

Filtre de Kalman d'ensemble et filtres particuliers pour le modèle de Lorenz

Vu Duc Tran*, Valérie Monbet*, François LeGland**

* Laboratoire de Statistiques Appliquées, Université de Bretagne Sud

** Irisa, Inria, Rennes

vtran@univ-ubs.fr, monbet@univ-ubs.fr, legland@irisa.fr

Résumé : Dans ce papier, nous comparons les performances de plusieurs algorithmes classiques d'assimilation de données pour le modèle de Lorenz. Ce modèle a la particularité d'être fortement non linéaire et chaotique. Nous introduisons de plus la borne de Cramer-Rao qui représente d'une certaine façon la plus faible erreur d'estimation qu'on peut espérer atteindre pour un problème d'assimilation donné.

Mots-clés : Assimilation de données, Filtre de Kalman d'ensemble, Filtre particulier, Oscillateur de Lorenz, Borne de Cramer-Rao.

1 INTRODUCTION

L'objectif de cet article est de comparer des méthodes d'assimilation de données de type Monte Carlo pour le modèle de Lorenz. On s'intéresse en particulier au filtre de Kalman d'ensemble, qui est très utilisé notamment en physique des océans, et à deux versions du filtre particulier, qui a été développé plus récemment et qui est utilisé par exemple en navigation.

Le modèle de Lorenz constitue un cas test intéressant car c'est un oscillateur chaotique et fortement non linéaire. Il est souvent utilisé dans la littérature pour tester les méthodes d'assimilation de données. Pour un historique complet de l'utilisation du modèle de Lorenz en assimilation de données séquentielles, on peut se reporter à [Evensen, 1997]. Les articles les plus marquants sont [Miller et al., 1994] qui compare les propriétés du filtre de Kalman étendu et de la méthode variationnelle, [Evensen, 1997] qui reprend le même exemple pour évaluer les performances du filtre de Kalman d'ensemble (EnKF), du lisseur de Kalman d'ensemble et d'une méthode d'optimisation et, plus récemment, [Pham, 2001] qui compare le filtre *bootstrap*, le filtre EnKF, le filtre EnKF du second ordre et le filtre *singular evolutive interpolated Kalman* (SEIK) pour reconstruire la trajectoire d'état du système de Lorenz dans \mathbb{R}^3 quand on ne dispose que d'une observation bruitée dans \mathbb{R} .

Ici, nous proposons de comparer le filtre EnKF avec deux filtres particuliers (filtre *bootstrap* et filtre *sequential importance resampling* (SIR)). Le filtre *bootstrap* est la version la plus simple des filtres particuliers et on peut la voir comme une extension directe du filtre EnKF. Le filtre SIR permet de résoudre des problèmes posés par le filtre *bootstrap*. Nous comparons les performances des

trois filtres pour l'estimation d'une trajectoire de l'oscillateur de Lorenz dont l'observation est discrète et bruitée. A notre connaissance il n'existe pas de résultat sur l'utilisation du filtre SIR pour le modèle de Lorenz. De plus, la variance des estimateurs proposés est comparée à la borne de Cramer-Rao correspondante ce qui n'a jamais été publié précédemment.

Dans la section 2, nous présentons brièvement les filtres que nous allons comparer dans un cadre général : le filtre EnKF, le filtre *bootstrap* et le filtre SIR. Puis, dans la section 3, nous introduisons la borne de Cramer-Rao. Enfin dans la section 4, nous décrivons la mise en oeuvre des filtres pour l'oscillateur de Lorenz, le calcul de la borne de Cramer-Rao et nous commentons les résultats obtenus.

2 DESCRIPTIONS DES FILTRES

Dans un premier temps, nous décrivons, dans un cadre général, les filtres dont les propriétés sont évaluées plus loin.

2.1 Généralités

Considérons le système dynamique discret suivant :

$$x_k = f(x_{k-1}) + w_k \quad (1)$$

$$y_k = Hx_k + v_k \quad (2)$$

où $x_k \in \mathbb{R}^n$ représente le vecteur d'état à l'instant k , f est une fonction non linéaire et H est un opérateur linéaire. Les bruits $w_k \in \mathbb{R}^n$ et $v_k \in \mathbb{R}^d$ sont des bruits blancs de loi $p_k(dw)$ et $q_k(v)dv$. Ils sont indépendants de la condition initiale x_0 distribuée suivant la loi $\mu_0(dx)$. Les observations $y_k \in \mathbb{R}^d$ sont indépendantes entre elles conditionnellement à l'état caché x_k .

On peut considérer ce modèle comme un modèle de Markov caché où les états cachés $\{x_k\}$ forment une chaîne de Markov et les observations $\{y_k\}$ vérifient l'hypothèse de canal sans mémoire. Le problème de filtrage consiste alors à estimer le vecteur aléatoire x_k connaissant une suite d'observations $y_{0:k} = (y_0, \dots, y_k)$.

