

Ensemble-Based Kalman Filters

Istvan Szunyogh

Texas A&M University
Department of Atmospheric Sciences

BIRS, Banff, 10-15 July 2011

Outline

- 1 Extended Kalman Filter
 - General Formulation and Notations
 - Example: Hénon mapping
- 2 Ensemble-Based Kalman Filters
 - General Formulation and Notations
 - EnKF Schemes
 - Frequently Asked Questions

The Data Assimilation Problem

- Suppose we are given **a set of noisy observations** of the atmosphere for a time interval τ
- Let $\gamma = \{\mathbf{x} = \mathbf{x}(t), t \in \tau\}$ be the trajectory in τ
- We want to determine **which trajectory γ fits the observations “best”**
- The **state estimate** $\mathbf{x}^a(t_a)$ for a given time t_a along the γ^a solution trajectory is called the **analysis** for analysis time t_a

Assumptions

- Divide τ into n Δt -long **observation time windows**:

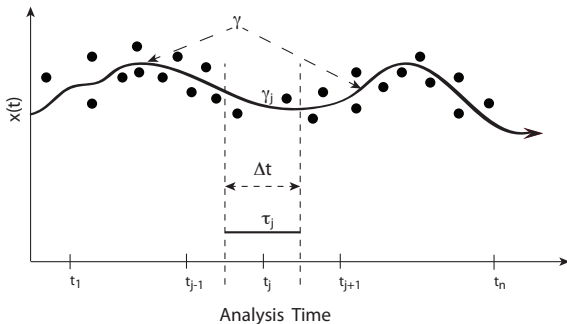
$$\tau_j = [t_j - \Delta t/2, t_j + \Delta t/2], \quad j = 1, \dots, n.$$

- **An observation** at time t_j is a **triple** $(\mathbf{y}_j^o, \mathcal{H}_j, \mathbf{R}_j)$, where \mathbf{y}_j^o is a **vector of observed values**, and \mathcal{H}_j and \mathbf{R}_j describe the relationship between \mathbf{y}_j^o and γ :

$$\mathbf{y}_j^o = \mathcal{H}_j(\gamma_j) + \varepsilon_j^o,$$

where γ_j is the part of the trajectory γ that falls into τ_j , ε_j is a Gaussian random variable with mean $\mathbf{0}$ and covariance matrix \mathbf{R}_j .

- We assume that the covariance between the errors of any two observations from two different time windows is zero.

Illustration for $m=1$: model with one variable

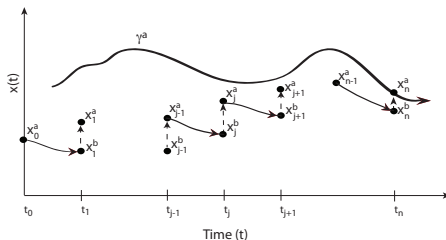
The most likely trajectory is the one that minimizes the “cost function”

$$J(\gamma) = \sum_{j=1}^n [\mathbf{y}_j^o - \mathcal{H}_j(\gamma_j)]^T \mathbf{R}_j^{-1} [\mathbf{y}_j^o - \mathcal{H}_j(\gamma_j)].$$

Generalized Observation Operator

- We wish to rewrite the cost function such that the independent variable is not a trajectory, but the state at a given time
- For a deterministic system this can be done
- **Generalized observation operator**, \mathcal{H}_{τ_n} : in addition to the spatial interpolation and the mapping of model variables into observables, it includes the mapping in time from the analysis time t_n to the observation times in τ_n (This is my terminology.)

Sequential Formulation



$$\begin{aligned}
 J(\delta \mathbf{x}_n) &= (\delta \mathbf{x}_n)^T (\mathbf{P}_n^b)^{-1} \delta \mathbf{x}_n \\
 &+ [\delta \mathbf{y}_n^o - \mathbf{H}_{\tau_n} \delta \mathbf{x}_n]^T \mathbf{R}_n^{-1} \\
 &\times [\delta \mathbf{y}_n^o - \mathbf{H}_{\tau_n} \delta \mathbf{x}_n].
 \end{aligned}$$

Remark: All practical approaches for data assimilation in the atmospheric sciences are based on this cost function

