We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

6,900
Open access books available

186,000
International authors and editors

200M
Downloads

Our authors are among the

154
Countries delivered to

TOP 1%
most cited scientists

12.2%
Contributors from top 500 universities

CLARIVATE ANALYTICS

BOOK CITATION INDEX

INDEXED

WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

A Comparison Study on Performance of an Adaptive Filter with Other Estimation Methods for State Estimation in High-Dimensional System

Hong Son Hoang and Remy Baraille

Additional information is available at the end of the chapter

Abstract

In this chapter, performance comparison between the adaptive filter (AF) and other estimation methods, especially with the variational method (VM), is given in the context of data assimilation problem in dynamical systems with (very) high dimension. The emphasis is put on the importance of innovation approach which is a basis for construction of the AF as well as the choice of a set of tuning parameters in the filter gain. It will be shown that the innovation representation for the initial dynamical system plays essential role in providing stability of the assimilation algorithms for stable and unstable system dynamics. Numerical experiments will be given to illustrate the performance of the AF.

Keywords: dynamical system, innovation process, filter stability, minimum mean square prediction error, simultaneous stochastic perturbation

1. Introduction

Consider the following data assimilation problem: Given the dynamical system

$$x_{k+1} = \phi(x_k) + w_k, \qquad (1.1)$$

and the observations

$$z_{k+1} = H_{k+1}x_{k+1} + v_{k+1}, k = 0, 1, 2, ..., N \qquad (1.2)$$

Here, $x_k \in R^n$ is the system state at k instant, $\phi(.) : R^n \to R^n$, $z_k \in R^p$ is observation vector, $H_k \in R^{p \times n}$ is the observation matrix, w_k, v_k are the model and observation uncorrelated noise

sequences which are mutually uncorrelated and uncorrelated with x_0. The statistical characteristics of the entering random variables are given as

$$E[x_0] = \overline{x}_0, E[x_0 x_0^T] = M_0,$$

$$E[w_k] = 0, E[w_k w_l^T] = \delta_{kl} Q,$$

$$E[v_k] = 0, E[v_k v_l^T] = \delta_{kl} R, E[w_k v_l] = 0,$$

$$E[(x_0 - \overline{x}_0) w_k^T] = 0, E[(x_0 - \overline{x}_0) v_k^T] = 0.$$

(1.3)

The problem we consider here is to estimate the system state x_k under the conditions that the dimension n of x_k is of order $10^6 - 10^8$, and there are uncertainties in statistics of the model and observational noises. Due to very large n, it is impossible to apply traditional estimation algorithms for producing the estimate $\hat{x}_k$ and that is the reason there exist different approximation algorithms for solving this estimation problem. Theoretically, the optimal in mean square error (MSE) estimate $\hat{x}_{k/N}$ based on the set of observations $Z[1, N] := \{z_1, ..., z_N\}$ is a filtered estimate for $N = k$ and smoothed estimate for $N > k$ [1, 2]. For the linear dynamical system Eqs. (**1.1**) and (**1.2**), the computation of $\hat{x}_k := \hat{x}_{k/k}$ can be efficiently performed using the Kalman filter (KF) [3] which is a sequential procedure. The KF provides also the equations for computation of estimation errors. If we are interested in obtaining $\hat{x}_{k/N}$—the best estimate for x_k based on $Z[1, N]$, the Kalman smoother can serve as an efficient algorithm for its computation. The KF approach, however, is inappropriate for solving estimation problems in high-dimensional systems. In this chapter, the high-dimensional system means the system whose state dimension is of order $10^6 - 10^8$. At the present and in the near future, the computer capacity, in both computational power and memory, is still very far to be sufficient to implement the KF in real time to produce the filtered estimate and to make corresponding forecast. For suboptimal schemes for atmospheric data assimilation based on the KF, see Ref. [4].

In this chapter, the emphasis is put principally on comparison of the AF with VM. For the review on the data assimilation methods in meteorology and oceanography, see Ref. [5]. To see more the advantages of the AF, we implement the extended KF (EKF) [6] in Section 6 and will compare its performance with that of the AF (the experiment with Lorenz system). The Cooper-Haines filter (CHF) [7], widely used in data assimilation in oceanography, is also applied in Section 7 to produce the estimate for the ocean state. It serves as a reference to be compared with that produced by the AF in high-dimensional setting.

In the next section, the variational method (VM), which is widely used in data assimilation for high-dimensional systems in meteorology and oceanography, is outlined. Section 3 provides the recently developed AF approach to data assimilation. The main idea of the AF is to take the innovation representation for the input-output system as a departure point to formulate the optimization problem, with the parameters of the filter gain as control variables. Section 4 presents the tools to implement the AF in a simple and efficient way which is adapted for high-dimensional setting. This includes the objective function, filter stability, structure of the error covariance matrix (ECM), gain parameterization, algorithm for optimization known as simultaneous perturbation stochastic approximation (SPSA). It is shown how the ECM is estimated

using an ensemble of samples of prediction error (PE) and the hypothesis on separation of vertical and horizontal variables structure (SeVHS) in the ECM. Computational comparison between the VM and AF is also given here. Section 5 includes a simple numerical experiment showing in detail how work the VM and AF, the difficulties of the VM in searching optimal solution and why no similar difficulties are encountered in the AF. The more complicated experiment with chaotic system known as Lorenz system is done in Section 6. The difficulties encountered here concern extreme sensitivity of its solution to small errors in the initial condition. Section 7 presents the performance of different filters in the data assimilation experiment with the high-dimensional ocean model MICOM with the North Atlantic configuration. The conclusions are given in Section 8.

Notation: In the chapter, A^T denotes the transpose of the matrix A; $E[.]$, $E[.|.]$ denote the expectation and conditional expectation, respectively; $||A||_F$ denotes the Frobenius norm of a matrix A.

2. Variational method (VM)

Consider the problem of estimating $\{x_k\}$ in Eqs. (1.1)–(1.3). The VM consists of minimizing the following objective function

$$J[x_0, ..., x_N] = e_0 M_0^{-1} e_0 +$$

$$\sum_{k=1}^{N} (z_k - H_k x_k)^T R^{-1} (z_k - H_k x_k), e_0 := x_0 - \overline{x}_0, \tag{2.1}$$

$$J[x_0, ..., x_N] \rightarrow \min_{[x_0,...,x_N]} \tag{2.2}$$

$$\text{under the constraints (1.1)} \tag{2.3}$$

Thus, in the VM, we seek optimal solutions in the functional space (space of functions $\{x_k\}$). For systems of high dimension, this task is impossible to perform. The simplification is required. Suppose the system Eq. (1.1) is linear and perfect, that is,

$$x_{k+1} = \Phi_k x_k, k = 0, 1, ... \tag{2.4}$$

Expressing all x_k as functions of the initial state x_0,

$$x_k = \Phi(k, 0) x_0,$$

$$\Phi(k, l) = \Phi_{k-1} ... \Phi_l, (k > l), \Phi(k, k) = I, \tag{2.5}$$

$$I \text{ is the identity matrix of appropriate dimension}$$

and substituting $x_k, \forall k$ Eq. (2.5) into Eq. (1.1), at each kth observation instant, the following set of observations is available for x_0,

$$z_k^1 = H_k^1 x_0 + v_1^k, k = 1, 2, \ldots$$
$$H_k^1 := [(H_1\Phi(1,0))^T, \ldots, (H_k\Phi(k,0))^T]^T, \qquad (2.6)$$
$$v_k^1 = [v_1^T, \ldots, v_k^T]^T.$$

Under the assumption on perfect model, the optimization problem Eqs. (2.1)–(2.3) is simplified as

$$J[x_0] \rightarrow \min_{[x_0]}, \qquad (2.7)$$

$$J[x_0] := e_0^T M_0^{-1} e_0 + \sum_{k=1}^{N} (z_k - H'_k x_0)^T R_k^{-1} (z_k - H'_k x_0), \qquad (2.8)$$
$$H'_k := H_k \Phi(k, 0).$$

We have now the unconstrained optimization problem Eqs. (2.7) and (2.8) with the vector of unknown parameters $\theta := x_0$—the initial state. This problem can be solved using standard optimization techniques [8].

It is not hard to write out a solution to the problem Eqs. (2.7) and (2.8). For high-dimensional systems, there is no computational and memory resources to handle such implementation. In practice, the solution to the problem Eqs. (2.7) and (2.8) is found by solving iteratively the equation

$$\nabla_\theta J[\theta] = 0, \qquad (2.9)$$
$$\nabla_\theta J[\theta] := [\partial J/\partial \theta_1, \ldots, \partial J/\partial \theta_1]^T.$$

Comment 2.1. Usually, finding a solution to Eqs. (**2.7**) and (**2.8**) is a heavy task: in addition to storing the model solution produced by the direct model, minimization requires 20–30 iterations to reach a relatively good approximate solution.

