Particle Filters for very-high-dimensional systems

Peter Jan van Leeuwen and Mel Ades
Data-Assimilation Research Centre DARC
University of Reading

Oxford-September 2012
Nonlinear filtering: Particle filter

\[ p(x|y) = \frac{p(y|x)p(x)}{\int p(y|x)p(x) \, dx} \]

Use ensemble

\[ p(x) = \sum_{i=1}^{N} \frac{1}{N} \delta(x - x_i) \]

\[ p(x|y) = \sum_{i=1}^{N} w_i \delta(x - x_i) \]

with

\[ w_i = \frac{p(y|x_i)}{\sum_j p(y|x_j)} \]

the weights.
Why are particle filters degenerate I

Probability space in large-dimensional systems is ‘empty’: the curse of dimensionality
Why are Particle Filters degenerate II

• The volume of a hypersphere of radius $r$ in an $M$ dimensional space is

\[ V \propto \frac{r^M}{\Gamma(M/2 - 1)} \]

• Taking for the radius $r \approx 3\sigma_y$ we find, using Stirling:

\[ V \propto \left( \frac{9\sigma_y}{M/2} \right)^{M/2} \]

• So very small indeed.
Why are Particle Filters degenerate III

For the optimal proposal density we find, for Gaussian process model and Gaussian observation errors:

\[ w_i \propto p(y^n | x_i^{n-1}) \]

\[ \propto \exp \left[ -\frac{1}{2} (y^n - H f(x_i^{n-1}))(H Q H^T + R)^{-1} \right. \]

\[ \left. \times (y^n - H f(x_i^{n-1})) \right] . \]

Ignoring covariances we find:

\[ \text{var}[-\log(w_i)] \propto \frac{M}{2} \left( \frac{V_x}{V_{\beta} + V_y} \right)^2 \left( 1 + 2 \left( \frac{V_y + V_{\beta}}{V_x} \right) \right) \]
Why are Particle Filters degenerate?

• ‘Number of particles needed grows exponentially with dimension of the state vector (Bickel et al, 2007).’

• A slightly different view: degeneracy due to number of independent observations.

• This is related to the extremely narrow likelihood, a tiny move of a particle gives a completely different weight.
The statistics

• The Stochastic PDE: $$x^n = f(x^{n-1}) + \beta^{n-1}$$

• Observations: $$y^n$$

• Relation between the two: $$y^n = H(x^n) + \epsilon^n$$

Assume: $$\beta \sim N(0, Q)$$
$$\epsilon \sim N(0, R)$$
$$H$$ is linear
Proposal density

We can run a different model with deterministic and Stochastic term drawn from proposal density $q$.

With a proposal density, the weights change to:

$$w_i = \frac{p(y^n | x^n_i)}{p(y^n)} \frac{p(x^n_i | x^{n-1}_i)}{q(x^n_i | x^{n-1}_i, y^n)}$$
The Equivalent-Weights Particle Filter

- Use simple proposal at each time step, e.g. ‘nudging’.

- Use different proposal at final time step to ensure that weights are very similar.
Proposal density between observations

We can explore the fact that the model needs several $O(100)$ time steps between observations, e.g. by using a relaxation term in the proposal:

$$q(x^n | x_{i}^{n-1}, y^m) = N \left( f(x_{i}^{n-1}) + S \left( y^m - H(x_{i}^{n-1}) \right), Q \right)$$

Corresponding to an evolution equation for each particle

$$x_{i}^{n} = f(x_{i}^{n-1}) + \hat{\beta}_{i}^{n} + S \left( y^n - H(x_{i}^{n-1}) \right)$$
Proposal density between observations

• One could also use the ‘optimal proposal density’ between observations.

• This can be implemented as a minimization method for each particle, and is also known as the Implicit Particle Filter.

• This is related to a method called 4DVar in meteorology and oceanography, which explores only the mode of the joint-in-time pdf.
Proposal density at observation time:
the essence of the Equivalent-Weights Particle Filter

The proposal density depends on the maximum weight from a deterministic particle can achieve during the last time step:

\[ q(x^n | x^{n-1}_i, y^n) = \begin{cases} q_1(x^n | x^{n-1}_i, y^n) & \text{if } w_i^{\text{max}} > w_{\text{target}} \\ q_2(x^n | x^{n-1}_i, y^n) & \text{if } w_i^{\text{max}} < w_{\text{target}} \end{cases} \]

The target weight is set by the user, as e.g. the weight that 80% of the particles can achieve.
The maximum weights

1. We know:

\[ w_i = \frac{p(y^n | x_i^n)}{p(y^n)} \frac{p(x_i^n | x_i^{n-1})}{q(x_i^n | x_i^{n-1}, y^n)} \]

2. Write down expression for each weight ignoring proposal:

\[ w_i \propto w_i^{rest} \exp \left[ -\frac{1}{2} \left( x_i^n - f(x_i^{n-1}) \right)^T Q^{-1} \left( x_i^n - f(x_i^{n-1}) \right) \right. \]

\[ \left. - \frac{1}{2} (y^n - H(x_i^n))^T R^{-1} (y^n - H(x_i^n)) \right] \]

3. When H is linear this is a quadratic function in \( x_i^n \) for each particle. Otherwise linearize.
The target weight

$- \log w_i$

$X_i^n$

Target weight
The Equivalent-Weights Particle Filter

The proposal density is chosen as:

\[
q(x^n|x_i^{n-1}, y^n) = \begin{cases} 
q_1(x^n|x_i^{n-1}, y^n) & \text{if } w_i^{\text{max}} > w^{\text{target}} \\
q_2(x^n|x_i^{n-1}, y^n) & \text{if } w_i^{\text{max}} < w^{\text{target}}
\end{cases}
\]

The target weight is set by the user, as e.g. the weight that 80% of the particles can achieve.

