Didier Auroux

University of Nice Sophia Antipolis

France

auroux@unice.fr

Data assimilation for geophysical fluids

Modelling and simulation of complex systems : stochastic and deterministic approaches

CEMRACS'13 July 22 - August 30, 2013

Talk overview

1. Data and models
2. Variational methods : 4D-VAR
3. Sequential methods : Kalman filter
4. Hybrid scheme : Back and Forth Nudging algorithm
5. Diffusive BFN algorithm

Motivations

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

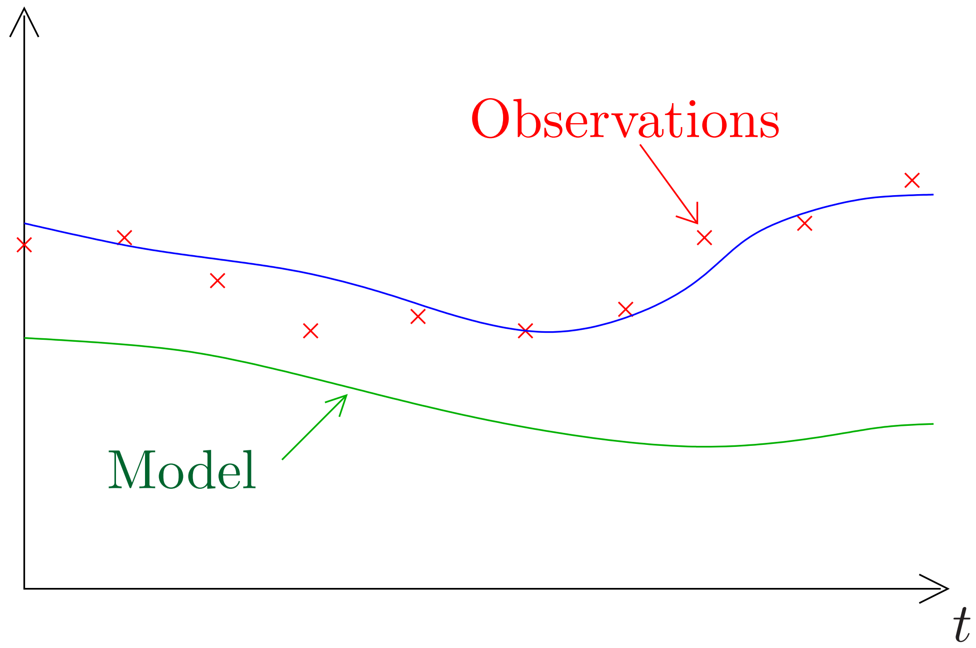
Geophysical fluids (atmosphere, oceans, ...) : turbulent systems $\Longrightarrow$ high sensitivity to the initial condition $\Longrightarrow$ need for a precise identification (much more than observations)

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.

$\Rightarrow$ 1. Data and models

2. Variational methods : 4D-VAR

3. Sequential methods : Kalman filter

4. Hybrid scheme : Back and Forth Nudging algorithm

5. Diffusive BFN algorithm

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 :

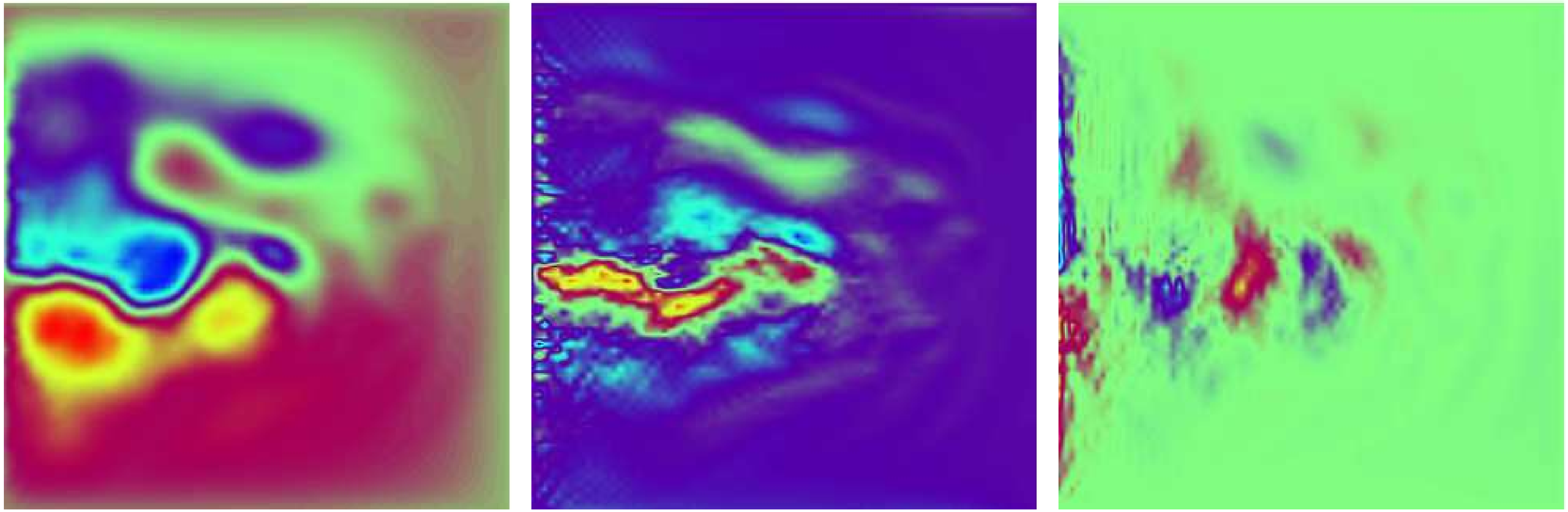
- The components of the velocity
- Pressure
- Temperature
- Humidity in the atmosphere, salinity in the ocean
- Concentrations for chemical species

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.

Shallow water model

$$
\begin{aligned}
\partial_t u - (f+\zeta)v + \partial_x B &= \frac{\tau_x}{\rho_0 h} - ru + \nu\Delta u \\
\partial_t v + (f+\zeta)u + \partial_y B &= \frac{\tau_y}{\rho_0 h} - rv + \nu\Delta v \qquad (1) \\
\partial_t h + \partial_x(hu) + \partial_y(hv) &= 0
\end{aligned}
$$

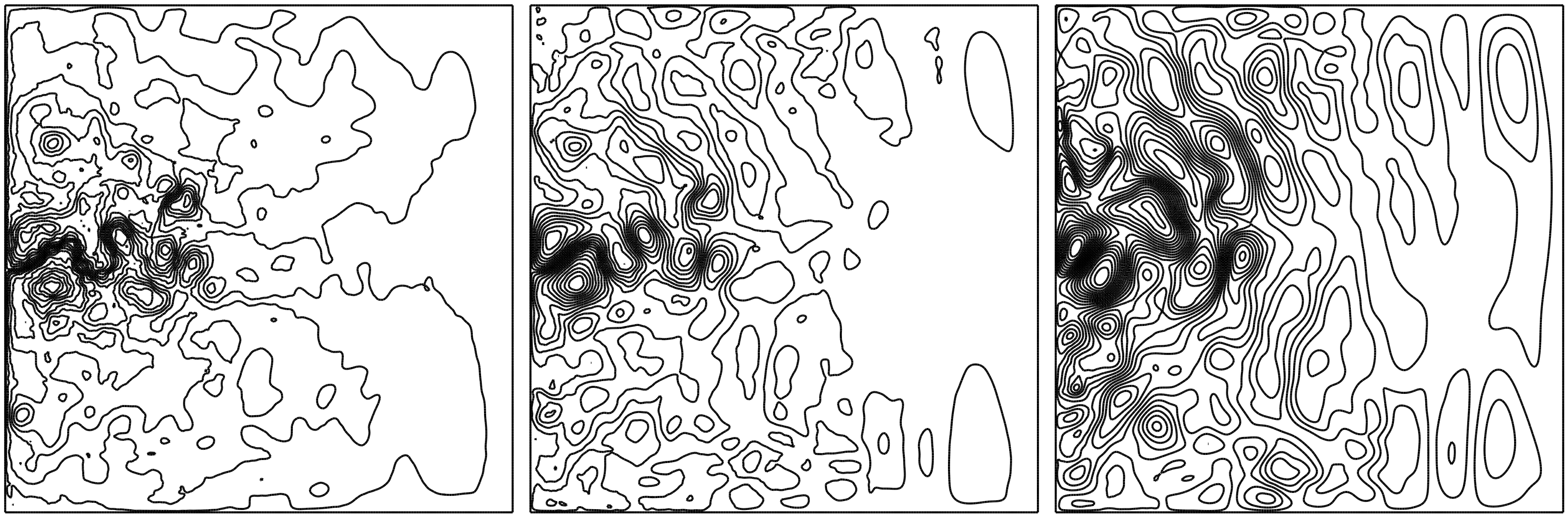


Quasi-geostrophic ocean model

$$k = 1 \ : \qquad \frac{D_1\left(\theta_1(\Psi) + f\right)}{Dt} - \beta\Delta^2\Psi_1 = F_1$$

$$2 \leq k \leq n-1 \ : \qquad \frac{D_k\left(\theta_k(\Psi) + f\right)}{Dt} - \beta\Delta^2\Psi_k = 0$$

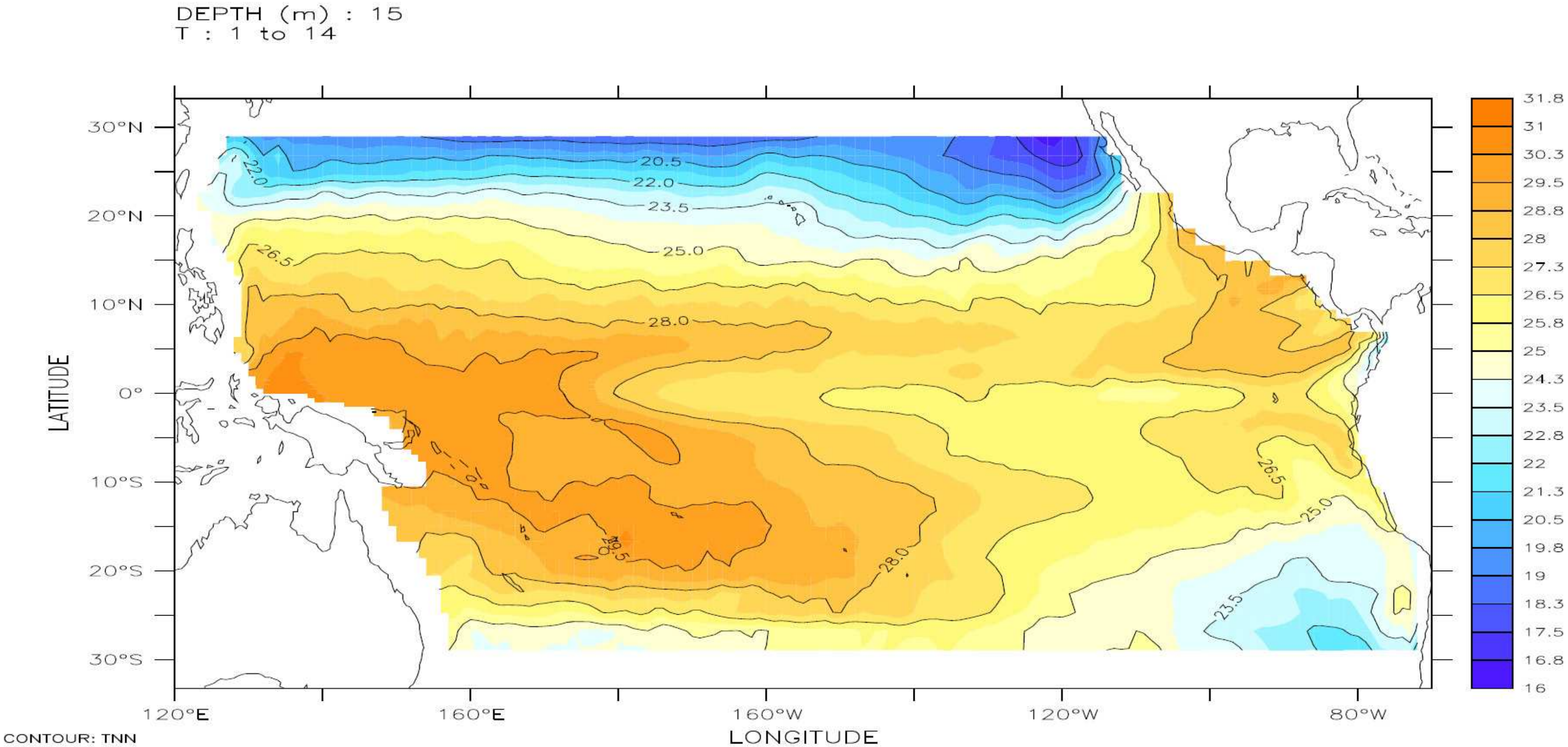
$$k = n \ : \qquad \frac{D_n\left(\theta_n(\Psi) + f\right)}{Dt} + \alpha\Delta\Psi_n - \beta\Delta^2\Psi_n = 0$$