Comment 2.2. Writing out $\nabla_\theta J[\theta]$ shows that solving Eq. (**2.9**) requires

$$(H'_k)^T y = \Phi_k^T \Phi_{k-1}^T \ldots \Phi_1^T H_k^T y \qquad (2.10)$$

for some y. As Φ_k^T is impossible to store, the approach known as adjoint equation (AE) is used which requires to construct AE code for computing the product $\Phi_k^T y$. Each iteration in minimization of Eqs. (2.7) and (2.8) thus requires one integration of the model over the assimilation period, followed by one adjoint integration. The cost of one adjoint integration is about twice the cost of one direct integration, so that one minimization requires the equivalent of between 50–100 integrations of the model over the assimilation period (p. 205 [9]). In the next section, we see that the SPSA can also be used to solve this problem at a much lower cost.

Comment 2.3 As θ—initial state—has a physical meaning, it is important to introduce constraints on the appropriate physically realistic structure of the correction for θ during the estimation process. A poor (in the physical sense) structure of the guess for the initial state can lead to large estimation errors.

3. Adaptive filtering (AF)

To overcome the difficulties listed in Comments 2.1–2.3, an adaptive filtering (AF) has been proposed in [10]. The main difference of the AF with the VM is lying in the choice of innovation representation for the original input-output system Eqs. (1.1) and (1.2) as a departure point to formulate the optimization problems. It is well known that under standard conditions, the optimal in MSE estimate $\hat{x}_k$ can be obtained by the KF. As the innovation process for the system output in the KF forms a white sequence, Kailath [11] has developed an innovation approach, in an elegant way, to derive the optimal filter for more general linear systems like nonstationary, filtering problems with Markovian processes for the model and observation errors. The innovation approach to linear least-squares approximation problems is first to "whiten" the output data and then to treat the resulting simpler white-noise observation problem. Consider the observation (output) sequence z_k. The innovation process, associated with z_k, is written as

$$\zeta_k = z_k - E[z_k|z_{k-1}^1], \tag{3.1}$$

Under standard conditions (Gaussianness, uncorrelated noise sequences …), $E[z_k|z_{k-1}^1] = H_k\hat{x}_{k/k-1}$ hence

$$\zeta_k = z_k - H_k\hat{x}_{k/k-1}, \hat{x}_{k/k-1} = \Phi_{k-1}\hat{x}_{k-1}, \tag{3.2}$$

where $\hat{x}_{k/k-1}$ is an optimal in MSE one-step ahead prediction for x_k given z_{k-1}^1. Using ζ_k instead of z_k, one can write out the formula for the estimate $\hat{x}_k$ and the KF under standard conditions. The filter has the form

$$\hat{x}_k = \Phi_k\hat{x}_{k-1} + K_k\zeta_k,$$
$$K_k = M_kH_k^T[H_kM_kH_k^T + R_k]^{-1} \tag{3.3}$$

where M_k is the ECM for the prediction $\hat{x}_{k/k-1}$. This matrix is found as a solution to the Riccati equation

$$M_k = \Phi_kP_k\Phi_k^T + Q_k, P_k = [I - K_kH_k]M_k. \tag{3.4}$$

Due to the very expensive computational burden in time stepping the ECM M_k in Eq. (**3.4**) as well as insufficient memory storage, the KF is impractical for solving data assimilation problems in very high-dimensional setting. The idea of the AF is based on the fact that when the filter is optimal, the innovation ζ_k has a minimum variance. If we assume that the gain K_k belongs to a set of parameterized gains, that is,

$$K_k = K_k(\theta), \theta \in \Theta, \tag{3.5}$$

the optimal AF can be considered as that in some class of parameterized filters of a given structure. The following objective function is introduced

$$J(\theta) = E[\Psi(\theta)] \to \min{}_{\theta \in \Theta}, \Psi(\zeta_k) = ||\zeta_k||^2_{\Sigma_k^{-1}}, ||\zeta_k||^2_{\Sigma_k^{-1}} := <\zeta_k, \Sigma_k^{-1} \zeta_k > . \tag{3.6}$$

In Ref. [12], the different classes of parameterized filters are found which belong to the class of stable reduced-order filters (ROF) [10, 13].

As an example for one class of ROFs, consider

$$K_k = P_{r,k} K_{e,k} \tag{3.7}$$

where $K_{e,k} : R^p \to R^{n_e}$ represents the gain, mapping the innovation vector from the observational space to the reduced space R^{n_e} of dimension $n_e \le n$; $P_{r,k}$ is mapping the reduced space R^{n_e} to the full space R^n. The choice of a reduced space is of primary importance since it depends on the main characteristics of the filter known as stability. As proved in [12], under detectability condition, stability of the filter is ensured by forming the columns of $P_{r,k}$ from unstable and stable eigenvectors (or singular vectors, Schur vectors) of the fundamental matrix Φ_k, and one can choose

$$K_{e,k} = H_{e,k}^T [H_{e,k} H_{e,k}^T(k) + R_k]^{-1}, H_{e,k} := H_k P_{r,k}, \tag{3.8}$$

One class of parameterized filters is (Section 5.2.2 in Ref. [12])

$$
\begin{aligned}
K_k(\theta) &= P_{r,k} \Lambda K_{e,k}(\theta), \\
\Lambda &= \text{diag } [\theta_1, \ldots, \theta_{n_e}], \\
1 - 1/|\phi_i| &< \eth_1(i) \le \theta_i \le \eth_2(i) < 1 + 1/|\phi_i|.
\end{aligned}
\tag{3.9}
$$

if ϕ_i is an unstable or neutral eigenvector of Φ. For the stable ϕ_i, we have

$$0 < \eth_1(i) \le \theta_i \le \eth_2(i) < 2, \tag{3.10}$$

4. Differences between VM and AF

4.1. Batch data formulation

We list now the main differences between two approaches VM and AF from which it becomes clear what are the advantages of the AF over the VM.

To make easier comparison between two approaches, let us write out the objective function Eq. (**3.6**) using a representation in a sample space

$$
\begin{aligned}
J_N(\theta) &\to \min{}_{\theta \in \Theta}, \\
J(\theta) \approx J_N(\theta) = \frac{1}{N} \sum_{k=1}^{N} (z_k - H\Phi_k \hat{x}_{k-1}(\theta))^T \Sigma_k^{-1} (z_k - H\Phi_k \hat{x}_{k-1}(\theta))
\end{aligned}
\tag{4.1}
$$

Mention that in practical implementation of the AF, the optimization algorithm is not constructed on the basis of Eq. (4.1), but on Eq. (3.6). That is, due to the fact that Eq. (**4.1**) is written in a batch form which requires to make optimization over the time interval $[1, N]$

resulting in a very high computational burden. Minimizing Eq. (**3.6**) allows to apply SPSA method which is much less consuming for both computational and memory requirements. Below the main differences between VM and AF are listed:

(D1) Dynamical system (DS): if in Eq. (**2.1**), the DS is the initial system Eq. (**1.1**); in Eq. (**4.1**), the DS is the filtering Eq. (**3.3**). This difference has an interesting consequence: if in practice, there is very little known about statistics of w_k, the sequence ζ_k is observed, and hence, it is possible to estimate the statistics of ζ_k.

(D2) The system noise w_k in Eq. (**1.1**) is white, while in Eq. (**3.3**), ζ_k is a white sequence only if the filter is optimal. That allows us to easily apply different statistical tests for verifying the optimality of the assimilation procedure.

(D3) Control variable x_0 in the VM is the initial state, whereas the control variable in Eq. (**3.6**) is the parameter vector θ.

This difference has an important consequence: as x_0 has to be of precise physical meaning (depending, for example, on the ocean domain of interest), the structure for the guess $\theta_0 := \hat{x}_0^0$ (for the initial state) as well as correction $\delta\hat{x}_0^\nu$, generated by iterative algorithm, must be chosen carefully so that at each ν iteration, the estimate $\hat{x}_0^\nu$, $\hat{x}_0^\nu = \hat{x}_0^{\nu-1} + \delta\hat{x}_0^\nu$, must be of physically realistic structure. This is not an easy task. On the other hand, in the AF, the parameters usually are immaterial [see θ Eq. in (**3.10**)]; hence, the choice of structure for θ is of no importance.

