OBSERVERS FOR DATA ASSIMILATION AND PARAMETER ESTIMATION

DIDIER AUROUX

Abstract. Nudging is a simple data assimilation method that uses dynamical relaxation to adjust a model towards observations. The standard nudging algorithm consists in adding a feedback term, proportional to the difference between the observations and the corresponding model state, to the model equations. Also known as the Luenberger (or asymptotic) observer, it theoretically requires an infinite time window to converge.

The Back and Forth Nudging (BFN) algorithm has been introduced in order to extend the efficiency of nudging to finite/small time windows. It consists in alternately solving the model forwards and backwards in time, with a nudging term in both cases, over the assimilation window. These approaches can be extended to more complex observers, for which non-observed variables can also be corrected with observed ones.

In this article, we give an overview of nudging, observers, and backward-forward algorithms, with applications to oceanography and fluid dynamics, for state and/or parameter estimation.

1. Introduction

The aim of data assimilation is to combine the observations and models, in order to retrieve a coherent and precise state of the system from a set of (usually) discrete spacetime data, and then to provide reliable forecasts of its evolution. Data assimilation covers all the mathematical and numerical techniques in which the observed information is accumulated into the model state by taking advantage of consistency constraints with laws of time evolution and physical properties, and which allow us to blend as optimally as possible all the sources of information coming from theory, models and other types of data [24, 14, 37].

Nudging is one of the very first data assimilation schemes, it has been successfully applied to geophysical systems almost 50 years ago. Nudging is a data assimilation method that uses dynamical relaxation to adjust a model towards observations. The standard nudging algorithm consists in adding to the state equations of a dynamical system a feedback term proportional to the difference between the observations and the equivalent quantities computed by integration of the state equations. The model appears then as a weak constraint, and the nudging term forces the state variables to fit as well as possible to the observations. This forcing term in the model dynamics has a tunable parameter that represents the relaxation time scale. In geophysical applications, it is usually chosen by numerical experimentation so as to keep the nudging terms small in comparison to the state equations, and large enough to force the model towards the observations. Using some theoretical considerations, it is possible to guess how the nudging coefficients should be chosen.

The nudging method is a flexible assimilation technique, and computationally much more economical than variational data assimilation methods [25] or sequential

data assimilation methods, based on Kalman filters [23, 24, 14]. It was first used in meteorology [22], and then it has been used with success in oceanography [38] and applied to a mesoscale model of the atmosphere [36]. Many results have also been carried out on the optimal determination of the nudging coefficients [41, 35, 39].

Several decades later, its ease of implementation and light computational cost, in comparison with 4D-Var or Kalman algorithms, make it still competitive. It indeed does not require any linearization step, nor adjoint model, nor manipulation of large covariance matrices ...

The main issues of data assimilation for geophysical systems are the huge dimension of the control vectors (and hence of the error covariance matrices) and the nonlinearities (most of the time, one has to linearize the model and/or some operators). The computation of the adjoint model is for example a difficult step in the variational algorithms. To get rid of these difficulties, we consider here nudging-based and more generally observers algorithms.

In Section 2, we introduce the notations, the standard nudging algorithm, and study its convergence in linear and nonlinear cases. In Section 3, we recall the backward nudging, back-and-forth nudging algorithm, and its extension to diffusive models. In Section 4, we introduce new developments of nudging or observer algorithms, for parameter estimation. In particular, we study the case of an unknown transport in an advection equation. Section 5 is devoted to conclusions and perspectives.

2. Forward nudging schemes

2.1. **Standard nudging.** The forward (or standard) nudging algorithm consists in adding to the state equations a feedback term, which is proportional to the difference between the observations and their equivalent quantities computed by the resolution of the state equations. The model appears then as a weak constraint, and the nudging term forces the state variables to fit as well as possible to the observations.

Let $X(t) \in \mathbb{R}^n$ be the model state at time t, and let consider a generic model equation:

$$\frac{dX}{dt} = F(X), \quad 0 < t < T, \tag{1}$$

where F represents the model, and with an initial condition $X(0) = X_0$. The assimilation window is assumed to be $t \in [0; T]$. Any (set of) ordinary differential equation(s) (ODE) can be written in such a way. Note also that after spatial discretization (the choice of discretization is not discussed here), any partial differential equation (PDE) can also be written as in Eq. (1). We denote by n the dimension of the (discrete or discretized) state vector at any time.

We assume that some observations or measurements $Y(t) \in \mathbb{R}^p$ of the state variable $X(t) \in \mathbb{R}^n$ are available. One generally assumes that $p \leq n$, as we expect the measurements to live in a smaller dimension space than the state space. For theoretical purposes, we assume that the observations are available at any time t, but of course, we will later give implementation details in more realistic situations where the measurements are only available at several discrete times.

Finally, let H be the observation operator, that maps a model state vector $X \in \mathbb{R}^n$ to an observation-like vector $H(X) \in \mathbb{R}^p$.

Then the nudging algorithm applied to Eq. (1) simply writes:

$$\frac{dX}{dt} = F(X) + K(Y - H(X)), \quad 0 < t < T, \tag{2}$$

with the same initial condition $X(0) = X_0$, and where K is the nudging operator from $\mathbb{R}^p$ to $\mathbb{R}^n$.

It is quite easy to understand that if K is *large* enough, then the state vector transposed into the observation space (i.e. through the observation operator) $H(X(t))$ will tend towards the observation vector $Y(t)$. Indeed, in the linear case (where the model operator F is linear), the nudging method is nothing else than the Luenberger observer, also called asymptotic observer, where the operator K can sometimes be chosen so that the error goes to zero when time goes to infinity [30].

2.2. **Convergence in the linear case.** We consider here a linear case: the model operator F, the observation operator H and the nudging operator K are linear operators, hence matrices.

Let assume that there are no errors here, which means that the observations perfectly fit the true state X_{true}: $Y = HX_{true}$. We also assume the model to be perfect, so that the true state is a model solution:

$$\frac{dX_{true}}{dt} = FX_{true}, \tag{3}$$

and the nudging system is then

$$\frac{dX}{dt} = FX + K(Y - HX). \tag{4}$$

Let $E = X - X_{true}$ be the error on the state. By considering the difference between Eq. (4) and Eq. (3), E is propagated in time by the following dynamical system:

$$\frac{dE}{dt} = (F - KH)E. \tag{5}$$

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 $E \to 0$, i.e. $X \to X_{true}$ when $t \to +\infty$. This gives the asymptotic convergence of the Luenberger (or asymptotic) observer.

