Hybrid particle-Kalman filter method for high-dimensional data assimilation problems

Amit Apte
International Centre for Theoretical Sciences (ICTS-TIFR)
Bangalore, India

Joint work with Laura Slivinski, Elaine Spiller, Bjorn Sandstede

Conference on stochastic systems and applications, 08 September 2014, IISc, Bangalore
Lagrangian data assimilation is the problem of using data from Lagrangian/passive instruments (e.g. drifters and gliders).

Particle filtering and Kalman filtering are two complementary data assimilation methods which are
- ineffective in high dimensional and nonlinear problems, respectively, but
- effective in nonlinear problems and high dimensions, respectively.

Hybrid particle-Kalman filter that I will discuss combines the strengths of both, for the Lagrangian data assimilation problem.

1. Lagrangian observations of the ocean
2. Two important methods of data assimilation - particle and Kalman filter
3. Hybrid Kalman-particle filter: a few results with model problems
4. Outlook
Outline

1. Lagrangian observations of the ocean
2. Two important methods of data assimilation - particle and Kalman filter
3. Hybrid Kalman-particle filter: a few results with model problems
4. Outlook
Most of the ocean data are skin-deep\(^1\)

- Sources of surface data: ships; buoys; drifters; and satellites (which give the largest quantity of data)
  - Measurements from ship: an array of instruments, sometimes reaching the sea floor is dropped to take measurements – very time and resource consuming and hence sparse temporal and spatial coverage
- Sources of subsurface data: Special instruments deployed for this specific purpose
  - Floats; Autonomous underwater vehicles (gliders): described in detail later – depth is limited but vast temporal(?) and spatial coverage

Most of the ocean data are skin-deep\textsuperscript{1}.

Left: four months of ship measurements; Right: satellite observations are much more numerous than other types

\textsuperscript{1}E.g. \url{http://ewoce.org/} for non-satellite data from 1988-1998
Section 1: Lagrangian observations of the ocean

Floats drift and drifters float\(^2\), and they are Lagrangian.

- Drifters are essentially rafts - they float at the surface and move with the flow, collecting data and transmitting them to the satellites.
- Floats have an ability to control their density, and thus their depth, but subsurface location measurements are difficult - they need sonars for acoustic communication and the uncertainty can be large.
- Most currently deployed floats do not measure subsurface location - they move around at a fixed depth and take measurements with the position being unknown.

\(^2\)Courtesy Chris Jones
Section 1:  Lagrangian observations of the ocean

Floats drift and drifters float\(^2\), and they are Lagrangian.

Typical Argo Float Mission
10 day cycle, repeated until batteries run out

- Descent to Drift Depth
- Passive Drift ~9 days
- Ascent to Surface
  Record Profile
  CTD powered on to measure
  *in situ* temperature,
  salinity and pressure.

\(^2\)Courtesy Chris Jones

Amit Apte (ICTS-TIFR, Bangalore)
Lagrangian floats move in a dynamic velocity field

- Given a velocity field $v(x, t)$, the position $x_d$ of Lagrangian floats can be described by the following ODE:

$$\dot{x}_d = v(x_d, t)$$

- The velocity field itself evolves in time and could be coupled to other variables such as temperature, etc.

- Denote $x_f$ to be all these variables – “f” stands for “flow” variables – and their evolution by $\dot{x}_f = m_f(x_f)$

- E.g. if we have a spectral model for solving Navier-Stokes equations, $x_f$ would consist of the Fourier components, so that

$$v(x_d, t) = m_d(x_f(t), x_d)$$

where the right hand side is linear in $x_d$ but nonlinear in $x_f$, e.g.,

$$v(x_d, t) = x_{f1}(t) \sin(x_d) + x_{f2} \cos(x_d) + \ldots$$
Augmented model for Lagrangian data assimilation

- Recall the equations for the flow and drifter variables:

\[
\dot{x}_f = m_f(x_f) \quad \text{and} \quad \dot{x}_d = m_d(x_f(t), x_d)
\]