Notations

The **background** \mathbf{x}_n^b is obtained by integrating the model from \mathbf{x}_{n-1}^a :

$$\mathbf{x}_n^b = \mathcal{M}_{t_{n-1}, t_n}(\mathbf{x}_{n-1}^a)$$

The **increment** is defined by the difference between the state estimate \mathbf{x}_n and the background:

$$\delta\mathbf{x}_n = \mathbf{x}_n - \mathbf{x}_n^b$$

\mathbf{H}_{τ_n} is the **linearization** of $\mathcal{H}_{\tau_n}(\mathbf{x}_n)$ about $\delta\mathbf{x}_n$:

$$\mathcal{H}_{\tau_n}(\mathbf{x}_n) \approx \mathcal{H}_{\tau_n}(\mathbf{x}_n^b) + \mathbf{H}_{\tau_n} \delta\mathbf{x}_n^b$$

$\delta\mathbf{y}_n^o$ is the **innovation**:

$$\delta\mathbf{y}_n^o = \mathbf{y}_n^o - \mathcal{H}_{\tau_n}(\mathbf{x}_n^b)$$

\mathbf{P}_n^b is the **background covariance matrix**

The Computation of \mathbf{P}_n^b

Linearizing the model dynamics about \mathbf{x}_{n-1}^a :

$$\mathcal{M}_{t_{n-1}, t_n}(\mathbf{x}_{n-1}) = \mathcal{M}_{t_{n-1}, t_n}(\mathbf{x}_{n-1}^a) + \mathbf{M}_{t_{n-1}, t_n} \varepsilon_{n-1}^a,$$

where ε_{n-1}^a is the random variable representing the analysis error at t_{n-1} , we obtain

$$\mathbf{P}_n^b = \mathbf{M}_{t_{n-1}, t_n} \mathbf{P}_{n-1}^a \mathbf{M}_{t_{n-1}, t_n}^T$$

If \mathbf{P}_{n-1}^a is known from the analysis process at t_{n-1} , the computation of \mathbf{P}_n^b requires **M integrations of the tangent-linear model**, $\mathbf{M}_{t_{n-1}, t_n}$, where M is the dimension of the state vector of the model (in a state-of-the-art model $M \sim 10^8$)

The Extended Kalman Filter (EKF)

- 1 $\mathbf{x}_n^b = \mathcal{M}_{t_{n-1}, t_n}(\mathbf{x}_{n-1}^a)$
- 2 $\mathbf{P}_n^b = \mathbf{M}_{t_{n-1}, t_n} \mathbf{P}_{n-1}^a \mathbf{M}_{t_{n-1}, t_n}^T$
- 3 $\delta \mathbf{x}_n^a = \mathbf{K}_n \delta \mathbf{y}_n^o$
- 4 $\mathbf{P}_n^a = (\mathbf{I} - \mathbf{K}_n \mathbf{H}_{\tau_n}) \mathbf{P}_n^b$

The **Kalman-gain matrix**, \mathbf{K}_n , can be written in a number of alternative forms; perhaps the most practical is

$$\mathbf{K}_n = \mathbf{P}_n^b \mathbf{H}_{\tau_n}^T (\mathbf{H}_{\tau_n} \mathbf{P}_n^b \mathbf{H}_{\tau_n}^T + \mathbf{R}_n)^{-1}$$

- **Forecast Step of EKF:** Equations 1 and 2
- **State Update Step of EKF:** Equation 3 and 4

Rudolf Kalman Receives the National Medal of Science And Technology on October 7, 2009



Example: Hénon mapping

Problem: *Derive the equations of the Extended Kalman Filter for the Hénon mapping for the case, in which the two components of the state vector are directly observed at each iteration step of the mapping.*

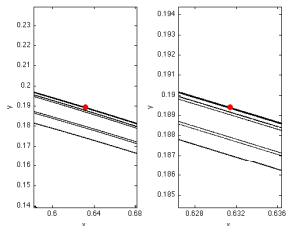
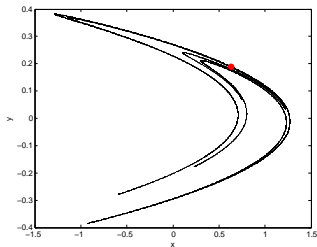
The equations of the Hénon mapping are

$$x_{j+1} = 1 + y_j - ax_j^2, \quad (1)$$

$$y_{j+1} = bx_j, \quad (2)$$

where x_j and y_j are the values of the two variables at “time” t_j , and a and b are scalar constants. For the usual choices $a = 1.4$, $b = 0.3$ of the parameters, the Hénon mapping shows complex chaotic behavior.