Selon le critère de minimum de variance, l'estimateur optimal pour ce problème est l'espérance conditionnelle $\mathbb{E}[x_k | y_{0:k}]$. Il est possible de calculer cette moyenne conditionnelle, en utilisant le filtre bayésien optimal [LeGland, 2004]. Son équation est facile à écrire, mais il est en général impossible de la résoudre, sauf dans le cas particulier des systèmes linéaires gaussiens, où elle se

ramène aux équations connues du filtre de Kalman-Bucy. Dans le cas général, on a recours à des approximations numériques de type Monte Carlo.

L'idée des méthodes d'assimilation de données séquentielles (ADs) de type Monte Carlo est d'approcher la moyenne conditionnelle $\mathbb{E}[x_k|y_{0:k}]$ par une espérance empirique calculée à partir d'un ensemble de N particules. Les particules sont soumises à un mécanisme d'évolution en deux étapes :

Etape de prédiction - Les particules explorent l'espace d'état de façon indépendante selon l'équation d'état (1).

Etape de correction - Lorsqu'une observation est disponible, les particules sont corrigées de façon à se 'rapprocher' de l'observation. L'étape de correction est illustrée, pour les trois filtres étudiés, figures 4 et 5.

Les variations entre les différents filtres viennent de la façon dont on réalise les étapes de prédiction et de correction.

2.2 Algorithmes des filtres

Avant d'écrire les algorithmes des filtres, il est nécessaire d'introduire quelques notations.

La moyenne et la variance de la variable aléatoire $x_k|y_{0:k}$ sont notées respectivement

$$x_k^a = \mathbb{E}[x_k|y_{0:k}] \quad (3)$$

$$P_k^a = \mathbb{E}[(x_k^a - x_k)(x_k^a - x_k)^T|y_{0:k}] \quad (4)$$

On définit également les quantités suivantes

$$x_k^f = \mathbb{E}[x_k|y_{0:k-1}] \quad (5)$$

$$P_k^f = \mathbb{E}[(x_k^a - x_k)(x_k^a - x_k)^T|y_{0:k-1}] \quad (6)$$

qui décrivent respectivement la moyenne et la variance de la variable aléatoire $x_k|y_{0:k-1}$. Les indices f et a représentent respectivement l'état prédit (forecast) et corrigé (analysed).

2.2.1 Filtre Kalman d'Ensemble (EnKF)

Supposons que w_k et v_k sont des bruits blancs gaussiens

$$p_k(dw) = \mathcal{N}(0, Q_k)$$

$$q_k(dv) = \mathcal{N}(0, R_k)$$

et que la loi initiale est aussi gaussienne

$$\mu_0(dx) = \mathcal{N}(0, P_0)$$

où $\mathcal{N}(\mu, \sigma)$ représente la loi de Gauss de moyenne μ et de variance σ .

Dans le filtre EnKF, l'étape de prédiction se fait d'après l'équation d'état (1) et l'étape de correction est réalisée comme dans le filtre de Kalman en calculant une moyenne pondérée entre la position de l'observation et la position des particules donnée par l'équation d'observation (2).

La figure 4 et 5 illustrent des itérations des filtres pour le modèle de Lorenz. On voit en particulier comment les particules réparties autour du lobe le plus bas sont corrigées et se regroupent autour de l'observation courante qui est représentée par l'intersection entre les 3 droites.

Initialisation - pour $i = 1, \dots, N$

- générer $x_0^{a,i} \sim \mathcal{N}(0, P_0)$.

Pour $k \geq 1$

Prédiction - pour $i = 1, \dots, N$

- générer $w_k^i \sim \mathcal{N}(0, Q_k)$

- $x_k^{f,i} = f(x_{k-1}^{a,i}) + w_k^i$

- $x_k^f \approx \frac{1}{N} \sum_{i=1}^N x_k^{f,i}$

- $P_k^f \approx \frac{1}{N-1} \sum_{i=1}^N [(x_k^{f,i} - x_k^f)(x_k^{f,i} - x_k^f)^T]$

Correction - pour $i = 1, \dots, N$

- générer $v_k^i \sim \mathcal{N}(0, R_k)$

- $\tilde{R}_k = \frac{1}{N} \sum_{i=1}^N (v_k^i)(v_k^i)^T$

- $K_k = P_k^f H^T (H P_k^f H^T + \tilde{R}_k)^{-1}$

- $x_k^{a,i} = x_k^{f,i} + K_k [y_k - H x_k^{f,i} + v_k^i]$

TAB. 1 – Filtre EnKF

Quand la fonction f dans le système dynamique est linéaire, alors [Evensen, 2003] affirme que le filtre EnKF converge exactement vers le filtre de Kalman-Bucy quand N tend vers l'infini.

Le principal inconvénient du filtre EnKF est qu'il nécessite de calculer des produits de matrices pour obtenir le gain de Kalman

$$K_k = P_k^f H^T (H P_k^f H^T + \tilde{R}_k)^{-1}$$

où l'indice T représente la transposée de la matrice.