(D4) Suppose the DS Eq. (**1.1**) is unstable. It implies that the error in estimating x_0 will grow during integration of the direct and AE. As for the AF, by its construction, the filtering system Eq. (**3.3**) remains stable. This can be seen by representing the filtering Eq. (**3.3**) through its fundamental matrix L_k,

$$
\begin{aligned}
\hat{x}_k &= L_k\hat{x}_{k-1} + K_k v_k \\
L_k &= (I - K_k H_k)\Phi_k.
\end{aligned}
\tag{4.2}
$$

As shown in Ref. [12], the filter Eq. (**4.2**) is stable under the conditions Eqs. (**3.9**) and (**3.10**). It means that the filtering error is bounded during model integration since the parameters θ_i are lying in the interval guaranteeing a stability of the filter Eq. (**4.2**).

(D5) Return to the objective function Eqs. (**2.1**)–(**2.3**). First taking the derivative of the objective function Eqs. (**2.1**)–(**2.3**) wrt (with respective to) x_0, we have

$$
\begin{aligned}
\frac{1}{2}\nabla_{x_0}J[\hat{x}_0^\nu] &= M_0^{-1}e_0^\nu - \sum_{k=1}^{N}\Phi^T(k,0)H_k^T R_k^{-1}(z_k - H_k\Phi(k,0)x_0^\nu) \\
&= M_0^{-1}e_0^\nu - \sum_{k=1}^{N}\Phi^T(k,0)H_k^T R_k^{-1}(H_k\Phi(k,0)e_0^\nu + v_k),\, e_0^\nu := x_0 - \hat{x}_0^\nu.
\end{aligned}
\tag{4.3}
$$

One sees that for a batch of N observations, Eq. (**4.3**) requires computation of N terms (without counting for the term $M_0^{-1}e_0^\nu$). The kth term is associated with the assimilation instant k, and one needs to compute first $\mu_k := \Phi(k,0)e_0^\nu$, that is, to integrate k times the direct model Φ_κ for $\kappa = 1,\ldots, k$ and next to integrate backward (k times also) the AE Φ_κ^T from $\kappa = k$ to $\kappa = 1$, that

is, to compute $\Phi^T(k,0)H_\kappa^T R_\kappa^{-1}(H_k\mu_k + v_k)$. The larger the k, the bigger the amplification of the initial error e_0^v and the observation error v_k. The error e_0^v is amplified doubly since it is integrated by the direct and adjoint models. But the amplification of v_k (and w_k when $w_k\neq 0$) is most worrying since it is integrated in the gradient estimate, making the gradient direction to be, possibly, completely erroneous.

4.2. Implementation of AF

4.2.1. The choice of criteria Eq. (3.6)

The choice of Eq. (3.6) is important in many aspects in order to obtain a simple and efficient data assimilation algorithm. The idea lying in the criteria Eq. (3.6) is to select some pertinent parameters as a control variables for minimizing the mean of the cost function $\Psi(\zeta_k)$. For example, for the class of filters Eqs. (3.3), (3.7)–(3.10) the vector $\theta = (\theta_1,\ldots, \theta_{n_e})^T$ can be chosen as control vector for the problem Eq. (3.6).

The solution to the problem Eq. (3.6) can be found iteratively using a stochastic optimization (SA) algorithm

$$\theta_{k+1} = \theta_k - a_k\nabla_\theta\Psi(\zeta_{k+1}) \tag{4.4}$$

where $\{a_k\}$ is a sequence of positive scalars satisfying some conditions to guarantee a convergence of the estimation procedure. The standard conditions are

$$a_k \to 0, \sum_{k=1}^{\infty} a_k = \infty, \sum_{k=1}^{\infty} a_k^2 < \infty \tag{4.5}$$

The algorithm Eq. (4.4) is much more simple [compared to the computation of Eq. (4.3)] since it requires, at the kth assimilation instant, to compute only the gradient of the sample cost function $\Psi(\zeta_k)$. The gradient $\nabla_\theta\Psi(\theta_k)$ of the sample objective function $\Psi(\theta_k)$ can be computed using the AE approach (in what follows, for simplicity, the subscript k will be omitted to shorten the notations).

$$\frac{1}{2}[\delta\Psi(\zeta_{k+1})]_{\theta_k} = -(H\Phi P_r\delta\Lambda K_e\zeta_k, \zeta_{k+1}) = -(\delta\Lambda K_e\zeta_k, \zeta_{k+1}'), \zeta_{k+1}' := P_r^T\Phi^T H^T\zeta_{k+1} \tag{4.6}$$

Thus, minimization of Eq. (3.6) by gradient-based SA algorithm requires only one integration of the direct model and one backward integration of the AE code: direct integration of $\hat{x}_k$ for producing the forecast $\hat{x}_{k+1/k} = \Phi\hat{x}_k$ and backward integration $\Phi^T H_k\zeta_{k-1}$ in computation of ζ_{k+1}'. For the structure of the gain Eq. (3.9), the objective function Ψ is quadratic wrt θ; hence, one can find easily the optimal parameters.

A less computational burden can be achieved by measuring the sample objective function (but not based on a gradient formula): instead of computing the gradient by Eq. (4.6) based on AE, one can approximate the gradient using the values of the cost function [on the basis of finite difference scheme (FDSA)]. Traditionally, the ith component of the gradient can be approximated by

$$\nabla_{\theta_i}\Psi(\theta_k) = g_i = [\Psi(\theta_k + c_k e_i) - \Psi(\theta_k - c_k e_i)]/(2c_k) \tag{4.7}$$

where e_i is the unit vector with 1 in the ith component, 0 otherwise.

It is seen that FDSA algorithms do not require the formula for the gradient. However, for the high-dimensional systems ($n \approx O(10^6) - O(10^7)$), this algorithm is inapplicable due to component-wise derivative approximation: for approximation of each partial derivative of the cost function, we need to make two integrations of the direct model.

In order to overcome the difficulties with very high dimension of θ, recently the class of algorithms known as simultaneous perturbation SA (SPSA) receives a great interest [14, 15]. The algorithm SPSA is of the same structure as that of FDSA Eq. (**4.7**), with the difference residing in the way to perturb stochastically and simultaneously all the components of θ. Concretely, let $\Delta_k = (\Delta_{k,1}, \ldots, \Delta_{k,n})^T$ be a random vector, $\Delta_{k,i}, i = 1, \ldots, n$ are Bernoulli independent identically distributed (iid). The gradient of the objective function is estimated as

$$\begin{aligned}
\nabla_\theta \Psi(\theta_k) &= g = (g_1, \ldots, g_n)^T, \\
g &= [\Psi(\theta_k + c_k \Delta_k) - \Psi(\theta_k - c_k \Delta_k)]\Delta_k^{-1}/(2c_k), \\
\Delta_k &= (\Delta_{k,1}, \ldots, \Delta_{k,n})^T, \Delta_k^{-1} := (1/\Delta_{k,1}, \ldots, 1/\Delta_{k,n})^T.
\end{aligned} \tag{4.8}$$

It is seen that in the SPSA, all the directions are perturbed at the same time (the numerator is identical in all n components). Thus, SPSA uses only two (or three) times integrations of the model, independently on the dimension of θ which makes it possible to apply to high-dimensional optimization problems. Generally, SPSA converges in the same number of iterations as FDSA, and it follows approximately the steepest descent direction, behaving like the gradient method [14]. On the other hand, SPSA, with the random search direction, does not follow exactly the gradient path. On average, though, it tracks the gradient nearly because the gradient approximation is an almost unbiased estimator of the gradient, as shown in Ref. [15].

For the SPSA algorithm, the conditions for $\{a_k\}$ and $\{c_k\}$ are

$$\begin{aligned}
&a_k > 0, c_k > 0, \quad a_k \to 0, \quad c_k \to 0, \\
&\sum_{k=1}^{\infty} a_k = \infty, \sum_{k=1}^{\infty} (a_k/c_k)^2 < \infty
\end{aligned} \tag{4.9}$$

4.2.2. On the operator P_r

As shown in Ref. [12], $span[P_r]$–the subspace, spanned by the columns of P_r, must be chosen so that the filter gain K ensures a stability of the filter. Mention that even the KF may suffer from instability. To ensure filter stability, P_r is constructed from all unstable and neutral eigenvectors of the fundamental matrix Φ (or real Schur vectors (ScVs), singular vectors). In practice, we choose P_r to be consisting of the column vectors of S (called *S-PE samples*)

$$S = \Phi X \tag{4.10}$$

which are results of integration of leading ScVs (columns of X). The columns in S have the meaning of the PE for the system state and are used to approximate the ECM M. As to the

ScVs, they are preferred to eigenvectors (or singular vectors) because the ScVs are real, and their computation is numerically stable. Mention that computation of singular vector requires also adjoint code. The ensemble of columns of S plays the same role as an ensemble PE samples in the ensemble-based filtering technique for approximating the background ECM [16, 17].

