Note on implicit filters

Florimond Guéniat,[1] Lionel Mathelin,[2] and Yussuf Hussaini[1]
[1)] *Department of mathematics, Florida State Univ., 32306-4510 Tallahassee, Fl, US*
[2)] *LIMSI-CNRS, 91403 Orsay cedex, France*

(Dated: 29 July 2015)

Keywords: Data assimilation, Model identification

I. INTRODUCTION

In this article, the numerical cost and the accuracy of Implicit Particle Filter (IPF) are quantified, using Burgers's equation and shallow water equations as scenarios. Identifying the uncertities in a data assimilation problem is specifically addressed.

IPF aims at estimating or the state, or the initial conditions or or some parameters of a system, by coupling a model and observations. A noise component can be present in both the observations and the model. Such an aim is of importance in many domains, most notably meteorologyref , where initial conditions compatible with observations leads to accurate predictions. If not the case, as expected from a chaotic system, the accuracy of the forecast can drop significantly. Seismographyref is an other field of importance for data assimilation, *e.g.*, to locate epicenter. It has been also successfully applied to nuclear fusion[1] and agronomy [2].

When the model is linear, and the noise Gaussian, the most suited class of methods are the Kalman filters. Uncertaintities are directly related to the parameters of the Gaussian, which are totally determined by the Kalman filter. The generalization of Kalman filters, ensemble Kalman filters,[1], can deal with non-linear model by using replica (particles) of the system, to represent the prior density of solutions. Nevertheless, it performs poorly when the errors and noises are not Gaussian,[2]. 4D-Var and cost function-minimization techniques are very popular methods, giving usually accurate results, but they fail at quantifying the uncertainties in the results, with respect to the noise in the model and observations. The sensitivity analysis cannot be considered as accurate, as soon as the noise cannot be considered as small cite Plessix, review . Moreover, results are actually biased if the posterior pdf is multimodal,[3].

Indeed, IPF have been successfully used for data assimilation,[4], system identification,[5], and parameter estimation. It is particularly relevant for multimodal pdf ; it provides an unbiased estimate of the solutions,[6]. As for Markov-Chain Monte Carlo method, IPF relies on a set of particles for the quantification of the conditional pdf of the system. The main difference lies in the fact that only a few, weighted hence carefully chosen particles are considered. Identifying these particles is done through the maximisation of a tailor-constructed probability. It ensures that each particle is associated with relevant informations of the pdf. The moments of the estimated pdf, as in standard MCMC methods, might be used for quantifying the uncertainties.

In this article, the precision and accuracy of IPF is qualitatively shown, and a roadmap for tackling the computational costs issue is proposed.

II. 4D-VAR

A. Preliminaries

Let's consider the state vector $\boldsymbol{x}$ of a system of interest, evolving on the $n_{\mathcal{D}}$ manifold $\mathcal{D}$. This field is modelized through the set of equations f:

$$f(\boldsymbol{x}, \dot{\boldsymbol{x}}, q, t) = 0. \tag{1}$$

The model might not be perfect. Flaws can come, from instance, from uncertainties in the model parameters q, or on some initial conditions. Consequently, a functional J is introduced, in order to fit the model by taking in account

---

[1] Cacuci, D. G., & Ionescu-Bujor, M. (2010). Best-Estimate model calibration and prediction through experimental data assimilation-I: Mathematical framework. Nuclear science and engineering, 165(1), 18-44.

[2] Guerif, M., & Duke, C. L. (2000). Adjustment procedures of a crop model to the site specific characteristics of soil and crop using remote sensing data assimilation. Agriculture, ecosystems & environment, 81(1), 57-69.

observations at hand:

$$J(\boldsymbol{x}, q) = \int_0^T j(\boldsymbol{x}, q, t)dt. \tag{2}$$

For instance, one may want to minimize the discrepancy between the model predictions and some observations y:

$$j(\boldsymbol{x}, q, t) = |h(\boldsymbol{x}, q, t) - y(t)|^2$$

In the data assimilation context, the initial conditions are actually derived from the array of parameters $q \in \mathbb{R}^{n_q}$, with a relation $g : \mathcal{D} \times \mathbb{R}^{n_q} \to \mathbb{R}$:

$$g(\boldsymbol{x}(0), q) = 0.$$

Minimizing J gives the best initial conditions, in the sense that Eq. (1) will produce fields as compatible as possible with the observations.

