Jacques Blum

University of Nice Sophia Antipolis

France

jblum@unice.fr

Data assimilation for geophysical problems : variational and sequential techniques

Modélisation du trafic aérien et Météorologie

5èmes rencontre Météo / Math. Appli., Météo France, Toulouse

8-9 Mars 2010

Talk overview

1. Data and models

2. 4D-VAR

3. Kalman filtering

4. Back and forth nudging

Motivations

Environmental and geophysical studies : forecast the natural evolution
$\rightsquigarrow$ retrieve at best the current state (or initial condition) of the environment.

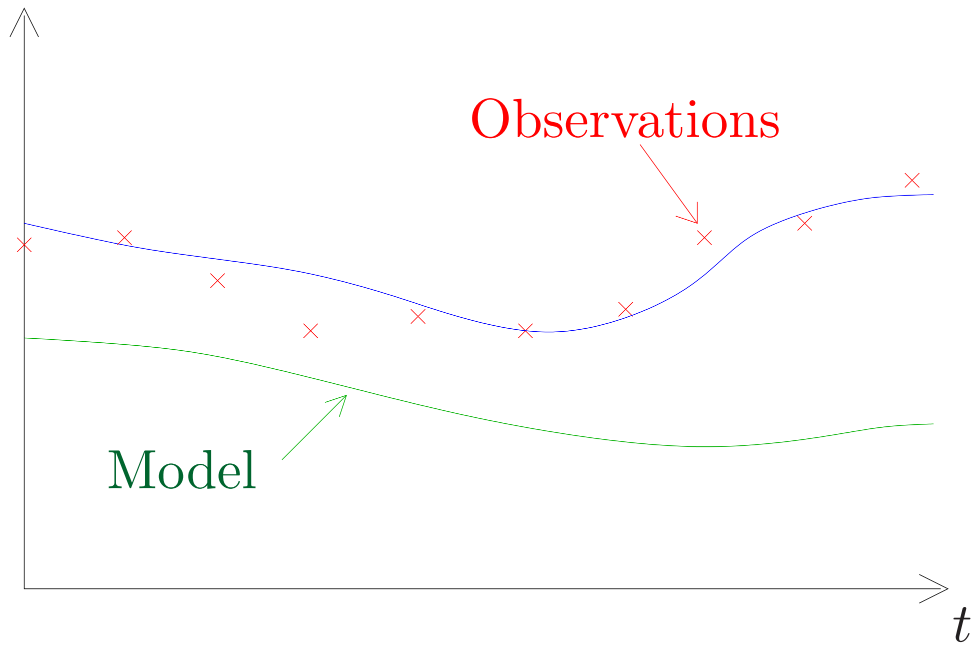
Geophysical fluids (atmosphere, oceans, ...) : turbulent systems $\Longrightarrow$ high sensitivity to the initial condition $\Longrightarrow$ need for a precise identification (much more than observations)

Environmental problems (ground pollution, air pollution, hurricanes, ...) : problems of huge dimension, generally poorly modelized or observed

Data assimilation consists in combining in an optimal way the observations of a system and the knowledge of the physical laws which govern it.

Main goal : identify the initial condition, or estimate some unknown parameters, and obtain reliable forecasts of the system evolution.

Data assimilation



combination

model + observations

$\Downarrow$

identification of the initial condition

in a geophysical system

Fundamental for a chaotic system (atmosphere, ocean, ...)

Issue : These systems are generally irreversible.

Goal : Combine models and data.

⇒ 1. Data and models

2. 4D-VAR

3. Kalman filtering

4. Back and forth nudging

Models

The equations governing the geophysical flows are derived from the general equations of fluid dynamics. The main variables used to describe the fluids are :

- The components of the velocity
- Pressure
- Temperature
- Humidity in the atmosphere, salinity in the ocean
- Concentrations for chemical species

The constraints applied to these variables are :

- Equations of mass conservation.
- Momentum equation containing the Coriolis acceleration term, which is essential in the dynamic of flows at extra tropical latitudes.
- Equation of energy conservation including law of thermodynamics.
- Law of behavior (e.g. Mariotte's Law).
- Equations of chemical kinetics if a pollution type problem is considered.

Models

Full model : primitive equations.

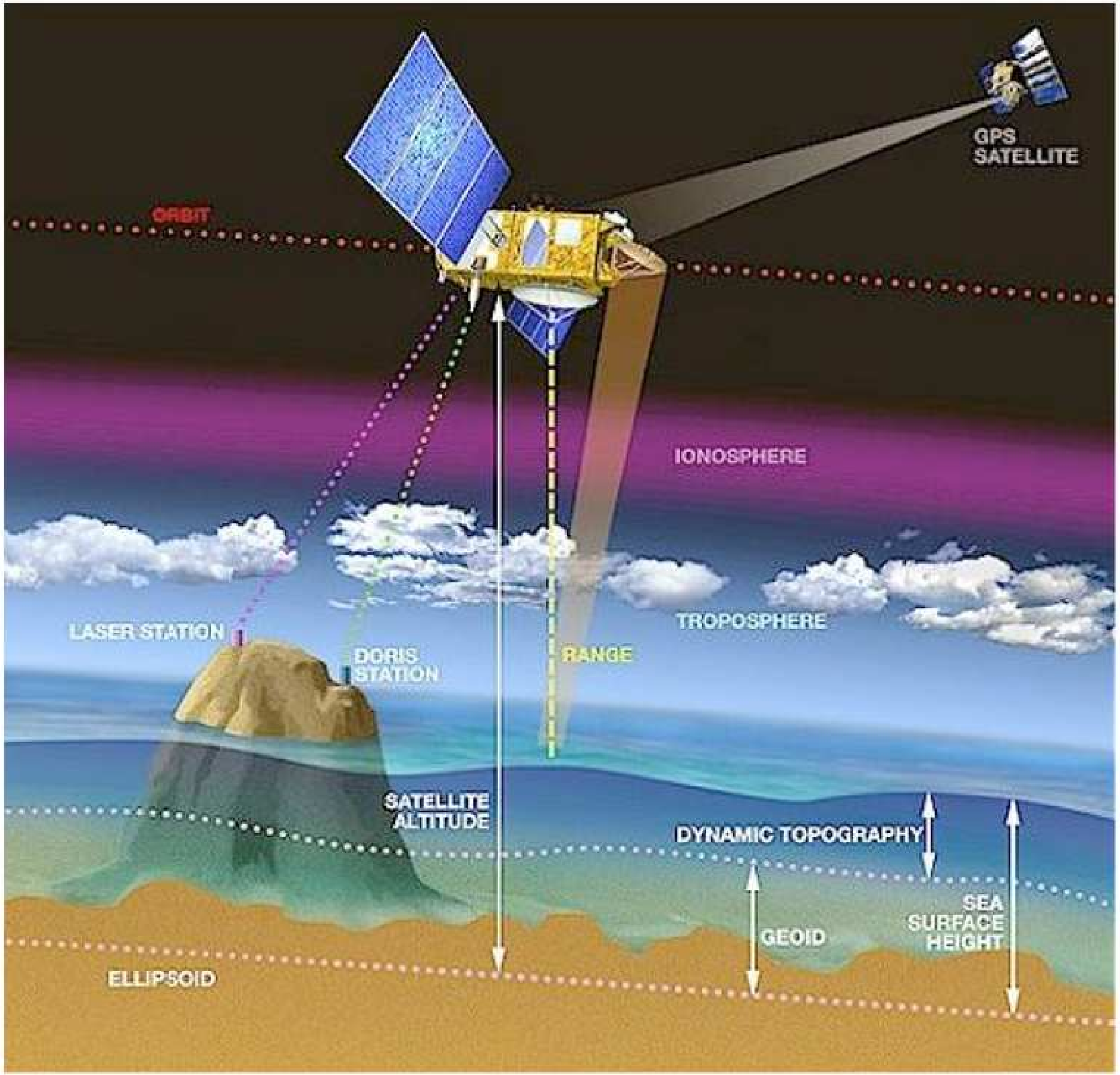
These equations are complex, therefore we cannot expect to obtain an analytical solution. Before performing a numerical analysis of the system it will be necessary to :

– **Simplify the equations.** This task will be carried out on physical basis. For example, three dimensional fields could be vertically integrated using hydrostatic assumptions in order to obtain a two dimensional horizontal field which is more tractable for numerical purposes : shallow-water equations.

