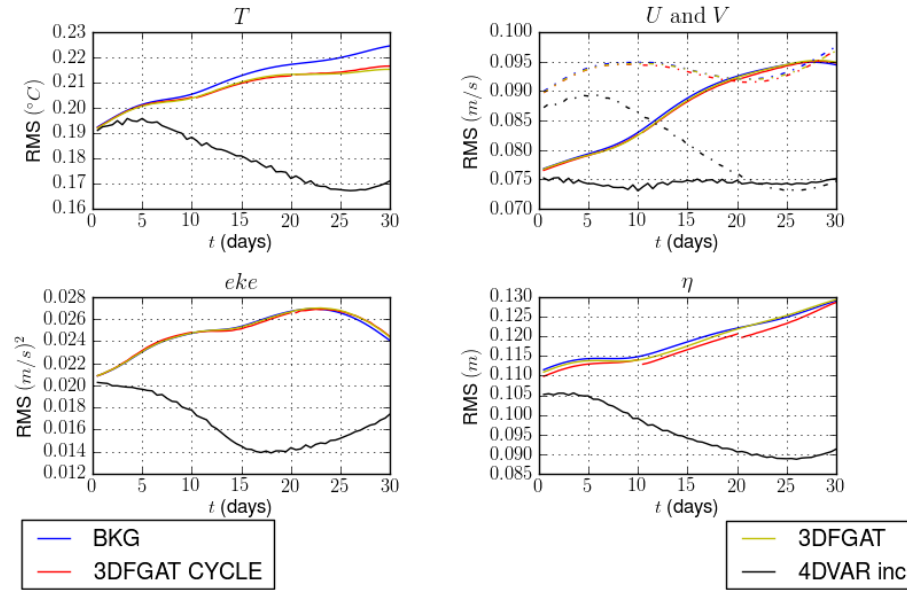


Figure 11: Comparison of RMS error with the true state for several assimilation strategies using PRF+SLA observations, for the 5 variables T , U , V , eke and η . Three kinds of assimilation are compared to the RMS error with background (blue): a cycle of 3 3DFGAT with 10 day assimilation window (red), a 3DFGAT assimilation inside the 30 day assimilation window (yellow) and the 4D-Var INC algorithm inside the 30 day assimilation window (black).

4.3 Assimilation of SLA+PRF observations

Figure 11 shows that use of temperature profiles (PRF) added to SLA observations gives a very good analysed state, for all variables. The apparent disconnection of T and the other variables finally allows good reconstruction for all variables at the end of the 4D-Var INC assimilation scheme.

Assimilation of PRF (resp. SLA) observations leads to a forecast improvement on the T variable (resp. U , V , η state variables). The use of both observations PRF+SLA does not degrade this complementarity. As a conclusion, we both need PRF and SLA observations to get a good forecast for all the state variables.

4.4 Assimilation of SLA+PRF+LAG observations

We now use a 15 day assimilation window, which is a good compromise between tangent linear hypothesis for \mathcal{H} (see section 2.4), and the sparse repartition of observations (each float gives information on its position each 10 or 15 days).

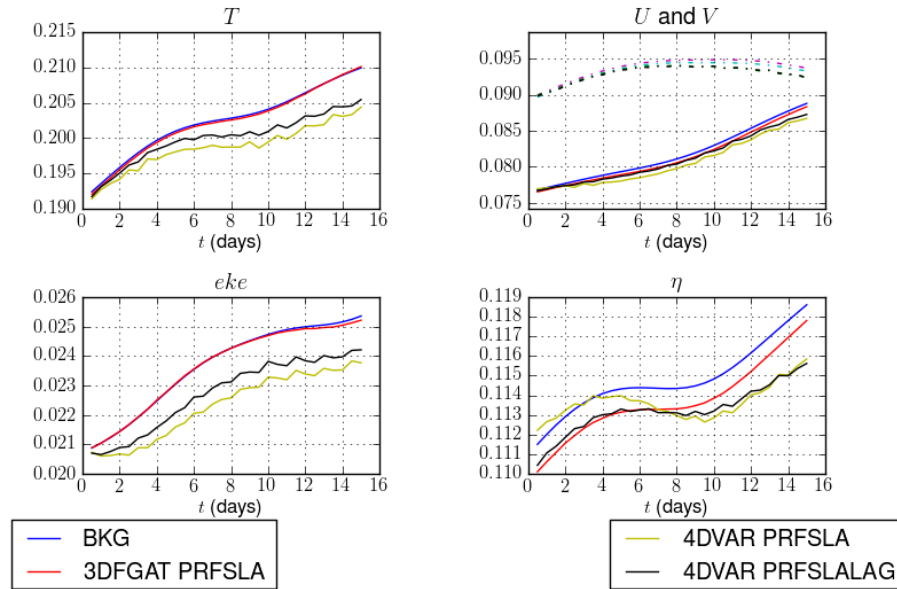


Figure 12: Comparison of RMS error for the 4D-Var INC algorithm using PRF+SLA obs and PRF+SLA+LAG observations, for the 5 variables T , U , V , eke and η . Three kinds of assimilation are compared to the RMS error with background (blue): a 3DFGAT with 30 days assimilation window using PRF+SLA observations (red), a 4D-Var INC assimilation inside the 30 day assimilation window using PRF+SLA observations (yellow) and the 4D-Var INC algorithm inside the 30 day assimilation window PRF+SLA+LAG observations (black).