Efficient and reliable methods for minimizing, such as the BFGSref algorithm, need the gradient of the functional with respect to the parameters. When $n_{\mathcal{D}}$ is large, approximating the gradient with finite differences become impractical. An powerful alternative to compute the gradient is to solve the adjoint equationref .

B. Computing the gradient of the cost functional

Identifying the minimum of the J relies on the gradient of the functional with respect to the parameters: $D_q J = \frac{D\,J}{Dq}$. For that, one can introduce the Lagrangian $\mathcal{L}$, associated with the two lagrangian parameters λ and μ:

$$\mathcal{L}\left(\boldsymbol{x}, \dot{\boldsymbol{x}}, q, \lambda, \mu\right) = \int_0^T \Big[j(\boldsymbol{x}, q, t) + \lambda^\dagger f(\boldsymbol{x}, \dot{\boldsymbol{x}}, q, t) \Big] dt + \mu^\dagger g(\boldsymbol{x}(0), q) \tag{3}$$

where † is the transpose operator. Naturally, both $\boldsymbol{x}$ and $\dot{\boldsymbol{x}}$ are considered as variables. As f and g are null by construction, λ and μ can be designed specifically to alleviate the computations.

The gradiant $D_q\mathcal{L} = \frac{D\mathcal{L}}{Dq}$ of $\mathcal{L}$ is:

$$\begin{aligned} D_q\mathcal{L}\left(\boldsymbol{x}, \dot{\boldsymbol{x}}, q, \lambda, \mu\right) = \int_0^T \Big[& \partial_{\boldsymbol{x}} j(\boldsymbol{x}, q, t) d_q\boldsymbol{x} + \partial_q j(\boldsymbol{x}, q, t) + \\ & \lambda^\dagger \partial_{\boldsymbol{x}} f(\boldsymbol{x}, \dot{\boldsymbol{x}}, q, t) d_q\boldsymbol{x} + \lambda^\dagger \partial_{\dot{\boldsymbol{x}}} f(\boldsymbol{x}, \dot{\boldsymbol{x}}, q, t) d_q\dot{\boldsymbol{x}} + \lambda^\dagger \partial_q f(\boldsymbol{x}, \dot{\boldsymbol{x}}, q, t) \Big] dt + \\ & \mu^\dagger \partial_{\boldsymbol{x}(0)} g(\boldsymbol{x}(0), q) dq_{\boldsymbol{x}(0)} + \mu^\dagger \partial_q g(\boldsymbol{x}(0), q). \end{aligned} \tag{4}$$

As a matter of facts, $D_q\mathcal{L} = D_q J$.

The term in $d_{\dot{\boldsymbol{x}}}$ cannot be easily estimated. An integration by parts gives:

$$\int_0^T \lambda^\dagger \partial_{\dot{\boldsymbol{x}}} f(\boldsymbol{x}, \dot{\boldsymbol{x}}, q, t) d_q\dot{\boldsymbol{x}} dt = \left[\lambda^\dagger \partial_{\dot{\boldsymbol{x}}} f(\boldsymbol{x}, \dot{\boldsymbol{x}}, q, t) d_q\boldsymbol{x}\right]_0^T - \int_0^T \left\{ \dot{\lambda}^\dagger \partial_{\dot{\boldsymbol{x}}} f(\boldsymbol{x}, \dot{\boldsymbol{x}}, q, t) + \lambda^\dagger d_t \partial_{\dot{\boldsymbol{x}}} f(\boldsymbol{x}, \dot{\boldsymbol{x}}, q, t) \right\} d_q\boldsymbol{x} dt$$