4.2.3. On separation of vertical and horizontal variables structure in ECM [18]

Let us consider the situation when the DS is described by PDEs. The state vector at the time instant k is $x_k = x_k(i, j, l)$ where (i, j, l) represents a grid point in three dimensional space. Introduce the stabilizing structure for the filter gain [12]

$$K_k = M_k H_k^T [H_k M_k H_k^T + R_k]^{-1},$$
$$M = M_d, M_d := P_r \Lambda P_r^T,$$

(4.11)

where Λ is symmetric positive definitive. As shown in Ref. [12], one can choose Λ to be diagonal with diagonal elements serving to regularize the amplitude of ECM. The matrix M in Eq. (**4.11**) is ECM, and if it is computed on the basis of the Riccati equation (**3.6**), K_k is known as the KF gain. For $M = M_d$, computation of M is realizable if the reduced dimension n_e is not too large. Actually the number n_e of the ensemble size is of order $O(100)$ which is too small for M to be a good approximation for the true ECM. In Ref. [18], it is assumed that the estimated ECM is a member of the class of ECMs with separation of vertical and horizontal variables structure (SeVHS). Mention that this hypothesis is not new and used in modeling the ECM in meteorological data assimilation [19]. The optimal ECM is found as a solution of the minimization problem

$$J(\theta) = E||M_d - M_v(\theta_1) \otimes M_h(\theta_2)||_F^2,$$
$$||.||_F \text{ denotes matrix Frobenious norm },$$
$$J(\theta) \to \min_\theta, \theta = (\theta_1^T, \theta_2^T)^T.$$

(4.12)

As the number of vertical layers in the today's numerical models is of order $O(10)$, all elements of the vertical ECM M_v (included in θ_1) can be considered as tuning to be estimated. As to M_h, it is often chosen in analytical form (e.g., the 1st or 2nd order autoregressive models). The parameters like correlation length can be selected as components of the control vector θ_2 in M_h.

Using dominant real Schur vectors has advantages that they are real and their computation is stable [12] while the computation of eigenvectors is unstable, and they may be complex.

4.3. Computational comparison between VM and AF

We give a brief comparison (computational burden) between the VM and AF algorithms. **Table 1** shows the number of elementary arithmetic operations required for implementation of VM and AF filtering algorithms based on AE tool. A smaller number of operations are required for the AF if the SPSA method is used (no need of Φ^T). Here, for simplicity, n^2

operations are accounted for the product $\Phi^T y$ (the same number as that required for Φy). In the AF, we assume that the full ECM M is used, whereas in the AROF (adaptive ROF), $M := P_r P_r^T$ as shown in Eq. (**4.6**). These numbers are calculated on the basis of Eqs. (**2.1**) and (**2.3**) since they represent the most computational burdens for these two algorithms. These numbers are rounded up to the dimensions of the entry matrices. Here, N_{it_o} is a number of iterations required to solve the minimization problem (**2.1**)–(**2.3**); N_{it} is a number of iterations required to solve the equation

$$\Xi y = \zeta_k, \Xi := [HMH + R]. \tag{4.13}$$

In the VM algorithm, the computation of M_0^{-1}, R^{-1} is not taken into account. In **Table 1**, there is also the number of operations required for the AROF, when the ECM M is given in the product decomposition form $M = P_r P_r^T$, $P_r \in R^{n \times n_e}$. In this situation, instead of n^2 operations in the AF, we need to perform $2nn_e$ operations. For $n_e << n$, much less computational and memory requirements are needed to perform the AROF.

VM	$(n^2(N^2 + N) + N(2np + p^2) + n^2)N_{it_o}$
AF	$((n^2 + 2np + p^2)N_{it} + 2(np + n^2))N$
AROF	$((2nn_e + 2np + p^2)N_{it} + np + 2nn_e)N$

Table 1. Number of elementary arithmetic operations.

To have the idea on how work in practice the VM and AF, here the examples of experiments with two numerical models MICOM (see Section 7) and HYCOM (Hybrid Coordinate Ocean Model) [20] developed at SHOM, Toulouse, France. The first experiment is performed with the MICOM model. The observations are available each 10 days (*ds*) during 2 years. For the MICOM (state dimension $n = 3 \times 10^5$), the 10 *ds* forecast requires 45 s (supercomputer Caparmor, IFREMER, France, sequential run). The 2-year integration takes 54.45 min (73 × 45 s). The AF needs 54.45 min × 3 = 164 min (or 2 h 45 min) to perform the assimilation experiment for the 2-year period. In this context, the VM requires between 5.7 ds and 11.4 ds to perform the experiment (hypothesis 50–100 times integration of the MICOM over the 2-year window, Section 2.2). As to the HYCOM (state dimension $n = 7 \times 10^7$), the observations are available each 5 ds. The 5 ds forecast requires 1 h (supercomputer Beaufix, Météo, France, parallel run, 62 processors). Two year integration requires 146 × 1 h = 146 h (or 6 ds). The AF needs, hence 18 ds to make the 2-year experiment. As to the VM, the experiment requires between 304 ds and 608 ds. That is one of the reasons why in operational setting one has to choose a short window for assimilating the observations by the VM.

Comment 4.1. Looking at **Table 1**, one sees that the dominant numbers n_d of operations in the VM and AF are $n_d(VM) = n^2 N^2 N_{it_o}$ and $n_d(NAF) = n^2 N N_{it}$. If we assume that $N_{it_o} \approx N_{it}$ (in fact, the number of iterations for solving the optimization problem Eqs. (**2.7**) and (**2.8**) is often larger than the number of iterations for solving the system of equations (**4.13**); it is more critical for the VM when the DS is nonlinear); the number $n_d(VM)$ is N times larger than $n_d(NAF)$.

5. Simple numerical examples: scalar case

5.1. Estimation problem

Consider the simple scalar dynamical process $x(t) = sin(t)$. As $\dot{x}(t) = cos(t)$, for $t_k = k\delta t$, using the approximation $x(t_k + \delta t) - x(t_k) = cos(t_k)\delta t$, one has the following discrete dynamical system

$$x_{k+1} = \phi_k x_k + u_k + w_k, \phi_k = \phi = 1, u_k = cos(k\delta t),$$
$$E(w_k) = 0, E(w_k v_l) = \sigma_w^2 \delta_{kl}, k = 0, 1, \ldots, N-1.$$

(5.1)

In Eq. (5.1), w_k represents the Gaussian model error. Suppose at each t_k moment, we observe the state x_k corrupted by the Gaussian noise v_k, hence

$$z_k = hx_k + v_k, h = 1, k = 1, \ldots, N,$$
$$E(v_k) = 0, E(v_k v_l) = \sigma_v^2 \delta_{kl}, k \neq l$$

(5.2)

where δ_{kl} is the Kronecker symbol. Suppose that the true initial state $x_0^* = 0.5$. The problem we study here is to estimate the system state x_k based on the set of observations $z_k, k = 1, \ldots, N$.

5.1.1. Experiment: cost functions

In the experiment, $\delta t = 0.01$, $N = 1000$. The two methods, VM and AF, will be implemented to produce the system estimates. We study two situations: (S1) the model is considered as perfect, that is, $w_k = 0$; and (S2) there exists the model error w_k with the variance $\sigma_w^2 = 0.001$. As to v_k, $\sigma_v^2 = 0.1$ in both cases.

Let $w_k = 0$. To see the advantages of the AF over VM in finding optimal solutions, **Figures 1** and **2** display the curves (time averaged variance of distances between the true trajectory and those resulting from varying the control variables) as functions of tuning parameters in these two methods: the initial state $\theta := x_0$ and the parameter $\theta := \lambda$ (see Eq. (3.9)). We remark that for the filtering system (5.1), (5.2), as $\phi = 1$, the system has one stable eigenvalue and one stable eigenvector (singular value and Schur decompositions are of the same structure). The filter fundamental matrix $L = (1-Kh)\phi = (1-K)$ is stable if $K \in (0, 2)$. For the gain structure (4.11), L is stable for any $M > 0$ since $K = \frac{M}{M+\sigma_r^2}$ for $h = 1$ and $\sigma_v^2 = 0.1 > 0$. We have then $K \in (0, 1) \subset (0, 2)$. Thus, one can choose $\theta := M > 0$ as tuning parameter. This structure is of less interest compared to Eq. (3.9) since θ enters in K in a nonlinear way, and in fact, K is allowed to vary only in the interval $(0, 1)$. For Eq. (3.9), $P_r = 1, H_e = 1$ hence $K_e = \frac{1}{1+\sigma_r^2} < 1$ and the filter is stable if θ satisfies Eq. (3.10). For this structure, the filter is stable for $K \in (0, 2)$. We will select the last structure as a departure point to optimize the AF performance.

