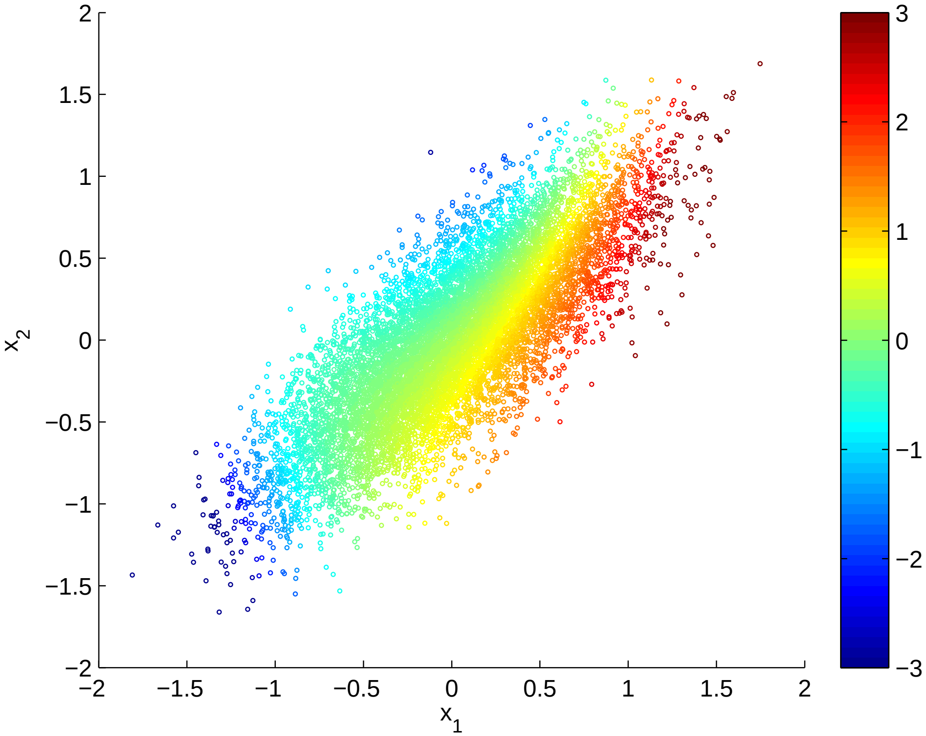
Ensemble Filtering in the Presence of Nonlinearity and Non-Gaussianity



- Chris Snyder
  National Center for Atmospheric Research, Boulder Colorado, USA

Preliminaries

Notation

- follow Ide et al. (1997) generally, except:
  $\ldots$ $\dim(\mathbf{x}) = N_x$, $\dim(\mathbf{y}) = N_y$
  $\ldots$ subscript $j{:}k$ indicates times $t_j, t_{j+1}, \ldots, t_k$,
  $\ldots$ superscripts index ensemble members, or iterations
- $\sim$ means "distributed as," e.g. $x \sim N(0,1)$
- state evolution: $\mathbf{x}_k = M(\mathbf{x}_{k-1}) + \eta_k$
- observations: $\mathbf{y}_k = H(\mathbf{x}_k) + \epsilon_k$

Basic Facts

1. Conditional pdf $p(\mathbf{x}_k|\mathbf{y}_{1:k})$ is the answer
   - ▷ summarizes everything that can be known about state
   - ▷ calculate sequentially, via Bayes rule,

$$p(\mathbf{x}_k|\mathbf{y}_{1:k}) = p(\mathbf{y}_k|\mathbf{x}_k)p(\mathbf{x}_k|\mathbf{y}_{1:k-1})/p(\mathbf{y}_{1:k})$$

   - ▷ algorithms that do not produce $p(\mathbf{x}_k|\mathbf{y}_{1:k})$ cannot be fully optimal

Basic Facts (cont.)

2. Linear, Gaussian systems are relatively easy
   - ▷ $p(\mathbf{x}_k|\mathbf{y}_{1:k})$ is Gaussian and thus determined by its mean and covariance
   - ▷ posterior (analysis) mean is linear in prior (background) mean and observations
   - ▷ no need to choose between posterior mean (min variance) and posterior mode (max likelihood) as "best" estimate; they are equal.
   - ▷ 4D-Var and Kalman filter (KF) agree; so does ensemble KF (EnKF) up to sampling error.

Basic Facts (cont.)

3. High-dimensional pdfs are hard
   - $p(\mathbf{x}_k|\mathbf{y}_{1:k})$ is a continuous fn of N_x variables. Direct approaches not feasible; discretization with n points per variable requires n^{N_x} d.o.f.
   - they are extraordinarily diffuse