- Combining these together, we obtain the dynamical model for the “augmented” state space \(x = (x_f, x_d)\):

\[
\dot{x} = \begin{pmatrix} \dot{x}_f \\ \dot{x}_d \end{pmatrix} = \begin{pmatrix} m_f(x_f) \\ m_d(x_f, x_d) \end{pmatrix} = m(x)
\]

Lagrangian data assimilation = using measurements of drifter positions \(x_d\) to “predict” the flow variables \(x_f\)
Outline

1. Lagrangian observations of the ocean

2. Two important methods of data assimilation - particle and Kalman filter

3. Hybrid Kalman-particle filter: a few results with model problems

4. Outlook
Consider a stochastic dynamical model

\[ x_{t+1} = m(x_t) + \zeta_t \quad \text{with } x_0 \text{ unknown} \]

Thus we assume a probability density \( p^a(x_0) \) for the initial condition.

We will consider the problem of “estimating” the state \( x \) at some time \( t \) given observations at times \( 1, 2, \ldots, N \).
Section 2: Particle and Kalman filter

Nonlinear filtering $\equiv$ data assimilation

- Consider a stochastic dynamical model

\[ x_{t+1} = m(x_t) + \zeta_t \quad \text{with } x_0 \text{ unknown} \]

Thus we assume a probability density $p^a(x_0)$ for the initial condition.

- We will consider the problem of "estimating" the state $x$ at some time $t$ given observations at times $1, 2, \ldots, N$.

  - **Smoothing**: Obtain a state estimate $x_t$ for $t < N$ using all the observations up to time $N$; in particular, determine $x_0$.
  - **Filtering**: Obtain a state estimate $x_N$ using observations up to time $N$.
  - **Prediction**: Obtain a state estimate $x_t$ for $t > N$ (the time horizon of prediction is important).
Consider a stochastic dynamical model

\[ x_{t+1} = m(x_t) + \zeta_t \quad \text{with } x_0 \text{ unknown} \]

Thus we assume a probability density \( p^a(x_0) \) for the initial condition.

- We will consider the problem of “estimating” the state \( x \) at some time \( t \) given observations at times 1, 2, \ldots, \( N \).

- In most applications in earth sciences, data is collected “all the time” so the most relevant problem is of filtering.

- Predictions are obtained by using the filtering solution as “initial conditions” for the appropriate PDE of interest (hence the common view that data assimilation is the problem of finding initial conditions).
Or data assimilation $\equiv$ determination of posterior i.e. conditional distribution given the observations

Observations $y_t$ at time $t$ depend on the state at that time.

$$y_t = h(x_t) + \eta_t \quad t = 1, \ldots, N$$

$h$ is called the observation operator. $\eta_t$ is observational noise. Eventually we will assume independence between $\eta_t$ and $\zeta_t$.

Probabilistic statement of Data assimilation problem: find the posterior distribution of the state conditioned on the observations

- **Smoothing:** $p(x_t|y_1, y_2, \ldots, y_N)$ for $t < N$
- **Filtering:** $p(x_N|y_1, y_2, \ldots, y_N)$
- **Prediction:** $p(x_t|y_1, y_2, \ldots, y_N)$ for $t > N$
Filtering density: obtained in a two step process

A notation: \( y_{1:t} = \{y_1, y_2, \ldots, y_t\} \) and \( x_{1:t} = \{x_1, x_2, \ldots, x_t\} \)

The first step is “prediction”

- Suppose we have the probability \( p^a(x_{1:t} \mid y_{1:t}) \) of states \( x_{1:t} \) up to time \( t \) conditioned on observations \( y_{1:t} \) up to time \( t \), and recalling that \( x_{t+1} = m(x_t) + \zeta_t \) (which is a Markov chain, with transition kernel \( p^m(x_{t+1} \mid x_t) \))
- Then the probability \( p^f(x_{1:t+1} \mid y_{1:t}) \) of the states \( x_{1:t+1} \) up to time \( t + 1 \) conditioned on observations \( y_{1:t} \) up to time \( t \), is obtained by:

\[
p^f(x_{1:t}, x_{t+1} \mid y_{1:t}) = p(x_{1:t} \mid y_{1:t}) \cdot p(x_{t+1} \mid x_{1:t}, y_{1:t})
\]

\[
\downarrow \quad \downarrow
\]

\[
= p^a(x_{1:t} \mid y_{1:t}) \cdot p^m(x_{t+1} \mid x_t)
\]
Filtering density: obtained in a two step process