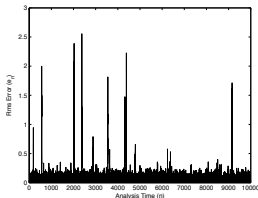
Hénon Attractor



Example: Implementation of the ETKF on the Hénon mapping

- Time series of “true” states was generated for $N = 10^4$ steps
- Observations were generated adding random observation noise with mean 0 and variance 0.01 to both variables
- Time series of analysis errors is computed by

$$e_n^a = \sqrt{(x_n^a - x_n^t)^2 + (y_n^a - y_n^t)^2}, \quad n = 1, \dots, N$$



Is It Possible to Do Better?

Two highly undesirable features of the results:

- Huge spikes in the error
- High overall rms error over time (0.12 vs. 0.14 for the observations)

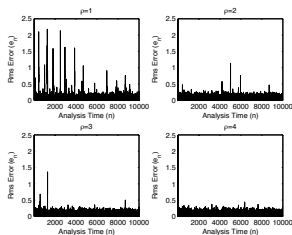
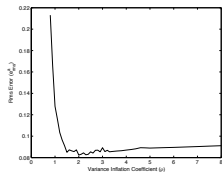
The only assumption made in this case was that the evolution of the error was linear.

Problem: *Show that the assumption of linear error dynamics introduces a bias into the background value of the x component and also leads to an underestimation of the background error variance for x .*

Variance Inflation

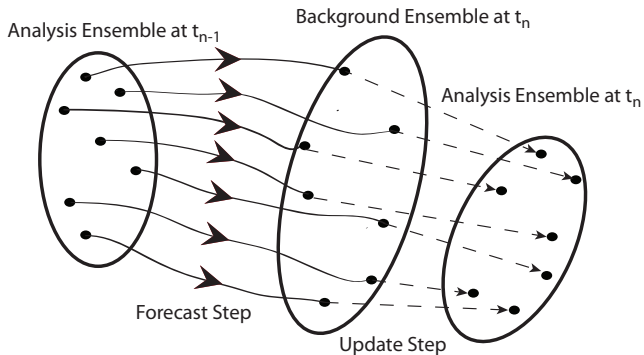
- This is a general problem with EKF, which has been known at least since Jazwinski (1970), where a simple approach was offered to correct the problem for the case in which the second derivative (Hessian) is available; but it is rarely mentioned in the EnKF literature, where the problem is usually attributed to an under-dispersive ensemble
- A simple approach to cope with the problem is **variance inflation** (e.g., $\mathbf{P}_n^b = \rho \mathbf{M}_{t_{n-1}, t_n} \mathbf{P}_{n-1}^a \mathbf{M}_{t_{n-1}, t_n}^T$, $\rho > 1$); it does not remove the bias or restore Gaussianity, but it has a large positive effect on the performance of the filter

Illustration of the Effect of Variance Inflation



The spikes disappear at a value of ρ , which is larger than the value that minimizes the rms error

Illustration for Two-Dimensional ($M = 2$) State Space



The Forecast Step of EnKF I

Given the $\{\mathbf{x}_{n-1}^{a(k)} : k = 1, 2, \dots, K\}$ ensemble of analyses, the forecast step of EnKF obtains an **ensemble of forecast trajectories**,

$$\{\gamma_n^{(k)} = \mathbf{x}_n^{(k)}(t) : t \in \tau_n, k = 1, 2, \dots, K\}$$

by an ensemble of nonlinear model integrations:

$$\mathbf{x}_n^{(k)}(t) = \mathcal{M}_{t_{n-1}, t}(\mathbf{x}_{n-1}^{a(k)}), \quad t \in \tau_n \quad k = 1, 2, \dots, K.$$