Ainsi dès que la dimension n du vecteur d'état x_k est grande, par exemple $n = 10^5$ pour les modèles physiques de l'océan, P_k, \tilde{R}_k sont des matrices de très grande dimension et le coût de calcul est important.

2.2.2 Filtre bootstrap

Dans les algorithmes reposant sur des techniques d'échantillonnage d'importance, on s'affranchit des produits de matrices.

Dans l'algorithme *bootstrap*, l'étape de prédiction est réalisée comme pour le filtre EnKF d'après l'équation d'état (1). Dans l'étape de correction aussi appelée étape de sélection, l'adéquation de chaque particule avec l'observation est évaluée grâce à la fonction de vraisemblance : un poids ω_k^i proportionnel à la vraisemblance est affecté à chaque particule $x_k^{f,i}$, et les particules sont éliminées ou dupliquées selon la valeur de leur poids, autrement dit les particules corrigées sont calculées en simulant N nouvelles particules selon la loi mélange $\sum_{i=1}^N \omega_k^i \delta_{x_k^{f,i}}(dx)$, où δ est la mesure de Dirac.

Cette étape de sélection constitue une amélioration significative par rapport à l'algorithme *sequential importance sampling* (SIS), dans lequel on se contente d'accumuler les poids au cours du temps. On constate dans ce cas un phénomène de dégénérescence des poids, puisque seules quelques particules conservent un poids significatif et contribuent utilement à l'estimation [Doucet et al. 2000]. Même avec l'étape de sélection, il peut néanmoins arriver que la plupart des particules disponibles à l'issue de l'étape de prédiction reçoivent un poids trop faible, de

Initialisation - pour $i = 1, \dots, N$

- générer $x_0^{f,i} \sim \mu_0(dx)$.

pour $i = 1, \dots, N$

- $\omega_0^i \propto q_0(y_0 - Hx_0^{f,i})$.

- générer $x_0^{a,i} \sim \sum_{j=1}^N \omega_0^j \delta_{x_0^{f,i}}$.

A tout instant $\mathbf{k} \geq 1$

Prédiction - pour $i = 1, \dots, N$

- générer $w_k^i \sim p_k(dw)$

- $x_k^{f,i} = f(x_{k-1}^{a,i}) + w_k^i$

Sélection - pour $i = 1, \dots, N$

- $\omega_k^i \propto q_k(y_k - Hx_k^{f,i})$.

- générer $x_k^{a,i} \sim \sum_{j=1}^N \omega_k^j \delta_{x_k^{f,i}}$.

TAB. 2 – Filtre *bootstrap*

sorte que le ré-échantillonnage ne concerne finalement que quelques particules. Ce phénomène est très clair sur l'exemple des figures 4 et 5. Deux solutions ont été proposées [Arulampalam, et al. 2002] :

1. choisir la densité d'importance telle que la variance des poids d'importance soit minimale. [Doucet et al. 2000] a montré que le choix optimal de la densité d'importance est la fonction de distribution de probabilité de x_k sachant (x_{k-1}, y_k) .
2. redistribuer les particules : on donne une mesure de génération $N_{\text{eff}} = \frac{N}{1 + \text{Var}(\omega_k^i)}$. Si N_{eff} est plus petit qu'une constante fixée N_T , c'est à dire que les poids sont très déséquilibrés, alors on redistribue les particules de façon à obtenir un nouveau système de particules équiréparties.

Le filtre SIR (pour *sequential importance resampling*) présenté ci-dessous reprend ces idées.

2.2.3 Filtre SIR (sequential importance resampling)

Dans le filtre SIR, l'étape de prédiction consiste à générer un ensemble de particules selon une densité d'importance qui n'est pas forcément celle qui correspond à l'équation d'état et qui peut dépendre de l'observation. L'étape de correction est similaire à celle du filtre *bootstrap*, mais on y ajoute une étape de redistribution adaptative comme indiqué ci-dessus.

Dans la table 3, on décrit le filtre SIR dans le cas d'un système non linéaire gaussien. On suppose que $p_k(dw)$ et $q_k(dv)$ sont des lois gaussiennes centrées de covariance Q_k et R_k respectivement, avec R_k inversible. On utilise de plus la loi d'importance optimale [Doucet et al. 2000] donnée dans ce cas par

$$p(x_k | x_{k-1}, y_k) = \mathcal{N}(m_k, \Sigma_k)$$

où

$$\Sigma_k^{-1} = Q_k^{-1} + H^T R_k^{-1} H$$

$$m_k = \Sigma_k [Q_k^{-1} f(x_{k-1}) + H^T R_k^{-1} y_k]$$

Les deux filtres *bootstrap* et SIR sont très simples à mettre en oeuvre, ils ne nécessitent pas d'opération sur

Initialisation - pour $i = 1, \dots, N$

- générer $x_0^{f,i} \sim \mu_0(dx)$.

- $\omega_0^i \propto q_0(y_0 - Hx_0^{f,i})$.