The pole assignment (or pole placement) method gives the existence of nudging gain matrices such that the observer system is stable, and hence the nudging solution converges towards the true state [4]: if a system (F, H) is observable, then there exists a matrix K such that $F - KH$ is stable, i.e. all eigenvalues have a strictly negative real part.

Note that for linear systems in finite dimension, the observability condition is ensured if and only if the rank of the observability matrix

$$\left(H; HF; HF^2; \ldots; HF^{n-1}\right) \tag{6}$$

is equal to n.

In order to illustrate the pole assignment method, we consider here a very simple example in dimension $n = 2$:

$$F = \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix}. \tag{7}$$

The eigenvalues of this matrix are 0 and 2, so that in one direction, the error on the state will remain constant in time, and in another direction, the error will increase

exponentially in time. Let assume that $p = 1$ and only the first component of the state is observed: $H = \begin{pmatrix} 1 & 0 \end{pmatrix}$. We are then looking for a matrix

$$K = \begin{pmatrix} k_1 \\ k_2 \end{pmatrix}$$

such that $F - KH$ is stable. The two eigenvalues of $F - KH$ are

$$\lambda = \frac{-(k_1 - 2) \pm \sqrt{k_1^2 - 4k_2 + 4}}{2}.$$

Hence, for any desired (negative) λ, we can set

$$k_1 = -2\lambda + 2, \quad k_2 = \frac{k_1^2}{4} + 1, \tag{8}$$

so that both eigenvalues of $F - KH$ are λ. For instance, $k_1 = 4$ and $k_2 = 5$ lead to $\lambda = -1$, so that the error will now decrease exponentially in time, with a decay rate of 1.

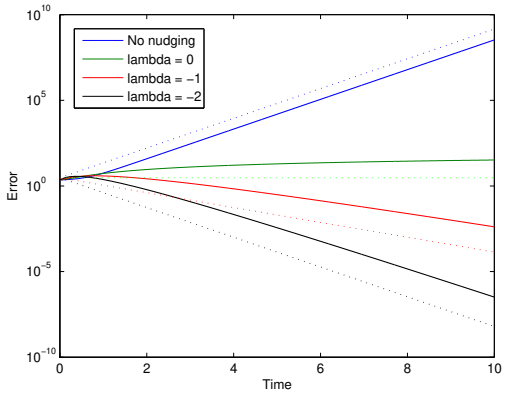

FIGURE 1. Evolution of the error norm $\|E\|$ versus time for the model given in Eq. (7), using various gain matrices K corresponding to various desired decay rates λ. The dotted lines correspond to a growth rate of 2 (blue), no variation (green), and decay rates of -1 (red) and -2 (black).

Fig. 1 shows the evolution of the error norm $\|E\|$ versus time, in logarithmic scale, for this simple linear model. The dotted lines correspond to constant growth or decay rates. In blue, there is no nudging, we use $K = 0$, so that the evolution of the error is driven by the largest eigenvalue of the matrix F, which is 2. In green, red and black, we use gain matrices K defined by Eq. (8), so that the observed decay rates of the error are consistant with the desired rates (given by λ).

2.3. **Extension to nonlinear cases.** We now consider a nonlinear system, where the (exact) model equation for the true state is:

$$\frac{dX_{true}}{dt} = FX_{true} + G(X_{true}), \tag{9}$$

where F stands for the linear part of the model, and $G(.)$ is a nonlinear function, assumed to be differential and Lipschitz continuous:

$$\|G(X_1) - G(X_1)\| \leq L\|X_1 - X_2\|, \quad \forall X_1, X_2, \tag{10}$$

for a given Lipschitz constant $L > 0$.

As in the previous section, we assume that the observation operator is linear, observations are perfect ($Y = HX_{true}$), and the linear part of the model (F, H) is observable. From the pole assignment method, we then know that there exists a matrix K such that $F - KH$ is stable. The standard Luenberger observer is then:

$$\frac{dX}{dt} = FX + G(X) + K(Y - HX), \tag{11}$$

so that the error $E = X - X_{true}$ is now solution of the following dynamical system:

$$\frac{dE}{dt} = (F - KH)E + G(X) - G(X_{true}). \tag{12}$$

Taking the inner product of Eq. (12) with E leads to

$$\frac{1}{2}\frac{d\|E\|^2}{dt} = E \cdot ((F - KH)E) + E \cdot (G(X) - G(X_{true}))\,,$$

that can be bounded by

$$\frac{1}{2}\frac{d\|E\|^2}{dt} \leq (\lambda_{max} + L)\|E\|^2, \tag{13}$$

thanks to Cauchy-Schwarz inequality, and Lipschitz continuity of G. Here, λ_{max} is the largest eigenvalue of $F - KH$. As (F, H) is observable, then $(F + LI, H)$ is also observable (I is the identity matrix), so that from the pole assignment method, we can choose K such that all eigenvalues of $F + LI - KH$ are (of real part) strictly smaller than 0, or that all eigenvalues of $F - KH$ are strictly smaller than $-L$, the opposite of the Lipschitz constant of G. Thus, the largest eigenvalue satisfies $\lambda_{max} < -L$ and this ensures the asymptotic decrease of the norm of the error: $X \to X_{true}$ when time goes to infinity.

2.4. **Example of nonlinear observers for the Lorenz model.** In order to illustrate this result, we consider here the Lorenz system [29]:

$$\begin{aligned} \frac{dx}{dt} &= \sigma(y - x), \\ \frac{dy}{dt} &= \rho x - y - xz, \\ \frac{dz}{dt} &= xy - \beta z, \end{aligned} \tag{14}$$

with the standard values of parameters $\sigma = 10$, $\rho = 28$ and $\beta = \frac{8}{3}$, and where $X = (x; y; z)$ is the state vector in $\mathbb{R}^3$.

The model (14) can be decomposed into a linear part F and a nonlinear part G as in Eq. (9):

$$F = \begin{pmatrix} -\sigma & \sigma & 0 \\ \rho & -1 & 0 \\ 0 & 0 & -\beta \end{pmatrix}, \qquad G(X) = \begin{pmatrix} 0 \\ -xz \\ xy \end{pmatrix}. \tag{15}$$

Assuming that all considered trajectories are bounded, then the function G is Lipschitz continuous.