The k -th **ensemble perturbation** at time t in τ_n is defined by

$$\mathbf{X}^{(k)}(t) = \mathbf{x}^{(k)}(t) - \bar{\mathbf{x}}(t),$$

where

$$\bar{\mathbf{x}}(t) = \frac{1}{K} \sum_{k=1}^K \mathbf{x}^{(k)}(t) = \frac{1}{K} \sum_{k=1}^K \mathcal{M}_{t_{n-1}, t}(\mathbf{x}_{n-1}^{a(k)}).$$

The Forecast Step of EnKF II

The ensemble perturbations are generated such that they satisfy

$$\mathbf{P}_{n-1}^a = \frac{1}{K-1} \sum_{k=1}^K \mathbf{x}_{n-1}^{a(k)} (\mathbf{x}_{n-1}^{a(k)})^T,$$

In addition, the EnKF assumes that

$$\begin{aligned} \mathbf{x}^{(k)}(t) &= \mathcal{M}_{t_{n-1}, t} \left(\bar{\mathbf{x}}_{n-1}^a + \mathbf{x}_{n-1}^{a(k)} \right) \\ &\approx \mathcal{M}_{t_{n-1}, t} \left(\bar{\mathbf{x}}_{n-1}^a \right) + \mathbf{M}_{t_{n-1}, t} \mathbf{x}_{n-1}^{a(k)}. \end{aligned}$$

Then, if $\{\mathbf{x}_{n-1}^{a(k)}\}$ samples a normal distribution with mean $\mathbf{0}$, $\{\mathbf{x}^{(k)}(t)\}$ samples a normal distribution with mean $\mathbf{0}$ and covariance matrix $\mathbf{P}(t) = \frac{1}{K-1} \sum_{k=1}^K \mathbf{x}^{(k)}(t) [\mathbf{x}^{(k)}(t)]^T$

The Direct Estimation of $\mathbf{P}_n^b \mathbf{H}_{\tau_n}^T$ and $\mathbf{H}_{\tau_n} \mathbf{P}_n^b \mathbf{H}_{\tau_n}^T$

$$\begin{aligned}\mathbf{P}_n^b \mathbf{H}_{\tau_n}^T &= E \left[\mathbf{x}_n^b \left(\mathbf{H}_{\tau_n} \mathbf{x}_n^b \right)^T \right] \\ &\approx \frac{1}{K-1} \sum_{k=1}^K \mathbf{x}_n^{b(k)} \left[\mathcal{H}_{\tau_n} \left(\mathbf{x}_n^{b(k)} \right) - \bar{\mathcal{H}}_{\tau_n} \left(\mathbf{x}_n^{b(k)} \right) \right]^T \\ &= \frac{1}{K-1} \sum_{k=1}^K \mathbf{x}_n^{b(k)} \left(\mathbf{y}_n^{b(k)} \right)^T, \\ \mathbf{H}_{\tau_n} \mathbf{P}_n^b \mathbf{H}_{\tau_n}^T &\approx \frac{1}{K-1} \sum_{k=1}^K \mathbf{y}_n^{b(k)} \left(\mathbf{y}_n^{b(k)} \right)^T\end{aligned}$$

Because of covariance localization, only small sub-matrices of $\mathbf{P}_n^b \mathbf{H}_{\tau_n}^T$ and $\mathbf{H}_{\tau_n} \mathbf{P}_n^b \mathbf{H}_{\tau_n}^T$ need to be computed

The State Update Step of EnKF

The main differences between the different formulations of the EnKF are in the update step:

- 1 **Perturbed Observations:** The analysis ensemble, $\{\mathbf{x}_n^{a(k)} : k = 1, 2, \dots, K\}$, is generated such that it satisfies the equations

$$\bar{\mathbf{x}}^a = \bar{\mathbf{x}}^b + \mathbf{K} \left[\mathbf{y}^o - \overline{\mathcal{H}(\mathbf{x}^b)} \right]$$

and

$$\mathbf{P}^a = (\mathbf{I} - \mathbf{K}\mathbf{H})\mathbf{P}^b.$$

- 2 **Square-Root Filters:** First $\bar{\mathbf{x}}_n^a$ and \mathbf{P}_n^a are computed, then, an ensemble of analysis perturbations, $\{\mathbf{X}_n^{a(k)} : k = 1, 2, \dots, K\}$, is generated such that it is consistent with \mathbf{P}_n^a