From **Figure 1**, it is seen that the curve "noise-free" is equal to 0 when $\hat{x}_0 = 0.5$ for S1, but the "noisy" attains the minimal value 0.121 at $\hat{x}_0 = 0.6$. We note that almost the same picture is obtained for the cost function (2.1) (time averaged variance of the distance between $\hat{x}_k, k = 1, \ldots, 1000$ and observations $z_k, k = 1, \ldots, 1000$) subject to $\hat{x}_0 \in [-1 : 1]$. Despite the fact

that the curves in **Figure 1** are quadratic, it is impossible to find the true initial state in the noisy situation since almost the same curves are obtained for the cost function Eq. (**2.1**).
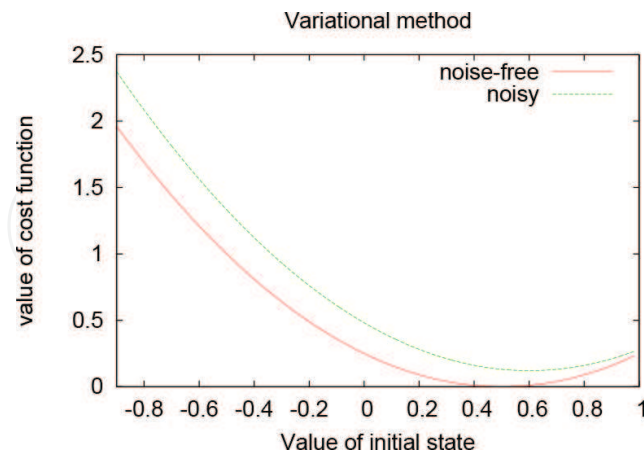

Figure 1. VM: cost functions resulting from perfect model and that with a model error.

Figure 2 presents the same curves as those in **Figure 1** resulting from application of the filter by letting the parameter θ in the gain vary in $\theta \in (0:2)$. **Figure 2** shows that for the both curves "noise-free" and "noisy", the minimal values are attained at $\theta = 1.1$ for both situations S1, S2. Moreover, the two minimal values are identical. The same picture is observed for the cost function Eq. (**4.1**) as function of $\theta \in (0:2)$. It means that independently on whether the model is perfect or not, AF formulation allows optimization algorithms to find the optimal value for θ, hence, to ensure optimality of the filter.

Two curves "noisy" in **Figures 1** and **2** show that when the model is noisy, the minimal value of the curve "noisy" (VM) in **Figure 1** is much higher (it is equal to 0.121) than that in **Figure 2** (0.009, for filtering). This fact is in favour of the choice of a short window for assimilating observations by the VM.
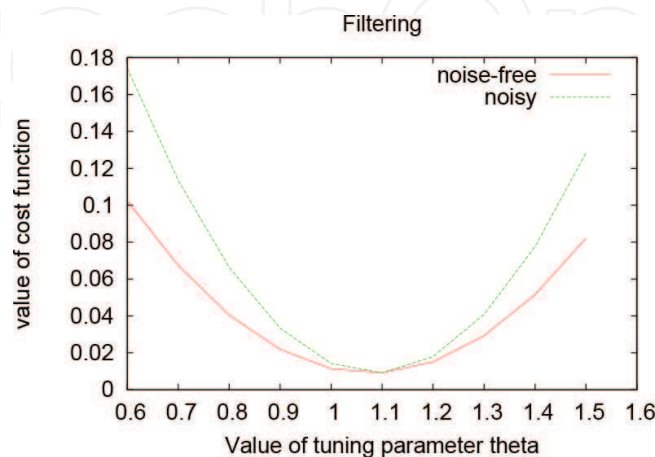

Figure 2. Filtering: cost functions resulting from perfect model and that with model error.

6. Numerical experiment: Lorenz system

6.1. Lorenz equations

The Lorenz attractor is a chaotic map, noted for its butterfly shape. The map shows how the state of a dynamical system evolves over time in a complex, non-repeating pattern [21].

The attractor itself and the equations from which it is derived were introduced by Edward Lorenz [21], who derived it from the simplified equations of convection rolls arising in the equations of the atmosphere.

The equations that govern the Lorenz attractor are:

$$\frac{dy_1}{dt} = -\sigma(y_1 - y_2), \frac{dy_2}{dt} = \rho y_1 - y_2 - y_1 y_3, \frac{dy_3}{dt} = y_1 y_2 - \beta y_3, \tag{6.1}$$

where σ is called the Prandtl number, and ρ is called the Rayleigh number. All $\sigma, \beta, \rho > 0$, but usually $\sigma = 10$, $\beta = 8/3$ and ρ is varied. The system exhibits chaotic behavior for $\rho = 28$ but displays knotted periodic orbits for other values of ρ.

6.2. Numerical model

In the experiments, the parameters σ, ρ, β are chosen to have the values 10, 28, and 8/3 for which the "butterfly" attractor exists.

The numerical model is obtained by applying the Euler method (first-order accurate method) to approximate Eq. (**6.1**). Symbolically, we have

$$y(t_{k+1}) = F^{'}(y(t_k)), y(t_k) := (y_1(t_k), y_2(t_k), y_3(t_k))^T, \tag{6.2}$$

where $\delta t := t_{k+1} - t_k$ is the *model* time step. The observations arrive at the moments T_k and $\Delta T_k := T_{k+1} - T_k$. The experiment setup is similar to that described in Ref. [22].

6.3. Observations: assimilation

The corresponding $\delta t = 0.005$, $\Delta T_k = 1$, hence the sequence of observations is given by $z(k) := z(T_k), k = 1, \ldots, N_o$. The dynamical system corresponding to the transition of the states between two time instants T_k and T_{k+1} is denoted as

$$x_{k+1} = F(x_k) + w_k \tag{6.3}$$

In Eq. (**6.3**), w_k simulates the model error. The sequence w_k is assumed to be a white noise having variance 2, 12.13 and 12.13 respectively. The observation system is then given by

$$z_k = Hx_k + v_k \tag{6.4}$$

where the operator $H = [h_1^T, h_2^T]^T$, $h_1 = (1, 0, 0), h_2 = (0, 0, 1)$, that is, the first and third components x_1, x_3 are observed at each time instant $k = 1, \ldots, 100$. The noise sequence v_k is white with

zero mean and variance $R = 2I_2$ where I_n is the unit matrix of dimension n. The initial estimate in all filters is given by the initial condition $\hat{x}(0) = (1, -1, 24)^T$.

The true system state x^* is modeled as the solution of Eq. (**6.3**) subject to $x_0^* = (1.508870, -1.531271, 25.46091)^T$.

The problem considered in this experiment is to apply the extended KF (EKF), nonadaptive filter (NAF), and adaptive filter (AF) to estimate the true system state using the observations z_k, $k = 1, 2, N_o$ and to compare their performances.

Here, the NAF is in fact the prediction error filter (PEF). Mention that the PEF is developed in Ref. [23] in which the prediction error ECM is estimated on the basis of an ensemble of PE samples, that is,

$$M = \frac{1}{T-1} \sum_{k=1}^{T} B_s(k), B_s(k) = \sum_{k=1}^{L} \delta x_k^{(l)} \delta x_k^{(l), T}, \tag{6.5}$$

where $\delta x_k^{(l)}, l = 1, \ldots, L$ are members of the set of L S-PE samples obtained by $L + 1$ integrations of the model from the reference state and L perturbed states which grow in the directions of the L dominant Schur vectors.

The filter gain is taken in the following form

$$K = MH^T \Sigma^{-1}, \Sigma = HMH^T + R \tag{6.6}$$

which is time invariant. At the same time, for the comparison purpose, the EKF is also used for assimilating the observations.

Figure 3 shows the evolution of the prediction errors resulting from three filters: NAF, AF, and EKF. It is of no surprise that the NAF has produced the estimates with larger estimation error. By adaptation, however, it is possible to obtain the AF, which improves significantly the PEF and even behaves better than the EKF.