If for instance we choose $H = (1 \quad 0 \quad 1)$, which means that there is only one observation of the system, which corresponds to the sum of the first and third components of the state, we can easily see that the observability matrix (6) is invertible, so that the system (F, H) is observable.

Note that for other choices of H, e.g. $H = (1 \quad 0 \quad 0)$ or $H = (0 \quad 1 \quad 0)$, the system is not observable, so that it is not possible to place the poles anywhere, but we can still prove that it is possible to find matrices K such that $F - KH$ is stable. As an example, with $H = (1 \quad 0 \quad 0)$, one can choose $K = (0 \quad \rho \quad 0)^T$ and the three eigenvalues of $F - KH$ are then -1, $-\beta$ and $-\sigma$.

We can also easily define nonlinear observers, where K is now a nonlinear operator. As an example, we assume here that only the first component x is observed, which means that $H = (1 \quad 0 \quad 0)$. In such case, we can consider the following observer system:

$$\begin{aligned} \frac{dx}{dt} &= \sigma(y - x), \\ \frac{dy}{dt} &= \rho x_{true} - y - x_{true} z, \\ \frac{dz}{dt} &= x_{true} y - \beta z, \end{aligned} \tag{16}$$

which is exactly the original Lorenz model where we replaced x by the (unnoisy) observation $x_{true} = HX_{true}$ in the y and z equations. The idea is quite easy to understand: the observation gives us the first component of the true state x_{true}, then we use it (instead of the observer's first component x) in the other equations of the observer, so that y and z will be controlled by x_{true}. Eq. (16) can be rewritten in a more explicit way:

$$\begin{aligned} \frac{dx}{dt} &= \sigma(y - x), \\ \frac{dy}{dt} &= \rho x - y - xz + \rho(x_{true} - x) - z(x_{true} - x), \\ \frac{dz}{dt} &= xy - \beta z + y(x_{true} - x), \end{aligned} \tag{17}$$

which is then the previous (linear) observer $K = (0 \quad \rho \quad 0)^T$ augmented by an additional nonlinear term $K(X) = (0 \quad -z \quad y)^T$.

The error E is then solution of

$$\frac{dE}{dt} = (F - KH)E + S(t)E, \tag{18}$$

where

$$F - KH = \begin{pmatrix} -\sigma & \sigma & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -\beta \end{pmatrix}, \qquad S(t) = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & -x_{true} \\ 0 & x_{true} & 0 \end{pmatrix}.$$

The eigenvalues of $F - KH + S(t)$ are the roots of the polynomial

$$(\lambda + \sigma)\left(\lambda^2 + \lambda(1 + \beta) + (\beta + x_{true}^2)\right),$$

and as $\beta > 0$ and $\sigma > 0$, then for any value of x_{true}, all three eigenvalues are (of real part) strictly negative, so that the error asymptotically decreases in time. Note that the largest real part oscillates between -1 (when $x_{true} = 0$, real eigenvalues) and $-\frac{11}{6}$ (when $|x_{true}| \geq \frac{5}{6}$, complex eigenvalues).

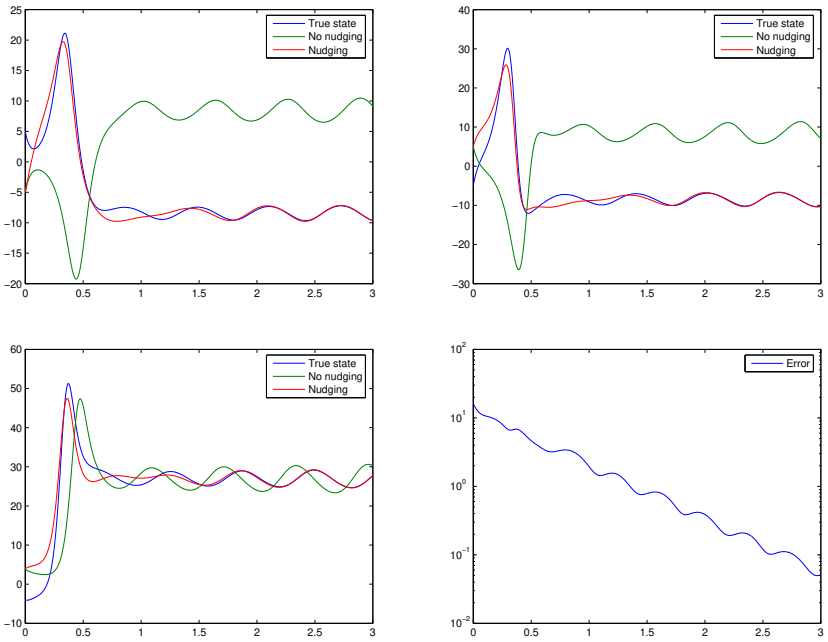

FIGURE 2. Evolution of the state versus time: x (top left), y (top right), and z (bottom left), for the true state X_{true} and the observer state X with or without the nudging term. Evolution of the norm of the error E versus time, in logarithmic scale (bottom right).

We now numerically illustrate this convergence by choosing the following starting points for the initialization of the Lorenz system:

$$X_{true}(0) = \begin{pmatrix} 5 \\ -5 \\ -4 \end{pmatrix}, \quad X(0) = -X_{true}(0) = \begin{pmatrix} -5 \\ 5 \\ 4 \end{pmatrix}.$$

Fig. 2 shows the evolution of the observer and true states versus time: the three components x, y and z are respectively shown on top left, top right, and bottom left figures. The true state X_{true} is represented in blue, the observer state without nudging is in green, and the observer state with nudging is in red. When there is no nudging, the state X remains far away from the true state X_{true}. But when we use the previously defined feedback term, the observer state quickly converges towards the true state.

The bottom right figure shows the evolution of $\|E\|$, the norm of the error, versus time in logarithmic scale. The slope approximately corresponds to a decay rate of 1, which is consistant with the largest eigenvalue of the system.

2.5. **Extension to more complex observers.** These ideas have been extended to more complex observers, with the aim of controlling non-observed variables with observed ones.