Si $N_{\text{eff}} < N_T$, alors pour $i = 1, \dots, N$

- générer $x_0^{a,i} \sim \sum_{j=1}^N \omega_0^j \delta_{x_0^{f,i}}$.

- $\omega_0^i = 1/N$

A tout instant $\mathbf{k} \geq 1$

Prédiction - indépendamment pour $i = 1, \dots, N$

- générer $x_k^{f,i} \sim \mathcal{N}(m_k^i, \Sigma_k)$

où $m_k^i = \Sigma_k [Q_k^{-1} f(x_{k-1}^i) + H_k^T R_k^{-1} y_k]$

Sélection - pour $i = 1, \dots, N$

- $\omega_k^i \propto \omega_{k-1}^i p(y_k | x_{k-1}^{a,i})$

où $p(y_k | x_{k-1}^{a,i}) = \mathcal{N}(Hf(x_{k-1}^{a,i}), HQ_{k-1}H^T + R_k)$

Si $N_{\text{eff}} < N_T$, alors pour $i = 1, \dots, N$

- générer $x_k^{a,i} \sim \sum_{j=1}^N \omega_k^j \delta_{x_k^{f,i}}$

- $\omega_k^i = 1/N$.

TAB. 3 – Filtre SIR

des matrices de grande dimension. De plus, ils convergent vers le filtre bayésien optimal lorsque le nombre N de particules tend vers l'infini [Crisan et Doucet 2002].

3 BORNE DE CRAMER-RAO

Pour évaluer les performances des filtres, dans les cas d'école où le système (1)-(2) est parfaitement connu, il est classique de comparer les erreurs en moyenne quadratique des estimateurs de l'état induit par chacun des filtres. Il est alors intéressant de comparer ces erreurs à la borne de Cramer-Rao (CRLB) qui est une borne inférieure pour l'erreur en moyenne quadratique des estimateurs bayésiens.

3.1 Cas général

Considérons de nouveau le système (1)-(2). Si \hat{x}_k^a est un estimateur du vecteur d'état x_k sachant l'observation $y_{0:k}$, alors la matrice de covariance de l'estimateur \hat{x}_k^a , notée C_k , admet une borne inférieure J_k^{-1} :

$$C_k = \mathbb{E}[(\hat{x}_k^a - x_k)(\hat{x}_k^a - x_k)^T] \geq J_k^{-1} \quad (7)$$

La matrice J_k est la matrice d'information, de dimension $n \times n$.

[Tichavsky et al., 1998] propose un algorithme récursif pour calculer la matrice d'information J_k

$$J_{k+1} = D_k^{22} - D_k^{21} (J_k + D_k^{11})^{-1} D_k^{12} \quad (8)$$

avec

$$D_k^{11} = -\mathbb{E}\{\nabla_{x_k} [\nabla_{x_k} \log p(x_{k+1} | x_k)]^T\}$$

$$D_k^{21} = -\mathbb{E}\{\nabla_{x_k} [\nabla_{x_{k+1}} \log p(x_{k+1} | x_k)]^T\}$$

$$D_k^{12} = [D_k^{21}]^T$$

$$D_k^{22} = -\mathbb{E}\{\nabla_{x_{k+1}} [\nabla_{x_{k+1}} \log p(x_{k+1} | x_k)]^T\}$$

$$-\mathbb{E}\{\nabla_{x_{k+1}} [\nabla_{x_{k+1}} \log p(y_{k+1} | x_{k+1})]^T\}.$$

où $\mathbb{E}\{\cdot\}$ représente l'espérance en fonction de x_k, x_{k+1} et y_{k+1} . ∇_{x_k} est la dérivée partielle d'ordre 1 en fonction de

x_k . Le schéma (8) démarre avec la matrice d'information initiale J_0 :

$$J_0 = \mathbb{E}\{[\nabla_{x_0} \log p_0(x_0)][\nabla_{x_0} \log p_0(x_0)]^T\}$$

où $\mathbb{E}\{\cdot\}$ représente l'espérance en fonction de x_0 , et p_0 représente la densité de x_0 .

3.2 Cas d'un modèle à bruit gaussien

Lorsque les bruits $\{w_k\}$ et $\{v_k\}$ du système (1)-(2) sont de loi gaussienne $\mu_0(dx) = \mathcal{N}(0, P_0)$, $p_k(dw) = \mathcal{N}(0, Q_k)$ et $q_k(dv) = \mathcal{N}(0, R_k)$, avec P_0 , Q_k et R_k inversibles, alors l'écriture de D_k^{11} , D_k^{21} , D_k^{22} et J_0 se simplifie et on obtient [Ristic et al., 2004]

$$J_0 = P_0^{-1}$$

$$D_k^{11} = \mathbb{E}\{F_k^T Q_k^{-1} F_k\} \quad (9)$$

$$D_k^{12} = \mathbb{E}\{F_k^T\} Q_k^{-1} \quad (10)$$

$$D_k^{22} = Q_k^{-1} + H^T R_k^{-1} H$$

où $F_k = [\nabla_{x_k} f^T(x_k)]^T$ est le jacobien de $f(x_k)$ évalué à la vraie valeur de x_k . On remarque que dans ce cas, le calcul de la borne de Cramer-Rao est similaire à celui du filtre de Kalman linéarisé.