  Other “toy” model : the quasi-geostrophic model obtained by a first-order expansion of the Navier-Stokes equation with respect to the Rossby number.

– **Discretize the equations.** The usual discretization methods are considered : finite differences, finite elements or spectral methods.

Data



Satellite altimetry (from AVISO web site).

Data assimilation methods

Data assimilation methods :

1. 4D-VAR : optimal control method, based on the minimization of a functional estimating the discrepancy between the model solution and the observations.

   [Le Dimet-Talagrand (Tellus, vol. 38A, 1986)]

2. Sequential methods : Kalman filtering, extended Kalman and ensemble Kalman filters.

   [Evensen (Ocean Dynamics, vol. 53, 2003)]

3. A new method : the Back and Forth Nudging.

   [Auroux-Blum (Nonlinear Processes in Geophysics, vol. 15, 2008)]

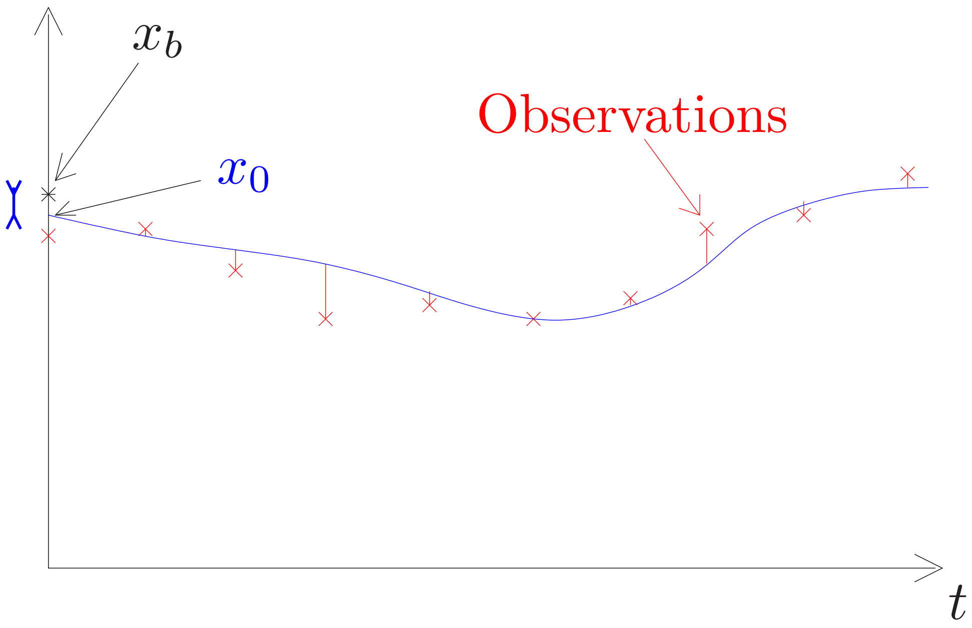
1. Data and models

$\Rightarrow$ 2. 4D-VAR

3. Kalman filtering

4. Back and forth nudging

4D-VAR



$$\begin{cases} \dfrac{dx}{dt} = F(x), \\ x(0) = x_0, \end{cases}$$

$x_{obs}(t)$: observations of the system, H : observation operator, x_b : background, B and R : covariance matrices of background and observation errors respectively.

$$\begin{aligned} J(x_0) &= \frac{1}{2}(x_0 - x_b)^T B^{-1}(x_0 - x_b) \\ &+ \frac{1}{2}\int_0^T [x_{obs}(t) - H(x(t))]^T R^{-1} [x_{obs}(t) - H(x(t))] \; dt \end{aligned}$$

Optimality system

Optimization under constraints :

$$\mathcal{L}(x_0, x, p) = J(x_0) + \int_0^T \left\langle p, \frac{dx}{dt} - F(x) \right\rangle dt$$

Direct model : $\begin{cases} \dfrac{dx}{dt} = F(x) \\ x(0) = x_0 \end{cases}$

Adjoint model : $\begin{cases} -\dfrac{dp}{dt} = \left[\dfrac{\partial F}{\partial x}\right]^T p + H^T R^{-1} \left[x_{obs}(t) - H(x(t))\right] \\ p(T) = 0 \end{cases}$

Gradient of the cost-function : $\dfrac{\partial J}{\partial x_0} = B^{-1}(x_0 - x_b) - p(0)$

Optimal solution : $x_0 = x_b + Bp(0)$ [Le Dimet-Talagrand (86)]

Example : Quasi-Geostrophic ocean model

We consider altimetric measurement of the surface of the ocean given by satellite observations (Topex-Poseidon, Jason). The observed data is the change in the surface of the ocean. According to the quasi geostrophic approximation it is proportional to the stream function in the surface layer :

$$h^{obs} = \frac{f_0}{g} \Psi_1^{obs}$$

Therefore we will assimilate surface data in order to retrieve the fluid circulation especially in the deep ocean layers.

The control vector is the initial state on the N layers :

$$u = \Big(\Psi_k(t=0)\Big)_{k=1,\dots,N} \in \mathcal{U}_{\text{ad}}$$

The state vector is

$$\Big(\Psi_k(t)\Big)_{k=1,\dots,N}$$

Example

We assume that the stream function is observed at each point of the surface layer at discrete times t_j. Then the cost function is defined by :

$$\mathcal{J}_\varepsilon(u) = \frac{1}{2}\sum_{j=1}^{n}\int_\Omega \Big(\Psi_1(t_j) - \Psi_1^{obs}(t_j)\Big)^2 \mathrm{d}s + \frac{\varepsilon}{2} \parallel R(u) \parallel_{\mathcal{T}}^2$$

The second term in the cost function is the regularization term in the sense of Tikhonov. It renders the inverse problem well posed, by taking into account the square of the potential vorticity of the initial state :

$$\parallel R(u) \parallel_{\mathcal{T}}^2 = \sum_{k=1}^{N} H_k \left[\int_\Omega \Big((\Delta\Psi_k)(0) - [W]_k.(\Psi)(0)\Big)^2 \mathrm{d}s\right]$$

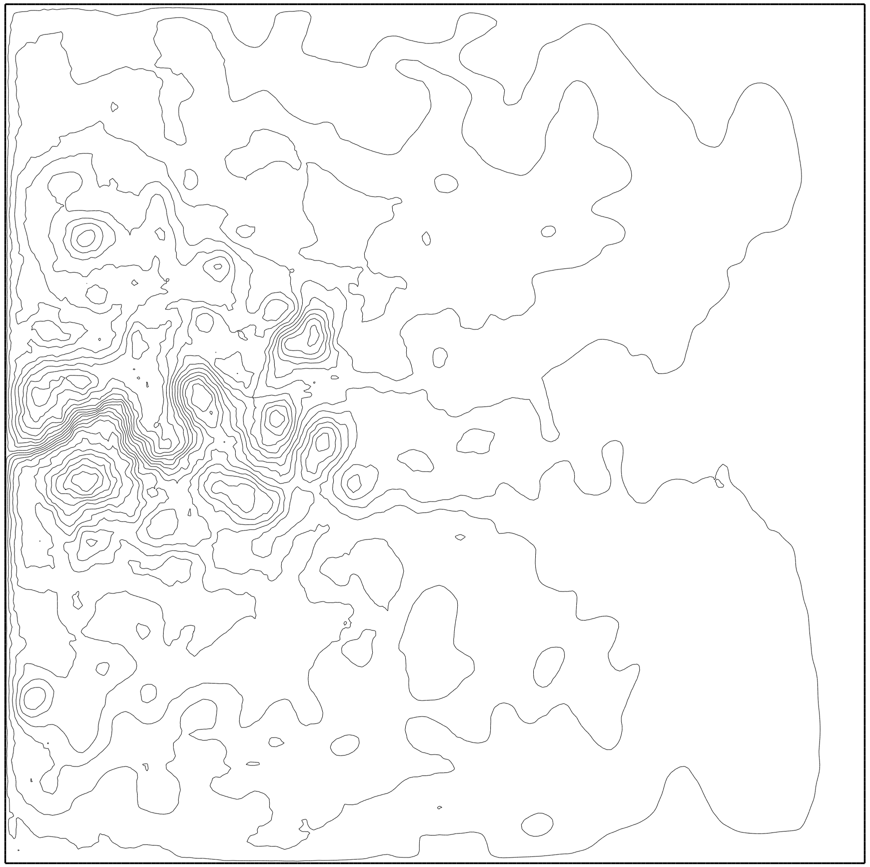
The parameter ε in the cost function is the relative weight of the regularization with respect to the quadratic distance between the observations and the computed state.