Figure 12 reveals that the good performance of the 4D-Var INC algorithm using PRF+SLA observations is degraded by the use of Lagrangian data. The reason is that the small scale phenomena that appear in the analysed state generate chaotic float trajectories.

We use another qualitative diagnostic to study the impact of Lagrangian data in assimilation. In the following figures 13 and 14, we compare the so-called Q-Q-plots ("Q" stands for quantile), which is a graphical method for comparing two probability distributions by plotting their quantiles against each other. A point (x,y) on the plot corresponds to one of the quantiles of the second distribution (y -coordinate) plotted against the same quantile of the first distribution (x -coordinate).

Figure 13 shows the quality of the forecast according to the observation distribution, for the variable SLA. Roughly, from figure 13, the analysed η distribution is well-centred around the observation distribution, meaning that our assimilation system introduces no bias when assimilating SLA. It seems moreover that positive values of SLA are more spread, and this phenomenon increases when adding other observations to SLA. Second column of Figure 13 reveals that the η probability distribution fits very well the observation probability distribution. This shows that the assimilated model represents in a proper way the observation distribution; no information is lost or added during the assimilation.

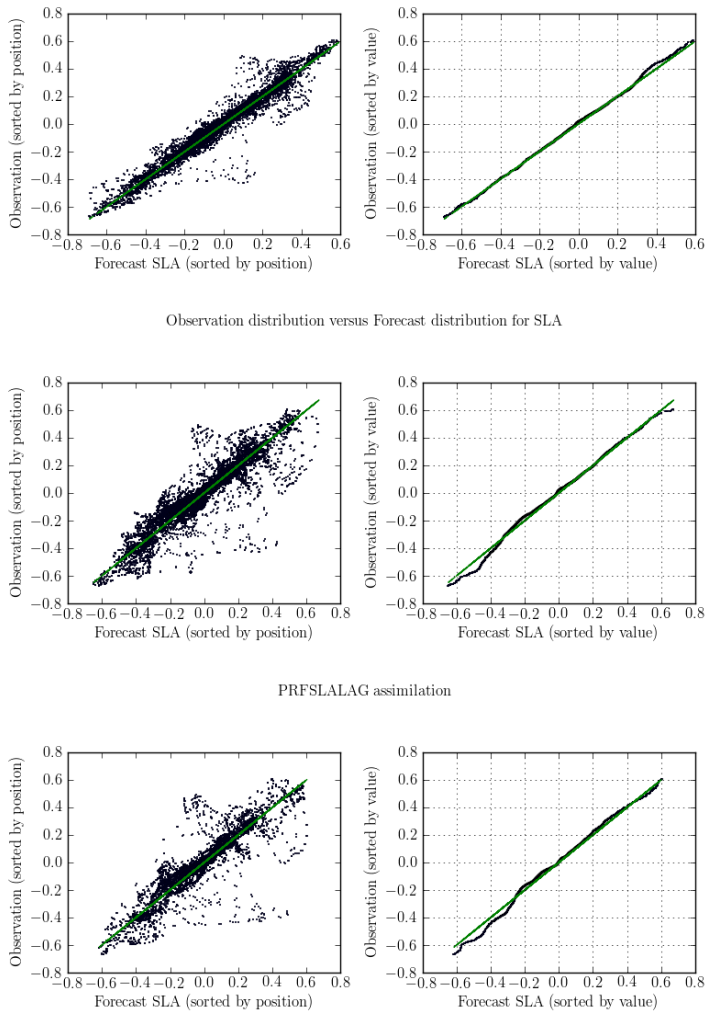


Figure 13: Q-Q plots for SLA observations, and for the three strategies of assimilation: using SLA (first line), PRF+SLA (second line) and PRF+SLA+LAG observations (third line). Two statistical characteristics are compared between analysed state, and observation distribution: on the left we study, according to each observation location, the quality of the analysed state; on the right, the comparison is done according to the value of each state variable; the spread of the points around the observation distribution line reveals the quality of the model counterparty statistics.