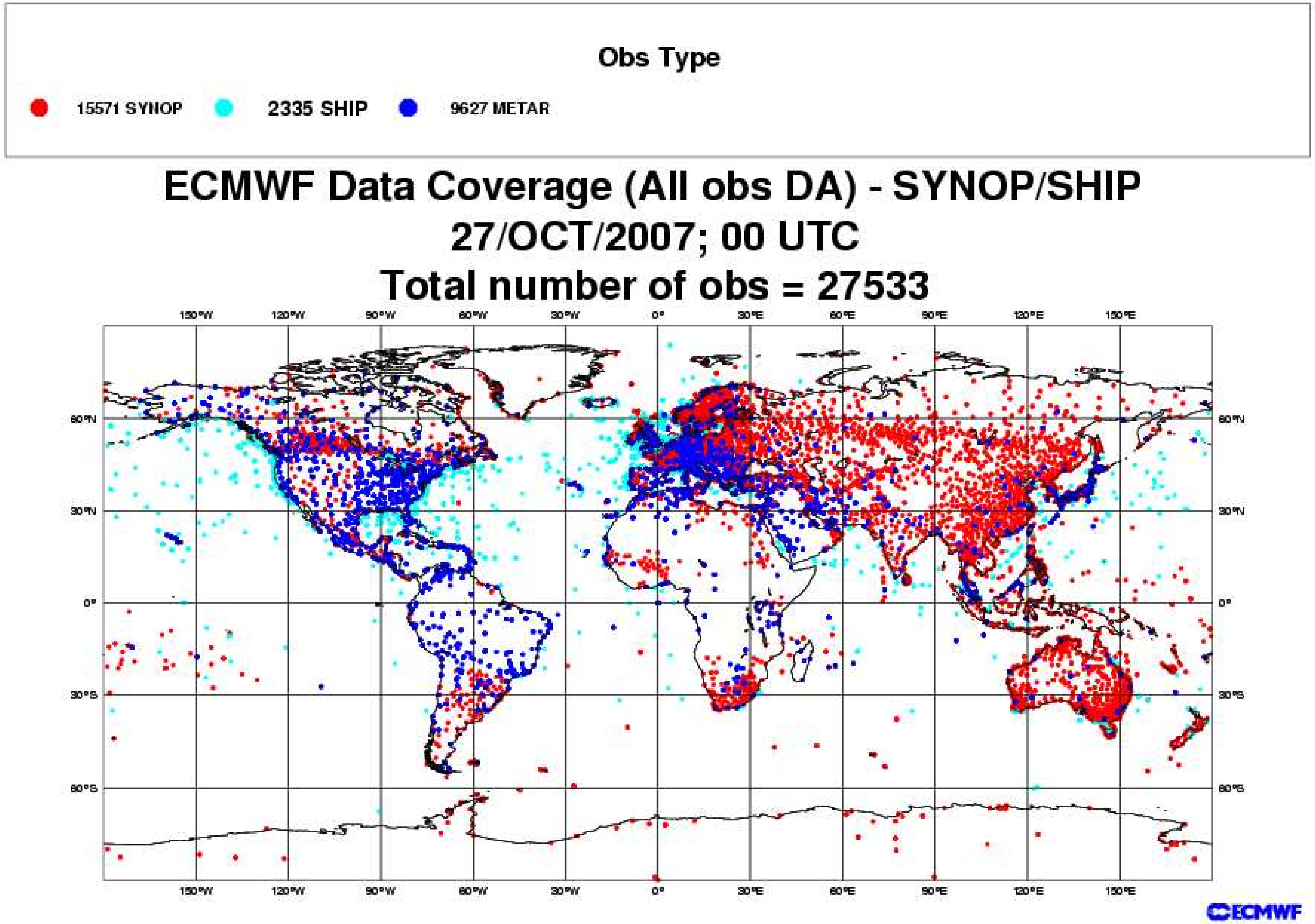
Primitive equations model

$$\frac{\partial U_{xy}}{\partial t} = -[U.\nabla U]_{xy} - \rho_0 \nabla_{xy} p + D^U \qquad \frac{\partial p}{\partial z} = -\rho g \qquad \nabla.U = 0$$

$$\frac{\partial T}{\partial t} = -\nabla.(TU) + D^T \qquad \frac{\partial S}{\partial t} = -\nabla.(SU) + D^S \qquad \rho = \rho(T,S,p)$$

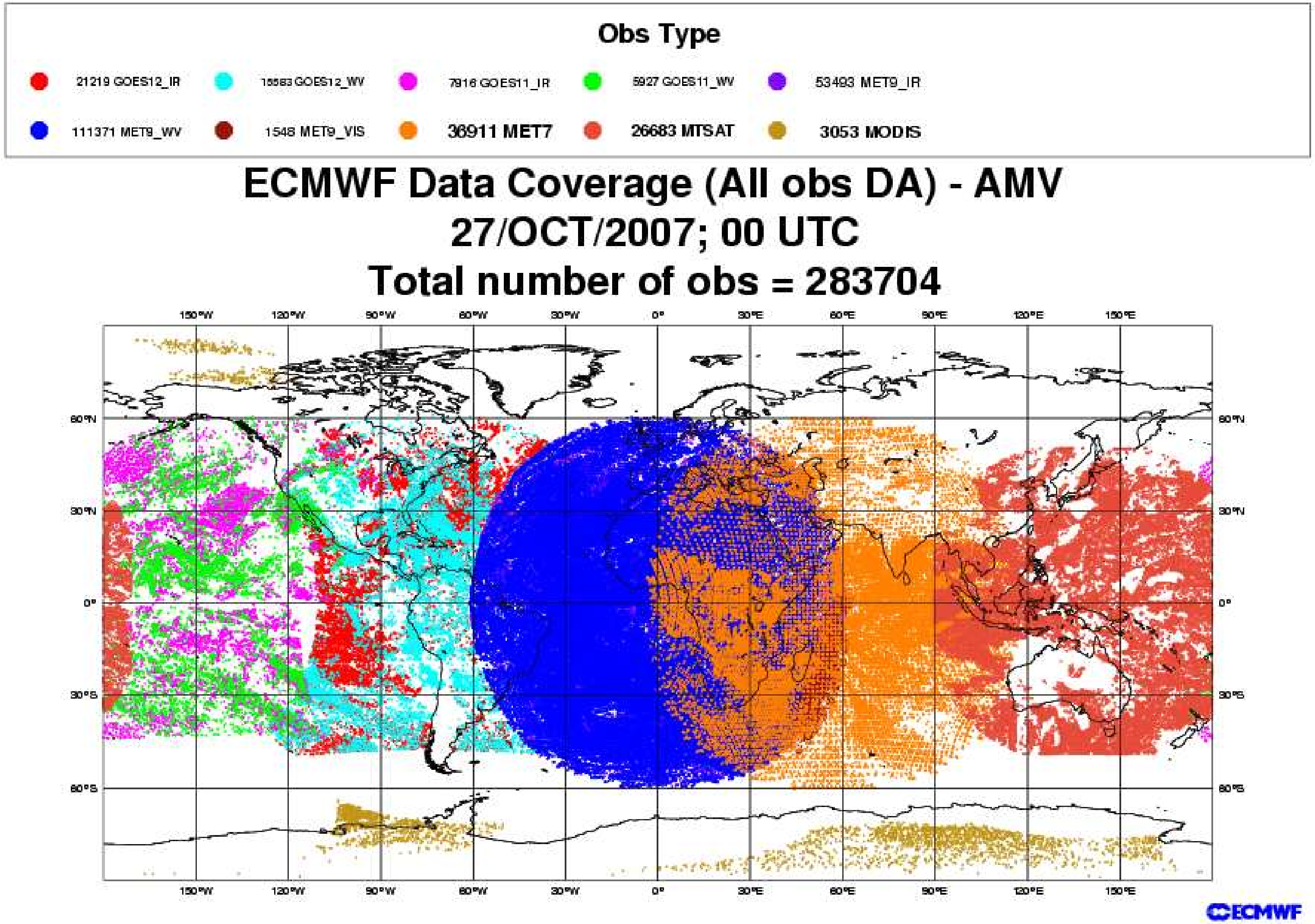


Data



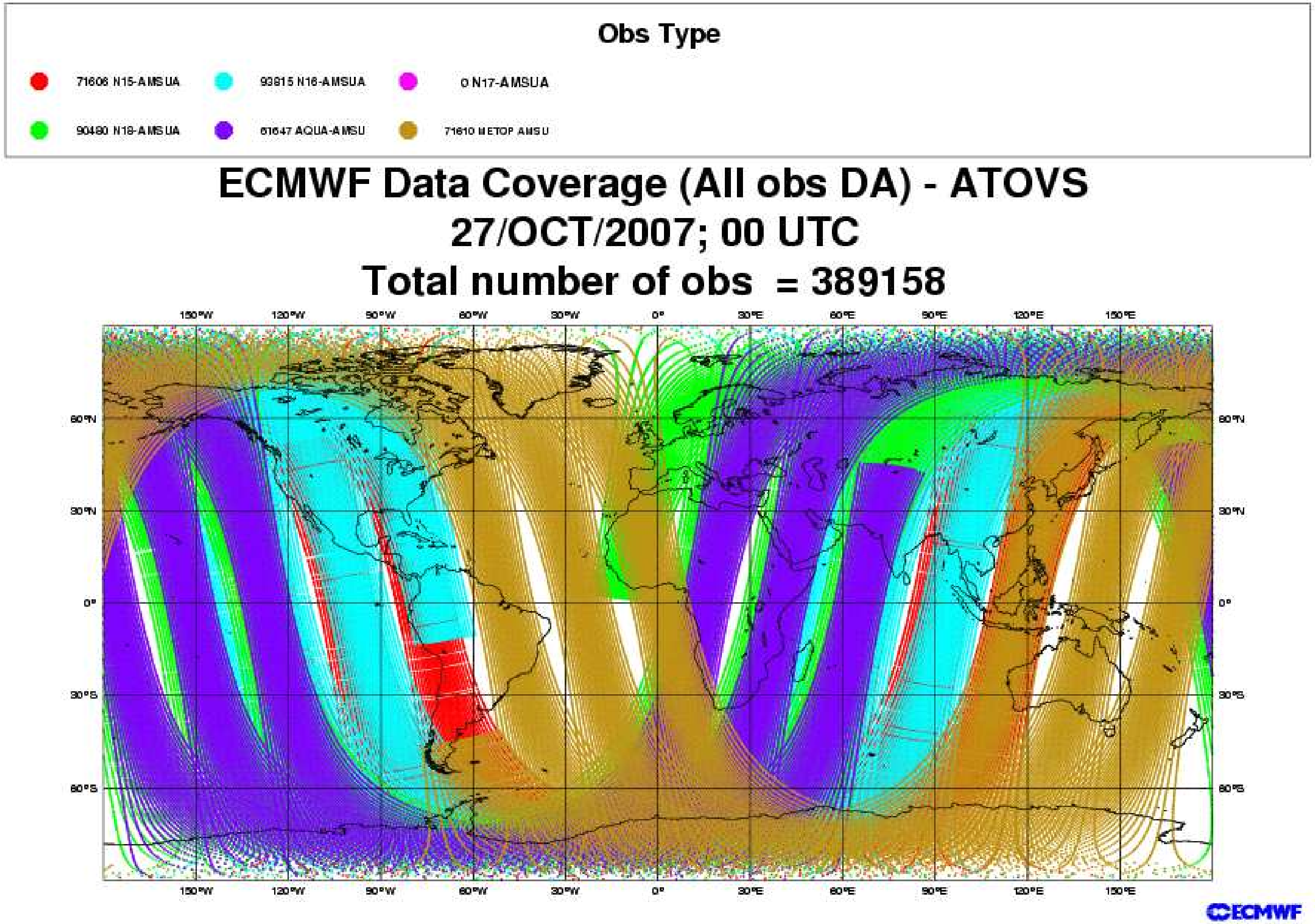
SYNOP/SHIP data : synoptic networks in red, airport data in blue, ship data in green.