4 APPLICATION

4.1 Modèle de Lorenz

L'oscillateur (11) introduit par Lorenz en 1963 [Lorenz, 1963] est un modèle simplifié qui représente un écoulement de fluide particulier. Il est célèbre pour son comportement chaotique et sa dynamique fortement non linéaire. La figure 4 montre une trajectoire de Lorenz. Il est couramment utilisé pour valider les méthodes d'assimilation de données car il permet notamment de reproduire l'imprédictabilité de l'évolution des systèmes météorologiques. Le système de Lorenz s'écrit en temps continu :

$$\begin{cases} \dot{x}_t = \sigma(y_t - x_t) + \dot{w}_t^x \\ \dot{y}_t = \rho x_t - y_t + x_t z_t + \dot{w}_t^y \\ \dot{z}_t = x_t y_t - \beta z_t + \dot{w}_t^z \end{cases} \quad (11)$$

où \dot{x}_t représente la dérivée de x_t par rapport au temps. On choisit habituellement $\sigma = 10$, $\rho = 28$, $\beta = 8/3$. Les bruits $\{w_t\}$ sont indépendants et tels que

$$\text{Pour tout } a, b, \int_a^b \dot{w}_t dt = (b - a)W$$

avec W une variable aléatoire gaussienne centrée de variance Q .

Nous supposons que nous disposons d'une suite d'observations bruitées $\{Y_{k\Delta t}\}_{k=1, \dots, n}$, $Y_{k\Delta t} \in \mathbb{R}^3$ des trois composantes d'un oscillateur de Lorenz et nous cherchons à reconstruire la trajectoire d'état correspondante.

4.2 Discrétisation du système dynamique

Pour appliquer les filtres décrits plus haut, il est nécessaire de discrétiser le système (11) qui représente l'équation d'état. Pour cela différents schémas

numériques peuvent être utilisés. Ici, nous avons choisi un schéma d'Euler pour sa simplicité mais aussi parce que ce schéma se parallélise facilement ce qui permet de réduire les coûts de calcul. Un point délicat dans la discrétisation est la prise en compte des bruits aléatoires $\{\dot{w}_t\}$.

Une première solution consiste à résoudre l'équation de Lorenz ordinaire (i.e. sans bruit) entre deux temps d'observation successifs $t_k = k\Delta t$ et $t_{k+1} = (k+1)\Delta t$ avec un pas de temps $\Delta\tau \ll \Delta t$ et d'ajouter, à chaque temps t_k , un bruit gaussien à la solution numérique. On notera ce schéma EDO dans la suite.

Une seconde solution consiste à résoudre numériquement l'équation différentielle stochastique et donc à ajouter un bruit aléatoire à chaque pas de temps du schéma numérique. Ce schéma sera noté EDS dans la suite.

4.3 Calcul de la borne de Cramer-Rao

En pratique, calculer la borne de Cramer-Rao revient à évaluer les espérances dans (9) et (10). Une approximation Monte Carlo est utilisée, des réalisations du vecteur d'état sont simulées et notées $\{x_k^i, i = 1, \dots, M\}$, puis les espérances dans (9) et (10) sont approchées par les moyennes empiriques

$$D_k^{11} \approx \frac{1}{M} \sum_{i=1}^M (F_k^i)^T Q_k^{-1} F_k^i$$

$$D_k^{12} \approx \frac{1}{M} \sum_{i=1}^M (F_k^i)^T Q_k^{-1}$$

avec $F_k^i = [\nabla_{x_k} f^T(x_k^i)]^T$ le jacobien de $f(x_k)$ évalué en x_k^i .

De la même façon, la matrice de covariance C_k dans l'équation (7) est estimée empiriquement à partir de M réalisations de $\hat{x}_k^{a,i}$ et x_k^i :

$$C_k \approx \frac{1}{M} \sum_{i=1}^M (\hat{x}_k^{a,i} - x_k^i)(\hat{x}_k^{a,i} - x_k^i)^T$$

Pour comparer, l'erreur en moyenne quadratique des estimateurs avec la borne de Cramer-Rao, on est amené à comparer 2 matrices. Nous avons choisi de comparer les traces des matrices :

$$\mathbb{E}[(\hat{x}_k^a - x_k)^T (\hat{x}_k^a - x_k)] \geq \text{tr}(J_k^{-1}) \quad (12)$$

où $\text{tr}(\cdot)$ est l'opération trace de la matrice.

4.4 Résultats

Pour étudier les performances des filtres, nous générons, avec un schéma d'Euler EDO (resp. EDS), une trajectoire du modèle de Lorenz que nous considérons comme la trajectoire de référence à estimer. La figure 4 montre une trajectoire de l'oscillateur de Lorenz avec pour condition initiale $(1.508870, -1.531271, 25.46091)$. Le schéma d'Euler est appliqué avec un pas de temps $\Delta\tau = 0.005$.