The term associated with $d_{\dot{\boldsymbol{x}}}$ can now be replaced in Eq. (4). Ordering terms in this equation leads to:

$$\begin{aligned} & D_q\mathcal{L}\left(\boldsymbol{x}, \dot{\boldsymbol{x}}, q, \lambda, \mu\right) = \\ & \int_0^T \Big[\left(\partial_{\boldsymbol{x}} j(\boldsymbol{x}, q, t) + \lambda^\dagger \partial_{\boldsymbol{x}} f(\boldsymbol{x}, \dot{\boldsymbol{x}}, q, t) - \left\{ \dot{\lambda}^\dagger \partial_{\dot{\boldsymbol{x}}} f(\boldsymbol{x}, \dot{\boldsymbol{x}}, q, t) + \lambda^\dagger d_t \partial_{\dot{\boldsymbol{x}}} f(\boldsymbol{x}, \dot{\boldsymbol{x}}, q, t) \right\} \right) d_q\boldsymbol{x} + \\ & \partial_q j(\boldsymbol{x}, q, t) + \lambda^\dagger \partial_q f(\boldsymbol{x}, \dot{\boldsymbol{x}}, q, t) \Big] dt + \\ & \left\{ \mu^\dagger \partial_{\boldsymbol{x}} g(\boldsymbol{x}, q) - \lambda^\dagger \partial_{\dot{\boldsymbol{x}}} f(\boldsymbol{x}, \dot{\boldsymbol{x}}, q, t) \right\} |_0 dq_{\boldsymbol{x}(0)} + \mu^\dagger \partial_q g(\boldsymbol{x}(0), q) + \left\{ \lambda^\dagger \partial_{\dot{\boldsymbol{x}}} f(\boldsymbol{x}, \dot{\boldsymbol{x}}, q, t) \right\} |_T d_q\boldsymbol{x}(T) \end{aligned} \tag{5}$$

Proper choices for λ and μ allow to simplify the expression of $D_q\mathcal{L}$. The choice of $\lambda(T) = 0$ actually nullifies the term $\left\{\lambda^\dagger \partial_{\dot{\boldsymbol{x}}} f(\boldsymbol{x}, \dot{\boldsymbol{x}}, q, t)\right\}|_T d_q \boldsymbol{x}(T)$. Then, λ can be chosen as the solution of the so-called adjoint equation:

$$\left(\partial_{\boldsymbol{x}} j(\boldsymbol{x}, q, t) + \lambda^\dagger \partial_{\boldsymbol{x}} f(\boldsymbol{x}, \dot{\boldsymbol{x}}, q, t) - \left\{\dot{\lambda}^\dagger \partial_{\dot{\boldsymbol{x}}} f(\boldsymbol{x}, \dot{\boldsymbol{x}}, q, t) + \lambda^\dagger d_t \partial_{\dot{\boldsymbol{x}}} f(\boldsymbol{x}, \dot{\boldsymbol{x}}, q, t)\right\}\right) = 0$$

integrated in backward time. Finally, the Lagrange parameter μ is set so it nullifies the component associated with $dq_{\boldsymbol{x}(0)}$:

$$\left\{\mu^\dagger \partial_{\boldsymbol{x}} g(\boldsymbol{x}, q) - \lambda^\dagger \partial_{\dot{\boldsymbol{x}}} f(\boldsymbol{x}, \dot{\boldsymbol{x}}, q, t)\right\}|_0 = 0.$$

Then, computing $D_q\mathcal{L}$, hence $D_q J$, is achieved by the integration of:

$$D_q J(\boldsymbol{x}, q) = \int_0^T \left[\partial_q j(\boldsymbol{x}, q, t) + \lambda^\dagger \partial_q f(\boldsymbol{x}, \dot{\boldsymbol{x}}, q, t)\right] dt + \mu^\dagger \partial_q g(\boldsymbol{x}(0), q). \tag{6}$$