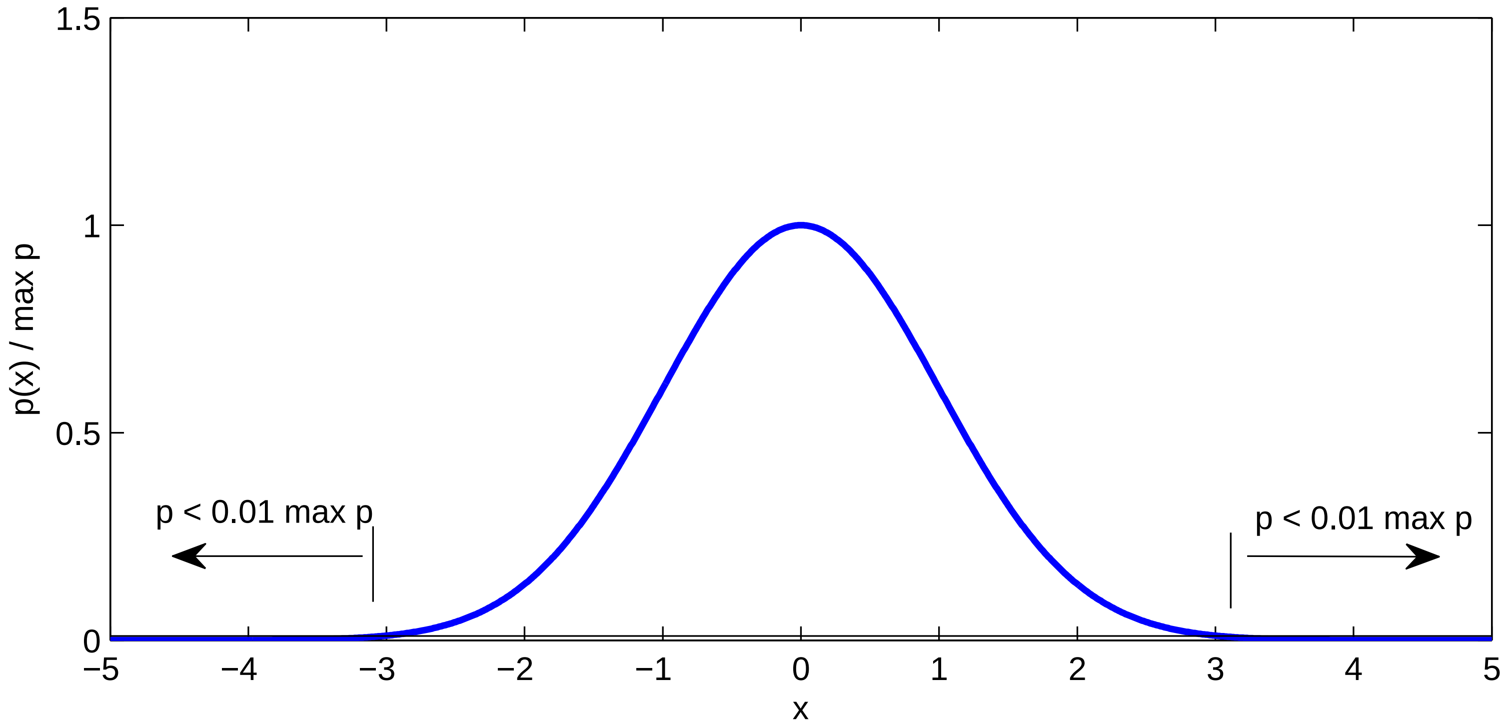
Basic Facts (cont.)

3. High-dimensional pdfs are hard

▷ $p(\mathbf{x}_k|\mathbf{y}_{1:k})$ is a continuous fn of N_x variables. Direct approaches not feasible; discretization with n points per variable requires n^{N_x} d.o.f.

▷ they are extraordinarily diffuse

Consider $\mathbf{x} \sim N(0, \mathbf{I})$.



Basic Facts (cont.)

3. High-dimensional pdfs are hard

   - $p(\mathbf{x}_k|\mathbf{y}_{1:k})$ is a continuous fn of N_x variables. Direct approaches not feasible; discretization with n points per variable requires n^{N_x} d.o.f.
   - they are extraordinarily diffuse

     Consider $\mathbf{x} \sim N(0, \mathbf{I})$.
     1 dimension: points with $p(\mathbf{x})$ less than 0.01 of max account for less than 1% of mass of pdf.
     10 dimensions: they account for about 1/2.

Outline

Nonlinearity and the ensemble Kalman filter (EnKF)

- Relation to the BLUE
- Iterative schemes

Particle filters

- Required N_e grows exponentially w/ "problem size"
- Importance sampling and the optimal proposal density

Outline

Nonlinearity and the ensemble Kalman filter (EnKF)

- ▷ Relation to the BLUE
- ▷ Iterative schemes

Particle filters

- ▷ Required N_e grows exponentially w/ "problem size"
- ▷ Importance sampling and the optimal proposal density

Not a comprehensive review!

The Best Linear Unbiased Estimator (BLUE)

Desire an estimate of $\mathbf{x}$ given observation $\mathbf{y} = H(\mathbf{x}) + \epsilon$

- Consider linear estimators, $\hat{\mathbf{x}} = \mathbf{A}\mathbf{y} + \mathbf{b}$
- Which $\mathbf{A}$ and $\mathbf{b}$ minimize $E(|\mathbf{x} - \hat{\mathbf{x}}|^2)$?

The BLUE (cont.)

The BLUE is the answer

- Let $\bar{\mathbf{x}} = E(\mathbf{x})$ and $\bar{\mathbf{y}} = E(\mathbf{y}) = E(H(\mathbf{x}))$
- Then BLUE is given by (e.g. Anderson and Moore 1979)

$$\hat{\mathbf{x}} = \bar{\mathbf{x}} + \mathbf{K}\left(\mathbf{y} - \bar{\mathbf{y}}\right), \quad \mathbf{K} = \mathrm{cov}\left(\mathbf{x}, \mathbf{y}\right)\mathrm{cov}(\mathbf{y})^{-1}$$

- Only need 1st and 2nd moments; no requirement that $\mathbf{x}$, ϵ are Gaussian or H is linear

Useful benchmark for nonlinear, non-Gaussian systems

- ... though $E(\mathbf{x}|\mathbf{y})$ has smaller expected squared error

Relation of EnKF to the BLUE

Start with $\mathbf{x}^f$ drawn from $p(\mathbf{x})$

EnKF update specifies a random, linear fn of $\mathbf{x}^f$ and $\mathbf{y}$

- EnKF:

$$\mathbf{x}^a = \mathbf{x}^f + \mathbf{K}\left(\mathbf{y} - H(\mathbf{x}^f) - \epsilon\right)$$

$$\mathbf{K} = \mathrm{cov}\left(\mathbf{x}_k, H(\mathbf{x}_k)\right)\left[\mathrm{cov}(H(\mathbf{x}_k)) + \mathbf{R}\right]^{-1}$$

- $\mathbf{x}^a$ has mean and covariance matrix given by BLUE formulas
- $\mathbf{x}^a$ need not be Gaussian
- in linear, Gaussian case, $\mathbf{x}^a$ has same distribution as $\mathbf{x}_k|\mathbf{y}_{1:k}$

Relation of EnKF to the BLUE

Start with $\mathbf{x}^f$ drawn from $p(\mathbf{x})$

EnKF update specifies a random, linear fn of $\mathbf{x}^f$ and $\mathbf{y}$

- EnKF:

$$\mathbf{x}^a = \mathbf{x}^f + \mathbf{K}\left(\mathbf{y} - H(\mathbf{x}^f) - \epsilon\right)$$

$$\mathbf{K} = \mathrm{cov}\left(\mathbf{x}_k, H(\mathbf{x}_k)\right)\left[\mathrm{cov}(H(\mathbf{x}_k)) + \mathbf{R}\right]^{-1}$$

- $\mathbf{x}^a$ has mean and covariance matrix given by BLUE formulas
- $\mathbf{x}^a$ need not be Gaussian
- in linear, Gaussian case, $\mathbf{x}^a$ has same distribution as $\mathbf{x}|\mathbf{y}$

> The EnKF is a Monte-Carlo implementation of the BLUE and, as $N_e \to \infty$, shares its properties.