Data



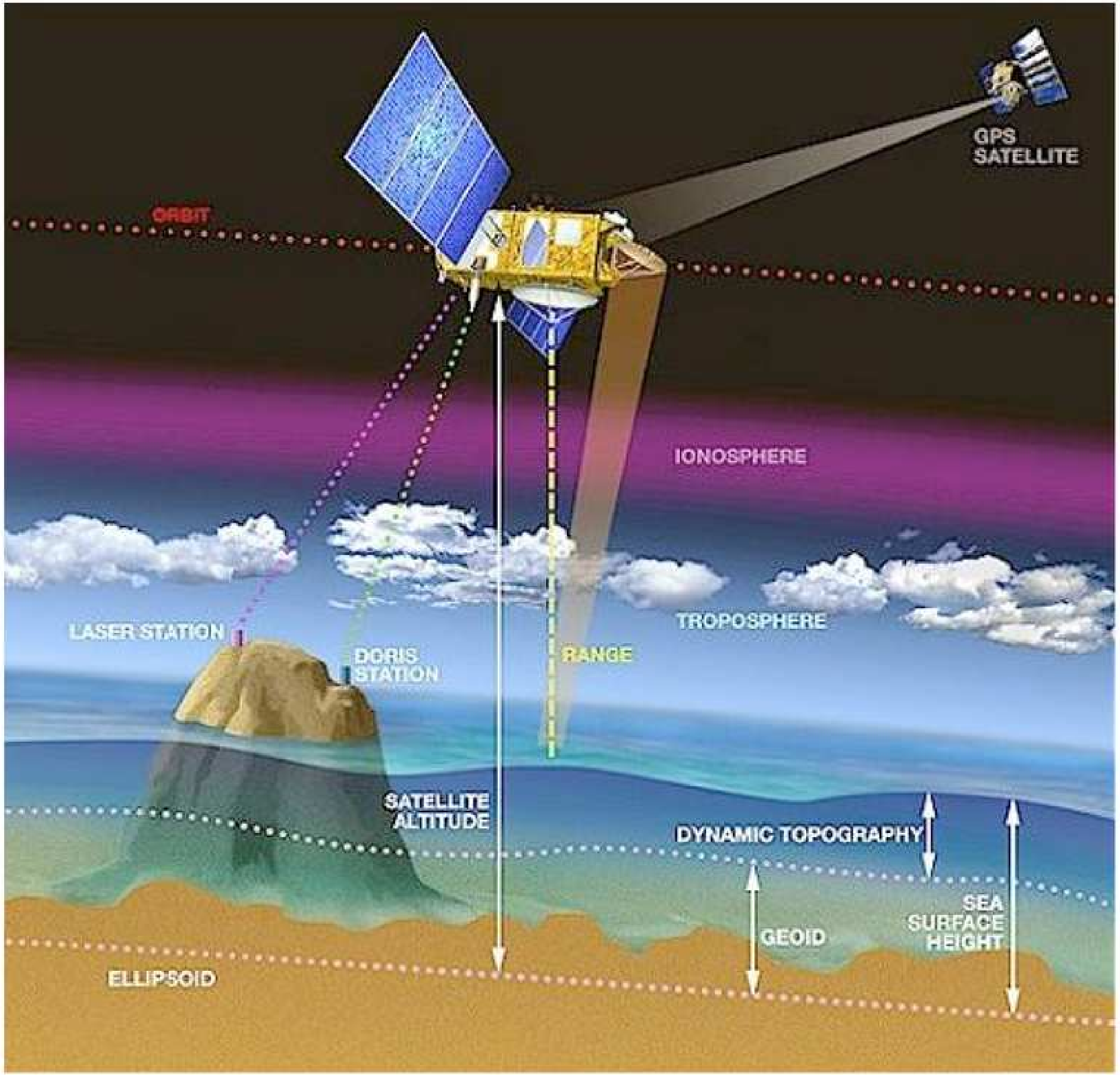
Observations from ten geostationary satellites.

Data



Trajectories of six polar orbiting satellites.

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. Hybrid method : the Back and Forth Nudging.

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

1. Data and models

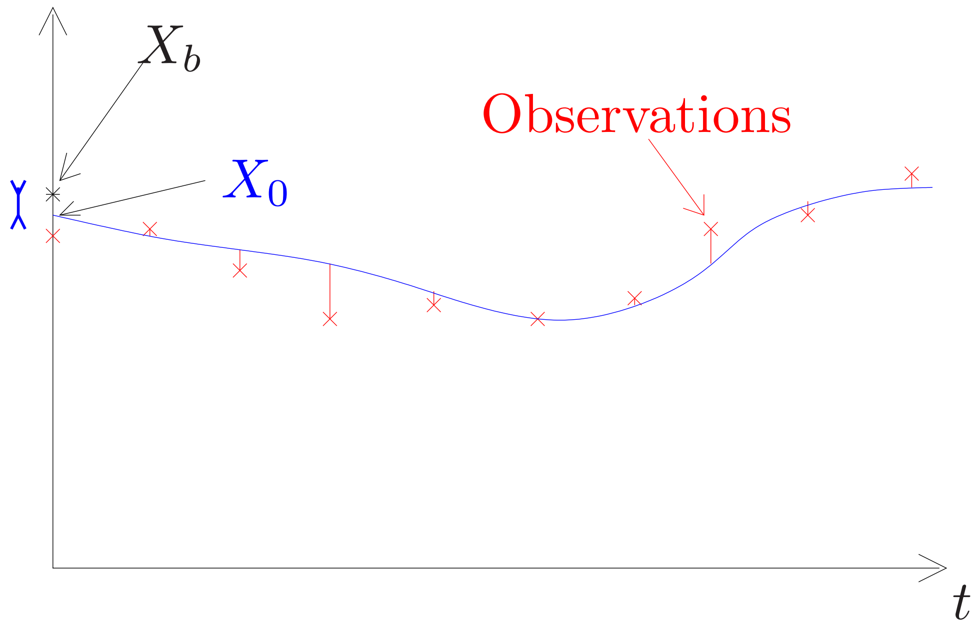
⇒ 2. Variational methods : 4D-VAR

3. Sequential methods : Kalman filter

4. Hybrid scheme : Back and Forth Nudging algorithm

5. Diffusive BFN algorithm

4D-VAR



$$\begin{cases} \dfrac{dX}{dt} = F(X), \\ X(0) = X_0 \end{cases}$$

$\mathcal{Y}(t)$: observations of the system, H : observation operator, X_b : background, B and R : covariance matrices of background and observation errors respectively.

$$\begin{aligned} J(X_0) &= \frac{1}{2}(X_0 - X_b)^T B^{-1} (X_0 - X_b) \\ &+ \frac{1}{2}\int_0^T [\mathcal{Y}(t) - H(X(t))]^T R^{-1} [\mathcal{Y}(t) - H(X(t))] \, dt \end{aligned}$$

Optimality system

Optimization under constraints :

$$\mathcal{L}\left(X_0, X, P\right) = J(X_0) + \int_0^T \left\langle P, \frac{dX}{dt} - F(X) \right\rangle dt$$

Direct model : $\begin{cases} \dfrac{dX}{dt} = F(X) \\ X(0) = X_0 \end{cases}$

Adjoint model : $\begin{cases} -\dfrac{dP}{dt} = \left[\dfrac{\partial F}{\partial X}\right]^T P + H^T R^{-1} \left[\mathcal{Y}(t) - H(X(t))\right] \\ P(T) = 0 \end{cases}$

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)]

4D-VAR algorithm computation

4D-VAR algorithm :
$$\left\{ \begin{array}{ccccc} X_0^k & \xrightarrow{direct} & X^k(t) & \xrightarrow{adjoint} & P^k(t) \\ & & & & \downarrow \\ X_0^{k+1} & \xleftarrow{descent} & \nabla J^k & = & \nabla J(X_0^k) \end{array} \right.$$

Each iteration = one integration of the direct model + one integration of the adjoint model.

Biggest issue : computation of the adjoint code!!!
Big issue : validation of the adjoint code
Smaller issue but still big : small number of iterations for operational DA
Smaller issue, ... actually really quite small : efficient optimization algo

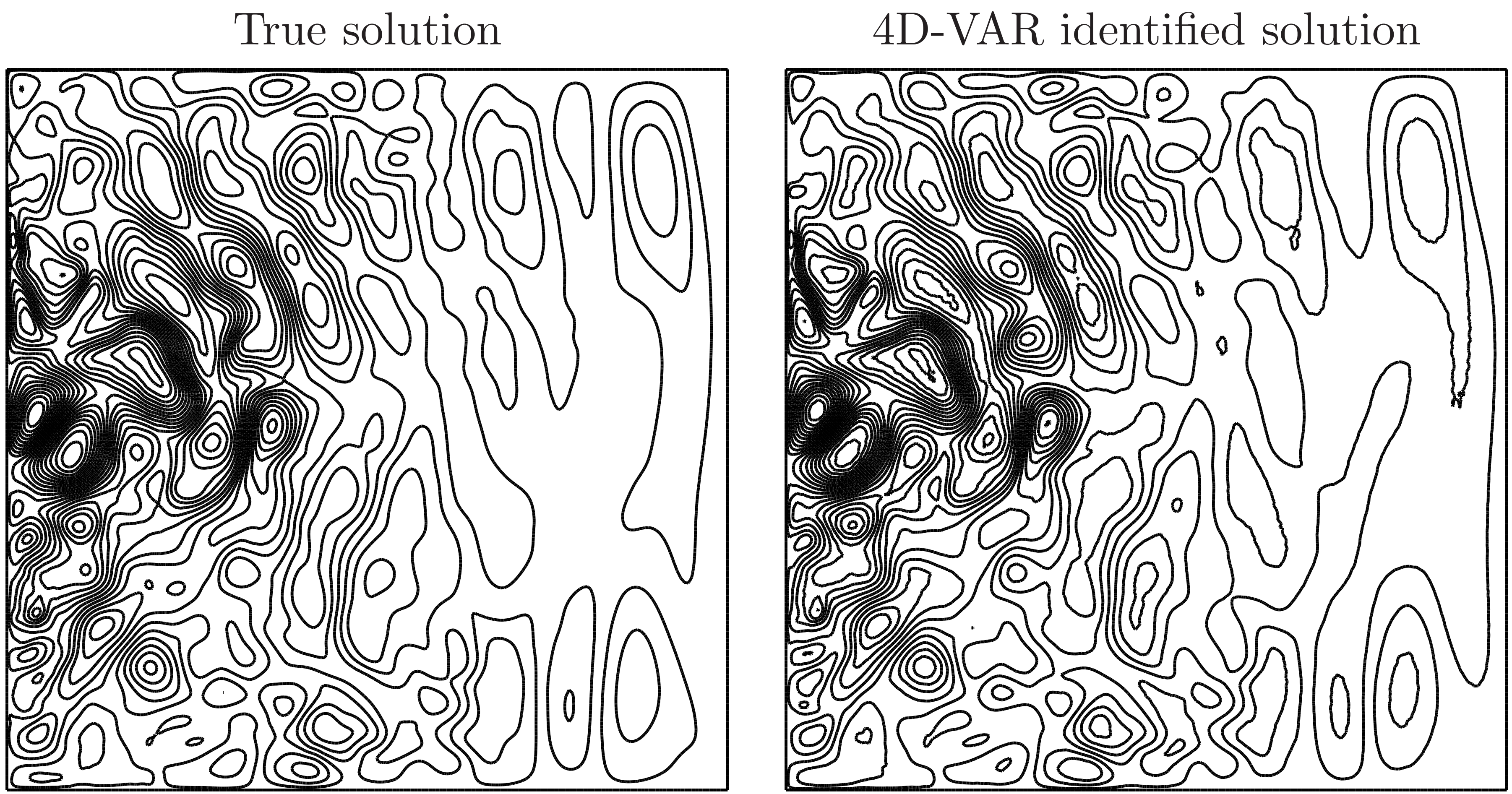
Example

Quasi-geostrophic ocean model :