A notation: \( y_{1:t} = \{y_1, y_2, \ldots, y_t\} \) and \( x_{1:t} = \{x_1, x_2, \ldots, x_t\} \)

The next step is “update”

- Given the above probability \( p_f(x_{1:t+1}|y_{1:t}) \) of the states \( x_{1:t+1} \) up to time \( t + 1 \) conditioned on observations \( y_{1:t} \) up to time \( t \), and recalling \( y_{t+1} = h(x_{t+1}) + \eta_{t+1} \)

→ Then the probability \( p_a(x_{1:t+1}|y_{1:t+1}) \) of the states \( x_{1:t+1} \) up to time \( t + 1 \) conditioned on observations \( y_{1:t+1} \) up to time \( t + 1 \) is given by Bayes’ theorem:

\[
p_a(x_{1:t+1}|y_{1:t}, y_{t+1}) = p(x_{1:t+1}|y_{1:t}) \cdot p(y_{t+1}|x_{1:t+1}, y_{1:t}) \frac{1}{p(y_{t+1}|y_{1:t})}
\]

\[\downarrow \quad \downarrow\]

\( \propto p_f(x_{1:t+1}|y_{1:t}) \cdot p_{\eta}(y_{t+1}|x_{t+1}) \)
Filtering density satisfies a recursion relation

Putting together the two relations from previous slide:

- “prediction” given by

\[ p^f(x_{1:t}, x_{t+1}|y_{1:t}) = p^a(x_{1:t}|y_{1:t}) \cdot p^m(x_{t+1}|x_t) \]

- “update” given by

\[ p^a(x_{1:t+1}|y_{1:t}, y_{t+1}) \propto p^f(x_{1:t+1}|y_{1:t}) \cdot p^a(x_{1:t}|y_{1:t}) \cdot p_\eta(y_{t+1}|x_{t+1}) \]

we obtain the following recursive relation for the posterior distribution

\[ p^a(x_{1:t+1}|y_{1:t+1}) \propto p^a(x_{1:t}|y_{1:t}) \cdot p^m(x_{t+1}|x_t) \cdot p_\eta(y_{t+1}|x_{t+1}) \]

where \( p_\eta(y_{t+1}|x_{t+1}) \) is the observational noise and \( p^m(x_{t+1}|x_t) \) is the Markov transition Kernel for the dynamical model.
Particle filter: a “weighted sample” representation of the filtering recursion

\[ p^a(x_{1:t+1}|y_{1:t+1}) \propto p^a(x_{1:t}|y_{1:t}) \cdot p^m(x_{t+1}|x_t) \cdot p_\eta(y_{t+1}|x_{t+1}) \]

- Suppose we have a weighted sample \( \{x_t^i, w_t^i\}, i = 1, \ldots, N \) from \( p^a(x_t|y_{1:t}) \), i.e., we approximate \( p^a(x_t|y_{1:t}) \approx \sum_{i=1}^N w_t^i \delta(x_t - x_t^i) \).
- If \( x_{t+1}^i \) is a sample from a “importance sampling density” \( q(x_{1+1}|x_t^i) \), then the weighted sample \( \{x_{t+1}^i, w_{t+1}^i\}, i = 1, \ldots, N \) approximates the posterior at time \( t + 1 \) if we choose

\[ w_{t+1}^i \propto w_t^i \cdot \frac{p^m(x_{t+1}^i|x_t^i) \cdot p_\eta(y_{t+1}|x_{t+1}^i)}{q(x_{1+1}^i|x_t^i)} \]

This is the main idea behind particle filtering.
Kalman filter: a “two moment” representation of the Gaussian posterior in case of linear model

Suppose the model is linear $m(x) = Mx$, the observation operator is linear $h(x) = Hx$, the initial distribution for $x_0$ is Gaussian, as are the stochasticity in the observations $\eta_t$ and in the dynamical model $\zeta_t$.