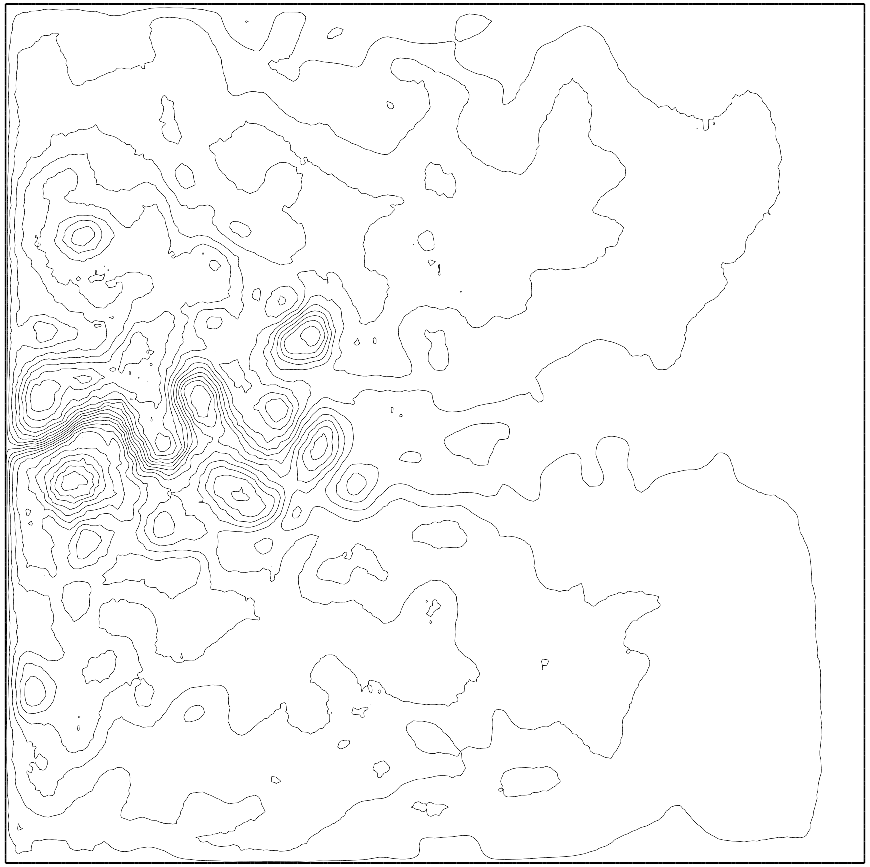
[Luong-Blum-Verron (98), Auroux-Blum (04)]

Example

True solution

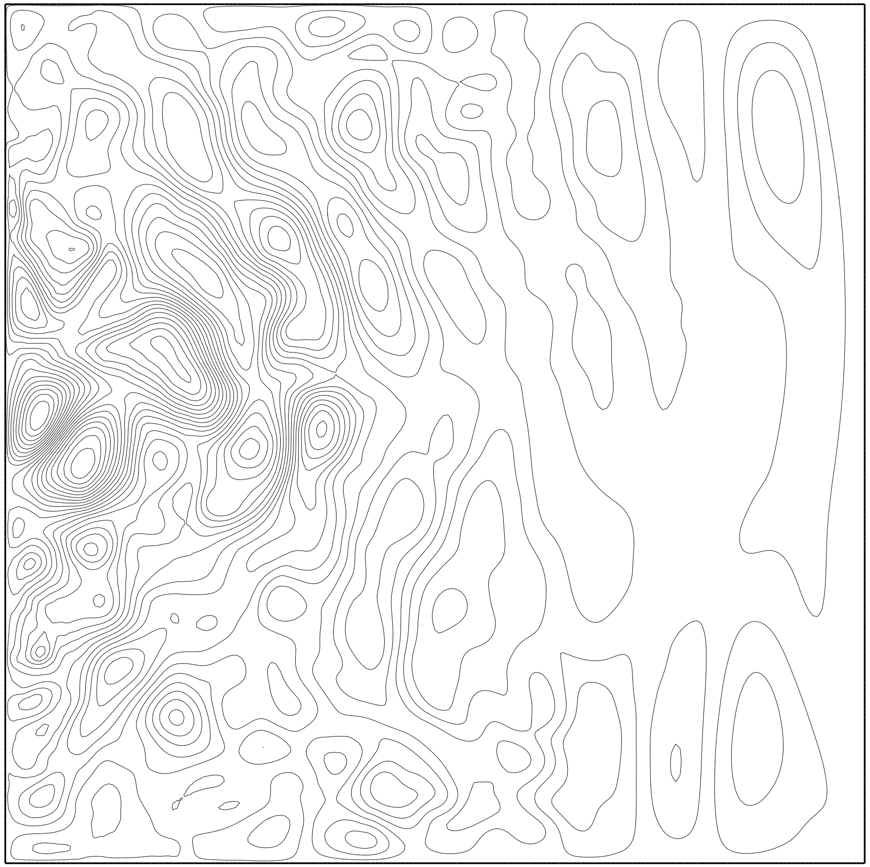
4D-VAR identified solution

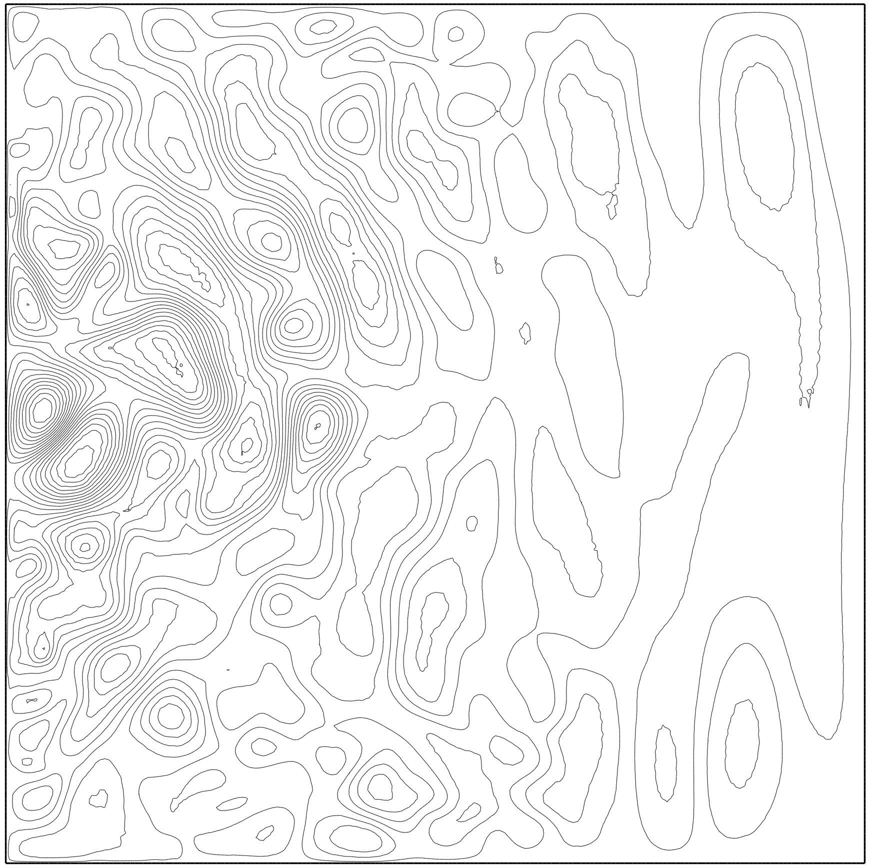




True initial condition (left) and identified initial condition by the 4D-VAR (right), for the upper layer.

Example





True initial condition (left) and identified initial condition by the 4D-VAR (right), for the bottom layer.

Kalman filter

The Kalman filter is a recursive filter that estimates the state of a dynamic system from a series of incomplete and noisy measurements.
We consider a discrete in time stochastic dynamic system for the true vector x^t :

$$x_k^t = M_{k-1} x_{k-1}^t + \eta_{k-1}$$

where k is the index of the observation time and M_k represents model dynamics while η_k is model error white in time with mean zero and covariance Q_k.