Relation of EnKF to the BLUE

Start with $\mathbf{x}^f$ drawn from $p(\mathbf{x})$

EnKF update specifies a random, linear fn of $\mathbf{x}^f$ and $\mathbf{y}$

- EnKF:

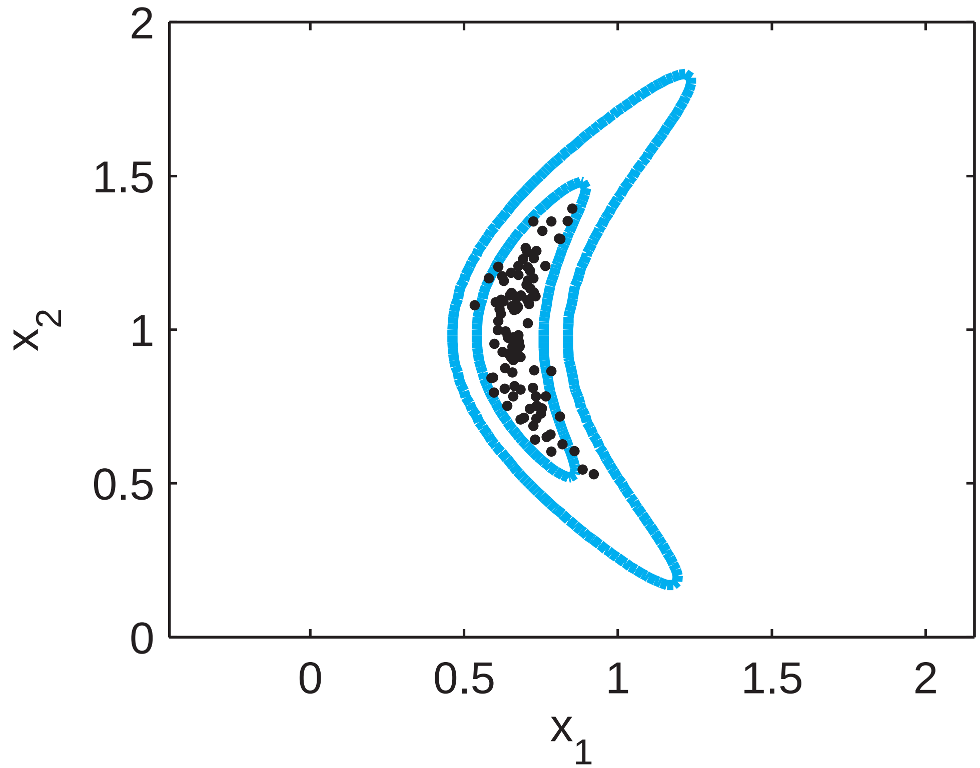
$$\mathbf{x}^a = \mathbf{x}^f + \mathbf{K}\left(\mathbf{y} - H(\mathbf{x}^f) - \epsilon\right)$$
$$\mathbf{K} = \mathrm{cov}\left(\mathbf{x}_k, H(\mathbf{x}_k)\right)\left[\mathrm{cov}(H(\mathbf{x}_k)) + \mathbf{R}\right]^{-1}$$

- $\mathbf{x}^a$ has mean and covariance matrix given by BLUE formulas
- $\mathbf{x}^a$ need not be Gaussian
- in linear, Gaussian case, $\mathbf{x}^a$ has same distribution as $\mathbf{x}|\mathbf{y}$

The EnKF is a linear method. It is optimal for linear, Gaussian systems but does *not* assume Gaussianity.

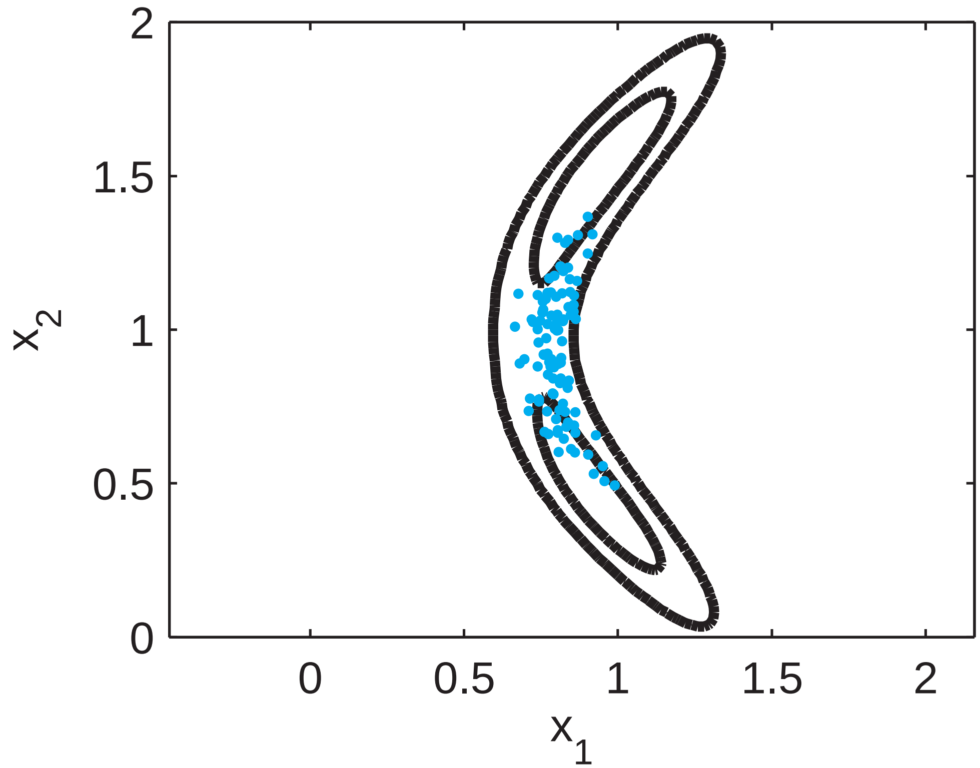
BLUE/EnKF Illustrated

- $p(\mathbf{x})$ and ensemble



BLUE/EnKF Illustrated

- $p(\mathbf{x}|y)$ for $y = x_1 + \text{noise} = 1.1$ and EnKF analysis ensemble (dots)