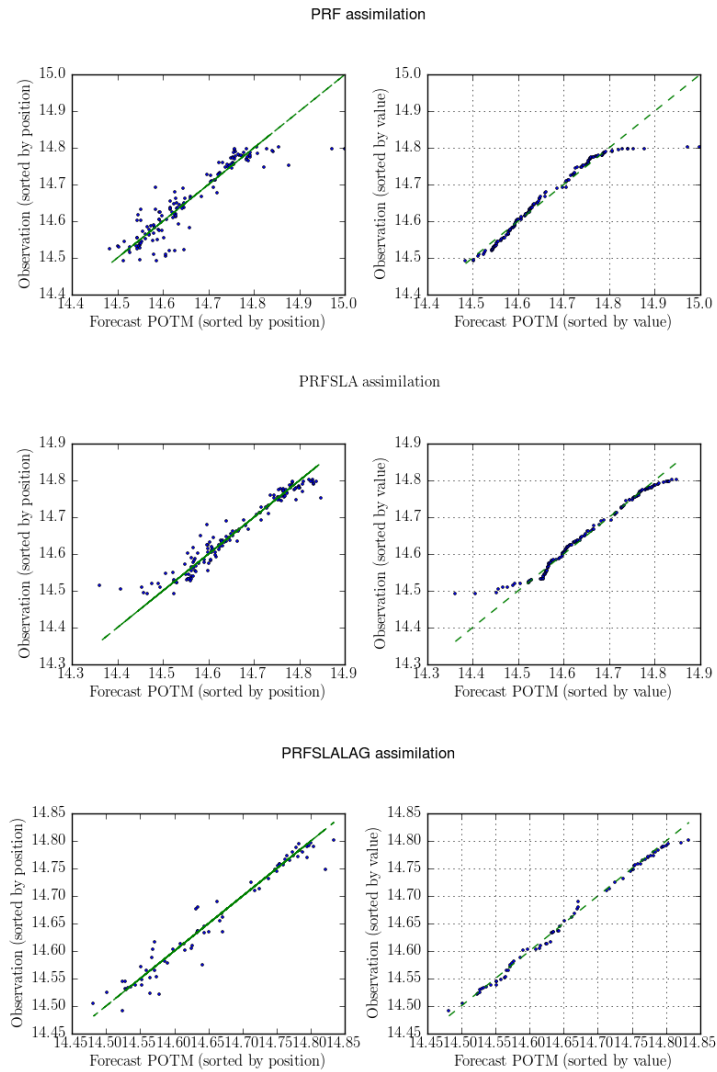


Figure 14: Q-Q plots for the PRF observations for the temperature. Three test cases are presented: first line: assimilation of PRF data; second line, assimilation of PRF+SLA data, and finally third line assimilation of PRF+SLA+LAG data. The figures of the two columns are constructed according to the same principle as for figure 13.

The addition of Lagrangian data assimilation introduces a disorder in both these properties: on right, the η distribution is more spread, and a small bias is introduced in the negative values of SLA.

Figure 14 reveals a different behaviour for T . The data sparsity makes the first column hard to interpret, but we can distinguish a tendency of the assimilated variable T to be biased for extreme values (higher values when assimilating PRF observations only, lower ones for PRF+SLA assimilation). Addition of Lagrangian data assimilation attenuates this phenomena, which is a good point. The analysed state distribution represents the right temperature observation distribution, in the whole domain (first column shows the good accordance of each model counterpart with its corresponding observation), as well as in all temperature definition interval (second column). Lagrangian data assimilation shows here its stronger benefit to assimilation system.

Conclusion

A new observation operator has been implemented in the NEMO / NEMOVAR framework, using the NEMO-OBS framework for Observation operator. This observation operator consists in computing trajectories of floats inside a domain, using model velocities. Its tangent requires a special treatment, since we have to differentiate the code with respect to observation position.

Conclusions made in another study (see [?]) are obtained here in a more realistic configuration. Moreover, our study goes one step further, by showing the impact of Lagrangian assimilation with realistic observation distribution. The specific nature of Lagrangian data requires a very high number of floats to give a real impact on the assimilation system. Nevertheless, we have shown that with a realistic distribution of floats, the temperature distribution was significantly improved, which can be very interesting in some applications.

Contents

1	The assimilation problem	4
1.1	Formulation of the incremental algorithm	4
1.2	3D-FGAT and 4D-Var incremental formulations	5
1.3	On the tangent linear hypothesis	5
1.4	Expressions of R and B	5
2	Observation operator for float trajectories	6
2.1	Main principles	7
2.2	The interpolation operator \mathbf{I}	8
2.3	Tangent and adjoint observation operators	9
2.4	Validation of the Lagrangian observation operator, its tangent and its adjoint	10
3	Experimental set up	11
3.1	Configuration	12
3.2	Identical twin experiments	12
3.3	Observations	14
3.4	Tangent Linear Hypothesis in this configuration	16
4	Numerical results on the assimilation of Lagrangian data	17
4.1	Assimilation of SLA observations	17
4.2	Assimilation of PRF observations	18
4.3	Assimilation of SLA+PRF observations	19
4.4	Assimilation of SLA+PRF+LAG observations	19



**RESEARCH CENTRE
GRENOBLE – RHÔNE-ALPES**

Inovallée
655 avenue de l'Europe Montbonnot
38334 Saint Ismier Cedex

Publisher
Inria
Domaine de Voluceau - Rocquencourt
BP 105 - 78153 Le Chesnay Cedex
inria.fr

ISSN 0249-6399