Nous disposons 320 observations qui sont réparties régulièrement sur l'intervalle $[0, 160]$. Les observations

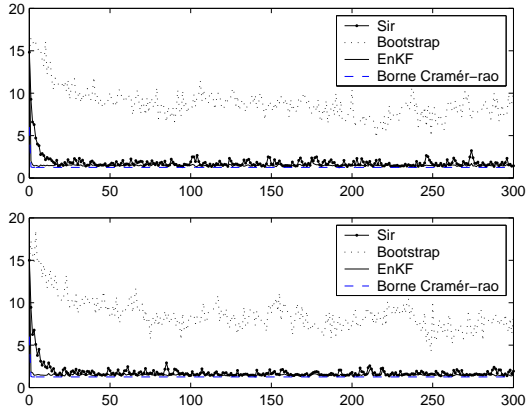


FIG. 1 – Variance des estimateurs et borne de Cramer-Rao - $Q = I, R = I$, Haut : EDO. Bas : EDS. ... *bootstrap*, .- SIR, — EnKF, - - - borne de Cramer-Rao.

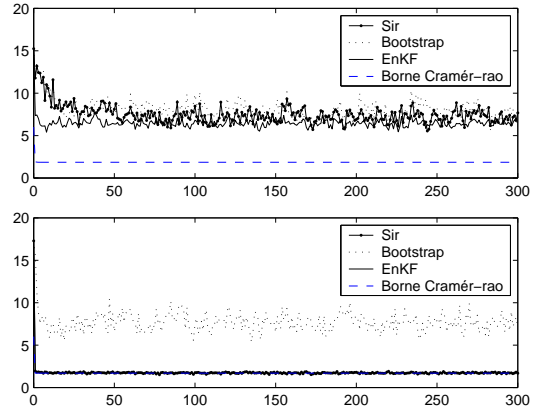


FIG. 3 – Variance des estimateurs et borne de Cramer-Rao - EDO, Haut : $Q = I, R = 25I$. Bas : $Q = 25I, R = I$ *bootstrap*, .- SIR, — EnKF, - - - borne de Cramer-Rao.

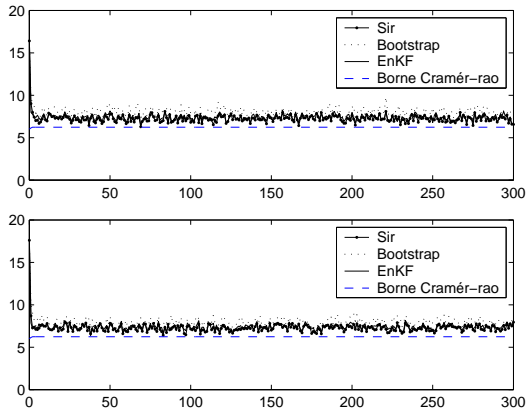


FIG. 2 – Variance des estimateurs et borne de Cramer-Rao - $Q = 25I, R = 25I$, Haut : EDO. Bas : EDS. ... *bootstrap*, .- SIR, — EnKF, - - - borne de Cramer-Rao.

sont simulées à partir de la solution de référence en ajoutant un bruit gaussien centré de variance $I_{3,3}$ aux valeurs de l'état (Figure 1) et $25I_{3,3}$ (Figure 2). On note $I_{n,n}$ la matrice identité sur $\mathbb{R}^{n,n}$. Dans le premier cas, l'erreur du modèle est un bruit gaussien centré de variance $Q = I_{3,3}$ et dans le second cas de variance $Q = 25I_{3,3}$.

Pour comparer les performances des filtres, 100 trajectoires du modèle de Lorenz ont été simulées et réestimées en utilisant les trois filtres avec 100 particules. Les figures 1 à 3 montrent l'erreur en variance des estimateurs ainsi que la trace de la borne de Cramer-Rao pour les deux schémas de discrétisation EDO et EDS. Pour faciliter les comparaisons, l'erreur en variance est approchée par la formule de l'équation (12). Nous avons vérifié que les biais sont nuls. Avant de discuter les figures, il faut remarquer que le temps de calcul nécessaire pour le filtre EnKF est en moyenne 3 fois plus important que celui des filtres SIR et *bootstrap* pour l'oscillateur de Lorenz.

Le fait de considérer une équation différentielle ordi-

naire ou stochastique ne modifie pas l'ordre de l'erreur en moyenne quadratique.

Les figures 1 et 2 montrent que dans tous les cas les performances des filtres SIR et EnKF sont similaires. Pour ces deux filtres l'erreur en moyenne quadratique, entre la trajectoire estimée et la trajectoire de référence, est proche de la borne de Cramer Rao. En revanche, on observe que le filtre *bootstrap* semble inadapté dans le cas où les variances sont faibles. Ce phénomène peut s'expliquer par le fait qu'après l'étape de prédiction les particules peuvent toutes se trouver sur un des lobes et alors que l'observation est sur l'autre lobe. L'étape de correction du bootstrap ne permet pas de déplacer les particules et on observe donc une forte erreur. Lorsque la variance de l'équation d'état est forte, le risque de se trouver dans cette situation est moins important car les particules sont plus dispersées.