Consider a linear observation process described by

$$y_k^0 = H_k x_k^t + e_k.$$

y_k^0 is the vector of observations while the vector e_k is an additive noise representing the error in observations due for instance to instrumental error.

Random noise e_k is assumed white in time with mean 0 and covariance R_k.

Kalman filter

– **Advance in time :**

$$\begin{cases} x_k^f = M_{k-1} x_{k-1}^a \\ \\ P_k^f = M_{k-1} P_{k-1}^a M_{k-1}^T + Q_{k-1} \end{cases}$$

where the forecast and analysis error covariance matrices at time k are given by :

$$\begin{cases} P_k^f = E\{(x_k^t - x_k^f)(x_k^t - x_k^f)^T\} \\ \\ P_k^a = E\{(x_k^t - x_k^a)(x_k^t - x_k^a)^T\} \end{cases}$$

Q_{k-1} is the model error covariance matrix at time $t = t_{k-1}$, M_{k-1} is the model dynamics. x_{k-1}^a and x_{k-1}^f are the analysis and the forecast at time $t = t_{k-1}$.

Kalman filter

– **Compute the Kalman gain :**

$$K_k = P_k^f H_k^T (H_k P_k^f H_k^T + R_k)^{-1}$$

The matrix K_k is the optimal weighting matrix known as the Kalman gain matrix.

– **Update the state :**

$$x_k^a = x_k^f + K_k(y_k^0 - H_k x_k^f)$$

Where y_k^0 is the observation at time $t = t_k$, and H_k is the observation matrix at time $t = t_k$.

– **Update error covariance matrix :**

$$P_k^a = (I - K_k H_k) P_k^f$$

Kalman filter

Computational cost of Kalman filter :

The Kalman filter assuming the dynamical model has n unknowns in the state vector then error covariance matrix has n^2 unknowns.

The evolution of the error covariance is very time consuming.

Thus KF in usual form can only be used for rather low dimensional dynamical models.

The basic Kalman filter is limited to a linear assumption. However, most non-trivial systems are non-linear. The non-linearity can be associated either with the process model or with the observation model or with both.

Extended Kalman filter

In extended Kalman filter (EKF) the state transition and observation models need not be linear functions of the state but may instead be non-linear functions.

The Jacobian or Tangent Linear Model is computed. At each time step the Jacobian is evaluated with current predicted states. These matrices can be used in the Kalman filter equations. This process essentially linearizes the non-linear function around the current estimate.

Shortcomings of the EKF :

Unlike its linear counterpart, the EKF is not an optimal estimator. In addition, if the initial estimate of the state is wrong, or if the process is modeled incorrectly, the filter may quickly diverge, owing to its linearization.

Usefulness of EKF will depend on properties of the model dynamics.

Ensemble Kalman filter

Ensemble Kalman filter (EnKF) :

The EnKF is a Monte Carlo approximation of the Kalman filter avoiding evolving the covariance matrix of the pdf of the state vector x. Instead the probability distribution is represented by a sample

$$X = [x_1, x_2, \cdots, x_N] = [x_i]$$

X is an $n \times N$ matrix whose columns are the ensemble members, and it is called the prior ensemble.

Ideally, ensemble members would form a sample from the prior distribution. However, the ensemble members are not in general independent except in the initial ensemble, since every EnKF step ties them together. They are deemed to be approximately independent, and all calculations proceed as if they actually were independent.

Ensemble Kalman filter

The initial ensemble should ideally be chosen to properly represent the error statistics of the initial condition.

The ensemble of model states is integrated forward in time according to the non-linear model equations, with a stochastic model error term.

The EnKF is now obtained simply by replacing the state covariance P in Kalman gain matrix :

$$K = PH^T(HPH^T + R)^{-1}$$

by the sample covariance C computed from the ensemble members (called the ensemble covariance).

[Evensen (03)]

1. Data and models
2. 4D-VAR
3. Kalman filtering
4. Back and forth nudging

Forward nudging

Let us consider a model governed by a system of ODE :

$$\frac{dX}{dt} = F(X), \quad 0 < t < T,$$

with an initial condition $X(0) = x_0$.

$X_{obs}(t)$: observations of the system
H : observation operator.

$$\begin{cases} \dfrac{dX}{dt} = F(X) + K(X_{obs} - H(X)), \quad 0 < t < T, \\ X(0) = x_0, \end{cases}$$

where K is the nudging (or gain) matrix.

In the linear case (where F is a matrix), the forward nudging is called Luenberger or asymptotic observer.

Forward nudging

– Meteorology : Hoke-Anthes (1976)

– Oceanography (QG model) : Verron-Holland (1989)

– Atmosphere (meso-scale) : Stauffer-Seaman (1990)

– Optimal determination of the nudging coeffcients :

Zou-Navon-Le Dimet (1992), Stauffer-Bao (1993),
Vidard-Le Dimet-Piacentini (2003)

Forward nudging : linear case

Luenberger observer, or asymptotic observer

(Luenberger, 1966)

$$\begin{cases} \dfrac{dX}{dt} = FX + K(X_{obs} - HX), \\ \dfrac{d\hat{X}}{dt} = F\hat{X}, \quad X_{obs} = H\hat{X}. \end{cases}$$

$$\frac{d}{dt}(X - \hat{X}) = (F - KH)(X - \hat{X})$$

If $F - KH$ is a Hurwitz matrix, i.e. its spectrum is strictly included in the half-plane $\{\lambda \in \mathbb{C}; Re(\lambda) < 0\}$, then $X \to \hat{X}$ when $t \to +\infty$.

BFN : Back and Forth Nudging algorithm

Iterative algorithm (forward and backward resolutions) :

$$\tilde{X}_0(0) = \tilde{x}_0 \text{ (first guess)}$$

$$\begin{cases} \dfrac{dX_k}{dt} = F(X_k) + K(X_{obs} - H(X_k)) \\ X_k(0) = \tilde{X}_{k-1}(0) \end{cases}$$

$$\begin{cases} \dfrac{d\tilde{X}_k}{dt} = F(\tilde{X}_k) - K'(X_{obs} - H(\tilde{X}_k)) \\ \tilde{X}_k(T) = X_k(T) \end{cases}$$

If X_k and $\tilde{X}_k$ converge towards the same vector X, and if $K = K'$, then X satisfies the state equation and fits to the observations.

Choice of the direct nudging matrix K

Implicit discretization of the direct model equation with nudging :

$$\frac{X^{n+1} - X^n}{\Delta t} = FX^{n+1} + K(X_{obs} - HX^{n+1}).$$

Variational interpretation : direct nudging is a compromise between the minimization of the energy of the system and the quadratic distance to the observations :

$$\min_X \left[\frac{1}{2} \langle X - X^n, X - X^n \rangle - \frac{\Delta t}{2} \langle FX, X \rangle + \frac{\Delta t}{2} \langle R^{-1}(X_{obs} - HX), X_{obs} - HX \rangle \right],$$

by choosing

$$K = H^T R^{-1}$$

where R is the covariance matrix of the errors of observation.

[Auroux-Blum (08)]

Choice of the backward nudging matrix K'

The feedback term has a double role :