III. SYSTEM

A. Burgers' equation

The Burgers' equation stochastic part ? is:

$$\partial_t \boldsymbol{x} + \frac{1}{2}\partial_z \boldsymbol{x}^2 - \nu \partial^2_{zz} \boldsymbol{x} = 0.$$

$\boldsymbol{x}$ evolves on an $n_{\mathcal{D}}$-dimensional manifold $\mathcal{D}$. We consider that it has been spatially discretized. The field $\boldsymbol{x}$ can hence be modelized by $\hat{\boldsymbol{x}}$ through a set of ordinary differential equations $f(\hat{\boldsymbol{x}}, \dot{\hat{\boldsymbol{x}}}, q, t) = 0$.

$$\dot{\hat{\boldsymbol{x}}} + L(\hat{\boldsymbol{x}}, q, t) + NL(\hat{\boldsymbol{x}}, q, t) = 0 \equiv f(\hat{\boldsymbol{x}}, \dot{\hat{\boldsymbol{x}}}, q, t),$$

where $q \in \mathbb{R}^q$ are the parameters of the model. The solution is discretized with n_x points linearly spaced: $\hat{\boldsymbol{x}} \in \mathcal{R}^{n_x}$, following an implicit scheme, in order to be solved by Newton iterations. The $\hat{}$ symbol is dropped in the following for the sake of readability.

$$\frac{\boldsymbol{x}_i^{n+1} - \boldsymbol{x}_i^n}{\Delta t} + \boldsymbol{x}_i^{n+1} \frac{\boldsymbol{x}_i^{n+1} - \boldsymbol{x}_{i-1}^{n+1}}{\Delta z} - \nu \frac{\boldsymbol{x}_{i+1}^{n+1} - 2\boldsymbol{x}_i^{n+1} + \boldsymbol{x}_{i-1}^{n+1}}{\Delta z^2} = 0 \tag{7}$$

More information can be found in Sec. A

B. Shallow water quations

We consider the SWE equations (in its Ito form ?):

$$\begin{aligned} \partial_t u &= -u\partial_x u - v\partial_y u - f\, v - g\partial_x h - b\, u \\ \partial_t v &= -u\partial_x v - v\partial_y v + f\, u - g\partial_y h - b\, v \\ \partial_t h &= -\partial_x\left((H + h)\, u\right) - \partial_y\left((H + h)\, v\right), \end{aligned} \tag{8}$$

Ito ? 0where u and v are the surface velocity, h (resp. H) is the deviation of height (resp. mean height) from the botton. f is the Coriolis coefficient and ν is the viscous drag coefficient. g is the acceleration due to gravity. This equation is spatially discretized.

IV. PARTICLE METHODS AND IMPORTANCE SAMPLING

A. Preliminaries

Consider a stochastic dynamical system, in the Ito formalism, evolving on an $n_{\mathcal{D}}$-dimensional manifold $\mathcal{D}$:

$$d\boldsymbol{x}(t) = \mathfrak{f}(\boldsymbol{x}(t))\, dt + \boldsymbol{G} dW, \qquad \boldsymbol{x} \in \mathcal{D}, \tag{9}$$

with x the state of the system and $\mathfrak{f} : \mathcal{D} \to \mathcal{D}$ the flow operator. dW is a an $n_{\mathcal{D}}$-dimensional Brownian motion (Wienner process) and $\boldsymbol{G}$ a diffusion $n_{\mathcal{D}} \times n_{\mathcal{D}}$ matrix. With a proper time scheme[3] associated with a time step δt, Eq. (9) becomes:

$$\boldsymbol{x}\left(t_0 + (n+1)\,\delta t\right) = \mathfrak{F}\left(\boldsymbol{x}\left(t_0 + n\delta t\right)\right) + \boldsymbol{G}\sqrt{\delta t}E, \tag{10}$$

where E is an $n_{\mathcal{D}}$-dimensional realization of the Wiener process. This model is to be understand as space and time discretization of a set of partial differential equations.