For instance, we refer the reader to [11] for an application to a shallow-water model, where the full state (water height and velocity) can be retrieved from observations of the height only. The proof of convergence in a linearized situation is based on a Fourier decomposition of the solution of a damped wave equation.

Another application is studied in [3], where the authors consider a compressible Navier-Stokes equation, written in conservation form. Either the pressure or velocity is observed, but not both. In both situations, the full state can also be retrieved, using an appropriate observer which is designed thanks to a Fourier analysis of the error. The authors prove that any desired decay rate can be achieved, and they also study the case where observations are not available everywhere in the space domain.

Remark. Observations are usually not available at any time step, nor any space gridpoint. There are many options, from simple to more complex, for dealing with this point. One possibility is to add the feedback term(s) only at the times/locations when/where observations are available. Otherwise, no feedback term is added to the model equations.

In order to avoid shocks, one usually filters, both in space and time, the feedback term, using smooth functions. Typically, in space, a convolution with e.g. a Gaussian kernel can smoothly spread the feedback term around the observation location. In time, similar ideas, or simpler trapezoidal filters, can be used to also spread the feedback over a small time sub-window around the observation time [13].

There are also other ways to deal with discrete data assimilation, typically by extending discrete observation values to piecewise constant functions of time, and by continuously assimilating the same observations, available at a given time, until the next observation time. We refer the reader to [20, 16, 21, 40] for more details.

3. Backward and forward observers

Observers usually converge in infinite time (asymptotically), so that in a standard data assimilation framework, where the time window is fixed (and possibly small), there is often not enough time to benefit from the exponential decrease of the error. It is then necessary to implement a new strategy, allowing one to perform multiple passes on the same assimilation window.

3.1. **Backward nudging.**

The backward nudging algorithm consists in solving the state equations of the model backwards in time, starting from a final state. A nudging term, with the opposite sign compared to the standard nudging algorithm, is added to the state equations, and the final obtained state is in fact an initial state of the system [8].

We assume that we have a final condition in the original model given by Eq. (1), instead of an initial condition. This leads to the following backward equation:

$$\frac{d\tilde{X}}{dt} = F(\tilde{X}), \quad T > t > 0, \tag{19}$$

with a *final* condition $\tilde{X}(T) = \tilde{X}_T$. If we apply nudging to this backward model with the opposite sign of the feedback term (in order to have a well-posed problem), we obtain

$$\frac{d\tilde{X}}{dt} = F(\tilde{X}) - \tilde{K}(Y - H(\tilde{X})), \quad T > t > 0, \tag{20}$$

where $\tilde{K}$ is the backward feedback. Solving Eq. (20) backwards in time, from $t = T$ to $t = 0$, allows to recover the state at the initial time $t = 0$.

3.2. **The BFN algorithm.** The Back and Forth Nudging (BFN) algorithm consists in solving first the forward (standard) nudging equation, and then the direct system backwards in time with a feedback term (which is the backward nudging). After resolution of this backward equation, one obtains an estimate of the initial state of the system. We repeat these forward and backward resolutions with the feedback terms until convergence of the algorithm [8].

The BFN algorithm is the following:

$$
\left\{
\begin{array}{l}
\dfrac{dX_k}{dt} = F(X_k) + K(Y - H(X_k)), \qquad 0 < t < T \\
X_k(0) = \tilde{X}_{k-1}(0),
\end{array}
\right.
\tag{21}
$$

$$
\left\{
\begin{array}{l}
\dfrac{d\tilde{X}_k}{dt} = F(\tilde{X}_k) - \tilde{K}(Y - H(\tilde{X}_k)), \qquad T > t > 0 \\
\tilde{X}_k(T) = X_k(T),
\end{array}
\right.
$$

with the notation $\tilde{X}_{-1}(0) = X_0$. Then, $X_0(0) = X_0$ the initial condition, and a resolution of the direct model gives $X_0(T)$ and hence $\tilde{X}_0(T)$. A resolution of the backward model provides then $\tilde{X}_0(0)$, which defines $X_1(0)$, and so on.

As we have seen in previous sections, the standard nudging converges asymptotically in time, so that a quite long time window can be necessary to get a strong decrease of the error. In many situations, this cannot be done as the time window is fixed and it can be relatively small. The idea of the Back and Forth Nudging is then to use this (small) time window to perform back and forth resolutions of the model, with nudging, so that the observations are assimilated multiple times over the window. It is a way to artificially extend the time window to a much larger one.

This algorithm can be compared to the 4D-Var algorithm [25], which also consists in a sequence of forward and backward resolutions. In the BFN algorithm, even for nonlinear problems, there is no need to linearize the system and the backward system is not the adjoint model (which can be difficult to implement) but simply the direct system, with an extra feedback term that stabilizes the resolution of the usually ill-posed backward resolution.

The BFN algorithm has been successfully tested for the system of Lorenz equations, Burgers equation and a quasi-geostrophic ocean model in [9], for a shallow-water model in [6] and compared with a variational approach for all these models. It has also been used to assimilate the wind data in a mesoscale model [15], for the reconstruction of quantum states in [28], for the assimilation of wide-swath altimetric data in a quasi-geostrophic ocean model [27, 26], ...

The convergence of the BFN algorithm has been proven in [8] for linear systems of ordinary differential equations and full observations, in [33] for reversible linear partial differential equations (wave and Schrödinger equations), in [19] for the reconstruction of quantum states... In [12], the authors consider the BFN algorithm on transport equations. They show that for non viscous equations (both linear transport and Burgers), the convergence of the algorithm holds under observability conditions. Convergence can also be proven for viscous linear transport equations

under some strong hypothesis, but not for viscous Burgers' equation. Moreover, the authors show that the convergence rate is always exponential in time [12].

Assuming that we set $K = \tilde{K}$, and assuming that the BFN converges, then the forward and backward solutions are equal at the limit, i.e $\tilde{X}_\infty = X_\infty$. If we take the sum of the two equations in (21) when the iteration number k goes to infinity, then the limit trajectory X_∞ satisfies the model equation (1). Moreover, the difference between the two equations in (21) shows that the limit trajectory is solution of the following equation:

$$K(Y - HX_\infty) = 0. \tag{22}$$

Eq. (22) shows that the limit trajectory perfectly fits the observations (through the observation operator, and the gain matrix). This simple calculation shows that the BFN algorithm over a finite time window $[0; T]$ is very similar to a simple direct nudging algorithm over an infinite time window, allowing then an asymptotic decay of the error when the number of backward-forward iterations goes to infinity (instead of time going to infinity).