- sample retains non-Gaussian curvature but does not capture bimodality

EnKF and Non-Gaussianity

Different EnKF schemes respond differently

- All variants of EnKF produce same sample mean and 2nd moment
- Other (non-Gaussian) aspects of updated ensemble depend on specific scheme
- Deterministic/"square root" filters are more sensitive to non-Gaussianity (Lawson and Hansen 2004, Lei et al. 2010)

Nonlinear update in observation space

- EnKFs that process obs one at a time can be written as update of observed quantity followed by regression onto state variables.
- Observation update is scalar and can use fully nonlinear techniques (Anderson 2010)

Iterative, Ensemble-Based Schemes

Motivation for iterations

- EnKF is a linear scheme
- Mean and mode of $\mathbf{x}_k | \mathbf{y}_{1:k}$ are nonlinear fns of $\mathbf{y}_{1:k}$; iteration is natural for weak nonlinearity (e.g. 4DVar)

Can EnKF be improved through iteration?

How to formulate iterations?

Iterative, Ensemble-Based Schemes (cont.)

Several ideas

- Minimize non-quadratic $J(\mathbf{x})$ with $\mathbf{x}$ restricted to ensemble subspace (Zupanski 2005)
- Perform series of N assimilations, each using same $\mathbf{y}_{1:k}$ but with obs-error covariance $N^{-1}\mathbf{R}$; first analysis provides prior for second, etc. (Annan et al. 2005)
- Repeated application of EnKF update, mimicking the outer loop of 4DVar (Kalnay and Yang 2010)

Iterative, Ensemble-Based Schemes (cont.)

Several ideas

- Minimize non-quadratic $J(\mathbf{x})$ with $\mathbf{x}$ restricted to ensemble subspace (Zupanski 2005)
- Perform series of N assimilations, each using same $\mathbf{y}_{1:k}$ but with obs-error covariance $N^{-1}\mathbf{R}$; first analysis is provides prior for second, etc. (Annan et al. 2005)
- **Repeated application of EnKF update, mimicking the outer loop of 4DVar (Kalnay and Yang 2010)**

4DVar and an Iterated Ensemble Smoother

Incremental 4DVar $\equiv$ sequence of Kalman smoothers

- Linearization of M and H about $\mathbf{x}^n$ makes inner-loop $\hat{J}(\delta x)$ quadratic; thus minimization of $\hat{J}$ is equivalent to Kalman smoother
- nth Kalman-smoother update is

$$\mathbf{x}_0^{n+1} = \mathbf{x}^f{}_0 + \mathbf{K}_{0|1:N_t}\left[\mathbf{y}_{1:N_t} - \left(H(\mathbf{x}^n_{1:N_t}) + \mathbf{H}(\mathbf{x}^f{}_{1:N_t} - \mathbf{x}^n_{1:N_t})\right)\right]$$

- see also Jazwinski (1970, section 9.7)

4DVar and an Iterated Ensemble Smoother

Incremental 4DVar $\equiv$ sequence of Kalman smoothers

- nth Kalman-smoother update is

$$\mathbf{x}_0^{n+1} = \mathbf{x}^f{}_0 + \mathbf{K}_{0|1:N_t}\left[\mathbf{y}_{1:N_t} - \left(H(\mathbf{x}_{1:N_t}^n) + \mathbf{H}(\mathbf{x}^f{}_{1:N_t} - \mathbf{x}_{1:N_t}^n)\right)\right]$$

Approximate iterated KS using ensemble ideas

- Make usual replacements:

$$\mathbf{H}\delta\mathbf{x}_k^f \approx H(\mathbf{x}^f{}_k) - H(\mathbf{x}_k^n),$$
$$\mathbf{K}_{0|1:N_t} \approx \hat{\mathbf{K}}_{0|1:N_t} = \mathrm{cov}(\mathbf{x}_0, H(\mathbf{x}_{1:N_t}))[\mathrm{cov}(H(\mathbf{x}_{1:N_t})) + \mathbf{R}_{1:N_t}]^{-1}$$

- Ensemble ICs drawn from $N(\mathbf{x}_0^n, \mathbf{P}_0^f)$ to approximate linearization about $\mathbf{x}^n$ in H and M.
- Ensemble mean at iteration $n+1$ given by

$$\mathbf{x}_0^{n+1} = \mathbf{x}^f{}_0 + \hat{\mathbf{K}}_{0|1:N_t}^n(\mathbf{y}_{1:N_t} - \overline{H(\mathbf{x}_{1:N_t})}\,)$$

- Same as usual update, but gain changes at each iteration

Kalnay-Yang Iteration for Ensemble KS

"Running in place" from Kalnay and Yang (2010)

- Ensemble mean at iteration $n+1$ given by
$$\mathbf{x}_0^{n+1} = \mathbf{x}_0^n + \hat{\mathbf{K}}_{0|1:N_t}^n(\mathbf{y}_{1:N_t} - \overline{H(\mathbf{x}_{1:N_t})}\,)$$
- Innovation is recalculated using most recent guess and gain changes at each iteration
- Intended to speed spin up of EnKS when initial estimate of $\mathbf{P}_0^f$ is poor

Converges to observations when H and M are linear

- Let $\mathbf{L}^n = \mathbf{I} - \mathbf{H}^T\hat{\mathbf{K}}_{0|1:N_t}^n$. Easy to show
$$\mathbf{H}\mathbf{x}_0^{n+1} = \left(\prod_{m=1}^{n}\mathbf{L}^m\right)\mathbf{H}\mathbf{x}^f{}_0 + \left(\mathbf{I} - \prod_{m=1}^{n}\mathbf{L}^m\right)\mathbf{y}$$
- Properties in nonlinear case are unclear