Kalman filter gives a recursion relation for the mean and covariance:

- (assumed) Gaussian posterior is expressed as (xₐ, Cₐ) for $p^a(x_t|y_{1:t})$ and (xᵢ, Cᵢ) for $p^f(x_{t+1}|y_{1:t})$:
  - “Update step” given by
    
    $$
    x^a_t = x^f_t + K(y_t - Hx^f_t) \quad \text{and} \quad C^a_t = (I - KH)C^f_t
    $$

  - Here $K = P^f_t H^T (HP^f_t H^T + R)^{-1}$ is the Kalman gain matrix
  - “Prediction step” given by
    
    $$
    x^f_{t+1} = Mx^a_t \quad \text{and} \quad C^f_{t+1} = MC^a_t M^T
    $$

- Ensemble Kalman filter is the Monte Carlo version of this filter, where the Gaussian distributions are represented using samples.
Ensemble based methods make use of an ensemble of states to represent the uncertainty.
Ensemble based methods make use of an ensemble of states to represent the uncertainty.
Outline

1. Lagrangian observations of the ocean
2. Two important methods of data assimilation - particle and Kalman filter
3. Hybrid Kalman-particle filter: a few results with model problems
4. Outlook
Contrasting the properties of particle and Kalman filters

On one hand,
- Ensemble Kalman filter is obviously designed for linear systems - for highly nonlinear systems, it fails to represent the true posterior distribution,
- but it seems to work well even in high \((10^3 - 10^6)\) dimensional systems with very small ensembles (\(\sim 100\)) [no theoretical understanding of this phenomena].

On the other hand,
- Particle filter is a sampling method, and its errors grow exponentially with the dimension of the state space,
- but clearly, it does not have any restrictions about the dynamics being linear (or even stochastic)
Some features of the augmented dynamical system for Lagrangian data assimilation

\[ \dot{x} = \begin{pmatrix} \dot{x}_f \\ \dot{x}_d \end{pmatrix} = \begin{pmatrix} m_f(x_f) \\ m_d(x_f, x_d) \end{pmatrix} = m(x) \]

- \( x_f \) is typically very high dimensional (its an approximation of an infinite dimensional dynamical system) – computationally “expensive” and large sampling errors – whereas \( x_d \) is 2-3 dimensional – computationally “cheap” and small sampling errors.

- The dynamics of the drifters \( x_d \) is highly nonlinear – large errors with a linear approximation – whereas that of the flow \( x_f \) is not so nonlinear (on the time scale of interest) – smaller errors with a linear approximation.

- It is natural to use EnKF for \( x_f \) while using a particle filter for \( x_d \)!

Q.: How do we consistently update weights and the mean and covariance to obtain a good approximation of the posterior?
The samples in the hybrid particle-Kalman filter scheme

The weighted sample consists of multiple drifters for each flow field:
\[ \{(x_f^i, x_d^{i,j}), w^{i,j}\} \text{ for } i = 1, \ldots, N_f \text{ and } j = 1, \ldots, N_d \]

- The area of the circle is proportional to its weight
- The shaded circles represent the marginal for \(x_f\) after integrating out (i.e., summing it) the other variables \(x_d\).

Thus, the full distribution is approximated by

\[
p(x_f, x_d) \approx \sum_{i,j=1}^{N_f,N_d} w^{i,j} \delta(x_f - x_f^i) \delta(x_d - x_d^{i,j})
\]
Filtering update in the hybrid scheme

1. Update the flow ensemble members $x^i_f$ using the observation $y$ with the EnKF update step given earlier.
2. Update the weights $w^{i,j}$ associated to each drifter sample $x^{i,j}_d$.
3. “Resample” the flow and drifter variables and set their weights to be constant $w = 1/(N_f N_d)$

We have compared this method with the standard particle filter and with ensemble Kalman filter in two model problems: a low-dimensional linear velocity field, and a high-dimensional nonlinear velocity flow.
Some results using linear shallow water equations