Note also that the backward-forward framework can be extended to any type of observers, and not only nudging or its extensions where non-observed variables are corrected by the observed ones. One can for instance think at BF-Kalman filters or smoothers, using full or reduced-order Kalman filters [18].

3.3. **Diffusive Back and Forth Nudging algorithm.** Another simple example of backward-forward observer is a more recent extension of the BFN algorithm. In the framework of oceanographic and meteorological problems, there is usually no diffusion in the physical process (e.g. Euler equation). However, the numerical equations that are solved contain some diffusion terms in order to both stabilize the numerical integration (or the numerical scheme is set to be slightly diffusive) and model some subscale turbulence processes. We can then separate the diffusion term from the rest of the model terms, and assume that the differential equation reads:

$$\frac{dX}{dt} = F(X) + \mu\,\Delta X, \quad 0 < t < T, \tag{23}$$

where we assume that F has no diffusive terms, μ is the diffusion coefficient, and we assume that the diffusion is here a standard second-order Laplacian (note that it could be a fourth or sixth order derivative as in some oceanographic quasi-geostrophic models [32]).

The idea of the D-BFN (Diffusive BFN) algorithm is to solve forward and backward the model F, and to use alternating signs for both the feedback term (as in the standard BFN) and the diffusion term. The D-BFN reads:

$$\begin{aligned} \frac{dX_k}{dt} &= F(X_k) + \mu\,\Delta X_k + K(Y - H(X_k)), \quad 0 < t < T, \\ &\qquad \text{with } X_k(0) = \tilde{X}_{k-1}(0), \end{aligned} \tag{24}$$

$$\begin{aligned} \frac{d\tilde{X}_k}{dt} &= F(\tilde{X}_k) - \mu\,\Delta \tilde{X}_k - \tilde{K}(Y - H(\tilde{X}_k)), \quad T > t > 0, \\ &\qquad \text{with } \tilde{X}_k(T) = X_k(T). \end{aligned} \tag{25}$$

It is straightforward to see that the backward equation (25) can be rewritten, using $t' = T - t$, as:

$$\frac{d\tilde{X}_k}{dt'} = -F(\tilde{X}_k) + \mu\,\Delta\tilde{X}_k + \tilde{K}(Y - H(\tilde{X}_k)), \quad 0 < t < T, \tag{26}$$

where $\tilde{X}$ is evaluated at time t' and with an *initial* (in t') condition $\tilde{X}_k(t' = 0) = X_k(t = T)$. Then the backward equation is well-posed, solved forwards in time with an initial condition, and with the same diffusion operator as in the forward equation (24). Then the diffusion term both takes into account the subscale processes and stabilizes the numerical (backward) integration, and the feedback term still controls the trajectory with the observations. Note that in this case, it is usually possible to set $\tilde{K} = K$ as only the non-diffusive terms are now reversed in the model equation.

In a similar way as in Section 3.2, we assume here the convergence of the D-BFN algorithm in a linear situation. Then, if we take the sum of Eqs. (24) and (25) at the limit when the number of iterations goes to infinity, we see that the limit trajectory X_∞ is now a solution of the model equation without diffusion. And the difference between Eqs. (24) and (25) shows that the limit trajectory satisfies the Poisson equation:

$$-\mu\,\Delta X_\infty = K(Y - HX_\infty) \tag{27}$$

which represents a well-known smoothing process on the observations, for which the degree of smoothness is given by μ divided by the order of magnitude of K [10, 5]. Eq. (27) indeed corresponds to the Euler equation of the minimization of the following cost function

$$J(X) = \big(R^{-1}(Y - HX)\big) \cdot (Y - HX) + \mu\|\nabla X\|^2 \tag{28}$$

if we set $K = H^T R^{-1}$ as in [9, 10], R^{-1} being the inverse of the covariance matrix of observation errors. The first term represents the quadratic distance to the observations (as in the 4D-Var cost function) and the second one is a first order Tikhonov regularisation term over the domain of resolution [25]. This is a nice increment to the BFN algorithm (in which the limit trajectory fits the observations, see Eq. (22)), as in the DBFN algorithm, the limit trajectory is the result of a smoothing process on the observations (which are often very noisy).

The D-BFN algorithm has been successfully tested on a linear transport equation in [10] and on a non-linear Burgers equation in [7]. It has also been used for the full primitive ocean model NEMO [34].

4. New developments: observers for parameter estimation

Among the new developments of observers (in the forward only, or backward-forward framework), we can cite the identification of model parameters, jointly with the model state.

Variational approaches are often used for model calibration and/or parameter estimation, as it can also quite easily identify the parameters (and not only the state), thanks to the same adjoint model. But observers have been recently developed in order to also identify the parameters.

4.1. **Ad-hoc parameter equation.** Using the idea developed in [31, 1], we can use sequential data assimilation methods (like Kalman filters, simple nudging, ...) for the identification of model parameters by simply adding an ad-hoc parameter equation. We consider the following coupled model:

$$\frac{dX}{dt} = F(X,c), \quad 0<t<T, \quad X(0)=X_0, \tag{29}$$

$$\frac{dc}{dt} = 0, \quad 0<t<T, \quad c(0)=c_0, \tag{30}$$

where c represents the model parameter(s). The first equation (29) is the standard model equation, where the parameter c explicitly appears in the model function F, and we add an ad-hoc equation (30) for the parameter, c_0 being an a priori estimation of the parameter. This system of equations is equivalent to solving only the model equation $\frac{dX}{dt} = F(X,c_0)$.

It is then straightforward, on the paper, to apply the nudging (or BFN, or other observer), to this coupled system, assuming the state X is observed, with the aim of recovering both the state and the parameter. In practice, the observer design will clearly depend on the model equation, as the keypoint is to find how to control or correct the parameter equation with the innovation vector on the state. The general and theoretical idea is to define a Lyapunov function, enabling us to obtain the expression of the nudging term to be added to the parameter equation in order to ensure the exponential decrease of the error [19].