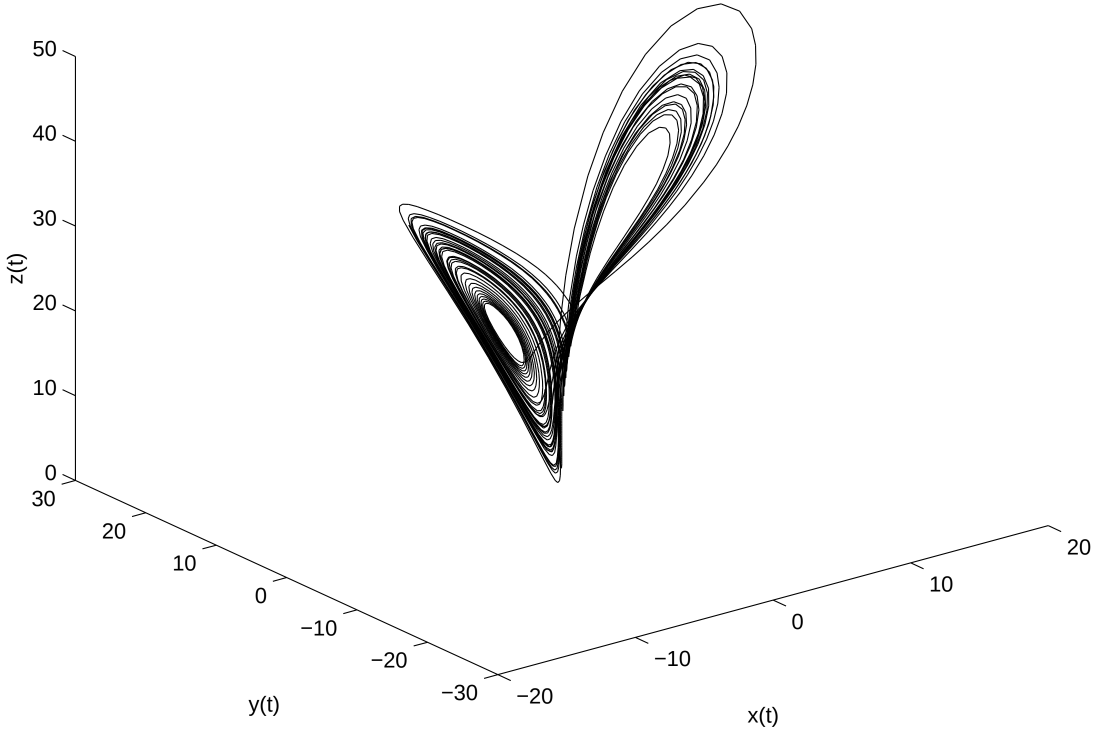
- stabilization of the backward resolution of the model (irreversible system)
- feedback to the observations

If the system is observable, i.e. $rank[H, HF, \dots, HF^{N-1}] = N$, then there exists a matrix K' such that $-F - K'H$ is a Hurwitz matrix (pole assignment method).

In practice, $K' = k'H^T$ and k' can be chosen as being the smallest value making the backward numerical resolution stable.

Lorenz' equations

$$\begin{cases} \dfrac{dx}{dt} = 10\ (y - x), \\[2ex] \dfrac{dy}{dt} = 28\ x - y - xz, \\[2ex] \dfrac{dz}{dt} = -\dfrac{8}{3}\ z + xy. \end{cases}$$



– Assimilation period : $[0; 3]$, forecast : $[3; 6]$.

– Time step : 0.001.

– 31 observations (every 100 time steps).

Lorenz - BFN convergence

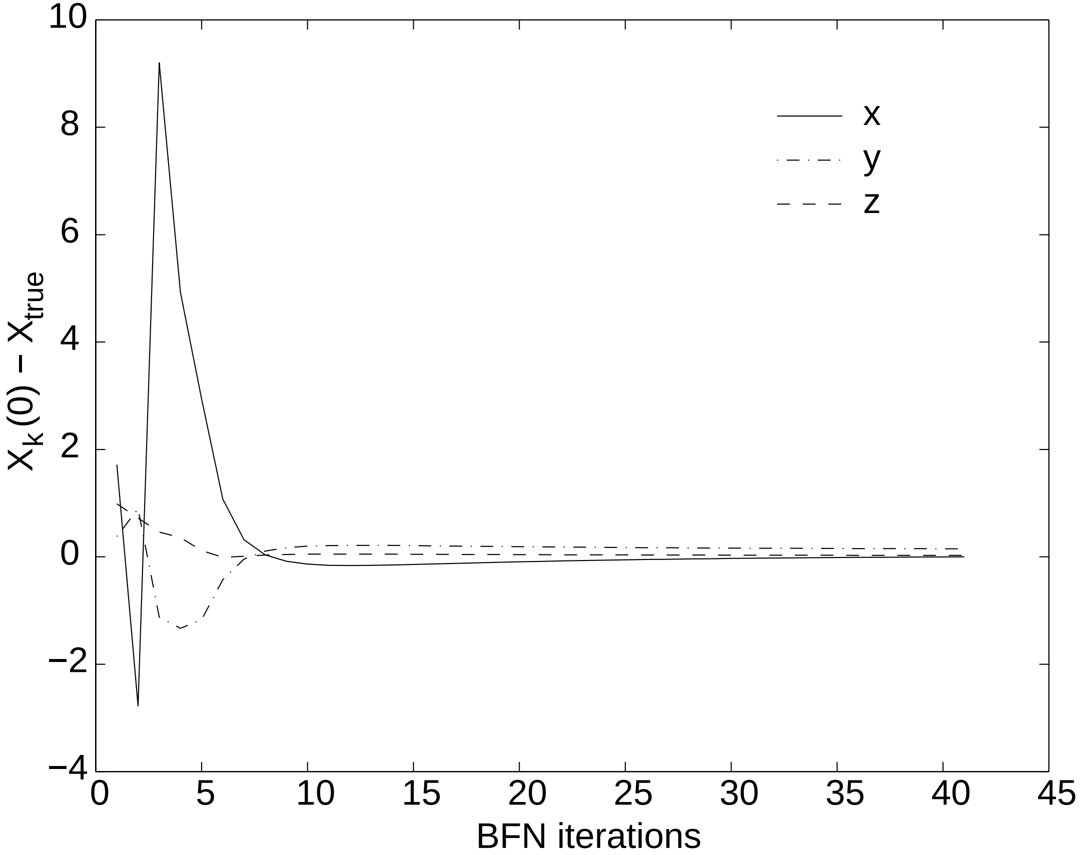


FIG. 1 – Difference between the kth iterate $X_k(0)$ and the exact initial condition x_{true} for the 3 variables versus the number of BFN iterations.

Lorenz - BFN convergence

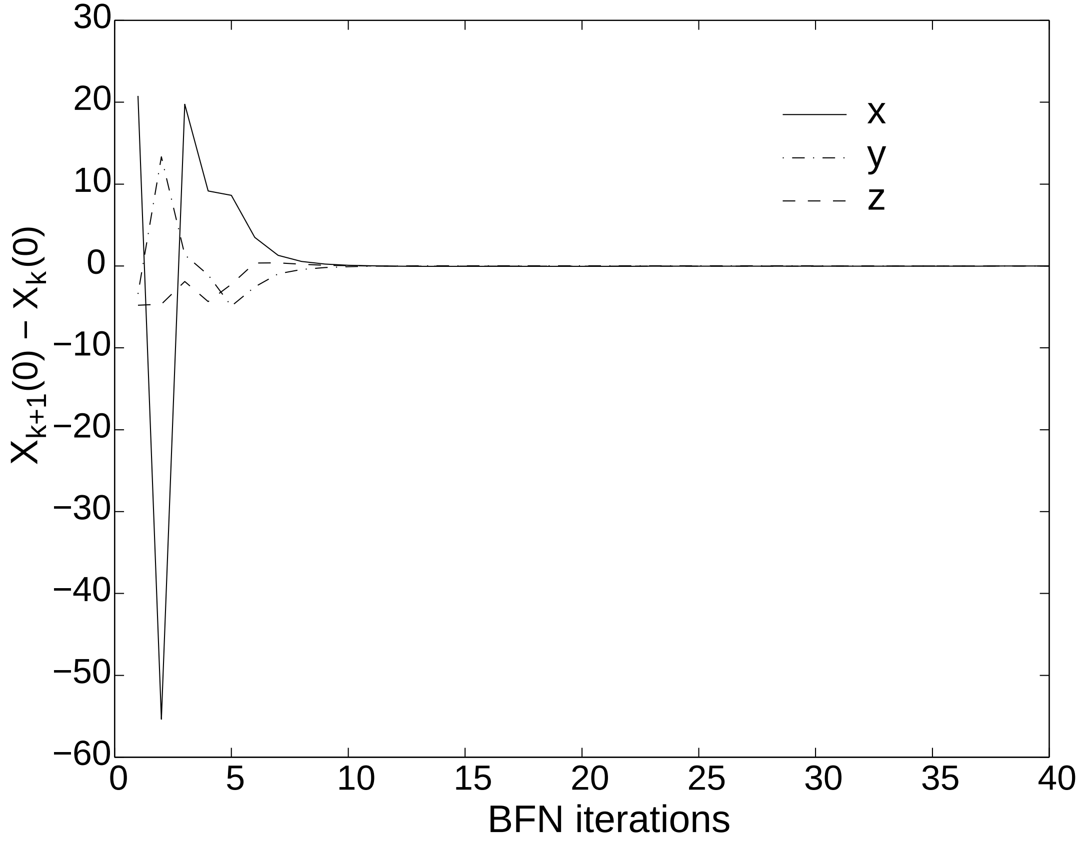


FIG. 2 – Difference between two consecutive BFN iterates for the 3 variables versus the number of BFN iterations.