Simple Example: Hénon Map

Hénon map

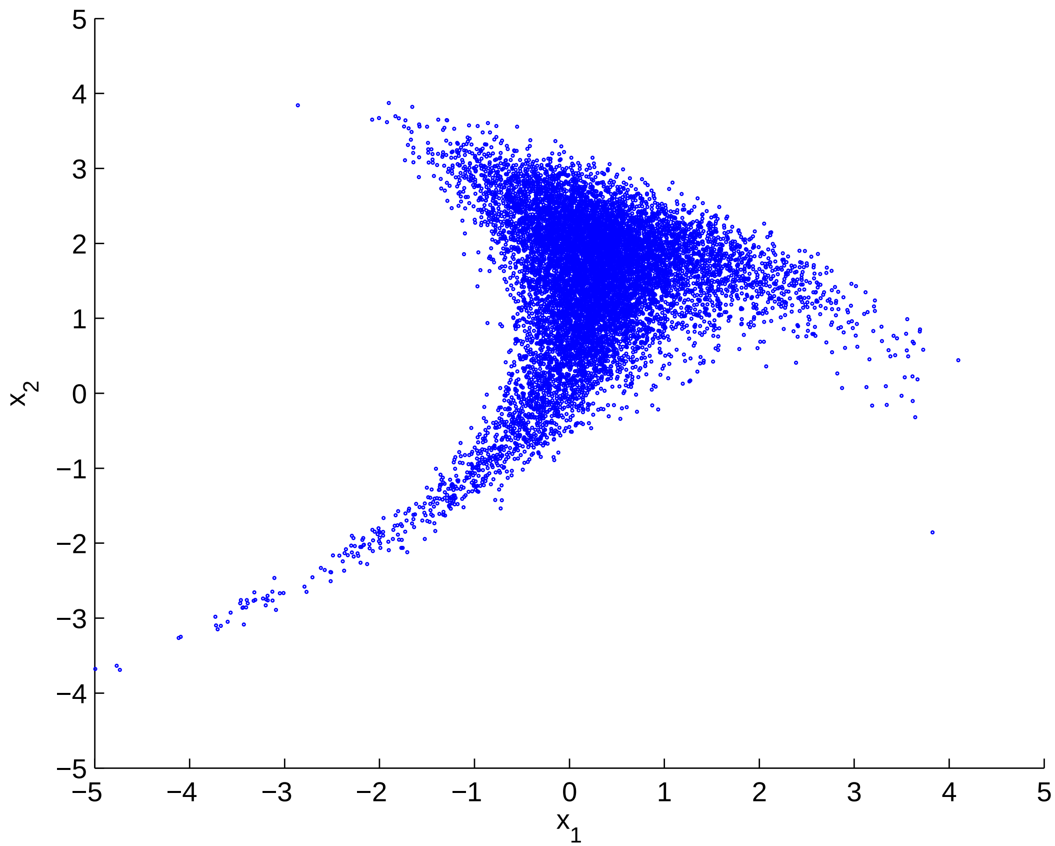
- state is 2d, $\mathbf{x} = (x_1, x_2)$
- iterate map twice in results here
- Note: subscripts denote components of $\mathbf{x}$!

An example

- Gaussian ICs at t_0 ("initial time")
- observe $y = x_1 + \epsilon$ at t_1 ("final time")
- update state at t_0, t_1

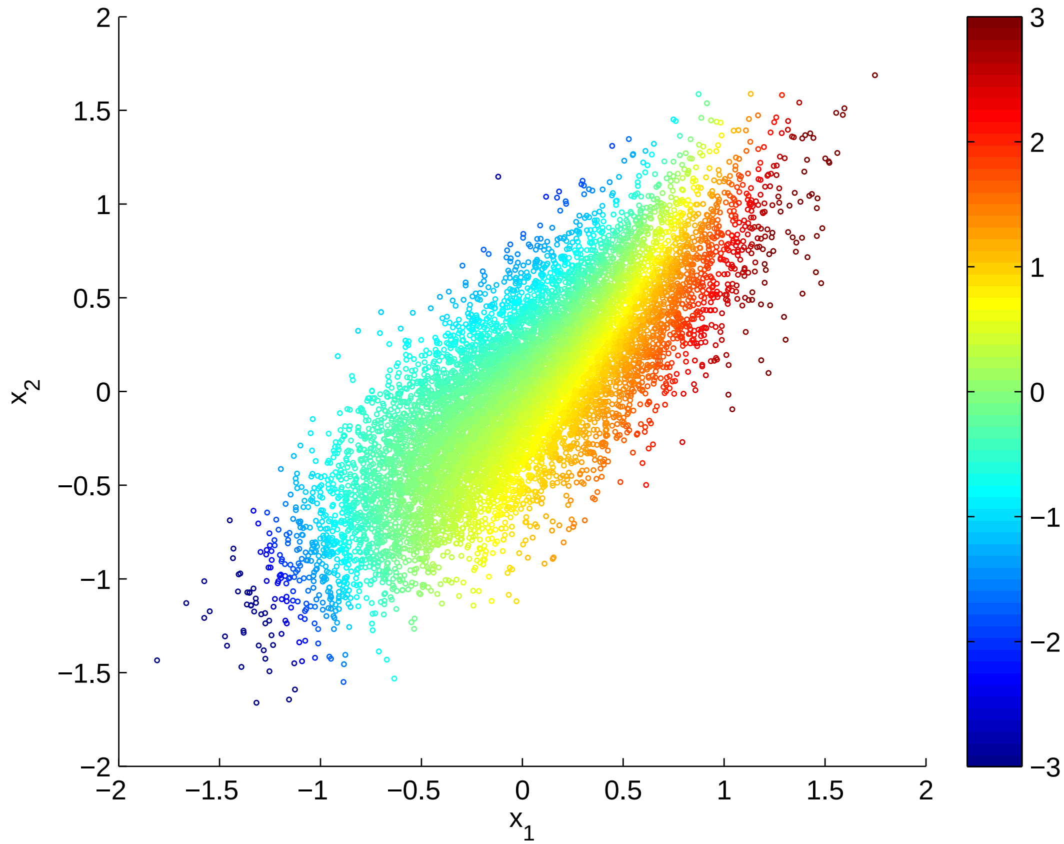
Simple Example (cont.)