4.2. **Identification of the transport velocity.** As an example, we consider here a one-dimensional inviscid linear transport equation:

$$\partial_t v(t,x) + a(x)\partial_x v(t,x) = 0, \quad 0<t<T, \quad 0<x<1, \tag{31}$$

with an initial condition $v(t=0,x)=v_0(x)$, and periodic boundary conditions in space. The parameter to be estimated is the transport coefficient $c=a$, and the state is the velocity $X=v$. For sake of simplicity, we assume that the velocity v is fully observed in time and space, so that v_{obs} is available for any t and x.

Applying (back-and-forth) nudging to the coupled state-parameter model gives the P-BFN (Parameter BFN) algorithm, where a is now assumed to be time-dependent. The direct nudging system reads:

$$\begin{aligned} \partial_t v(t,x) + a(t,x)\partial_x v(t,x) &= k_v(v_{obs}(t,x)-v(t,x)), \quad 0<t<T, \\ v(t=0,x) &= v_0(x), \end{aligned} \tag{32}$$

$$\begin{aligned} \partial_t a &= k_a \mathcal{F}(v_{obs}(t,x)-v(t,x)), \quad 0<t<T, \\ a(t=0,x) &= a_0(x), \end{aligned} \tag{33}$$

where k_v and k_a are standard nudging parameters, and $\mathcal{F}$ is a feedback function, possibly involving differential operators, such that there exists a Lyapunov function that exponentially decreases in time, leading to the convergence of both the state and parameter observers. The idea is quite similar to what has been done e.g. in [11].

For the sake of simplicity, we can assume that $v_{obs}=0$. This can be done by linearizing the model around a reference solution (both state and parameter), and then by considering the equation of the error, which is the difference between the observer and observations (true state) equations.

By multiplying the first (state) equation by $v(t,x)$ and by integrating over the space domain $\Omega =]0;1[$, we get:

$$\int_\Omega \partial_t v(t,x)\, v(t,x)\, dx + \int_\Omega a(t,x)\partial_x v(t,x)\, v(t,x)\, dx = -k_v \int_\Omega v(t,x)^2\, dx.$$

Let define $\mathcal{L}_v(t) = \dfrac{1}{2}\displaystyle\int_\Omega v(t,x)^2\, dx$.

Then we obtain:

$$\partial_t \mathcal{L}_v(t) = -\int_\Omega a(t,x)\partial_x v(t,x)\, v(t,x)\, dx - 2k_v \mathcal{L}_v(t).$$

Similarly, by multiplying the second (parameter) equation by $a(t,x)$ and integrating over the space domain, if we define $\mathcal{L}_a(t) = \dfrac{1}{2}\displaystyle\int_\Omega a(t,x)^2\, dx$, then we have:

$$\partial_t \mathcal{L}_a(t) = -k_a \int_\Omega a(t,x)\mathcal{F}(v(t,x))\, dx.$$

By combining these two equations:

$$\partial_t \mathcal{L}_v(t) = -\int_\Omega a(t,x)\partial_x v(t,x)\, v(t,x)\, dx - 2k_v \mathcal{L}_v(t),$$

$$\partial_t \mathcal{L}_a(t) = -k_a \int_\Omega a(t,x)\mathcal{F}(v(t,x))\, dx,$$

we see that if we define:

$$\mathcal{F}(v(t,x)) = \partial_x v(t,x)\, v(t,x), \tag{34}$$

then the cross terms cancel and we deduce that

$$\frac{1}{2}\partial_t \left(\|v(t,.)\|^2 + \frac{1}{k_a}\|a(t,.)\|^2 \right) = -k_v \|v(t,.)\|^2,$$

so that the function $\mathcal{L}(t) = \mathcal{L}_v(t) + \dfrac{1}{k_a}\mathcal{L}_a(t)$ is a Lyapunov function, decreasing in time. This gives the joint convergence of the state v and parameter a towards the solution.

Fig. 3 shows the initial guess, the true solution and the solution identified by the P-BFN algorithm after 5 iterations, for both the state (left) and parameter (right). The bottom figure shows the decrease of the Lyapunov function versus time, during the back and forth iterations. Note that in this simulation, the observations are only available at some subperiods in time over the time window $[0;0.2]$ (and also only on some part of the space domain). This explains all the plateaux in the bottom figure: no observations are available at those times, so that no feedback terms are added to the equations, and thus the error does not decrease. The idea is similar to Kalman filters: when there are no available observations, only the model is used, while when observations are available, some correction is done. We can see that the parameter is very well recovered after just a few iterations.

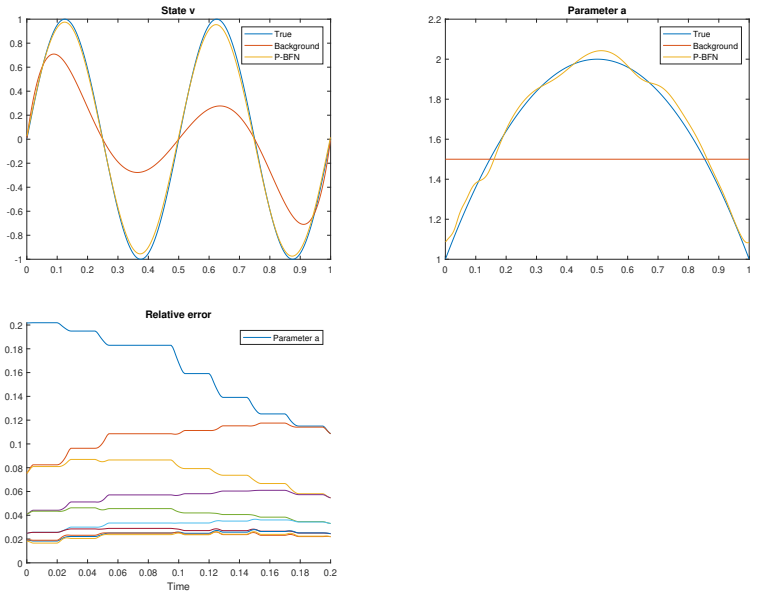

FIGURE 3. Initial guess (green), true solution (blue), and P-BFN identified solution (red) for the state (top left) and parameter (top right); Decrease of the error versus time during the P-BFN iterations (bottom).