La figure 3 donne les résultats obtenus par un schéma EDO dans les cas où le rapport de la variance de l'équation d'état sur la variance de l'équation d'observations est $1/25$ (haut) puis 25 (bas). On remarque que dans le cas où on a une forte incertitude sur les observations, les erreurs en moyenne quadratique s'éloignent significativement de la borne de Cramer-Rao. On constate l'inverse lorsque qu'on a une forte confiance dans les observations. En effet dans ce dernier cas, les observations sont proches de la trajectoire de référence et, à chaque étape d'assimilation, l'observation attire fortement le nuage de particules.

5 CONCLUSION

Dans ce papier, nous comparons trois méthodes usuelles d'assimilation de données séquentielle pour estimer la trajectoire d'un oscillateur de Lorenz dont l'observation est discrète et bruitée. On met en évidence que les filtres EnKF et SIR ont des performances similaires. Ces deux filtres conduisent à une erreur en moyenne quadratique très proche de la borne de Cramer-Rao, ce qui peut être

considéré comme un critère d'optimalité. Il est surprenant d'observer que le filtre *bootstrap* conduit à une erreur en moyenne quadratique plus importante, en particulier quand les variances d'état et d'observation sont faibles.

REMERCIEMENTS

Cette étude a bénéficié du soutien de l'INRIA, dans le cadre de l'Action de Recherche Coopérative *Assimilation de données pour la qualité de l'air* (ADOQA), et de la région Bretagne.

Bibliographie

- [Arulampalam, et al. 2002] M. S. Arulampalam, S. Maekell, N. J. Gordon, and T. Clapp, A tutorial on particle filters for online nonlinear / non-Gaussian Bayesian tracking. *IEEE Transactions on Signal Processing*, vol. 50, no.2, 174-188.
- [Bergman, 1999] N. Bergman, Recursive bayesian estimation : Navigation and tracking applications. PhD thesis, Linköping University, Sweden.
- [Crisan et Doucet 2002] D. Crisan, A. Doucet, A survey of convergence results on particle filtering methods for practitioners. *IEEE Transactions on Signal Processing*, vol. 50, no.3, 736-746.
- [Doucet et al. 2000] A. Doucet, S. Godsill, C. Andrieu, On sequential Monte Carlo sampling methods for Bayesian filtering. *Statistics and Computing*, vol. 10, 197-208.
- [Evensen, 1997] G. Evensen, Advanced data assimilation for strongly non-linear dynamics. *Mon Weather Rev* 125, 1342-1354.
- [Evensen, 2003] G. Evensen, The ensemble Kalman filter : theoretical formulation and practical implementation. *Ocean Dynamics*, 53, pp.343-367.
- [LeGland, 2004] F. LeGland, Introduction au filtrage en temps discret : Filtre de Kalman, Filtrage Particulaire, Modèles de Markov Cachés, polycopie pour Master Recherche STI, 2004-05.
- [Lorenz, 1963] Lorenz, E., Deterministic non periodic flow, *JAS*.
- [Miller et al., 1994] Miller, M. Ghil, and F. Gauthiez, 1994, Advanced data assimilation in strongly nonlinear dynamical systems. *J. Atmos. Sci.*, 51, 1037-1056.
- [Pham, 2001] DT. Pham, Stochastic methods for sequential data assimilation in strongly nonlinear systems. *Mon Weather Rev* 129, 1194-1207.
- [Ristic et al., 2004] B. Ristic, S. Arulampalam, and N. Gordon, *Beyond the Kalman Filter : Particle Filters for Tracking Applications*. Artech House, 2004.
- [Tichavsky et al., 1998] P. Tichavsky, C. H. Muravchik, and A. Nehorai, Posterior Cramér-Rao bounds for discrete-time nonlinear filtering. *IEEE Transactions on Signal Processing*, vol. 46, 1386-1396.
- [VanTrees, 1968] H. L. Van Trees, *Detection, Estimation and Modulation Theory*. John Wiley & Sons, 1968.