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. [Luong-Blum-Verron (98), A.-Blum (04)]

1. Data and models

2. Variational methods : 4D-VAR

⇒ 3. Sequential methods : Kalman filter

4. Hybrid scheme : Back and Forth Nudging algorithm

5. Diffusive BFN algorithm

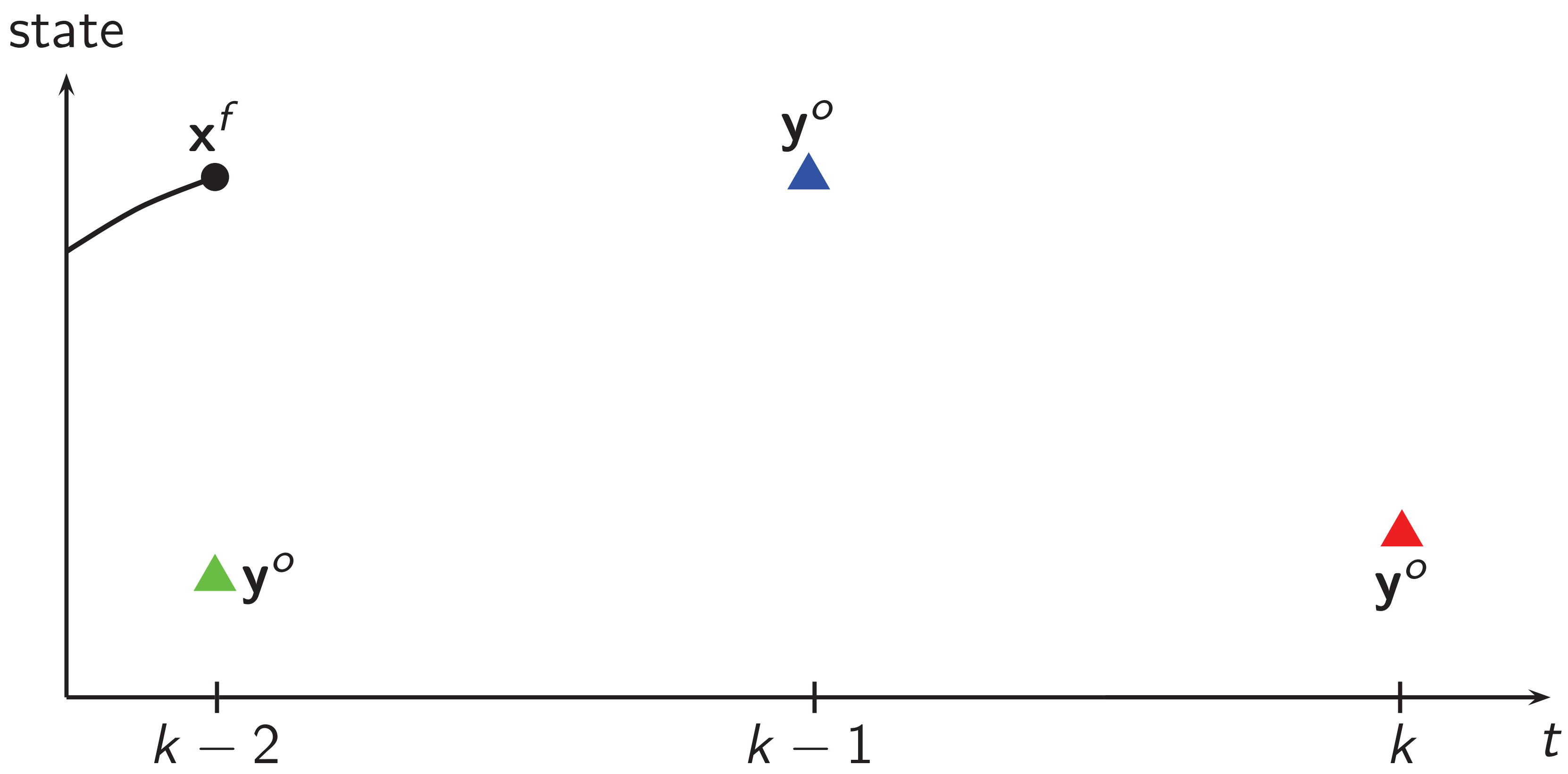
Kalman filter

The Kalman filter is a recursive filter that estimates the state of a dynamic system from a series of incomplete and noisy measurements.

It is an extension to dynamical systems of the optimal interpolation, or BLUE (Best Linear Unbiased Estimator) between the observation and the forecast state.

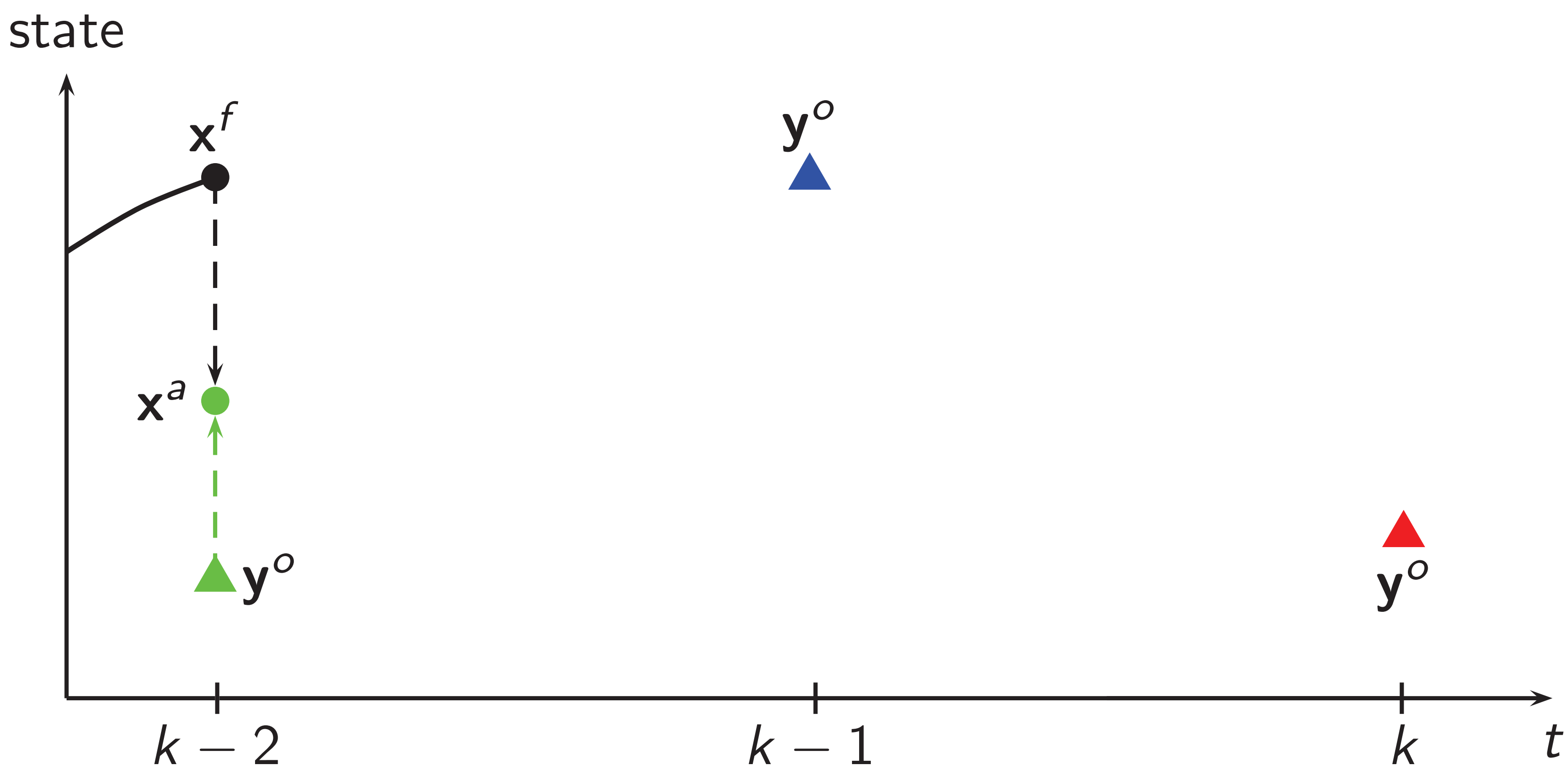
Two steps, repeated for each observation time : prediction and correction.

Kalman filter :

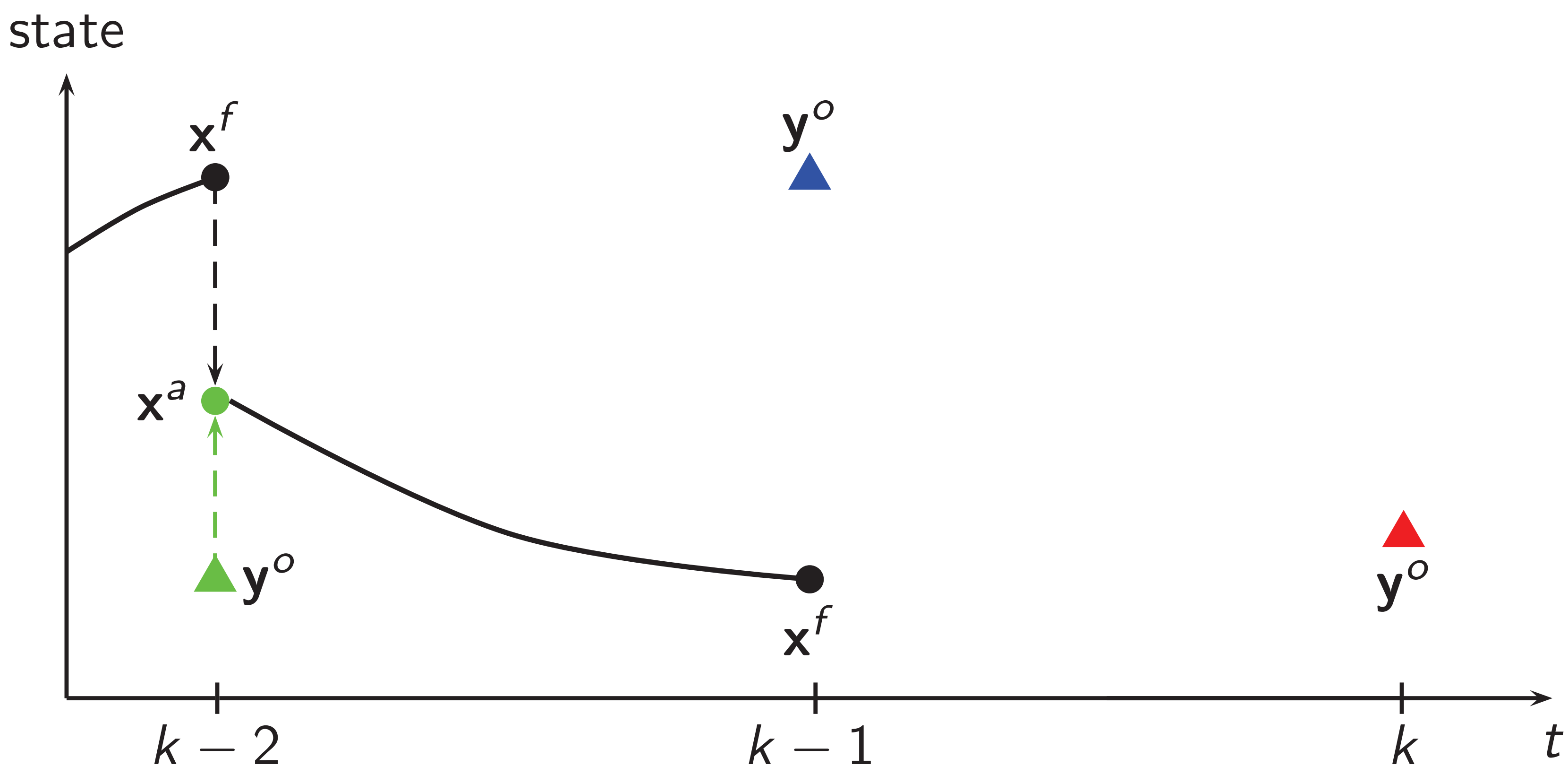


Kalman filter

Kalman filter :