Mention that the VM is much less appropriate for assimilating the observations in the Lorenz model due to the choice of the initial state as control vector. For simplicity, we simulate the situation when all three components of the system state are observed in additive noise, i.e. with $H = I_3$. **Figure 4** displays time averaged variances of the difference between the true trajectory and model trajectory (denoted as $AV(x^*, \hat{x})$), resulting from varying the third component of the initial state for two situations of noise-free and noisy models. Namely, we initialize the model by the initial state, which is the same as the true one $x_0^* = (1.508870, -1.531271, 25.46091)^T$, with the difference, that the third component $\hat{x}_3(0)$ is varying in the interval $[24.5 : 26.5]$. The global minimum is attained at $x_0^*(3) = 25.46091$ as expected. However, if the system is initialized by the estimate in a vicinity, even not so far from $x_3^*(0)$, there is no guarantee that the VM can approach the true initial condition. For the noisy model, the global minimum is not attained at $x_0^*(3)$. As for the PEF, the function $AV(x^*, \hat{x})$ is quadratic wrt to the gain parameter, for both situations of noise-free or noisy models, as seen in **Figure 5**: here, the sample cost function (**4.1**) is computed over all assimilation period, by varying the third parameter θ_3 in the gain (related to the third observed component of the system state).
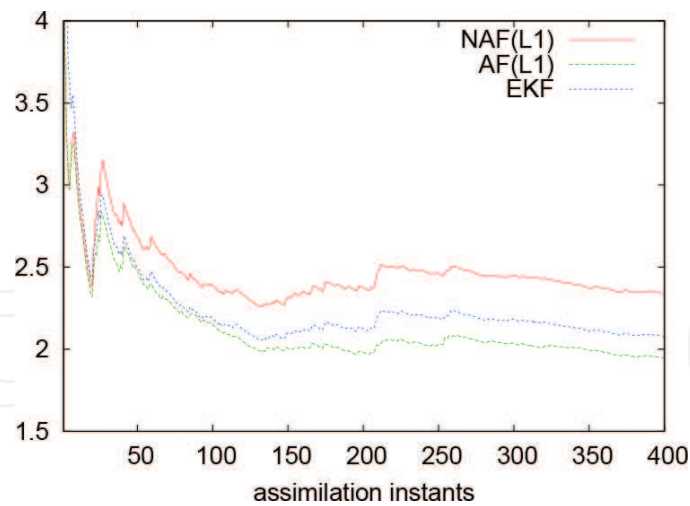
Figure 3. Prediction errors resulting from three filters: nonadaptive filter (NAF), adaptive filter (AF), and EKF.
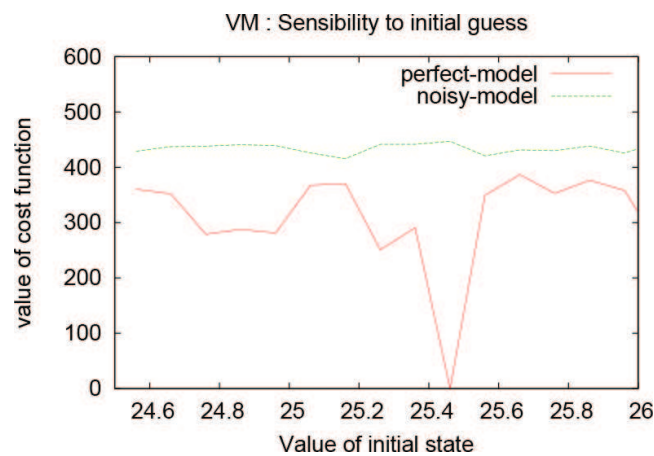

Figure 4. Time averaged variance between the true trajectory and model trajectory in the VM as a function of perturbed third component of the initial state. The global minimum is attained at the true initial condition, but there is no guarantee for the VM to approach the true initial state (no-noisy model). For noisy model, the global minimum is not attained at the true initial state. The curve "noisy model" is scaled by the factor $C = 1/15$.
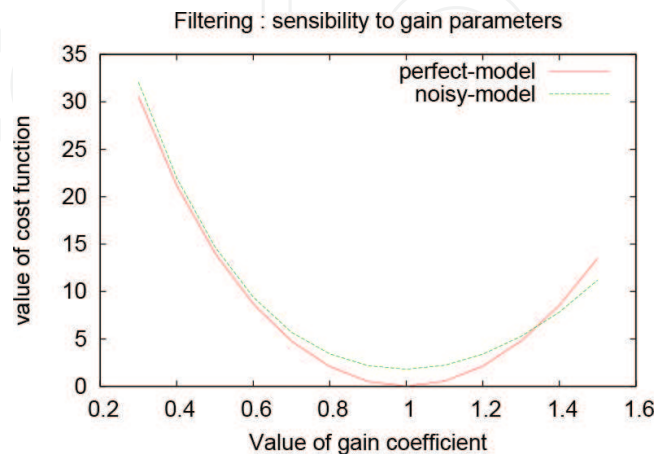

Figure 5. Cost function in the PEF as a function of perturbed third gain parameter θ_3. It is seen that in the PEF, the cost function is quadratic wrt to the gain parameter in both situations of noise-free and noisy models. The curve "noisy model" is scaled by the factor $C = 1/50$.

7. Assimilation in high-dimensional model

7.1. MICOM model and assimilation problem

In this section, we show how the AF can be designed in a simple way to produce the high performance estimates for the ocean state in the high-dimensional ocean model MICOM. For details on the Miami Isopycnal Coordinate Ocean Model (MICOM) used here, see Ref. [24]. The model configuration is a domain situated in the North Atlantic from 30°N to 60°N and 80°W to 44°W; for the exact model domain and some main features of the ocean current (mean, variability of the SSH, velocity) produced by the model, see Ref. [24]. The grid spacing is about 0.2° in longitude and in latitude, requiring $N_h = II \times JJ = 25200$ ($II = 140$, $JJ = 180$) horizontal grid points. The number of layers in the model is $KK = 4$. It is configured in a flat bottom rectangular basin (1860 km $\times$ 2380 km $\times$ 5 km) driven by a periodic wind forcing. The model relies on one prognostic equation for each component of the horizontal velocity field and one equation for mass conservation per layer. We note that the state of the model is $x := (h, u, v)$ where $h = h(i, j, lr)$ is the thickness of lrth layer; $u = u(i, j, lr), v = v(i, j, lr)$ are two velocity components. The layer stratification is made in the isopycnal coordinates, that is, the layer is characterized by a constant potential density of water. The model is integrated from the state of rest during 20 years. Averaging the sequence of states over 2 years 17 and 18 gives a so-called *climatology*. During the period of 2 years 19 and 20, every 10 days (10 ds), we calculate the sea surface height (SSH) from the layer thickness h which will serve as a source for generating observations to be used in the assimilation experiments (in total, there are 72 observations).

7.2. Different filters

The filter used for assimilating SSH observations is of the form

$$\hat{x}_{k+1} = F[\hat{x}_k] + K\zeta_{k+1}, k = 0, 1, \ldots \tag{7.1}$$

where $\hat{x}_{k+1}$ is the filtered estimate for x_{k+1}, $x_{k+1} = [h_{k+1}, u_{k+1}, v_{k+1}]$ is the system state at $(k + 1)$ assimilation instant, $F(.)$ represents the integration of the nonlinear MICOM model over 10 days, K is the filter gain, ζ_{k+1} is the innovation vector. The gain K is of the form (**4.11**) where the ECM M will be estimated from the MICOM model. In the experiment, to be closed to realistic situations, only SSH at the grid points $i = 1, \ldots, 140, j = 1, \ldots, 180$ are collected as observations. Thus, the observations are available not at all model grid points. The gain K is symbolically written as $K = (K_h, K_u, K_v)^T$ with K_u, K_v representing the operators which produce the correction for the velocity (u, v) from the layer thickness correction $K_h\zeta$ using the geostrophy hypothesis. The filter thus is a reduced order which has the gain K_h to be estimated from S-PE samples.

7.2.1. PEF: computation of ECM

In the experiment, two assimilation methods will be implemented. First the PEF is designed. To do that, the data ECM M_d (see Eqs. (**6.5**) and (**7.4**), below) is performed by generating an

ensemble of PE samples (as done in the experiment with Lorenz system, see the sampling procedure in Ref. [23] for more detail). As the number of elements of ECM is of order 10^{10} (for only the layer thickness component h), it is impossible to simulate a sufficient number of PE samples so that M_d would be a good estimate for the ECM. The matrix M_d will be used only as data to estimate the parameters in a parametrized ECM as follows (see Ref. [18]):

Let $M \in R^{n_h \times n_h}$ be the ECM for the layer thickness h, that is, $M = M(\mathbf{s}, s')$. One useful and efficient way to simplify the filter structure is to assume that the ECM M has a SeVHS. Assuming there exist two covariance matrices, M_v and M_h such that

$$M(s, s') = M_v(s_v, s_{v'}) \otimes M_h(s_h, s_{h'}), s_v := l, s_h := (i, j), \tag{7.2}$$

where $\otimes$ denotes the Kronecker product between two matrices [25],

$$M_v(s_v, s_{v'}) \otimes M_h(s_h, s_{h'}) = M(i, j, l; i', j', l') =$$

$$\begin{pmatrix} m_v(1,1)M_h & m_v(1,2)M_h & \dots & m_v(1, n_v)M_h \\ m_v(2,1)M_h & m_v(2,2)M_h & \dots & m_v(2, n_v)M_h \\ \dots & \dots & \dots & \dots \\ m_v(n_v,1)M_h & m_v(n_v,2)M_h & \dots & m_v(n_v, n_v)M_h \end{pmatrix} \tag{7.3}$$

The main advantage of the separability hypothesis is that the number of parameters to be estimated in the covariance matrix is reduced drastically. As a consequence, even an ensemble of PE samples with small size can serve as a large data set for estimating the unknown parameters. This results in a fast convergence of the estimation procedure. In addition, introducing the SeVHS hypothesis allows to avoid the rank deficiency problem in the estimation of the ECM. In fact, as only a few numbers of ScVs can be computed in very high-dimensional systems, approximation of the ECM M by Eq. (6.5) results in rank deficiency for M. With such an ECM, the resulting filter will probably produce worse results, not to say on instability which may occur during the filtering process.