Let $h : \mathcal{D} \to \mathbb{R}^{n_p}$ be a sensor function, so an observation on the system is:

$$y(t) = h\left(x, t\right) + \sqrt{\boldsymbol{S}}Y, \tag{11}$$

Y is an n_p-dimensional, standard-normal random variable, and $\sqrt{\boldsymbol{S}}$ a covariance matrix. Let also consider, without any loss of generality, that $\boldsymbol{G} = g\boldsymbol{Id}$ and $\sqrt{\boldsymbol{S}} = \sqrt{s}\boldsymbol{Id}$. In the following, the time will be discretized accordingly to δt. For any variable $\star$, $\star\left(t_0 + n\delta t\right)$ will be shortened as $\star_n$.

B. Probabilities derived from the Ito formulation

The fact that noises in Eqs. (10) and (11) are Gaussian does not mean that the probability density function (pdf) associated with the state of the system is Gaussian, as it will be shown in the following.

The probability that the $n_{\mathcal{D}}$-dimensional random variable X^{n+1} coincide with a solution of the system described in Eq. (9), at time $n+1$ and with the state $\boldsymbol{x}_n$ as initial condition, is given by:

$$P_{int}^{\boldsymbol{x}_n, n+1}\left(X^{n+1}\right) = P\left(X^{n+1} | \boldsymbol{x}_n\right).$$

This probability only depends on the system equations, *i.e.* Eqs. (9) and (10). When dealing with a Wiener process, the conditional probability is:

$$P_{int}^{\boldsymbol{x}_n, n+1}\left(X^{n+1}\right) \propto \exp\left(-\left\|X^{n+1} - \mathfrak{F}\left(\boldsymbol{x}_n\right)\right\|_2^2 / 2\delta t g^2\right). \tag{12}$$

Note that if the noise is not Gaussian (*i.e.* not described by a Wiener process), it only affects the right side of Eq. (12).

On an other hand, the probability that the observation of the variable X^{n+1} is compatible with the observation of the system is given by:

$$P_{obs}^{n+1}\left(X^{n+1}\right) = P\left(h\left(X^{n+1}\right) | y_{n+1}\right).$$

Again, this probability only depends on the sensor function, see Eq. (11):

$$P_{obs}^{n+1}\left(X^{n+1}\right) \propto \exp\left(-\left\|h\left(X^{n+1}\right) - y_{n+1}\right\|_2^2 / 2s\right). \tag{13}$$

From the Bayes' theorem, by combining Eq. (12) and Eq. (13), the probability $P_{SI}^{\boldsymbol{x}_n, n+1}$ that X^{n+1} is a possible outcome from the previous state $\boldsymbol{x}_n$, given the observation y_{n+1}, is:

$$P_{SI}^{\boldsymbol{x}_n, n+1}\left(X^{n+1}\right) \propto P_{int}^{\boldsymbol{x}_n, n+1}\left(X^{n+1}\right) P_{obs}^{n+1}\left(X^{n+1}\right) \tag{14}$$

From this writing, as a product, the pdf of $P_{SI}^{\boldsymbol{x}_n, n+1}$ is not normal.

Generalization is straightforward. Having several observations $\{y_{n+i}\}_{\{i\in\{1,\ldots,r\}\}}$ available at hand, between time t_n and t_{n+r}, is common. Consequently, Eq. (14) becomes:

$$P_{SI}^{\boldsymbol{x}_n, n+r}\left(\boldsymbol{X}_{n+1}^{n+r}\right) \propto \prod_{i=1}^{r} P_{int}^{X^{n+i-1}, n+i}\left(X^{n+i}\right) \prod_{i\in\{1,\ldots,r\}} P_{obs}^{n_i}\left(X^{n+i}\right), \tag{15}$$

where $\boldsymbol{X}_{n+1}^{n+r} = \left\{X^{n+1}, \ldots, X^{n+r}\right\}$ is now a $r \times n_{\mathcal{D}}$-dimensional random variable, coinciding with a potential trajectory of the state between $n+1$ and $n+r$. and with $X^n = \boldsymbol{x}_n$.