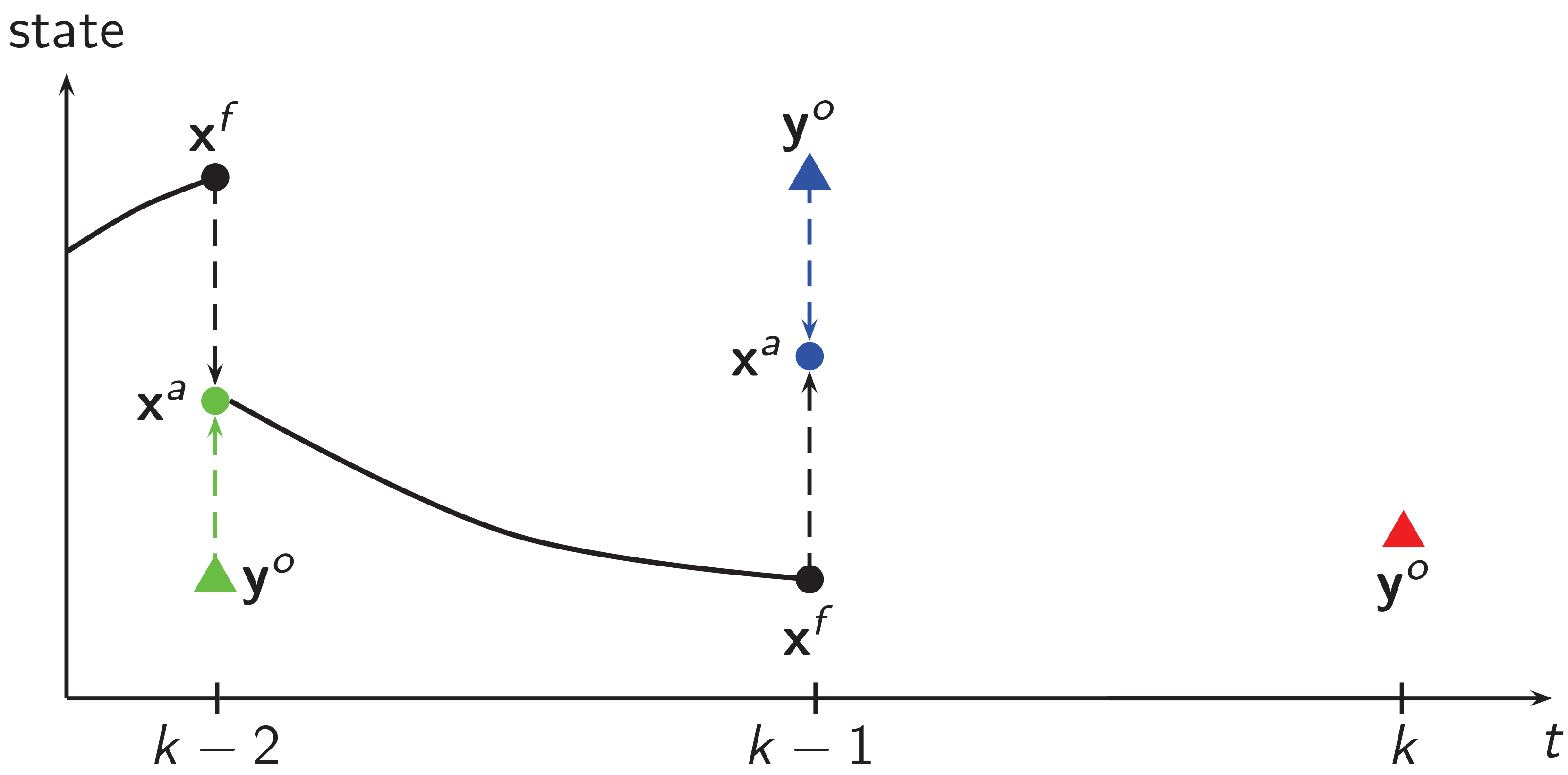


Kalman filter :



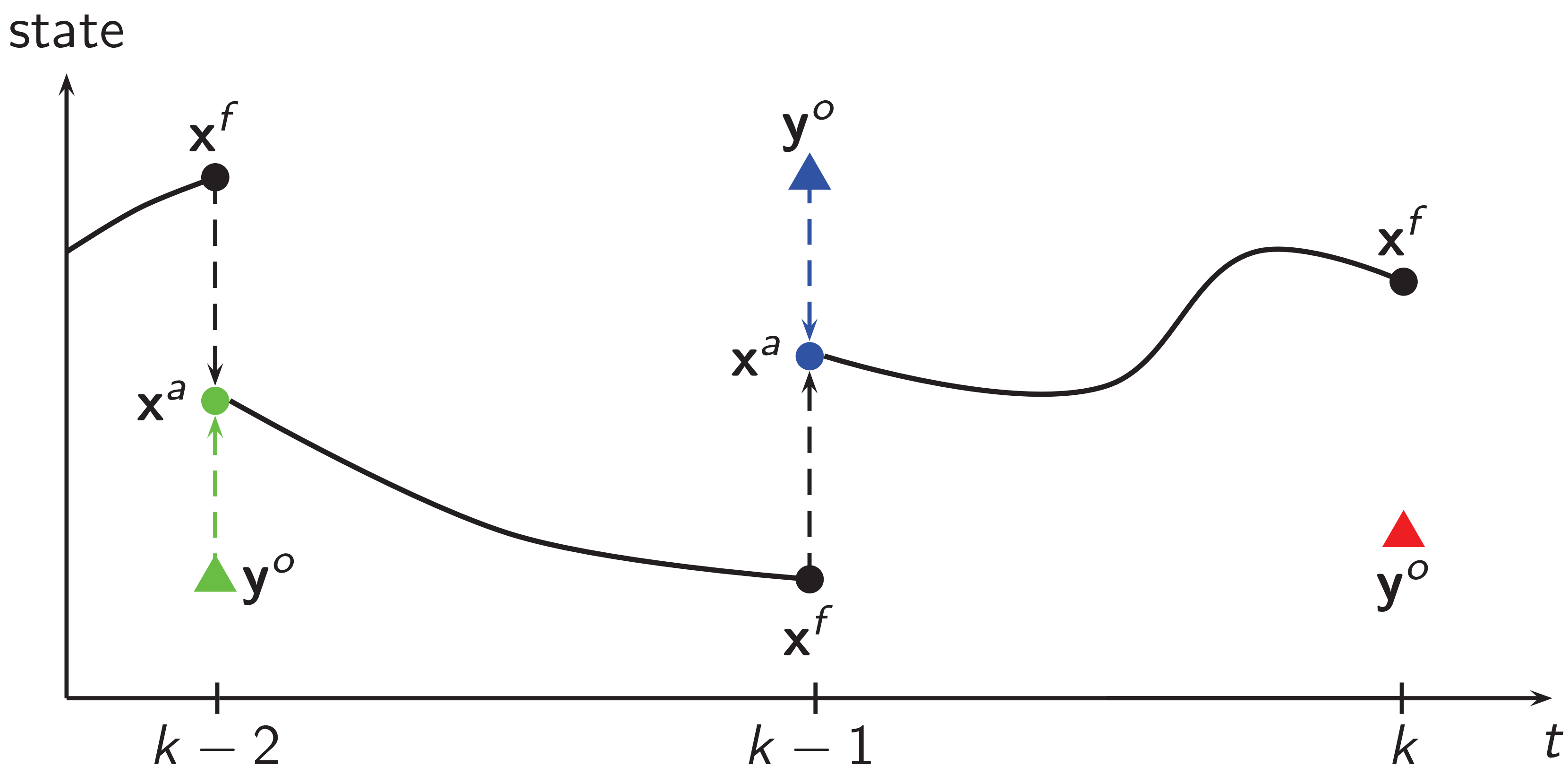
Kalman filter

Kalman filter :

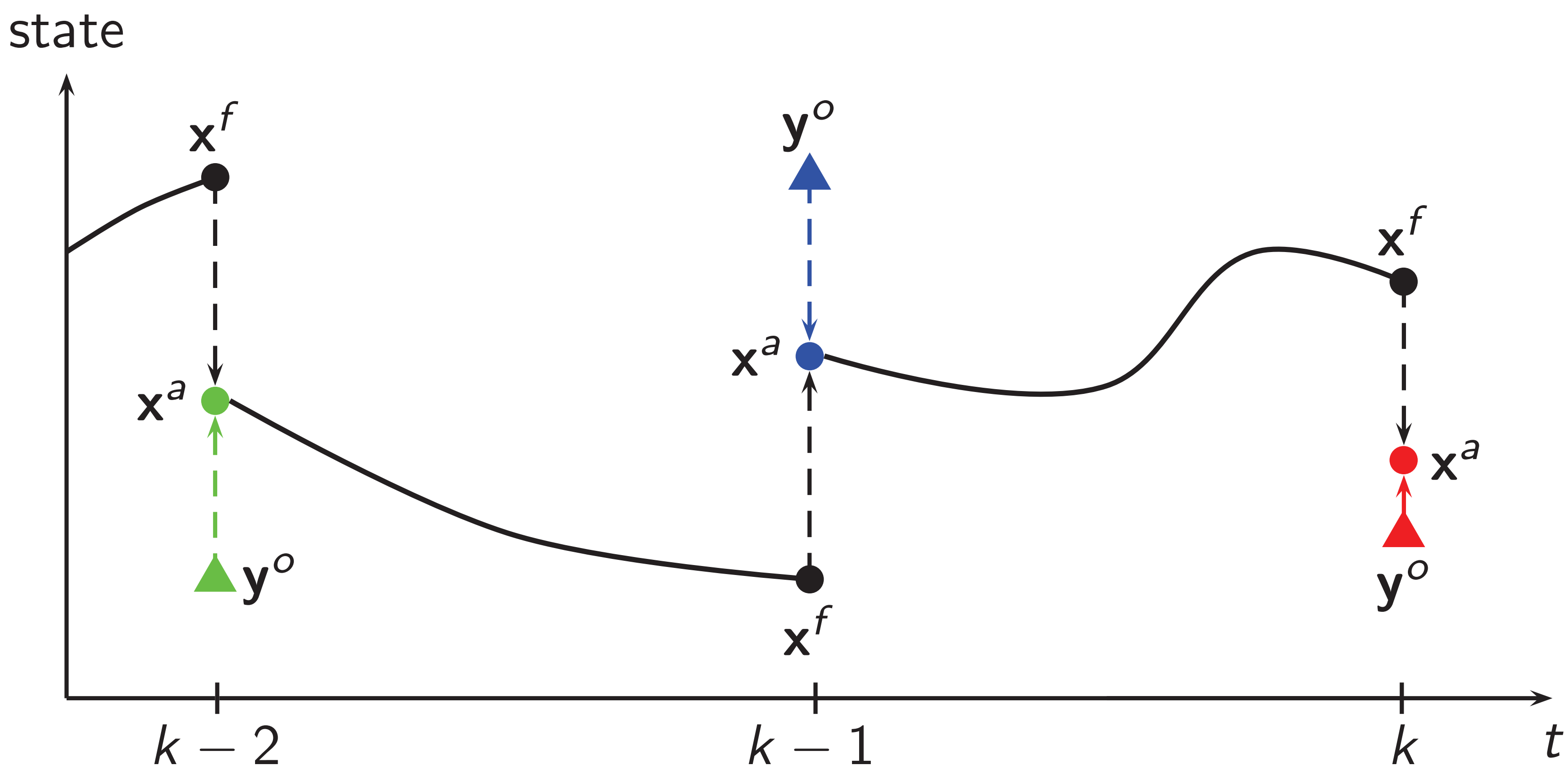


Kalman filter

Kalman filter :



Kalman filter :



Prediction step

We denote by X_n^a the **analyzed state** at time t_n, and M_n the system resolvent from time t_n to t_{n+1}.

Prediction/forecast step : get a background state at time t_{n+1}

$$X_{n+1}^f = M_n X_n^a$$

We denote by P_n^f the covariance matrix of forecast error $X_n^f - X_n^t$ and P_n^a the covariance matrix of analysis error $X_n^a - X_n^t$.

$$\begin{aligned} P_{n+1}^f &= E\left((X_{n+1}^f - X_{n+1}^t)(X_{n+1}^f - X_{n+1}^t)^T \right) \\ &= E\left((M_n X_n^a - M_n X_n^t - \varepsilon_n)(M_n X_n^a - M_n X_n^t - \varepsilon_n)^T \right) \end{aligned}$$

$$P_{n+1}^f = M_n P_n^a M_n^T + Q_n$$

where Q_n is the model error covariance matrix at time t_n.