For particles that cannot reach the target weight we use:

\[
q_2(x^n|x_i^{n-1}, y^m) = N\left(f(x_i^{n-1}), Q\right)
\]
The two proposal densities

For particles that can reach the target weight we use:

\[
q_1(x^n|x_{i}^{n-1}, y^m) = (1-\epsilon)U(\hat{x}_{i} - \gamma_U Q^{1/2}1, \hat{x}_{i} + \gamma_U Q^{1/2}1) \\
+ \epsilon N(\hat{x}_{i}, \gamma_N^2 Q)
\]

This is a mixture density:
The deterministic move

Determine $\alpha$ at crossing of line with target weight contour in:

$$\hat{x}_i = f(x_i^{n-1}) + \alpha_i K \left( y^n - H(f(x_i^{n-1})) \right)$$

with

$$K = QH^T(HQH^T + R)^{-1}$$
The stochastic part of the proposal

A draw from the uniform density gives:

\[ w_i = \frac{|Q|^{1/2} (2\gamma_U)^k}{1 - \epsilon} w_i^{\text{rest}} p(x_i^n | x_i^{n-1}) p(y^n | x_i^n) \]

A draw from the Gaussian density gives:

\[ w_i = \frac{w_i^{\text{rest}} p(x_i^n | x_i^{n-1}) p(y^n | x_i^n)}{(2\pi)^{k/2} |\gamma_N^2 Q|^{1/2} \exp \left( -\frac{1}{2} \gamma_U d_i \beta_i^n (\gamma_U^2 Q)^{-1} \gamma_U d_i \beta_i^n \right)} \]

The ratio between the two is (ignoring the exp part):

\[ \frac{(2\pi)^{k/2} |\gamma_N^2 Q|^{1/2}}{(1 - \epsilon) \frac{\epsilon}{|Q|^{1/2} (2\gamma_U)^k}} \]

which can be made equal to one when:

\[ \gamma_N = \frac{2^{k/2} \epsilon}{\pi^{k/2} (1 - \epsilon) \gamma_U^k} \]
Equivalent-Weights Particle Filter

- Use relaxation-term proposal up to last time step
- Calculate $w_i^{max}$ and target weight (e.g. 80%)
- Calculate deterministic moves for high-weight particles:
  \[ \hat{x}_i = f(x_{i}^{n-1}) + \alpha_i K \left( y^n - H(f(x_{i}^{n-1})) \right) \]
- Determine stochastic move
  \[ p(\hat{\beta}_i^{n-1}) \propto (1 - a)U[-b, b] + aN(0, \hat{Q}) \]
- Calculate new weights and resample ‘lost’ particles
How essential are Gaussian assumptions?

• Allows for analytical expressions.
• But no real need.
• $w_{i}^{max}$ calculations do not have to be very accurate.
• Same for $w^{target}$.
• Deterministic move has to be very accurate, good iterative schemes should be used.
Application: the barotropic vorticity equation

- Stochastic barotropic vorticity equation:
  \[
  \frac{\partial q}{\partial t} + u \cdot \nabla q = F
  \]

- 256 by 256 grid - 65,536 variables
- Doubly periodic boundary conditions
- Semi-Lagrangian time stepping scheme
- Twin experiments
- Observations every 50 time steps – decorrelation time of 42
- 32 particles
Fully observed system

True model state

Mean of particles

Time step 600

Time step 1150
\( \frac{1}{4} \) Observations over half of state
Individual particles are not smooth.
The update of the unobserved part

Particle 23 before update  Particle 23 after update  Difference
Time evolution for different relaxation strengths
¼ observations over half of state

Variance

(Mean – Truth)²

Time step 600

Time step 1150
Convergence of the pdf’s

32 particles  128 particles  512 particles
Rank histograms

Full state observed

¼ of half state observed
Miss-specification of process noise

Truth: $L_Q = 5$ gridpoints

PF: $L_Q = 9$ gridpoints
Miss-specification of process noise

Truth: Q=0.025

EW-PF: Q=0.1
Conclusions

• Particle filters do not need state covariances.

• Degeneracy is related to number of observations, not to size of the state space.

• Proposal density allows enormous freedom.

• Equivalent-weights scheme solves dimensionality problem?

• Other efficient schemes are being derived.

• Future plans: numerical weather prediction, climate forecasting.
We need more people!

• We still have places on the *Data Assimilation and Inverse Methods in Geosciences MSc programme*