Suppose we are given the ensemble of S-PE samples $S_\tau[L] = [\delta h_\tau^{(1)}, \dots, \delta h_\tau^{(L)}]$ which are obtained at the τ time instant by applying the sampling procedure in Ref. [23] subject to L perturbations. For $\tau = 1, \dots, T$, the ECM M_d in Eq. (4.12) is estimated as

$$M_d = \frac{1}{T} \sum_{\tau=1}^{T} M_\tau,$$

$$M_\tau := \frac{1}{L} S_\tau(L) S_\tau^T(L) \tag{7.4}$$

For the problem Eqs. (4.3)–(4.4) let us define the vector of unknown parameters in the ECM $M(s, s')$ as

$$\theta := (c_{11}, ..., c_{1n_v}, c_{21}, ..., c_{2n_v}, c_{n_v 1}, ..., c_{n_v n_v}, L_d)^T, c_{kl} := m_v(k, l). \tag{7.5}$$

where $c_{kl} := m_v(k, l)$. As to the parameter L_d, it represents the correlation length of the horizontal ECM M_h,

$$M_h(y, y') = exp(-d(y, y')/L_d), \tag{7.6}$$

$d(y, y')$ is the distance between two horizontal points $y := (i, j)$ and $y' := (i', j')$.

Considering M_d as data matrix, optimization problem for determining the vector θ looks like

$$J[\theta] = E[\Psi(M_t, \theta)] \rightarrow \min_\theta,$$
$$\Psi(M_t, \theta)] := ||M_t - M_v(s_v, s_{v'}) \otimes M_h(s_h, s_{h'})||_F^2, \tag{7.7}$$

Mention that the problem Eq. (**7.4**) is somewhat closely related to the Nearest Kronecker Product (NKP) problem [25]. In the experiment, the correlation length L_d is not estimated and is taken identical in two filters PEF and CHF, $L_d = 25$.

7.2.2. PEF: computation of gain

As to the computation of the gain, introduce the notations: at the instant k, let $x(i, j, l)$ be the value of the system state defined at the grid points (i, j, l). Let $\vec{x} = (\vec{x}_1^T, \vec{x}_2^T, ..., \vec{x}_{n_v}^T)^T$ be a vector representation for x where $\vec{x}_l$ is a vector whose components are the values of x at all the horizontal grid points (ordered in some way) at the lth, vertical layer.

Consider the ECM Eq. (**3.10**) and the observation equation (**1.2**). Represent the observation matrix H in a block-matrix form

$$H = [H_1, ..., H_{n_v}] \tag{7.8}$$

which corresponds to the vector representation $\vec{x}$, that is,

$$H\vec{x} = \sum_{v=1}^{n_v} H_v \vec{x}_v \tag{7.9}$$

Compute the gain according to Eq. (**2.4**). We have

$$MH^T = M_v \otimes M_h H^T = [\Sigma_1^T, ..., \Sigma_{n_v}^T]^T,$$
$$\Sigma_l = M_h \sum_{k=1}^{n_v} c_{lk} H_k^T = M_h G_{v,l},$$
$$MH^T = M_d G_v, M_d = \text{block diag } [M_h, ..., M_h],$$
$$G_v = [G_{v,1}^T, ..., G_{v,n_v}^T]^T, G_{v,l} := \sum_{k=1}^{n_v} c_{lk} H_k^T,$$
$$\Sigma := HMH^T + R = \sum_{k=1, l=1}^{n_v} c_{lk} H_l M_h H_k^T + R, \tag{7.10}$$

As proved in Ref. [18], in this case, the gain has the following form

$$K = M_d K_v, K_v = [K_{v,1}^T, \ldots, K_{v,n_v}^T]^T = G_v \Sigma^{-1},$$

$$K_{v,l} = G_{v,l} \Sigma^{-1}, l = 1, \ldots, n_v,$$

(7.11)

where $G_{v,l}, \Sigma$ are defined in Eq. (**7.9**).

7.2.3. Cooper-Haines filter: CHF

To see the performance of the AF, we implement also a so-called CHF. The CHF [7] is obtained from (**7.9**), (**7.10**) under three hypotheses [24]: (**H1**) the analysis error of the system output is canceled in the case of noise-free observations; (**H2**) conservation of the linear potential vorticity (PV); (**H3**) there is no correction for the velocity at the bottom layer. The AF in Ref. [24] is obtained by relaxing one or several hypotheses (**H1**)–(**H3**). From the filtering theory, the difference between the PEF and CHF is lying in the way we estimate the elements of the ECM.

For the choice of the tuning parameters in the PEF, see Ref. [24].

7.3. Numerical results

First, we run the model initialized by the climatology. This run is different from that used for modeling the sequence of true ocean states only by changing the initial state by climatology. This run is denoted as *model*. **Figure 6** shows the (spatial) averaged variance between SSH observations and that produced by the *model*. We see the error grows as time progresses, meaning instability of the numerical model wrt the perturbed initial condition. That fact signifies that the VM will have difficulties in producing high performance estimates if the assimilation window is long.
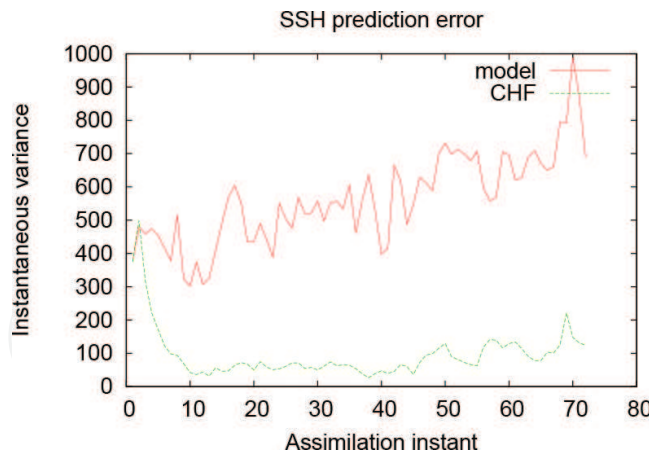

Figure 6. SSH prediction errors resulting from "model" and CHF: growing of the prediction error in the "model" signifies instability of the numerical model wrt specification of the initial system state.

Next the two filters, CHF and PEF, are implemented under the same initial condition as those carried out with the experiment *model*. It is seen from **Figure 6** that initialized by the same initial condition, and the CHF is much more efficient than the *model* in reducing the estimation error. The performance comparison between the CHF and PEF is presented in **Figure 7**. Here, the (spatially averaged variance) SSH prediction errors, resulting from two

filters CHF and PEF, are displayed. The superiority of the PEF over the CHF is undoubted. It is clear from **Figure 7** that the PEF is capable of producing the better estimates, with lower error level, along all assimilation period. On the other hand, if the estimation error in CHF decreases continuously at the beginning of the assimilation and is stabilized more or less after (during the interval $k \in (10 : 45)$), it becomes to increase considerably at the end of the assimilation period. It means that the PEF is more efficient than the CHF and the fact that the ECM in the PEF, constructed on the basis of the S-PE samples, has the effect to stabilize its behavior.
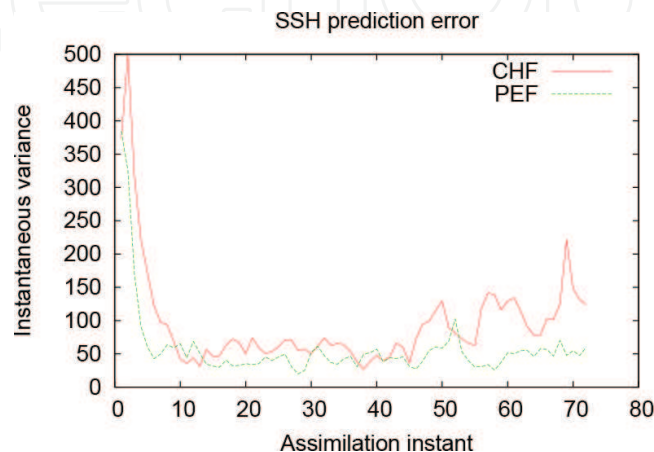

Figure 7. SSH prediction errors resulting from two filters: CHF and PEF. The PEF is capable of producing the estimates with lower estimation errors and is stable along all assimilation period, whereas the CHF has a difficulty to maintain the same performance at the end of the assimilation period. The PEF is much better than the CHF in providing better estimates for the system states.