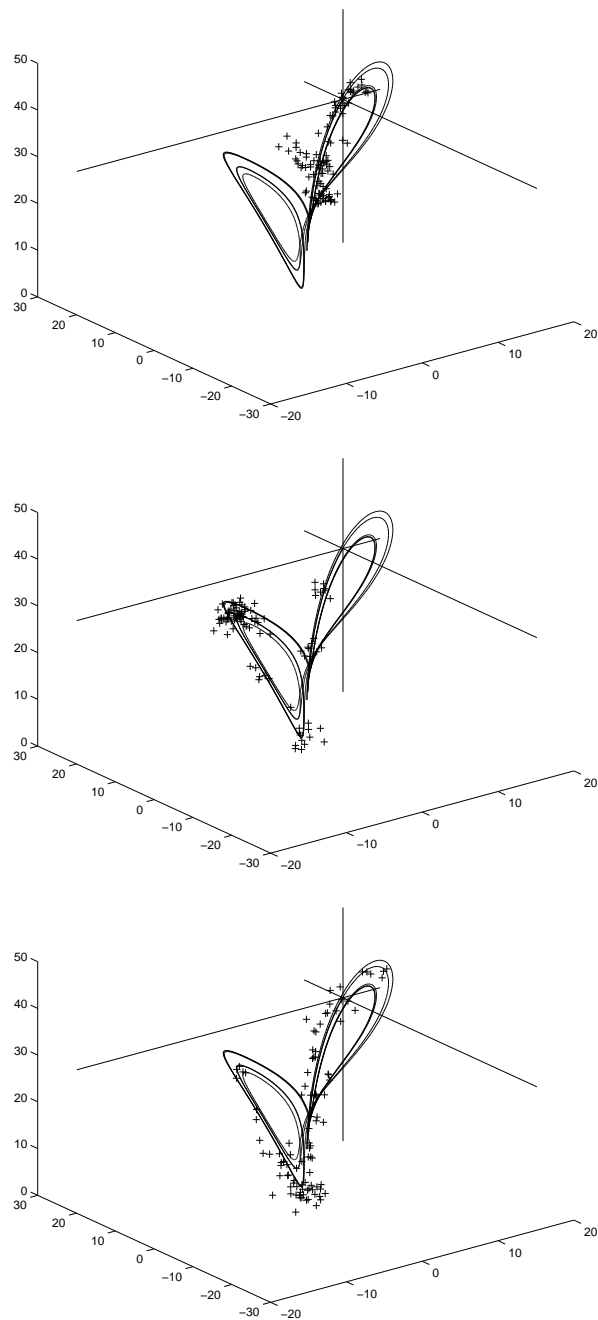


FIG. 4 – Trajectoire de Lorenz et particules du filtre SIR(haut), *bootstrap* (milieu) et EnKF(bas) avant l'étape de correction.

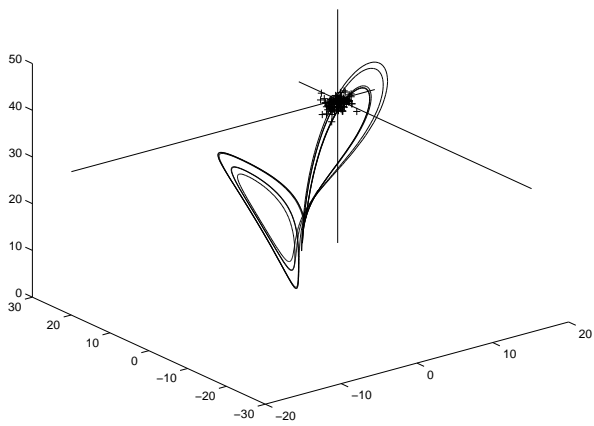
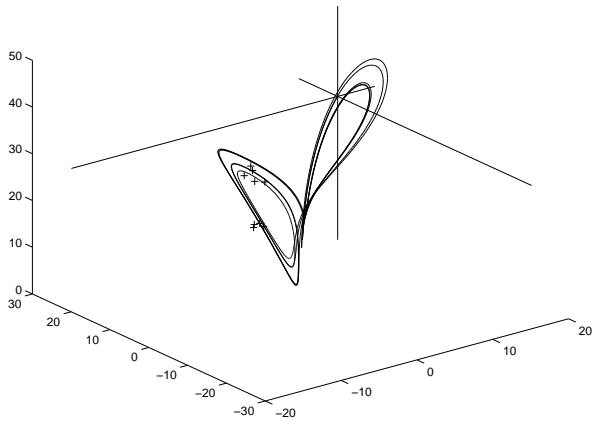
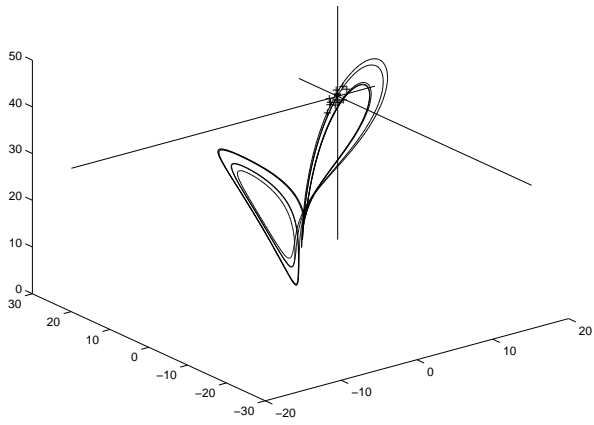


FIG. 5 – Trajectoire de Lorenz et particules du filtre SIR(haut), *bootstrap* (milieu) et EnKF(bas) après l'étape de correction.