4.3. **QG model with SWOT-like data.** As another example, the P-BFN algorithm has been tested on a quasi-geostrophic ocean model in the framework of SWOT data (see [2] for more details). We assume here that the phase speed c, or the barotropic deformation wavenumber $\kappa = \dfrac{f^2}{c^2}$ (f is the Coriolis parameter), is unknown (see e.g. [17]). We then add an ad hoc equation to the quasi-geostrophic model, claiming that the time derivative of c is equal to 0. And we apply the P-BFN algorithm to the coupled system in order to identify both the state and the parameter.

Fig. 4 shows on the top the result of BFN iterations when the parameter is fixed to the exact value (left) and to a wrong value (right). The forward (blue) and backward (red) iterations are done on the time period $[0; 20]$ days, and then the forecast period is $[20; 85]$ days. Using the exact parameter, the SSH (sea-surface height) is very well identified and the forecast remains excellent. Using a wrong parameter, it is still possible to reasonably identify the state, but once the assimilation window is finished, the forecast uses a wrong model parameter, so that the SSH error increases quite quickly.

Using now the P-BFN algorithm to jointly identify the state and parameter (see bottom left figure), the SSH is quickly corrected, and the parameter is also well identified, so that the SSH is small at the end of the assimilation window (almost as in the top left figure), and it remains small as well during the forecast, the parameter being very well estimated. Finally, the bottom right figure shows the decrease of the Lyapunov function describing the parameter error during the P-BFN

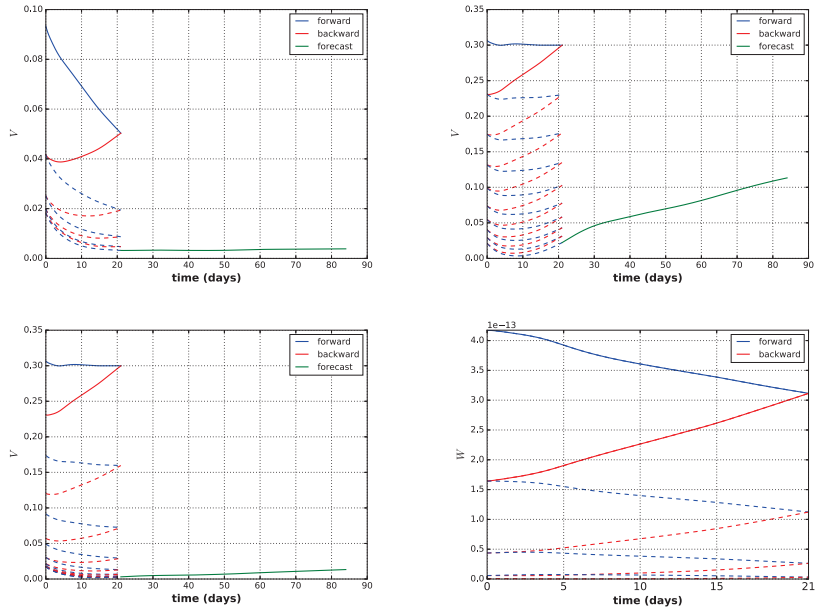

FIGURE 4. SSH error versus time using BFN with the true parameter (top left), BFN with a fixed wrong parameter (top right), and P-BFN for jointly estimating the state and parameter (bottom left); Lyapunov function describing the parameter error versus time during the P-BFN iterations (bottom right).

iterations. As with the standard BFN, the P-BFN only requires a few iterations to reach convergence, and to identify here both the parameter and the state.

5. Conclusions and perspectives

Nudging-based algorithms and observers have been proven to be very efficient and easy to implement, at least in quite simple situations. Recent improvements have been introduced in order to account for specific - but quite generic - issues like finite-time and small windows, diffusive models, reconstruction of non observed variables, parameter estimation with the aim of model calibration, ...

Several developments are now in progress with (forward-backward) observers in this framework, mainly still with the idea of identifying the full state of a system from a very partial knowledge, sometimes using indirect variables like passive tracers in a flow, or indirect observations (e.g. with images).

Also, as seen in Section 4, the choice of the observer feedback in the parameter equation is highly dependent on the state equation, so that further studies are currently ongoing for application to other types of equations (heat equation, wave equation, ...).

References

[1] C. Afri, V. Andrieu, L. Bako, and P. Dufour. State and parameter estimation: A nonlinear luenberger observer approach. *IEEE Trans. Automat. Control*, 62:973–980, 2017.

[2] S. Amraoui, D. Auroux, J. Blum, and E. Cosme. Back-and-Forth Nudging for the quasi-geostrophic ocean dynamics with altimetry: theoretical convergence study and numerical experiments with the future SWOT observations. *Discrete & Continuous Dynamical Systems - S*, 2022. http://dx.doi.org/10.3934/dcdss.2022058.

[3] A. Apte, D. Auroux, and M. Ramaswamy. Observers for compressible Navier-Stokes equation. *SIAM J. Control Optim.*, 56(2):1081–1104, 2018.

[4] M. Arnold and B. N. Datta. Single-input eigenvalue assignment algorithms: A close look. *SIAM J. Matrix Anal. Appl.*, 19(2):444–467, 1998.

[5] G. Aubert and P. Kornprobst. *Mathematical problems in image processing*, volume 147 of *Applied Mathematical Sciences*. Springer-Verlag, 2001.

[6] D. Auroux. The Back and Forth Nudging algorithm applied to a shallow water model, comparison and hybridization with the 4D-VAR. *Int. J. Numer. Methods Fluids*, 61(8):911–929, 2009.

[7] D. Auroux, P. Bansart, and J. Blum. An evolution of the Back and Forth Nudging for geophysical data assimilation: application to burgers equation and comparisons. *Inv. Prob. Sci. Eng.*, 21(3):399–419, 2013.

[8] D. Auroux and J. Blum. Back and forth nudging algorithm for data assimilation problems. *C. R. Acad. Sci. Paris, Ser. I*, 340:873–878, 2005.

[9] D. Auroux and J. Blum. A nudging-based data assimilation method for oceanographic problems: the Back and Forth Nudging (BFN) algorithm. *Nonlin. Proc. Geophys.*, 15:305–319, 2008.

[10] D. Auroux, J. Blum, and M. Nodet. Diffusive Back and Forth Nudging algorithm for data assimilation. *C. R. Acad. Sci. Paris, Ser. I*, 349(15-16):849–854, 2011.