---

[3] Kloeden, P. E., Platen, E. (1999). Numerical solution of stochastic differential equations. Berlin: Springer, Klauder Petersen scheme

The probability P^n_{DA} that the system is in a given state X^n at the given time n can be derived from Eq. (15):

$$P^n_{DA}\left(\boldsymbol{X}^{n+r}_n\right) \propto \prod_{i=1}^{r} P_{int}^{X^{n+i-1},n+i}\left(X^{n+i}\right) \prod_{i\in\{1,\dots,r\}} P^{n_i}_{obs}\left(X^{n+i}\right), \tag{16}$$

with $\boldsymbol{X}^{n+r}_{n+1} = \{X^n, \dots, X^{n+r}\}$.

Data assimilation, as system identification, aims at identifying the state of the considered system, at any desired time. Consequently, $\chi \equiv \text{argmax}\left(P^n_{DA}\right)$ (resp. $\chi \equiv \text{argmax} P_{SI}^{\boldsymbol{x}_n,n+r}$) corresponds to the best solution of the data assimilation (resp. system identification) problem,[4–6]. The form of Eqs. (12) and (13) implies that minimizing $F_{SI} = -\log P_{SI}^{\boldsymbol{x}_n,n+r}$ and $F_{DA} = -\log P^n_{DA}$ is numerically preferable.

In the following, the notation P (resp. F) will refer indistinctly to $P_{SI}^{\boldsymbol{x}_n,n+r}$ or P^n_{DA} (resp. F_{SI} or F_{DA}), when no confusion is possible. Similarly, X will refer to the considered random variable, and its dimension will be short-named $n_{\mathcal{D}}$.

Identifying the maximum can be, alternatively, efficiently done by 4D-VAR, refs , when the distribution associated with P_{DA} is known to be mono-modal[4].

For state identification, as for data assimilation, the initial conditions $\boldsymbol{x}_n$ in Eq. (15) are often unknown. This problem is usually tackled by considering m particles, *i.e.* replicas of the system[5]. For each particle i, an initial condition $\boldsymbol{x}^i_n$ is guessed. Maxima are computed from Eq. (15), for all the initial guesses.

By the law of large numbers, it allows to estimate the moments of the pdf, that is, for system identification, the mean state $\boldsymbol{x}_n$, by averaging the results.

Estimating the moments of the a given pdf might need a consequent number of particlesrefs , and both the storage and the identification of samples from that pdf might be near to impossible. Actually, when strictly following the law of large numbers, all the particles has the same importance. An option is, instead of drawing random particles, to identify particles more representative of the probability density function (pdf) associated with P. Hence, less particles are needed for an accurate estimation of the pdf properties.

C. Importance sampling

Intuitively, a particle carries more information on the probability, as soon as it is associated with a high probability in Eq. (14) (resp. (15) or (16)). Moreover, estimations of the pdf properties rely on drawing samples from the pdf. A sample has to be a possible outcome of the physical system described by Eq. (9) for being meaningful.

Importance sampling,[4,5], ref is allows to tackle both these issues. The key is to consider an easy-to-sample pdf p. Some samples X_i are drawn from it. These samples are mapped with the the actually hard-to-sample pdf P. It also provides a quantification of the importance of each sample with respect to the pdf, namely the weight of each particle, so the estimation of the moments are not biased.

The weights of X_i are:

$$w\left(X_i\right) = \frac{P\left(X_i\right)}{p\left(X_i\right)}. \tag{17}$$

For being efficient, it is necessary that the maxima of p and P coincide. If not, some particles will be associated with low weights, and are hence of a low importance. Because of potential storage limitations and computational costs, such particles are to be avoided, and the objective is to identify particles with comparable weights.

In order to construct a p density, let consider ξ, an intermediate random variable. ξ is drawn from a $n_{\mathcal{D}}$-dimensional Gaussian pdf. Such a variable is, of course, far from being a possible outcome of the physical system. It will be used as an intermediate for identifying a high probability sample X, solution of the algebraic equation,[5]:

$$F\left(X\right) - \Phi_F = \xi^t\xi/2, \tag{18}$$

where $\Phi_F = F\left(\chi\right)$ is the given minimum of F, identified for instance with a 4D-VAR method.