Correction step

Analysis step : Correction of the background vector X^f_{n+1} with the innovation vector $\mathcal{Y}_{n+1} - H_{n+1}X^f_{n+1}$:

$$X^a_{n+1} = X^f_{n+1} + K_{n+1}(\mathcal{Y}_{n+1} - H_{n+1}X^f_{n+1})$$

The new analysis error $e^a_{n+1} = X^a_{n+1} - X^t_{n+1}$ is :

$$e^a_{n+1} = e^f_{n+1} + K_{n+1}(\epsilon_{n+1} - H_{n+1}e^f_{n+1})$$

where ϵ is the observation error (of covariance matrix R) and e^f is the forecast error (of covariance matrix P^f).

$$P^a_{n+1} = [I - K_{n+1}H_{n+1}]P^f_{n+1}[I - K_{n+1}H_{n+1}]^T + K_{n+1}R_{n+1}K^T_{n+1}$$

One chooses K_{n+1} such that the variance of analysis error is minimum :

$$K_{n+1} = P^f_{n+1}H^T_{n+1}[H_{n+1}P^f_{n+1}H^T_{n+1} + R_{n+1}]^{-1}$$

Then,
$$P^a_{n+1} = [I - K_{n+1}H_{n+1}]P^f_{n+1}$$

KF algorithm

- Initialization :

$$X_0^f \text{ and } P_0^f \text{ given}$$

- Analysis :

$$\begin{aligned}
K_n &= P_n^f H_n^T [H_n P_n^f H_n^T + R_n]^{-1} \\
X_n^a &= X_n^f + K_n(\mathcal{Y}_n - H_n X_n^f) \\
P_n^a &= [I - K_n H_n] P_n^f
\end{aligned} \tag{2}$$

- Forecast :

$$\begin{aligned}
X_{n+1}^f &= M_{n;n+1} X_n^a \\
P_{n+1}^f &= M_{n;n+1} P_n^a M_{n;n+1}^T + Q_n
\end{aligned}$$

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.

(remind that $n \simeq 10^9 \ldots$)

Thus KF in usual form can only be used for rather low dimensional dynamical models (use of model reduction : POD, SVD, ...).

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.

$\Rightarrow$ extended Kalman filter, ensemble Kalman filter, reduced KF, local ensemble KF, evolutive reduced extended local ensemble KF, ...

1. Data and models

2. Variational methods : 4D-VAR

3. Sequential methods : Kalman filter

$\Rightarrow$ 4. Hybrid scheme : Back and Forth Nudging algorithm

5. Diffusive BFN algorithm

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 $X(0) = x_0$.

$\mathcal{Y}(t)$: observations of the system
H : observation operator.

$$\begin{cases} \dfrac{dX}{dt} = F(X) + K(\mathcal{Y} - H(X)), \quad 0 < t < T, \\ X(0) = X_0, \end{cases}$$

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) : De Mey et al. (1987), Verron-Holland (1989)

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

– Optimal determination of the nudging coefficients :

Zou-Navon-Le Dimet (1992), Stauffer-Bao (1993),
Vidard-Le Dimet-Piacentini (2003)
Lakshmivarahan-Lewis (2011)

Forward nudging : linear case

Luenberger observer, or asymptotic observer

(Luenberger, 1966)

$$\begin{cases} \dfrac{dX_{true}}{dt} = FX_{true}, \quad \mathcal{Y} = HX_{true}, \\ \dfrac{dX}{dt} = FX + K(\mathcal{Y} - HX). \end{cases}$$

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

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 $X \to X_{true}$ when $t \to +\infty$.

Backward nudging

How to recover the initial state from the final solution ?

Backward model :

$$\begin{cases} \dfrac{d\tilde{X}}{dt} = F(\tilde{X}), \quad T > t > 0, \\ \tilde{X}(T) = \tilde{X}_T. \end{cases}$$

If we apply nudging to this backward model :

$$\begin{cases} \dfrac{d\tilde{X}}{dt} = F(\tilde{X}) - K(\mathcal{Y} - H\tilde{X}), \quad T > t > 0, \\ \tilde{X}(T) = \tilde{X}_T. \end{cases}$$

BFN : Back and Forth Nudging algorithm

Iterative algorithm (forward and backward resolutions) :

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

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

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

[A.-Blum, C. R. Acad. Sci. Math. 2005]

If X_k and $\tilde{X}_k$ converge towards the same limit 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 :

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

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

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

by chosing

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

where R is the covariance matrix of the errors of observation, and k is a scalar.

[A.-Blum, Nonlin. Proc. Geophys. 2008]

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).

Simpler solution : one can define $K' = k'H^T R^{-1}$, where k' is e.g. the smallest value making the backward numerical integration stable.

Example of convergence results

Viscous linear transport equation :

$$\begin{cases} \partial_t u - \nu \partial_{xx} u + a(x)\partial_x u = -K(u - u_{obs}), & u(x, t=0) = u_0(x) \\ \partial_t \tilde{u} - \nu \partial_{xx} \tilde{u} + a(x)\partial_x \tilde{u} = K'(\tilde{u} - u_{obs}), & \tilde{u}(x, t=T) = u_T(x) \end{cases}$$

We set $w(t) = u(t) - u_{obs}(t)$ and $\tilde{w}(t) = \tilde{u}(t) - u_{obs}(t)$ the errors.

• If K and K' are constant, then $\forall t \in [0, T]$: $\widetilde{w}(t) = e^{(-K-K')(T-t)} w(t)$

(still true if the observation period does not cover $[0, T]$)

• If the domain is not fully observed, then the problem is ill-posed.

Error after k iterations : $w_k(0) = e^{-[(K+K')kT]} w_0(0)$

$\rightsquigarrow$ exponential decrease of the error, thanks to :

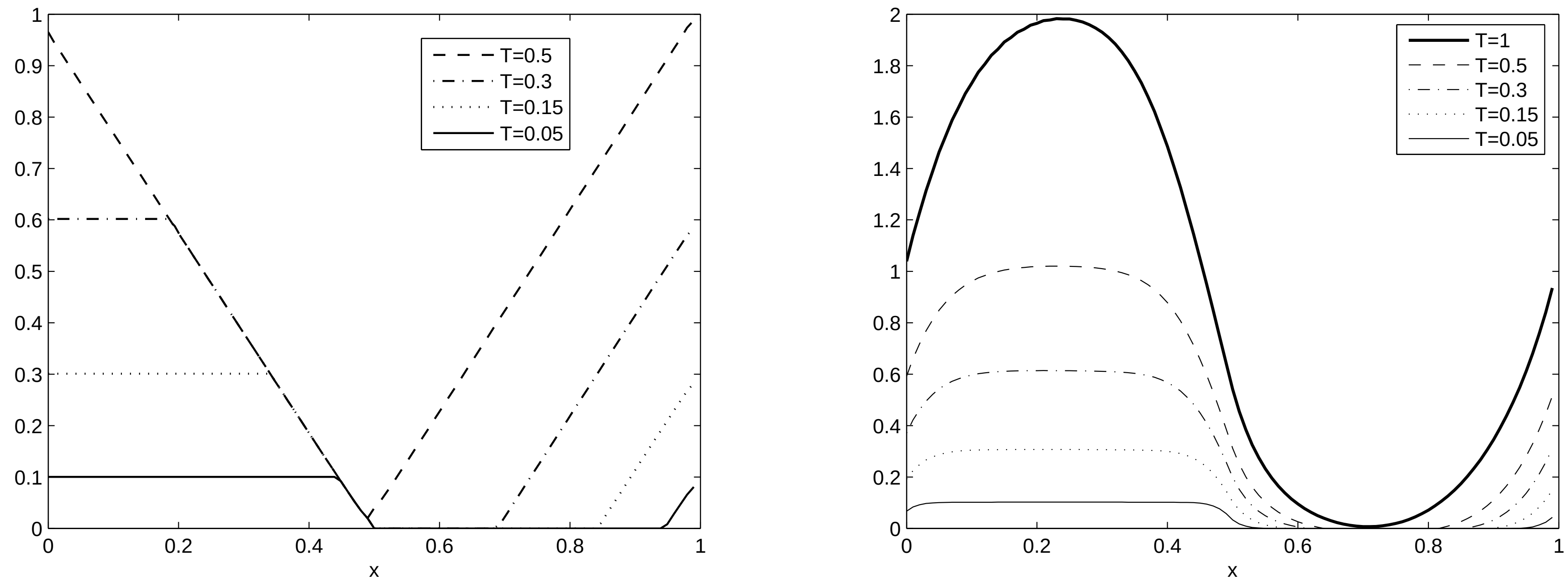
- $K + K'$: infinite feedback to the observations (not physical)
- T : asymptotic observer (Luenberger)
- k : infinite number of iterations (BFN)

[A.-Nodet, COCV 2012]

Observability condition

Let $\chi(x)$ be the time during which the characteristic curve with foot x lies in the support of K. Then the system is observable if and only if $\min_x \chi(x) > 0$.

Partial observations in space : half of the domain is observed.



Decrease rate of the error after one iteration of BFN as a function of the space variable x, for various final times T.

Linear case (left) : theoretical observability condition $= T > 0.5$

Nonlinear case (right) : numerical observability condition $= T > 1$

Shallow water model

$$
\begin{aligned}
\partial_t u - (f+\zeta)v + \partial_x B &= \frac{\tau_x}{\rho_0 h} - ru + \nu\Delta u \\
\partial_t v + (f+\zeta)u + \partial_y B &= \frac{\tau_y}{\rho_0 h} - rv + \nu\Delta v \\
\partial_t h + \partial_x(hu) + \partial_y(hv) &= 0
\end{aligned}
$$

- $\zeta = \partial_x v - \partial_y u$ is the relative vorticity ;
- $B = g^* h + \dfrac{1}{2}(u^2 + v^2)$ is the Bernoulli potential ;
- $g^* = 0.02\ m.s^{-2}$ is the reduced gravity ;
- $f = f_0 + \beta y$ is the Coriolis parameter (in the β-plane approximation), with $f_0 = 7.10^{-5}\ s^{-1}$ and $\beta = 2.10^{-11}\ m^{-1}.s^{-1}$;
- $\tau = (\tau_x, \tau_y)$ is the forcing term of the model (e.g. the wind stress), with a maximum amplitude of $\tau_0 = 0.05\ s^{-2}$;
- $\rho_0 = 10^3\ kg.m^{-3}$ is the water density ;
- $r = 9.10^{-8}\ s^{-1}$ is the friction coefficient.
- $\nu = 5\ m^2.s^{-1}$ is the viscosity (or dissipation) coefficient.

Shallow water model

2D shallow water model, state = height h and horizontal velocity (u, v)

Numerical parameters : (run example)

Domain : $L = 2000$ km $\times$ 2000 km ; Rigid boundary and no-slip BC ; Time step = 1800 s ; Assimilation period : 15 days ; Forecast period : 15 + 45 days

Observations : of h only ($\sim$ satellite obs), every 5 gridpoints in each space direction, every 24 hours.

Background : true state one month before the beginning of the assimilation period + white gaussian noise ($\sim$ 10%)

Comparison BFN - 4DVAR : sea height h ; velocity :u and v.

Diffusion problem

Backward model and diffusion :

The main issue of the BFN is : how to handle diffusion processes in the backward equation ?

Let us consider only diffusion : heat equation (in 1D)

$$\partial_t u = \partial_{xx} u$$

The backward nudging model will be :

$$\partial_t \tilde{u} = \partial_{xx} \tilde{u} + K(\tilde{u} - u_{obs})$$

from time T to 0. By using a change of variable $t' = T - t$, we can rewrite the backward model as a forward one :

$$\partial_{t'} \tilde{u} = -\partial_{xx} \tilde{u} - K(\tilde{u} - u_{obs}),$$

and we can see that even if the nudging term stabilizes the model, the backward diffusion is a real issue (unbounded eigenvalues, except for discrete Laplacian).

1. Data and models

2. Variational methods : 4D-VAR

3. Sequential methods : Kalman filter

4. Back and forth nudging algorithm

$\Rightarrow$ 5. Diffusive BFN algorithm

Diffusive BFN

Diffusive free equations in the geophysical context :

In meteorology or oceanography, theoretical equations are usually diffusive free (e.g. Euler's equation for meteorological processes).

In a numerical framework, a diffusive term is added to the equations (or a diffusive scheme is used), in order to both stabilize the numerical integration of the equations, and take into consideration some subscale phenomena.