[11] D. Auroux and S. Bonnabel. Symmetry-based observers for some water-tank problems. *IEEE Trans. Automat. Contr.*, 56(5):1046–1058, 2011.

[12] D. Auroux and M. Nodet. The Back and Forth Nudging algorithm for data assimilation problems: theoretical results on transport equations. *ESAIM Control Optim. Calc. Var.*, 18(2):318–342, 2012.

[13] J. W. Bao and R. M. Errico. An adjoint examination of a nudging method for data assimilation. *Month. Weather Rev.*, 125:1355–1373, 1997.

[14] A. F. Bennett. *Inverse modeling of the ocean and atmosphere*. Cambridge University Press, 2002.

[15] A. Boilley and J.-F. Mahfouf. Assimilation of low-level wind in a high resolution mesoscale model using the back and forth nudging algorithm. *Tellus A*, 64:18697, 2012.

[16] E. Burman, J. Ish-Horowicz, and L. Oksanen. Fully discrete finite element data assimilation method for the heat equation. *ESAIM Math. Model. Numer. Anal.*, 52(5):2065–2082, 2018.

[17] D. B. Chelton, R. A. Deszoeke, M. G. Schlax, K. El Naggar, and N. Siwertz. Geographical variability of the first baroclinic Rossby radius of deformation. *J. Phys. Ocean.*, 28:433–460, 1998.

[18] E. Cosme, J. Verron, P. Brasseur, J. Blum, and D. Auroux. Smoothing problems in a Bayesian framework and their linear Gaussian solutions. *Month. Weath. Rev.*, 140(2):683–695, 2012.

[19] A. Donovan, M. Mirrahimi, and P. Rouchon. Back and Forth Nudging for quantum state reconstruction. In *4th Int. Symp. Communications Control Signal Proc.*, pages 1–5, 2010.

[20] C. Foias, C. F. Mondaini, and E. S. Titi. A discrete data assimilation scheme for the solutions of the two-dimensional navier-stokes equations and their statistics. *SIAM J. Appl. Dyn. Syst.*, 15(4):2109–2142, 2016.

[21] B. García-Archilla and J. Novo. Error analysis of fully discrete mixed finite element data assimilation schemes for the navier-stokes equations. *Adv. Comput. Math.*, 46(61), 2020.

[22] J. Hoke and R. A. Anthes. The initialization of numerical models by a dynamic initialization technique. *Month. Weaver Rev.*, 104:1551–1556, 1976.

[23] R. E. Kalman. A new approach to linear filtering and prediction problems. *Trans. ASME - J. Basic Engin.*, 82:35–45, 1960.

[24] E. Kalnay. *Atmospheric modeling, data assimilation and predictability*. Cambridge University Press, 2003.

[25] F.-X. Le Dimet and O. Talagrand. Variational algorithms for analysis and assimilation of meteorological observations: theoretical aspects. *Tellus*, 38A:97–110, 1986.

[26] F. Le Guillou, N. Lahaye, C. Ubelmann, S. Metref, E. Cosme, and A. Ponte. Joint estimation of balanced motions and internal tides from future wide-swath altimetry. *J. Adv. Model. Earth Syst.*, 13(12), 2021. https://doi.org/10.1029/2021MS002613.

[27] F. Le Guillou, S. Metref, E. Cosme, C. Ubelmann, M. Ballarotta, J. Le Sommer, and J. Verron. Mapping altimetry in the forthcoming SWOT era by Back-and-Forth Nudging a one-layer quasigeostrophic model. *J. Atmos. Ocean. Tech.*, 38(4):697–710, 2021.

[28] Z. Leghtas, M. Mirrahimi, and P. Rouchon. Observer-based quantum state estimation by continuous weak measurement. In *American Control Conference (ACC)*, pages 4334–4339, 2011.

[29] E. N. Lorenz. Deterministic non periodic flow. *J. Atmos. Sci.*, 20:130–141, 1963.

[30] D. Luenberger. Observers for multivariable systems. *IEEE Trans. Autom. Contr.*, 11:190–197, 1966.

[31] P. Moireau and D. Chapelle. Reduced-order Unscented Kalman Filtering with application to parameter identification in large-dimensional systems. *ESAIM: Control, Optimisation and Calculus of Variations*, 17(2):380–405, 2011.

[32] J. Pedlosky. *Geophysical fluid dynamics*. Springer-Verlag, New-York, 1979.

[33] K. Ramdani, M. Tucsnak, and G. Weiss. Recovering the initial state of an infinite-dimensional system using observers. *Automatica*, 46(10):1616–1625, 2010.

[34] G. A. Ruggiero, Y. Ourmières, E. Cosme, J. Blum, D. Auroux, and J. Verron. Data assimilation experiments using the diffusive back and forth nudging for the nemo ocean model. *Nonlin. Proc. Geophys.*, 22:233–248, 2015.

[35] D. R. Stauffer and J. W. Bao. Optimal determination of nudging coefficients using the adjoint equations. *Tellus A*, 45:358–369, 1993.

[36] D. R. Stauffer and N. L. Seaman. Use of four dimensional data assimilation in a limited area mesoscale model - part 1: Experiments with synoptic-scale data. *Month. Weather Rev.*, 118:1250–1277, 1990.

[37] O. Talagrand. Assimilation of observations, an introduction. *J. Met. Soc. Japan*, 75(1B):191–209, 1997.

[38] J. Verron and W. R. Holland. Impact de données d'altimétrie satellitaire sur les simulations numériques des circulations générales océaniques aux latitudes moyennes. *Ann. Geophys.*, 7(1):31–46, 1989.

[39] P. A. Vidard, F.-X. Le Dimet, and A. Piacentini. Determination of optimal nudging coefficients. *Tellus A*, 55:1–15, 2003.

[40] B. You. A discrete data assimilation algorithm for the three dimensional planetary geostrophic equations of large-scale ocean circulation. *J. Dyn. Differ. Equ.*, 2022.

[41] X. Zou, I. M. Navon, and F.-X. Le Dimet. An optimal nudging data assimilation scheme using parameter estimation. *Quart. J. Roy. Meteorol. Soc.*, 118:1163–1186, 1992.

Université Côte d'Azur, CNRS, LJAD, France
Email address: didier.auroux@univ-cotedazur.fr