The sample X has then an high probability to remains close to χ, as ξ is drawn for a 0 mean pdf.

---

[4] if note, the estimation might be biased. Weir 2013

[5] *e.g.*, in Markov Chain Monte Carlo simulations

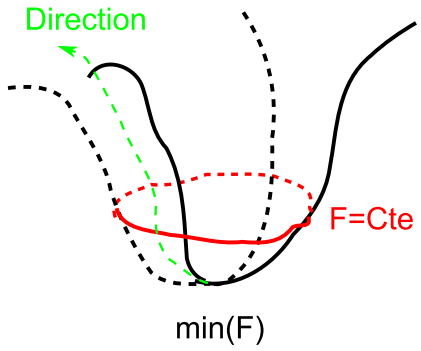

FIG. 1. Illustration of the methodology for drawing samples. The red curve corresponds to iso-values of $F(X) = Fmin+1/2\xi^T\xi$. The green curve illustrate the random direction associated with ξ. The sample is found at the intersection of these two curves.

In order to identify the weight from Eq. (17), one has to compute the probability $p_X(X)$ of drawing X from Eq. (18). By the substitution rule, p_X is related to the probability $P_\xi(\xi)$ of drawing the sample ξ by:

$$p_X(X)\left|\det\frac{\partial X}{\partial \xi}\right| = P_\xi(\xi).$$

Consequently, with Eq. (17), the weight associated to the sample X is:

$$w(X) = \frac{P(X)}{P_\xi(\xi)/\left|\det\frac{\partial X}{\partial \xi}\right|}. \tag{19}$$

D. Random map

Solving Eq. (18) is a necessary step. For that, Morzfeld *et al.*[5] proposed to use the following ansatz:

$$X = X(\lambda) = \chi + \lambda L^t\xi/\sqrt{\xi^t\xi}, \tag{20}$$

where λ is a real parameter, and L^t is the Choleski decomposition of H^{-1}, H being the Hessian of F, evaluated in χ. It minimizes the dispersion of the jacobian $\left|\det\frac{\partial X}{\partial \xi}\right|$ when considering different particles, and hence the dispersion of weights,[5].

A BFGS-algorithm is then used for identifying the λ minimizing Eq. (18).

Computing the weight needs the evaluation of the jacobian of the maps between the reference sample X and the random variable ξ. Following[5], let be $\rho = \sqrt{\xi^t\xi}$.

Differentiating Eq. (20) leads to:

$$\begin{aligned}\frac{\partial X}{\partial \xi} &= \frac{\partial}{\partial \xi}\lambda L^t\xi/\sqrt{\xi^t\xi} \\ &= L^t\left(\frac{\partial \lambda}{\partial \xi}\xi/\sqrt{\xi^t\xi} + \lambda\frac{\partial}{\partial \xi}\xi/\sqrt{\xi^t\xi}\right)\end{aligned} \tag{21}$$

V. RESULTS

A. Burgers' equation

To illustrate the methodology discussed above, we now consider the Burgers' equation presented is Sec. III A. The objective is the identification of the initial conditions, knowing only some observations. The accuracy and the computational costs are estimated. Three main parameters are allowed to variate: the noises in the model g, the noises in the observations s, and finally the dimensions (spatial ? temp?).