Lorenz - BFN convergence

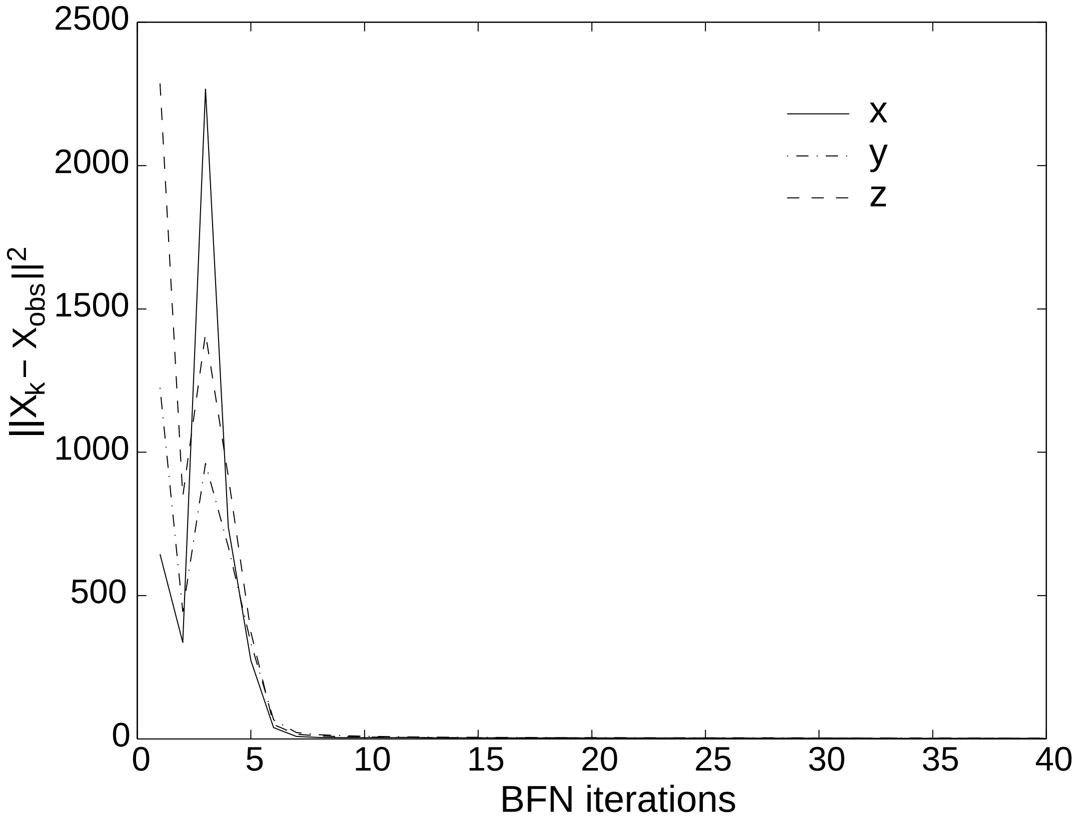


FIG. 3 – RMS difference between the observations and the BFN identified trajectory versus the BFN iterations.

Lorenz - Comparison BFN/4D-VAR

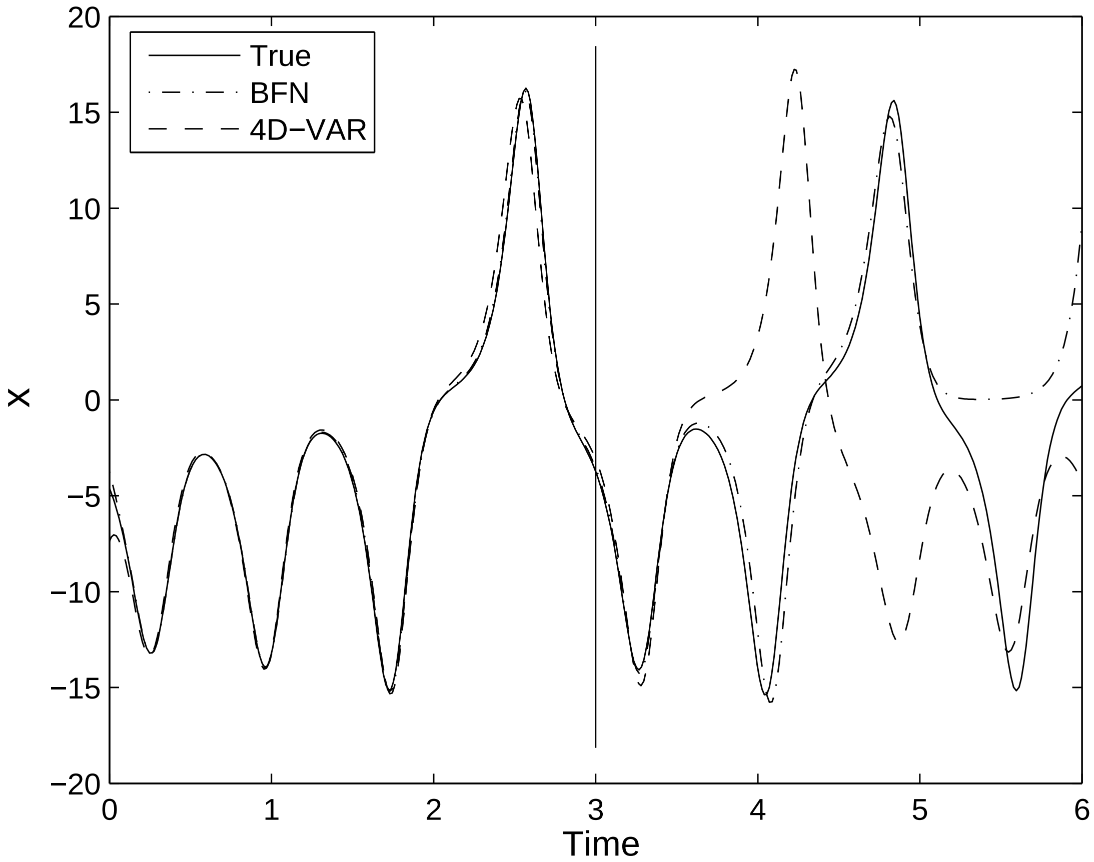


FIG. 4 – Evolution in time of the reference trajectory (plain line), and of the trajectories identified by the 4D-VAR (dashed line) and the BFN (dash-dotted line) algorithms, in the case of perfect observations and for the first Lorenz variable x.