Example : weather forecast is done with Euler's equation (at least in Météo France...), which is diffusive free. Also, in quasi-geostrophic ocean models, people usually consider ∇^4 or ∇^6 for dissipation at the bottom, or for vertical mixing.

Diffusive BFN

Standard BFN algorithm :

Original model :

$$\partial_t X = F(X), \quad 0 < t < T.$$

Corresponding BFN algorithm :

$$\begin{cases} \partial_t X_k = F(X_k) + K(\mathcal{Y} - H(X_k)), \\ X_k(0) = \tilde{X}_{k-1}(0), \quad 0 < t < T, \end{cases}$$

$$\begin{cases} \partial_t \tilde{X}_k = F(\tilde{X}_k) - K'(\mathcal{Y} - H(\tilde{X}_k)), \\ \tilde{X}_k(T) = X_k(T), \quad T > t > 0, \end{cases}$$

with the notation $\tilde{X}_0(0) = x_0$.

Diffusive BFN

Addition of a diffusion term :

$$\partial_t X = F(X) + \nu \Delta X, \quad 0 < t < T,$$

where F has no diffusive terms, ν is the diffusion coefficient, and we assume that the diffusion is a standard second-order Laplacian (could be a higher order operator).

We introduce the D-BFN algorithm in this framework, for $k \geq 1$:

$$\begin{cases} \partial_t X_k = F(X_k) + \nu \Delta X_k + K(\mathcal{Y} - H(X_k)), \\ X_k(0) = \tilde{X}_{k-1}(0), \quad 0 < t < T, \end{cases}$$

$$\begin{cases} \partial_t \tilde{X}_k = F(\tilde{X}_k) - \nu \Delta \tilde{X}_k - K'(\mathcal{Y} - H(\tilde{X}_k)), \\ \tilde{X}_k(T) = X_k(T), \quad T > t > 0. \end{cases}$$

Diffusive BFN

It is straightforward to see that the backward equation can be rewritten, using $t' = T - t$:

$$\partial_{t'} \tilde{X}_k = -F(\tilde{X}_k) + \nu \Delta \tilde{X}_k + K'(\mathcal{Y} - H(\tilde{X}_k)), \quad \tilde{X}_k(t' = 0) = X_k(T),$$

where $\tilde{X}$ is evaluated at time t'. As it is now forward in time, this equation can be compared with the forward nudging equation :

$$\partial_t X_k = F(X_k) + \nu \Delta X_k + K(\mathcal{Y} - H(X_k)), \quad X_k(0) = \tilde{X}_{k-1}(t' = T).$$

Then the backward equation can easily be solved, with an initial condition, and the same diffusion operator as in the forward equation. Only the physical model has an opposite sign.

The diffusion term both takes into account the subscale processes and stabilizes the numerical backward integrations, and the feedback term still controls the trajectory with the observations.

Linear transport equation

$$\partial_t u + a(x)\,\partial_x u = 0, \quad t \in [0,T],\ x \in \Omega, \quad u(t=0) = u_0 \in L^2(\Omega)$$

with periodic boundary conditions, and we assume that $a \in W^{1,\infty}(\Omega)$.

Numerically, for both stability and subscale modelling, the following equation would be solved :

$$\partial_t u + a(x)\,\partial_x u = \nu \partial_{xx} u, \quad t \in [0,T], x \in \Omega, \quad u(t=0) = u_0 \in L^2(\Omega),$$

where $\nu \geq 0$ is assumed to be constant.

Linear transport equation

Let us assume that the observations satisfy the physical model (without diffusion) :

$$\partial_t u_{obs} + a(x)\,\partial_x u_{obs} = 0, \quad t \in [0,T], x \in \Omega, \quad u_{obs}(t=0) = u_{obs}^0 \in L^2(\Omega).$$

We assume in this idealized situation that the system is fully observed (and H is then the identity operator).

Then the D-BFN algorithm applied to this problem gives, for $k \geq 1$:

$$\begin{cases} \partial_t u_k + a(x)\,\partial_x u_k = \nu \partial_{xx} u_k + K(u_{obs,k} - u_k), \\ t \in [2(k-1)T, 2(k-1)T + T], x \in \Omega \\ u_k(2(k-1)T, x) = \tilde{u}_{k-1}(2(k-1)T, x) \end{cases}$$

$$\begin{cases} \partial_t \tilde{u}_k - a(x)\,\partial_x \tilde{u}_k = \nu \partial_{xx} \tilde{u}_k + K(\tilde{u}_{obs,k} - \tilde{u}_k), \\ t \in [2kT - T, 2kT], x \in \Omega \\ \tilde{u}_k(2kT - T, x) = u_k(2kT - T, x). \end{cases}$$

Smoothing equation

At the limit $k \to \infty$, v_k and $\tilde{v}_k$ tend to $v_\infty(x)$ solution of

$$\nu \partial_{xx} v_\infty + K(u^0_{obs}(x) - v_\infty) = 0,$$

or equivalently

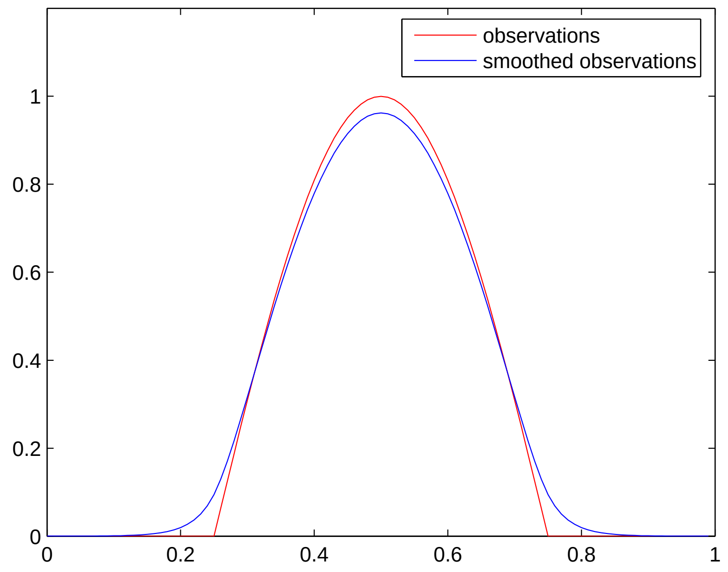
$$-\frac{\nu}{K} \partial_{xx} v_\infty + v_\infty = u^0_{obs}.$$

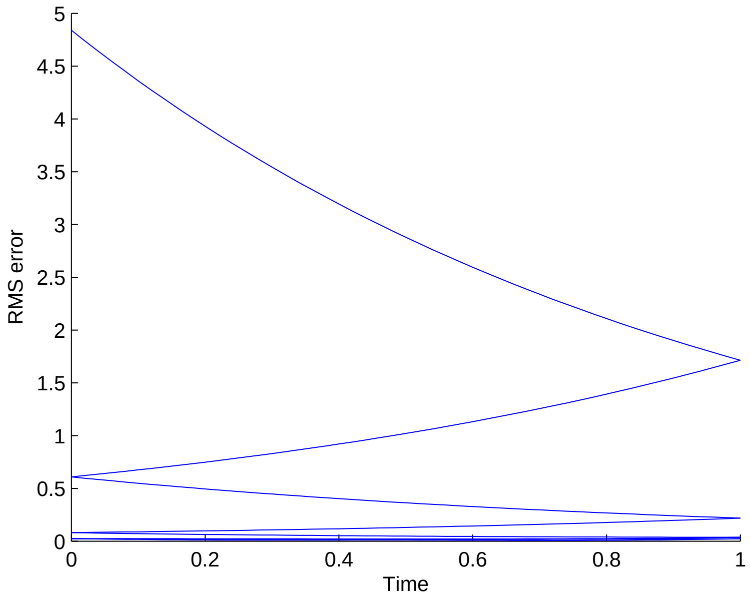
This equations is well known in signal or image processing, as being the standard linear diffusion restoration equation. In some sense, v_∞ is the result of a smoothing process on the observations u_{obs}, where the degree of smoothness is given by the ratio $\frac{\nu}{K}$.

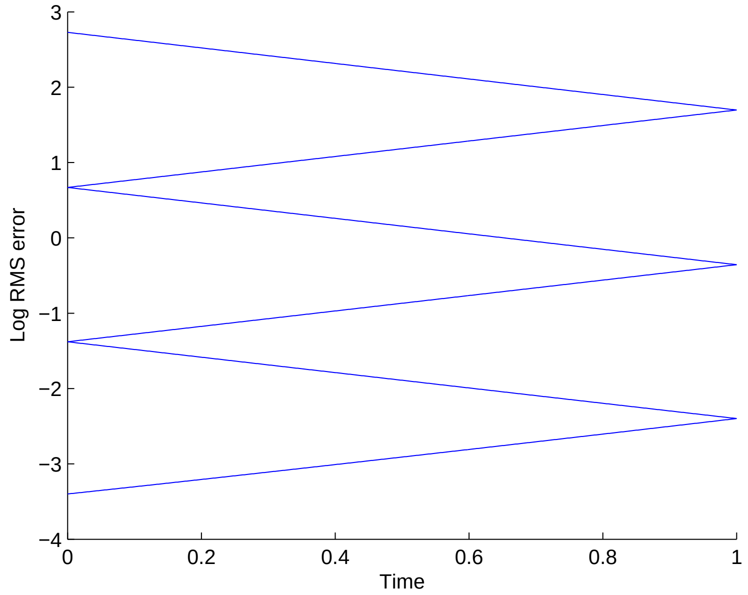
Convergence result for constant advection equation.

[A.-Blum-Nodet, CRAS 2011]

Numerical experiments



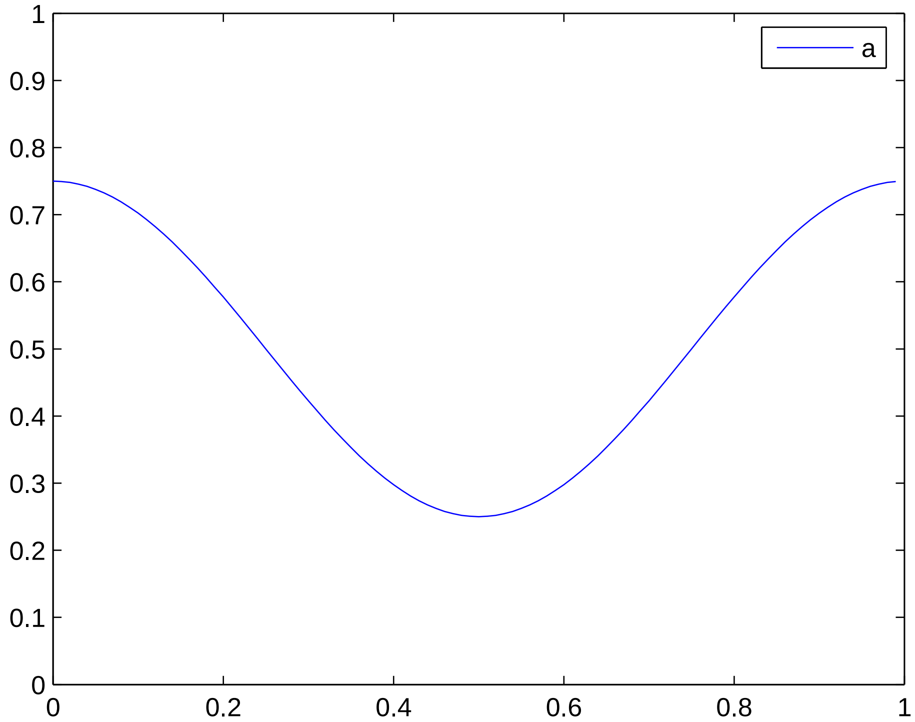




Initial condition of the observation and corresponding smoothed solution ; RMS difference between the BFN iterates and the smoothed observations ; same in semi-log scale. Movie

Numerical experiments

Linear transport equation with non-constant transport :