1. accuracy

- noise in model
- noise in observation
- dimensions

2. Computational costs

- noise in model
- noise in observation
- dimensions

BIBLIOGRAPHY

[1]G. Evensen, *Data assimilation: the ensemble Kalman filter*. Springer Science & Business Media, 2009.
[2]R. Miller, E. Carter, and S. Blue, "Data assimilation into nonlinear stochastic models," *Tellus A*, vol. 51, no. 2, pp. 167–194, 1999.
[3]F. Rabier and P. Courtier, "Four-dimensional assimilation in the presence of baroclinic instability," *Quarterly Journal of the Royal Meteorological Society*, vol. 118, no. 506, pp. 649–672, 1992.
[4]A. Chorin, M. Morzfeld, and X. Tu, "Implicit particle filters for data assimilation," *Communications in Applied Mathematics and Computational Science*, vol. 5, no. 2, pp. 221–240, 2010.
[5]M. Morzfeld, X. Tu, E. Atkins, and A. Chorin, "A random map implementation of implicit filters," *Journal of Computational Physics*, vol. 231, no. 4, pp. 2049–2066, 2012.
[6]E. Atkins, M. Morzfeld, and A. Chorin, "Implicit particle methods and their connection with variational data assimilation.," *Monthly Weather Review*, vol. 141, no. 6, 2013.
[7]E. Lorenz, "Deterministic nonperiodic flow," *J. Atmos. Sci.*, vol. 20, no. 2, pp. 130–141, 1963.
[8]T. El Moselhy and Y. Marzouk, "Bayesian inference with optimal maps," *Journal of Computational Physics*, vol. 231, no. 23, pp. 7815–7850, 2012.

Appendix A: Burgers' equation and adjoint equations

We recall here the expression from Eq. (7):

$$\frac{\boldsymbol{x}_i^{n+1}-\boldsymbol{x}_i^n}{\Delta t}+\boldsymbol{x}_i^{n+1}\frac{\boldsymbol{x}_i^{n+1}-\boldsymbol{x}_{i-1}^{n+1}}{\Delta z}-\nu\frac{\boldsymbol{x}_{i+1}^{n+1}-2\boldsymbol{x}_i^{n+1}+\boldsymbol{x}_{i-1}^{n+1}}{\Delta z^2}=0$$

The ith component of f is then:

$$f_i\left(\boldsymbol{x},\dot{\boldsymbol{x}},q,t\right)=\dot{\boldsymbol{x}}_i+\boldsymbol{x}_i\frac{\boldsymbol{x}_i-\boldsymbol{x}_{i-1}}{\Delta z}-\nu\frac{\boldsymbol{x}_{i+1}-2\boldsymbol{x}_i+\boldsymbol{x}_{i-1}}{\Delta z^2}.$$

Consequently, the partial derivatives with respect to $\boldsymbol{x}$ are:

$$\partial_{\boldsymbol{x}_j}f_i\left(\boldsymbol{x},\dot{\boldsymbol{x}},q,t\right)=\frac{1}{\Delta z}\left(2\boldsymbol{x}_i\delta_{i,j}-\delta_{i-1,j}\right)+\frac{1}{\Delta z^2}\left(\delta_{i+1,j}-2\delta_{i,j}+\delta_{i+1,j}\right).$$

The partial derivative with respect to $\dot{\boldsymbol{x}}$ is:

$$\partial_{\dot{\boldsymbol{x}}}f\left(\boldsymbol{x},\dot{\boldsymbol{x}},q,t\right)=Id,$$

and then

$$dt\partial_{\dot{\boldsymbol{x}}}f\left(\boldsymbol{x},\dot{\boldsymbol{x}},q,t\right)=0.$$

The cost function is associated with:

$$j\left(\boldsymbol{x},q,t\right)=||\boldsymbol{x}-y(t)||^2.$$

Hence, the partial derivatives with respect to q and $\boldsymbol{x}$ are:

$$\partial_q j\left(\boldsymbol{x}, q, t\right) = 0,$$

$$\partial_{\boldsymbol{x}} j\left(\boldsymbol{x}, q, t\right) = 2 * \left(\boldsymbol{x} - y(t)\right).$$

The parameters q are actually the initial conditions. Hence, n_q is equal to $n_{\mathcal{D}}$. As a matter of fact, g is:

$$g(\boldsymbol{x}(0), q) = \boldsymbol{x}(0) - q = 0.$$

Then, the partial derivatives of g follow:

$$\partial_q g(\boldsymbol{x}(0), q) = -Id,$$

and

$$\partial_u g(\boldsymbol{x}, q)|_0 = Id.$$