For two dimensional velocity \((u, v)\) and height \(h\) fields:

\[
\begin{align*}
\frac{\partial u}{\partial t} &= v - \frac{\partial h}{\partial s_1}, \\
\frac{\partial v}{\partial t} &= -u - \frac{\partial h}{\partial s_2}, \\
\frac{\partial h}{\partial t} &= -\frac{\partial u}{\partial s_1} - \frac{\partial v}{\partial s_2},
\end{align*}
\]

We seek periodic solutions on \(\mathbb{R}^2\) in \(u, h\), specifically, the following Fourier modes:

\[
\begin{align*}
u(s_1, s_2, t) &= -2\pi l \sin(2\pi k s_1) \cos(2\pi l s_2) u_0 + \cos(2\pi m s_2) u_1(t) \\
v(s_1, s_2, t) &= 2\pi k \cos(2\pi k s_1) \sin(2\pi l s_2) u_0 + \cos(2\pi m s_2) v_1(t) \\
h(s_1, s_2, t) &= \sin(2\pi k s_1) \sin(2\pi l s_2) u_0 + \sin(2\pi m s_2) h_1(t)
\end{align*}
\]
The amplitudes satisfy the following:

\[ \dot{u}_0 = 0, \quad \dot{u}_1 = v_1, \]
\[ \dot{v}_1 = -u_1 - 2\pi m h_1, \quad \dot{h}_1 = 2\pi m v_1. \]

The observations are the positions of the drifters that satisfy:

\[ \dot{s}_1(t) = u(s_1(t), s_2(t), t), \quad \dot{s}_2(t) = v(s_1(t), s_2(t), t). \]

Numerical experiments consisted of the following:

- Generate noisy observations from a long trajectory of the drifter
- Assimilate these observations with
  - A hybrid filter
  - An Ensemble Kalman filter
  - A particle filter with a large number of samples
- Compare the posterior mean with the true values
Hybrid filter performs almost as well as particle filter

- Two top right panels: errors in flow and position
- Two bottom panels: true trajectory and mean values for the three filters
High-dimensional model problem

Quasi-geostrophic shallow water equations for the velocity field

\[
\left[ \frac{\partial}{\partial t} - \frac{\partial \eta}{\partial y} \frac{\partial}{\partial x} + \frac{\partial \eta}{\partial x} \frac{\partial}{\partial y} \right] \Delta \eta = F(x, y, t)
\]

\[ u(x, y, t) = -\frac{\partial \eta}{\partial y}, \quad v(x, y, t) = \frac{\partial \eta}{\partial x} \]

The observations are again the positions of the drifters that satisfy:

\[ \dot{s}_1(t) = u(s_1(t), s_2(t), t), \quad \dot{s}_2(t) = v(s_1(t), s_2(t), t) \]

Numerical experiments again consisted of

- A hybrid filter
- An Ensemble Kalman filter
- A “free run” without any filtering
Hybrid filter is effective in tracking the “truth”

True height and velocity fields

Mean of the free run

True drifter trajectory

Mean of the EnKf

Mean of the hybrid filter
Hybrid filter is effective in tracking the “truth”
Outline

1. Lagrangian observations of the ocean
2. Two important methods of data assimilation - particle and Kalman filter
3. Hybrid Kalman-particle filter: a few results with model problems
4. Outlook
Summary:

- Lagrangian data assimilation is the problem of using data from Lagrangian/passive instruments (e.g. drifters and gliders)
- **Particle filtering** and **Kalman filtering** are two complementary data assimilation methods which are
  - ineffective in high dimensional and nonlinear problems, respectively, but
  - effective in nonlinear problems and high dimensions, respectively.
- Hybrid particle-Kalman filter that I discussed combines the strengths of both, for the Lagrangian data assimilation problem.

Some questions:

- high (infinite) dimensional methods for resampling velocity field
- Use of dynamical information, e.g., unstable manifolds; Lyapunov exponents and vectors; etc. for improving filter performance

---

2 Laura Slivinski et al. “A Hybrid Particle-Ensemble Kalman Filter for Lagrangian Data Assimilation”. In: Tellus ?? (2014), ??
3 Laura Slivinski, Elaine Spiller, and Amit Apte. “A hybrid particle-ensemble Kalman filter for high dimensional Lagrangian data assimilation”. In: (). submitted