Perturbed Observations

The main steps of the computation

- 1 K vectors of perturbed observations, $\{\mathbf{y}^{o(k)} : k = 1, \dots, K\}$ are generated by adding random observation noise to the vector of observations \mathbf{y}^o :

$$\mathbf{y}^{o(k)} = \mathbf{y}^o + \mathbf{e}^{(k)}, \quad k = 1, \dots, K$$

The observation noise samples a distribution with mean $\mathbf{0}$ and covariance matrix \mathbf{R}

- 2 The analysis ensemble is generated by assimilating a different vector of perturbed observations into each background ensemble member:

$$\mathbf{x}^{a(k)} = \mathbf{x}^{b(k)} + \mathbf{K} \left[\mathbf{y}^{o(k)} - \mathcal{H} \left(\mathbf{x}^{b(k)} \right) \right], \quad k = 1, \dots, K$$

A Few Comments on the Perturbed Observation Schemes

- The first correct formulations of EnKF were perturbed observation schemes (Houtekamer and Mitchell 1998; Burgers et al. 1998)
- The acronym EnKF in many papers refers to only perturbed observation schemes
- In general, the easiest to implement among all ensemble-based Kalman filters
- Usually implemented as a serial assimilation scheme (the observations are assimilated one-by-one or in small batches)

Square-Root Filters

- Recall that

$$\mathbf{P}_n^a = \frac{1}{K-1} \sum_{k=1}^K \mathbf{x}_{n-1}^{a(k)} (\mathbf{x}_{n-1}^{a(k)})^T$$

- We can form the \mathbf{X}_{n-1}^a **matrix of ensemble perturbations** with the help of the ensemble perturbations: the k -th column of \mathbf{X}_{n-1}^a is $(K-1)^{-1/2} \mathbf{x}_{n-1}^{a(k)}$
- Then

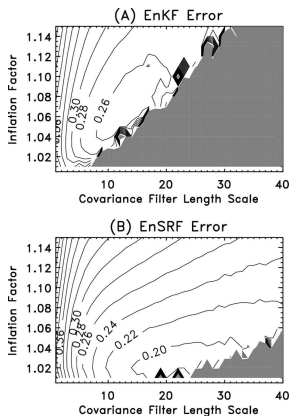
$$\mathbf{P}_n^a = \mathbf{X}_{n-1}^a (\mathbf{X}_{n-1}^a)^T.$$

- Thus the analysis ensemble perturbations can be generated by computing a square-root of \mathbf{P}_n^a (the square-root of a matrix is not uniquely defined!)

A Few Comments on the Square-Root Filter Schemes

- Schemes that belong to this group are the **Ensemble Square-Root Filter** of Whitaker and Hamill (2002), the **Ensemble Adjustment Filter** (Anderson 2001), the **Ensemble Transform Kalman Filter** (Bishop et al. 2001), the **Local Ensemble Kalman Filter** (Ott et al. 2004) and the **Local Ensemble Transform Kalman Filter** (Hunt et al. 2007)
- The **LETKF** is **not serial scheme**: it estimate the state vector components independently, assimilating all observations that may affect the analysis of a component in one step
- Generally, they are **more accurate** than the perturbed observations schemes **for small ensembles**

Comparison for 40-Variable Lorenz Model



Corrected version of figure from Whitaker and Hamill (2002)

Frequently Asked Questions

- **Q: Which EnKF scheme should I use in my application?**
- **A:** The main differences between the schemes are in the computational efficiency: for a large model and few observations serial schemes are more efficient; when the model is large and there are many observations the LETKF is becoming more advantageous. For radiance observations, the LETKF have some potential advantage
- **Q: Can EnKF schemes assimilate radiance observations?**
- **A:** Yes, but you have to find an acceptable localization and bias correction strategies. Recommended reading on the topic: Fertig et al. (Tellus, 2007, 2009); Miyoshi et al. (2010); Aravequia et al. (MWR, 2011); Campbell et al. (MWR, 2010)