Movie

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

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

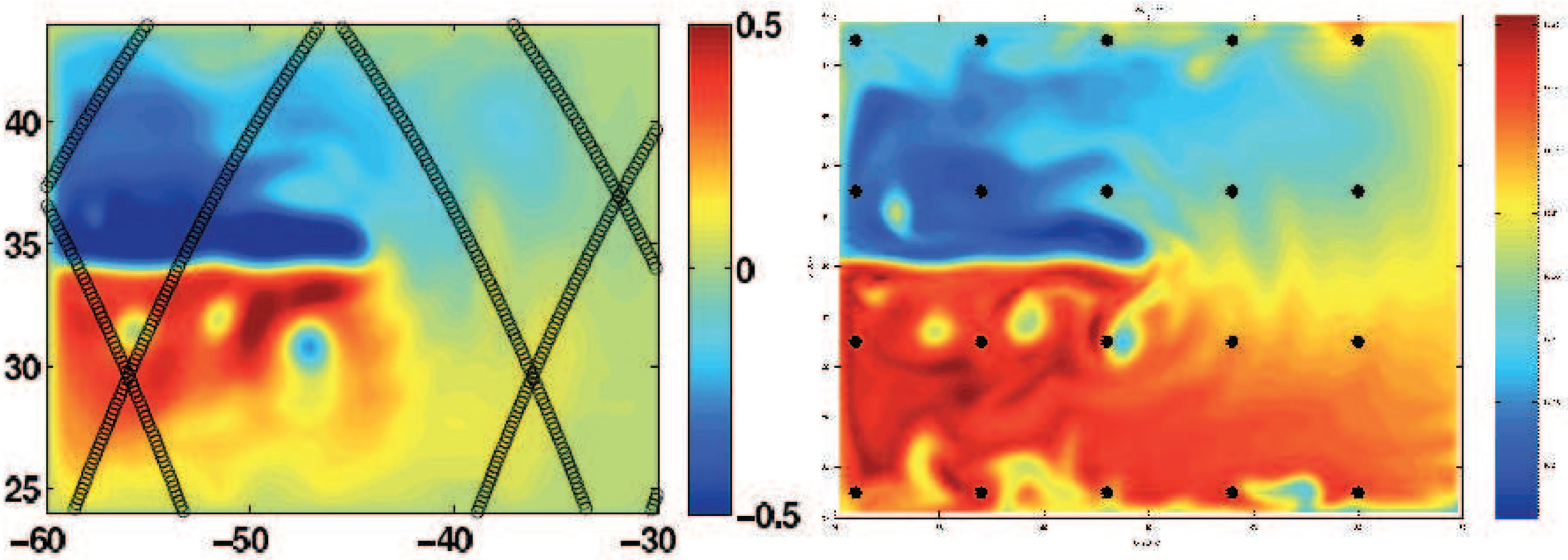
The surface pressure is given by : $p_s = \rho g \eta.$

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.

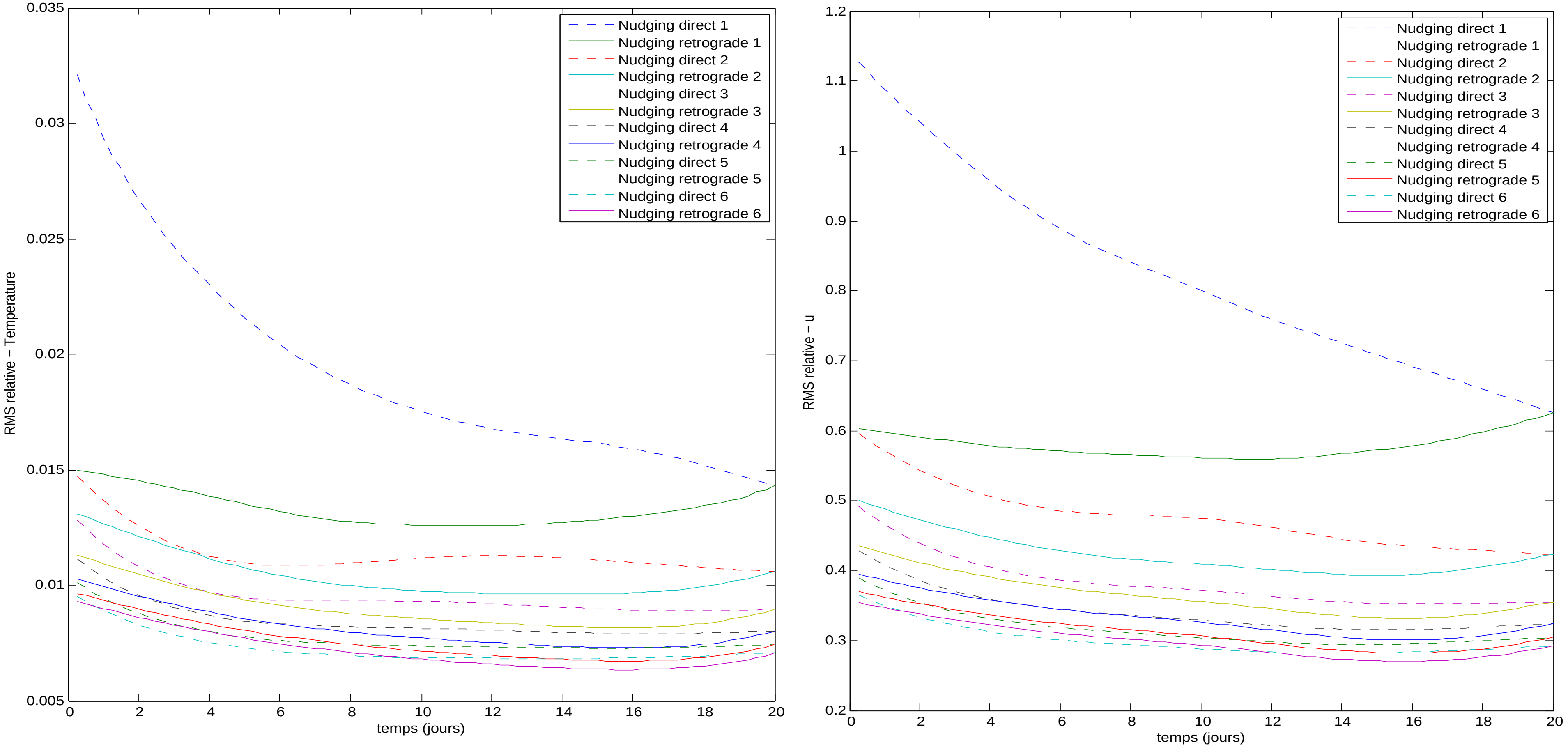
Twin experiments : observations are extracted from a reference run, according to networks of realistic density : SSH is observed similarly to TOPEX/POSEIDON, and temperature is observed on a regular grid that mimics the ARGO network density.

Full primitive ocean model



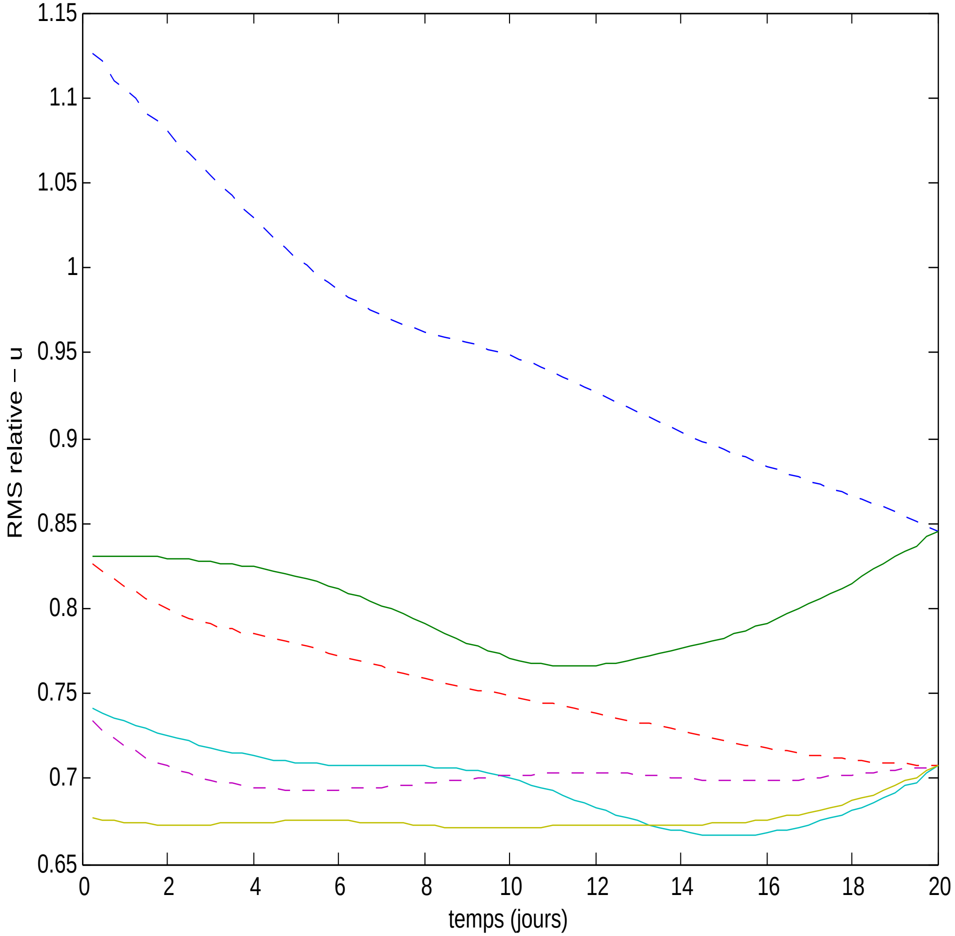
Example of observation network used in the assimilation : along-track altimetric observations (Topex-Poseidon) of the SSH every 10 days ; vertical profiles of temperature (ARGO float network) every 18 days.

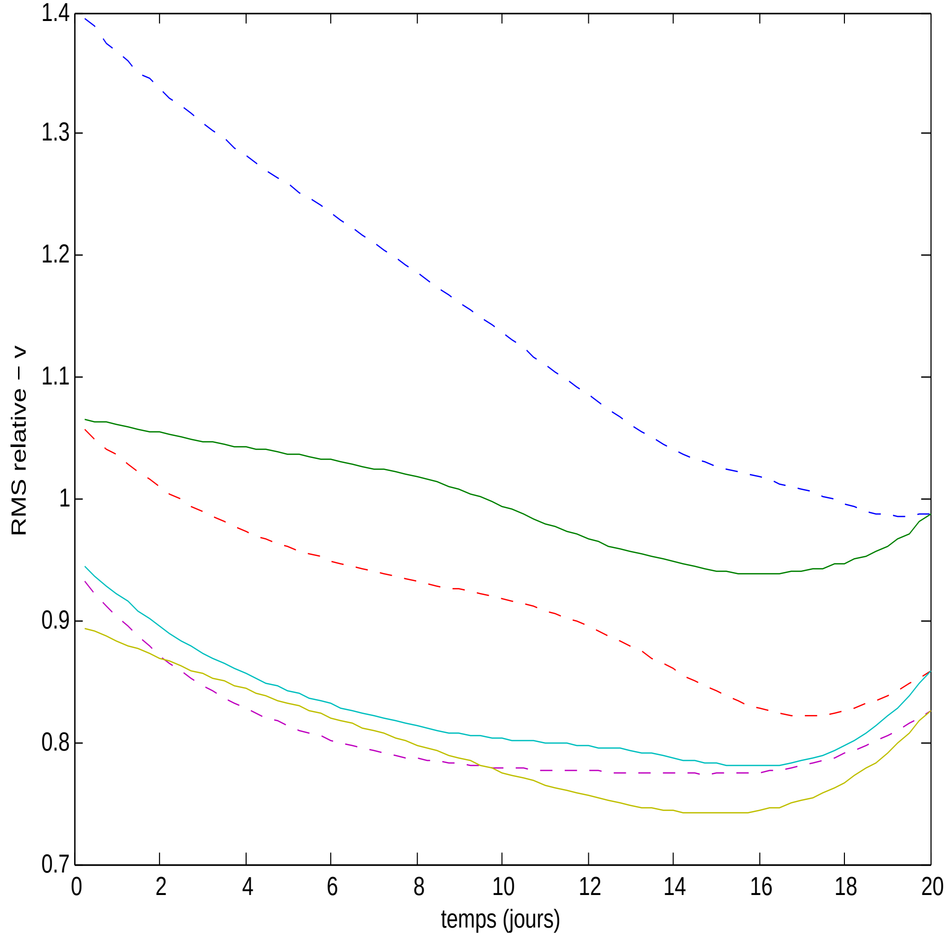
Numerical results



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
- Excellent preconditioner for other DA methods

Thank you for your attention !