To see the effect of adaptation, **Figure 8** displays the filtered errors for the u-velocity component estimates at the surface, produced by the PEF and APEF, respectively. The APEF is an AF, which is an adaptive version of the PEF. Here, the tuning parameters are optimized by the SPSA method. From the computational point of view, the SPSA requires much less time
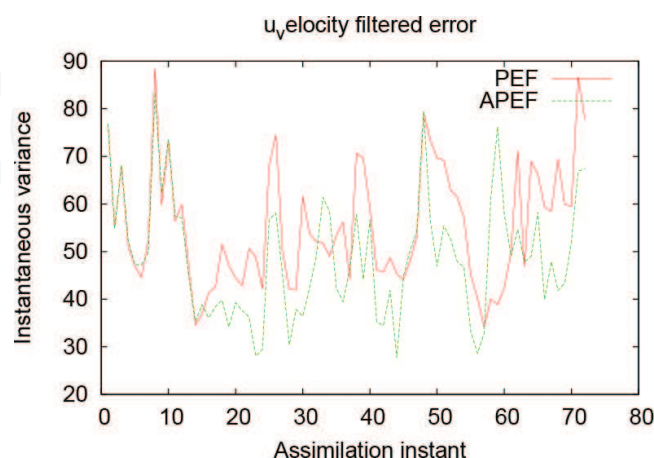

Figure 8. Filtered errors for the u-velocity component estimate, resulting from PEF and APEF (AF based on PEF). Optimization is performed by SPSA. By tuning the parameters in the filter gain, one can improve considerably the performance of the PEF.

integration and memory storage compared with the traditional AE method. At each assimilation instant, we have to make only two integrations of the MICOM for approximating the gradient vector. From **Figure 8**, one sees that the adaptation allows to reduce significantly the estimation errors produced by the PEF.

8. Conclusions

In this chapter, a comparison study on the performance of the AF with other existing methods is presented in the context of its application to data assimilation problems in high-dimensional numerical models. As it is seen, in comparison with the standard VM, the AF is much simpler to implement and produces better estimates. The advantages of the AF over other methods such as EKF, CHF are also demonstrated. The principal reason for high performance of the AF is lying in the choice of innovation representation for the initial input-output system and selection of pertinent gain parameters as control variables to minimize the MSE of the innovation process. If in the VM, the choice of the structure for the initial state is the most important thing to do (but that is insufficient for guaranteeing its high performance), in the AF, however, the initial state has a little impact on the performance of the AF. This happens because the AF is selected as optimal in the class of stable filters, and as a consequence, the error in the initial estimate is attenuated during assimilation process. In contrary, in the VM, the error in the specification of the initial guess is amplified during assimilation if the numerical model is unstable.

In conclusion, it is important to emphasize that the AF approach, presented in this chapter, is consolidated by exploiting the following road map: (i) generate data ECM from S-PE samples which grow in the directions of dominant Schur vectors; (ii) select a parametrized structure for the ECM under the hypothesis SeVHS; (iii) make the choice of tuning parameters in the gain by minimizing the distance between the data ECM and that having the SeHVS structure; (iv) adjust the unknown parameters in the gain in order to minimize the PE error of the system output by applying the SPSA algorithm.

There are a wide variety of engineering problems to which the AF is applicable and that could be worthy of further study. Depending on particular problems, undoubtedly, the other modifications would be helpful to improve the filter performance, to simplify its implementation. But the main features of the AF, presented in this chapter, remain as the key points to follow in order to preserve a high performance of the AF.

Author details

Hong Son Hoang* and Remy Baraille

*Address all correspondence to: hhoang@shom.fr

SHOM/HOM/REC, Toulouse, France

References

[1] Bucy R.S. and Joseph D.P.: Filtering for Stochastic Processes, with Applications to Guidance. New York: Wiley; 1968.

[2] Maybeck P.S.: Stochastic Models, Estimation, and Control. Vol. 1, Academic Press; 1979.

[3] Kalman R.E.: A new approach to linear filtering and prediction problems. Trans ASME J. Basic Eng. 1960; 82: 35–45.

[4] Todling R. and Cohn S.E.: Suboptimal schemes for atmospheric data assimilation based on the Kalman filter. Monthly Weather Rev. 1994; **122**: 2530–2557.

[5] Ghil M. and Manalotte-Rizzoli P.: Data assimilation in meteorology and oceanography. Adv Geophys. 1991; **33**: 141–266.

[6] Simon D.: Optimal State Estimation. Hoboken, NJ: John Wiley and Sons; 2006. ISBN 978-0-471-70858-2.

[7] Cooper M. and Haines K.: Altimetric assimilation with water property conservation. J Geophys Res. 1996; **101**: 1059–1077.

[8] Nocedal J. and Wright S.: Numerical Optimization. New York: Springer-Verlag; 1999. ISBN 0-387-9873-2.

[9] Talagrand O.: Data assimilation problems. In: Raschke E. and Jacob D., editors. Energy and Water Cycles in the Climate System. NATO ASI Series I, Vol. **5**; 1993, pp. 187–213.

[10] Hoang H.S., De Mey P., Talagrand O. and Baraille R.: A new reduced-order adaptive filter for state estimation in high dimensional systems. Automatica. 1997; **33**: 1475–1498.

[11] Gevers M. and Kailath T.: An innovations approach to least-squares estimation, Pt. VI: discrete-time innovations representations and recursive estimation. IEEE Trans Autom Control. 1973; 18(6), **12**: 588–600.

[12] Hoang H.S., Talagrand O. and Baraille R.: On the design of a stable adaptive filter for high dimensional systems. Automatica. 2001; **37**, 341–359.

[13] Baram Y. and Kalit G.: Order reduction in linear state estimation under performance constraints. IEEE Trans Autom Control. 1987; AC-32, **11**: 983–989.

[14] Spall J. C. Accelerated second-order stochastic optimization using only function measurements. In: Proceedings of 36th IEEE Conference on Decision and Control, 14171424; 1997.

[15] Hoang H.S. and Baraille R.: Stochastic simultaneous perturbation as powerful method for state and parameter estimation in high dimensional systems. In: Albert R. Baswell, editors. Advances in Mathematics Research. Vol 20, Nova Science Publishers.; 2015, pp. 117–148.

[16] Evensen G.: The ensemble Kalman filter: theoretical formulation and practical implementation. Ocean Dyn. 2003; **53**: 343–367.

[17] Hamill T.M.: Ensemble-based atmospheric data assimilation. In: Palmer T., editor. Predictability of Weather and Climate. Cambridge Press; 2006; pp. 124–156.

[18] Hoang H.S. and Baraille R.: A low cost filter design for state and parameter estimation in very high dimensional systems. In: Proceedings of the 19th IFAC Congress; August, 2014, pp. 3256–3261.

[19] Daley R.: The effect of serially correlated observation and model error on atmospheric data assimilation. Monthly Weather Rev. 1992; **120**: 165–177

[20] Baraille R.: Modélisation numérique avec le modèle HYCOM au SHOM. At "http://mathocean.math.cnrs.fr/presentations/Baraille.pdf"

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

[22] Kivman G.A.: Sequential parameter estimation for stochastic systems. Nonlinear Process Geophys. 2003; **10**: 253–259. doi:10.5194/npg-10-253-2003

[23] Hoang H.S. and Baraille R.: Prediction error sampling procedure based on dominant Schur decomposition. Application to state estimation in high dimensional oceanic model. J Appl Math Comput. 2011; **12**: 3689–3709.

[24] Hoang H.S., Baraille R. and Talagrand O.: On an adaptive filter for altimetric data assimilation and its application to a primitive equation model MICOM. Tellus, 57A, 2005, **2**: 153–170.

[25] Golub G.H. and Van Loan C.F.: Matrix Computations. Johns Hopkins University Press; 1996.