Lorenz - Comparison BFN/4D-VAR

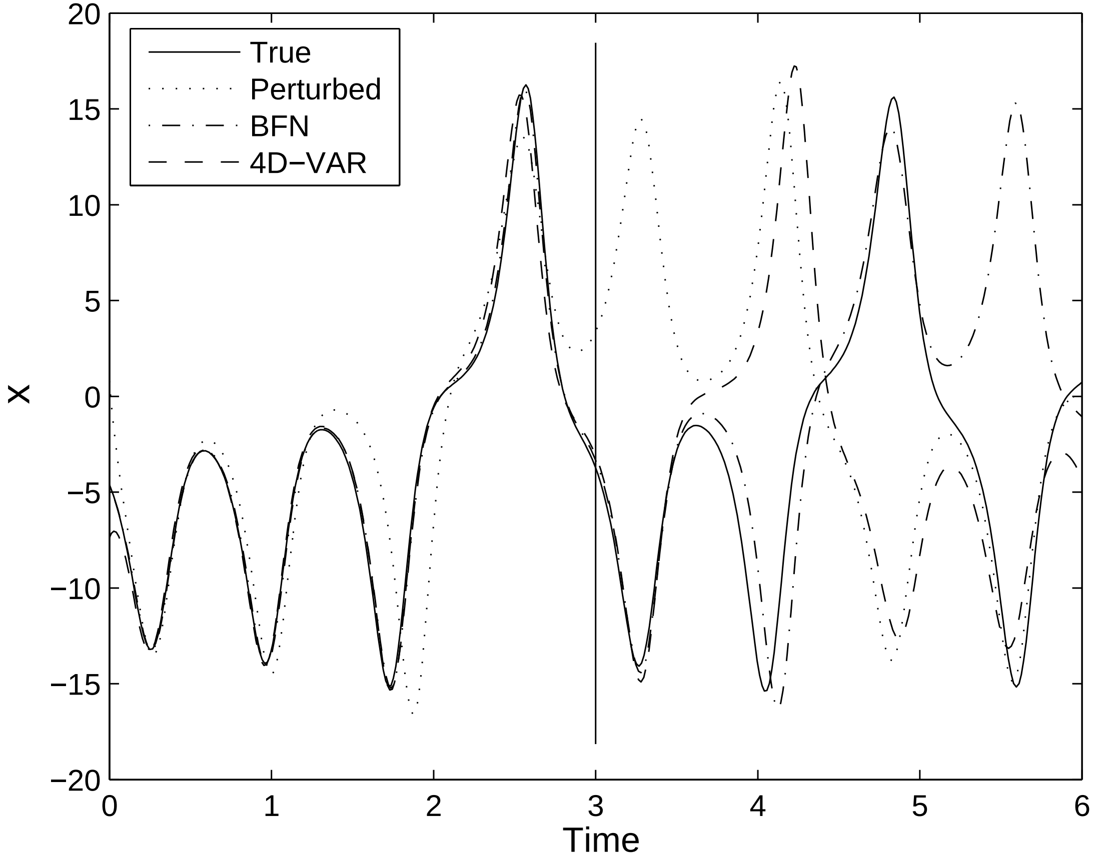


FIG. 5 – Evolution in time of the reference trajectories (plain line), and of the trajectories identified by the 4D-VAR (dashed line) and the BFN (dash-dotted line) algorithms, in the case of noised observations (with a 10% gaussian blank noise) and for the first Lorenz variable x.

Full primitive ocean model

Primitive equations : Navier-Stokes equations (velocity-pressure), coupled with two active tracers (temperature and salinity).

Momentum balance :

$$\frac{\partial U_h}{\partial t} = -\left[(\nabla \wedge U) \wedge U + \frac{1}{2}\nabla(|U|^2)\right]_h - f.z \wedge U_h - \frac{1}{\rho_0}\nabla_h p + D^U + F^U$$

Incompressibility equation : $\nabla . U = 0$

Hydrostatic equilibrium : $\frac{\partial p}{\partial z} = -\rho g$

Heat and salt conservation equations :

$$\frac{\partial T}{\partial t} = -\nabla.(TU) + D^T + F^T \quad \text{(+ same for S)}$$

Equation of state : $\rho = \rho(T, S, p)$

Full primitive ocean model

Free surface formulation : the height of the sea surface η is given by

$$\frac{\partial \eta}{\partial t} = -div_h((H+\eta)\bar{U_h}) + [P-E]$$

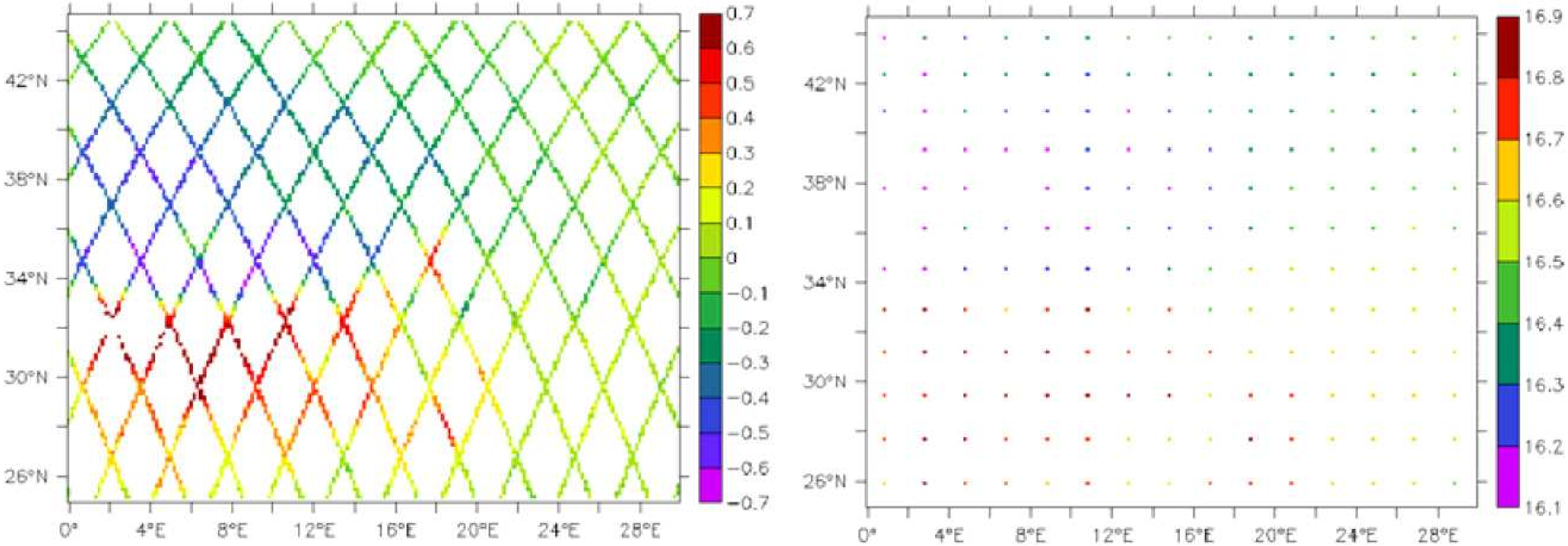
The surface pressure is given by : $\quad p_s = \rho g \eta.$

This boundary condition is then used for integrating the hydrostatic equilibrium and calculating the pressure.

Numerical experiments : double gyre circulation confined between closed boundaries (similar to the shallow water model). The circulation is forced by a sinusoidal (with latitude) zonal wind.

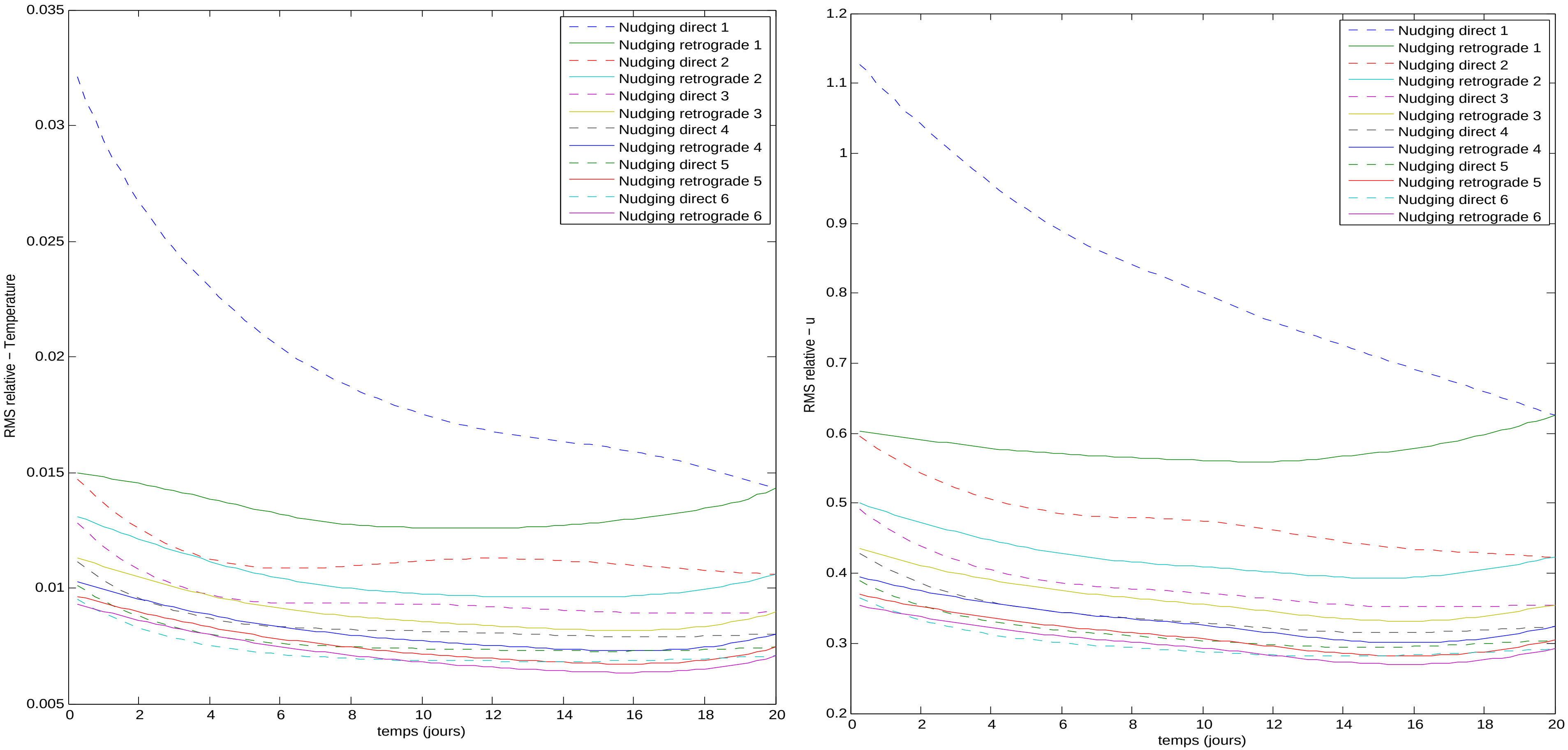
Twin experiments : observations are extracted from a reference run, according to networks of realistic density : SSH is observed similarly to TOPEX/POSEIDON, and temperature is observed on a regular grid that mimics the ARGO network density.