- prior at t_1



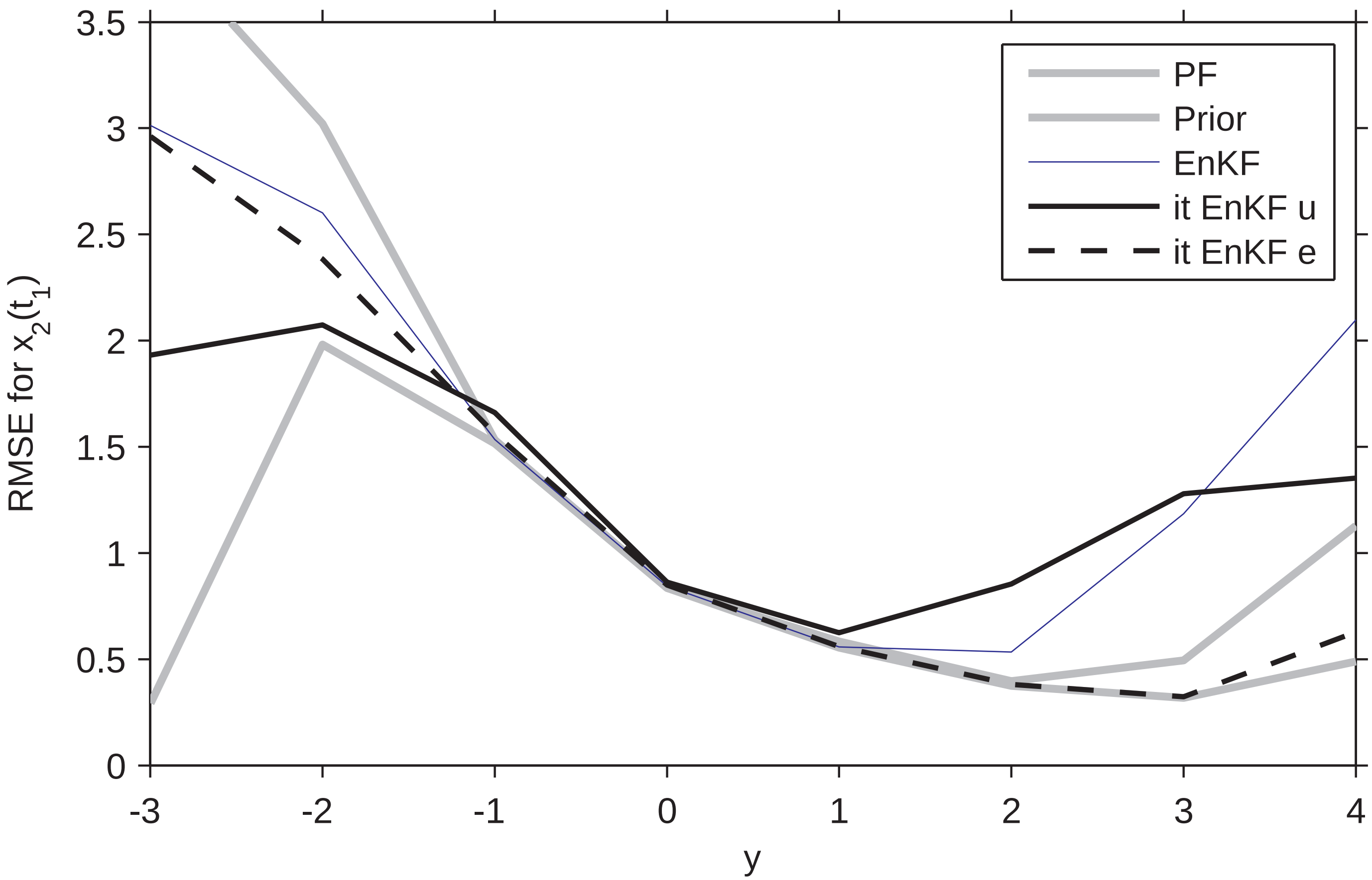
Simple Example (cont.)

- prior at t_0, with value of $x_1(t_1)$ shown by colors



Simple Example (cont.)

- ▷ RMS estimation error, averaged over realizations as fn of y



Particle Filters (PFs)

Sequential Monte-Carlo method to approximate $p(\mathbf{x}_k|\mathbf{y}_{1:k})$

- particles $\equiv$ ensemble members
- like EnKF, generates samples from desired pdf, rather than pdf itself

Particle Filters (cont.)

The simplest PF

- given $\{\mathbf{x}_{k-1}^i,\, i = 1, \ldots, N_e\}$ drawn from $p(\mathbf{x}_{k-1}|\mathbf{y}_{1:k-1})$
- $\mathbf{x}_k^i = M(\mathbf{x}_{k-1}^i) + \epsilon_k$; this gives a sample from $p(\mathbf{x}_k|\mathbf{y}_{1:k-1})$.
- approximate this prior as sum of point masses,

$$p(\mathbf{x}_k|\mathbf{y}_{1:k-1}) \approx N_e^{-1} \sum_{i=1}^{N_e} \delta(\mathbf{x} - \mathbf{x}_k^i)$$

- Bayes $\Rightarrow$

$$p(\mathbf{x}_k|\mathbf{y}_{1:k}) \propto p(\mathbf{y}_k|\mathbf{x}_k) \sum_{i=1}^{N_e} \delta(\mathbf{x} - \mathbf{x}_k^i) = \sum_{i=1}^{N_e} p(\mathbf{y}_k|\mathbf{x}_k^i)\delta(\mathbf{x} - \mathbf{x}_k^i)$$

- thus, posterior pdf approximated by weighted sum of point masses

$$p(\mathbf{x}_k|\mathbf{y}_{1:k}) \approx \sum_{i=1}^{N_e} w_i \delta(\mathbf{x} - \mathbf{x}_k^i), \quad \text{with} \quad w_i = \frac{p(\mathbf{y}_k|\mathbf{x}_k^i)}{\sum_{j=1}^{N_e} p(\mathbf{y}_k|\mathbf{x}_k^i)}$$

Particle Filters (cont.)