Full primitive ocean model



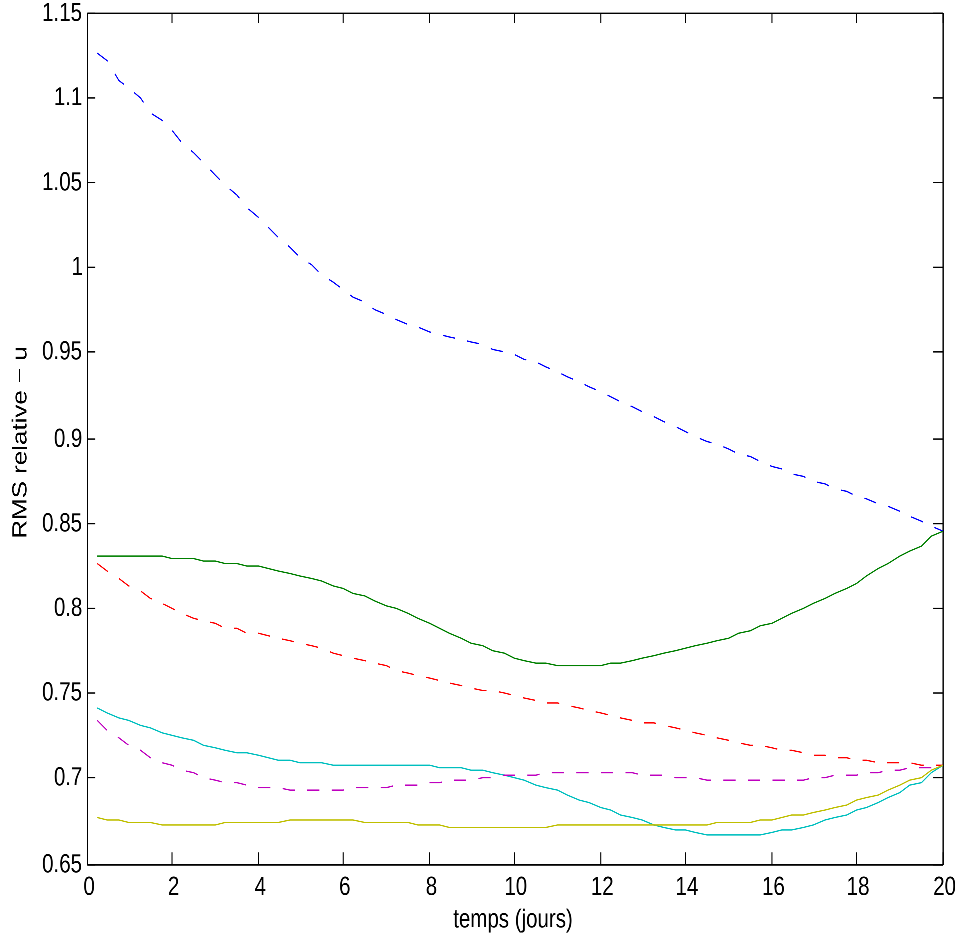
Example of observation network used in the assimilation : along-track altimetric observations (Topex-Poseidon) of the SSH every 10 days ; vertical profiles of temperature (ARGO float network) every 18 days.

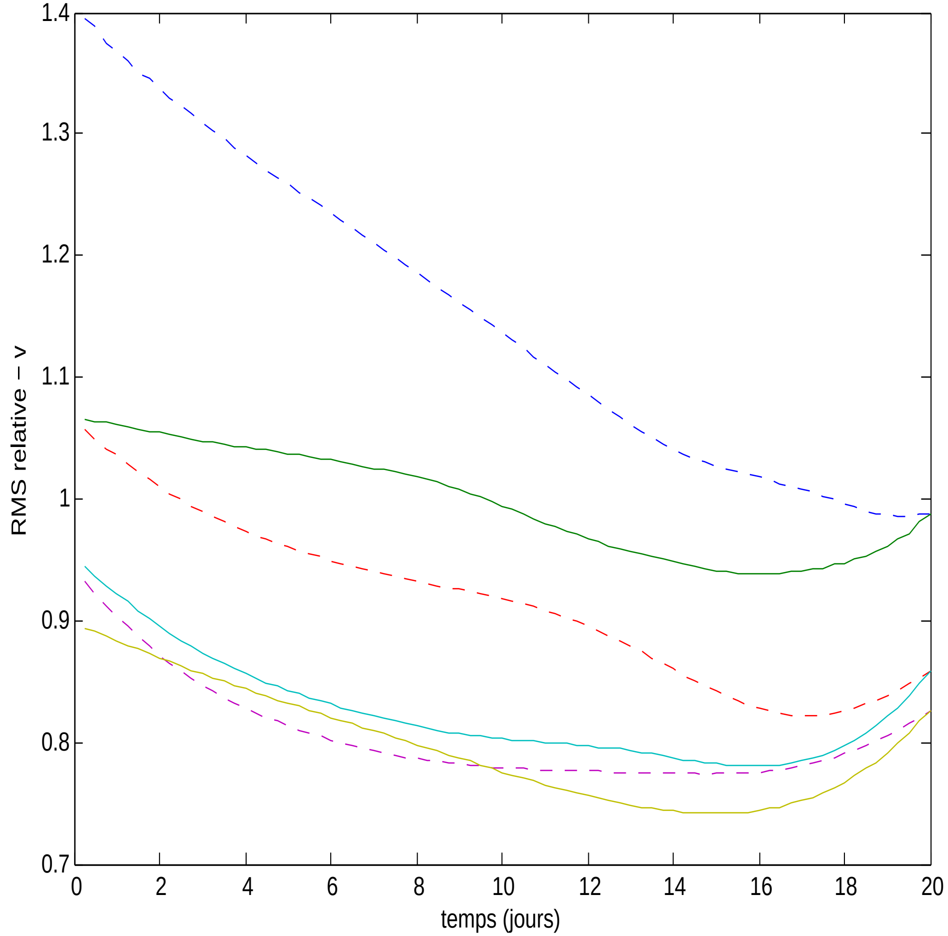
Numerical results



Relative RMS error of the temperature (left) and longitudinal velocity (right), 6 iterations of BFN (nudging terms in the temperature and SSH equations only), with full and unnoisy SSH observations every day.

Numerical results





Relative RMS error of the longitudinal and transversal velocities, 3 iterations of BFN (nudging terms in the temperature and SSH equations only), with “realistic” SSH observations (T/P track + 15% noise).

Conclusions

4D-VAR :

- Requires linearization of the model, computation of the adjoint state and an optimization algorithm
- Requires covariance error matrices on observations and background.
- Model is a strong constraint
- Advantage : robustness and global optimization (reanalysis)

Kalman filtering :

- No adjoint state
- Drawback : huge error covariance matrices
- Ensemble Kalman filter becomes more realistic for the implementation
- Requires simulation of model errors

Conclusions

BFN :

- Easy implementation (no linearization, no adjoint state, no minimization process)
- Very efficient in the first iterations
- Converges more rapidly than 4D-VAR
- Lower computational and memory costs than 4D-VAR
- Model is a weak constraint
- Could be an excellent preconditioner for 4D-VAR