Asymptotically convergent to Bayes rule

- PF yields an exact implementation of Bayes' rule as $N_e \to \infty$; no approximations other than finite ensemble size

Can be exceedingly simple

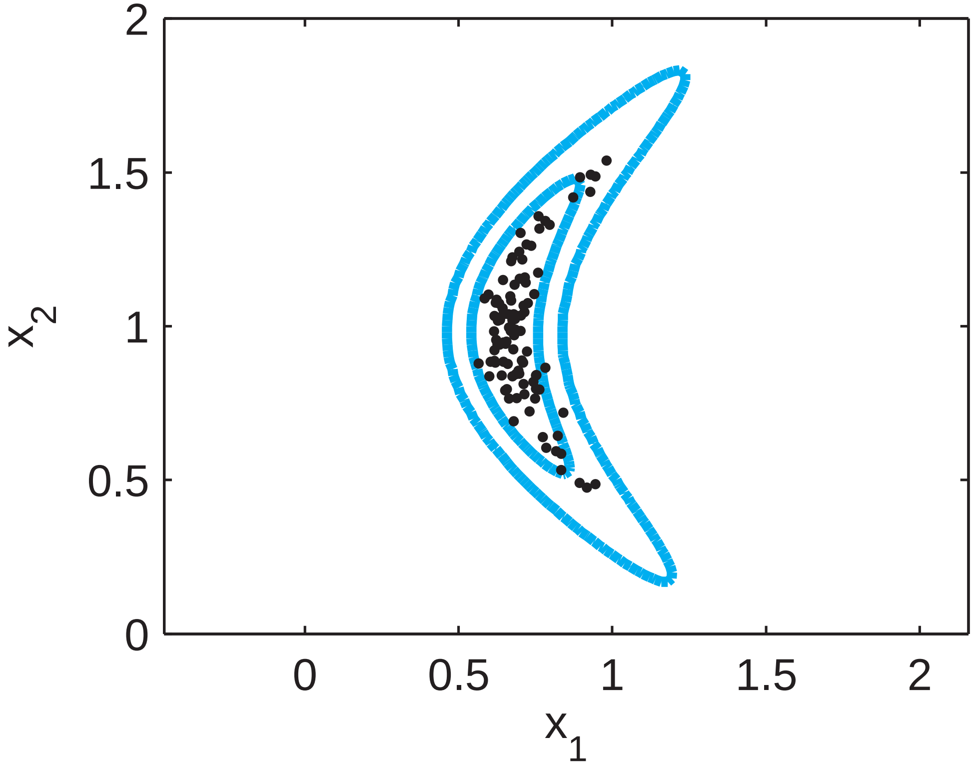
- main calculations are for w_i, e.g. $p(\mathbf{y}|\mathbf{x}_k^i)$ for $i = 1, \ldots, N_e$.

Widely applied, and effective, in low-dim'l systems

- Interest for geophysical systems too: van Leeuwen (2003, 2010), Zhou et al. (2006), Papadakis et al. (2010), hydrology

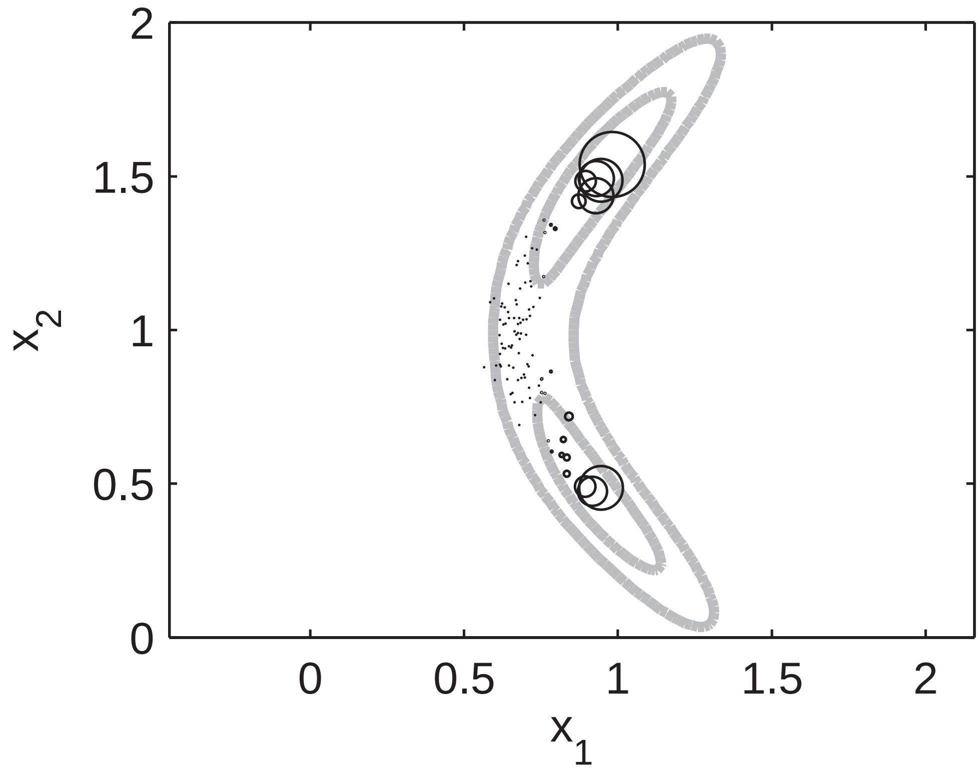
PF Illustrated

- $p(\mathbf{x})$, as before, and prior ensemble



PF Illustrated

- $p(\mathbf{x}|\mathbf{y})$ and "weighted" ensemble (size $\propto$ weight)



- weighted ensemble captures bimodality
- particles don't move; assimilation is just re-weighting

“Collapse” of Weights

A generic problem for PF

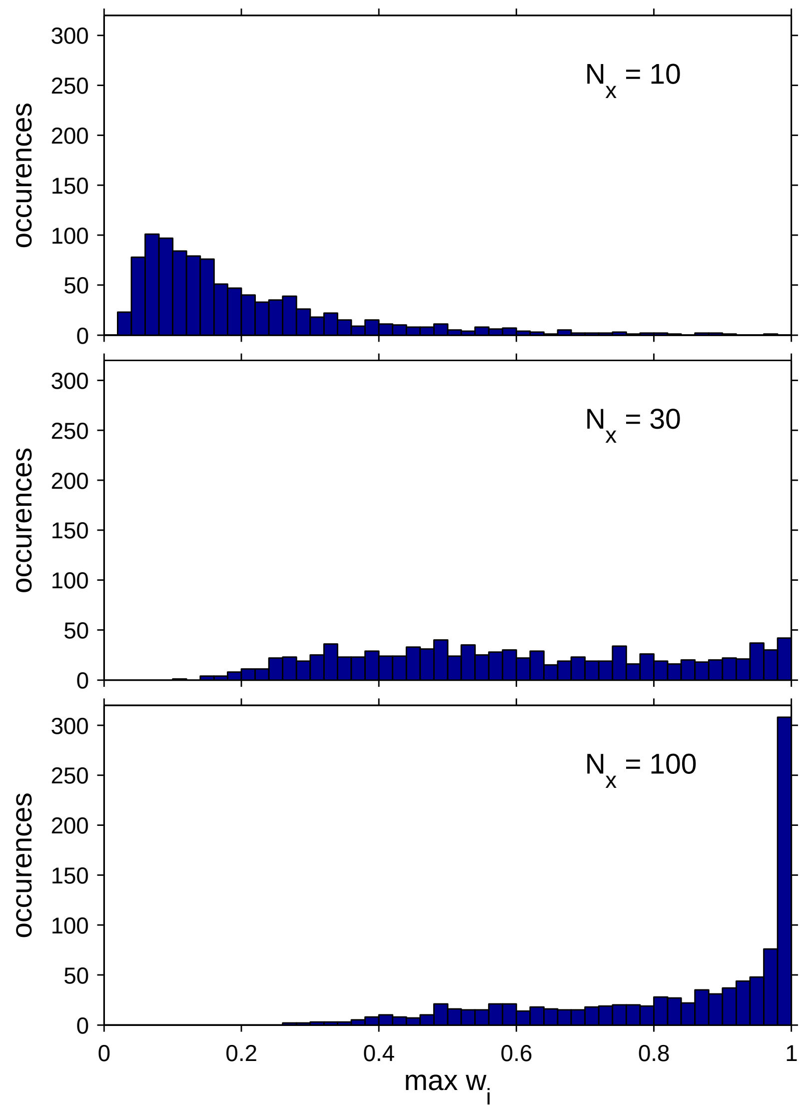
- $\max w^i \longrightarrow 1$ as N_x, N_y increase with N_e fixed
- when cycling over multiple observation times, tendency for collapse increases with t

Simple Example

- prior: $\mathbf{x} \sim N(0, \mathbf{I})$
- identity observations: $N_y = N_x$, $\mathbf{H} = \mathbf{I}$
- observation error: $\epsilon \sim N(0, \mathbf{I})$

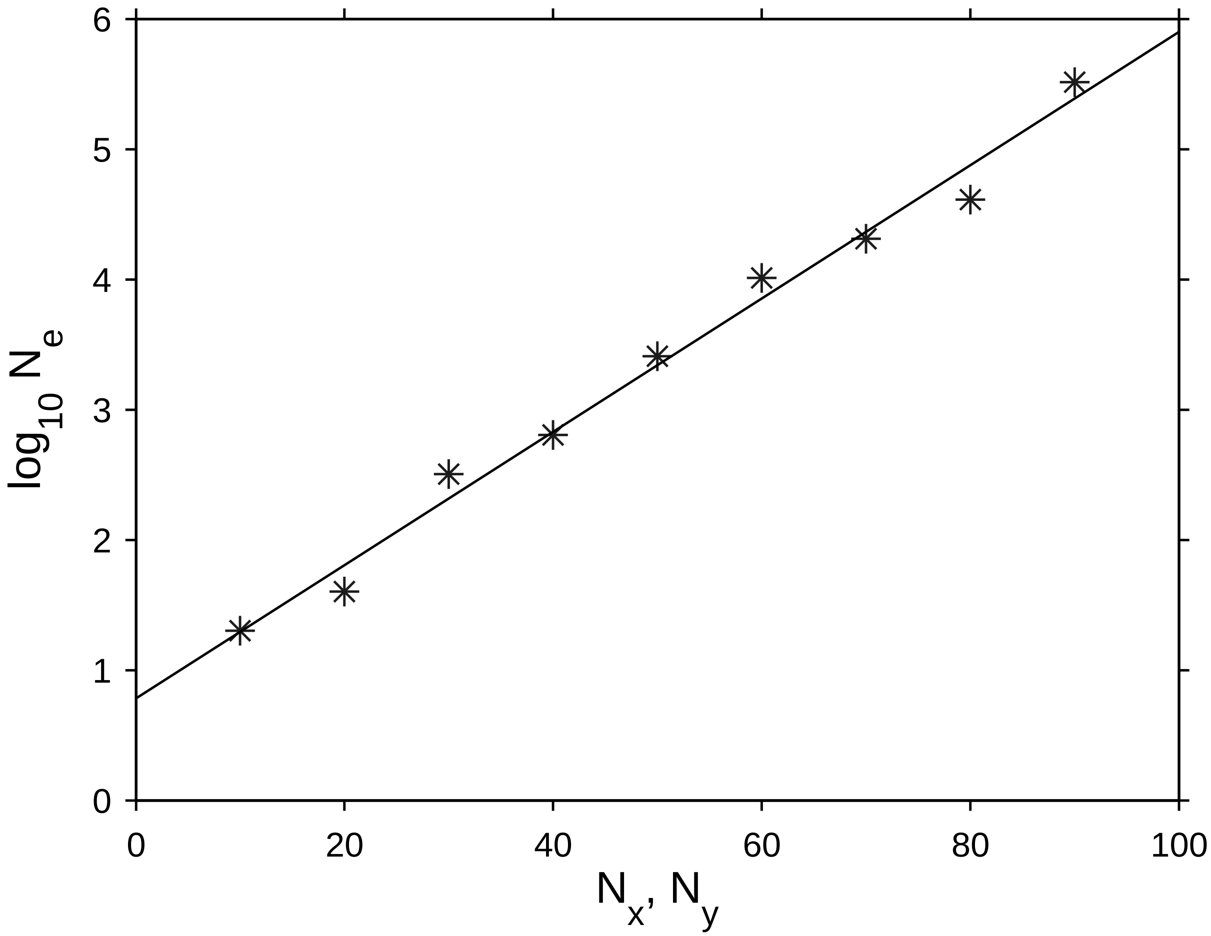
Behavior of $\max w^i$

- $N_e = 10^3$; $N_x = 10, 30, 100$; 10^3 realizations



Required ensemble size

- N_e s.t. PF mean has expected error less than obs



Required ensemble size (cont.)

Collapse occurs because w_k^i varies (a lot) with i

- variance of weights (over particles, given $\mathbf{y}$) is controlled by

$$\tau^2 = \mathsf{var}\left(-\log(p(\mathbf{y}_k|\mathbf{x}_k))\right)$$

- involves only obs-space quantities—no direct dependence on N_x

Conditions for collapse

- if $N_e \to \infty$ and $\tau^2/\log(N_e) \to \infty$,

$$E(1/\max w^i) \sim 1 + \frac{\sqrt{2\log N_e}}{\tau}$$

- see Bengtsson et al. (2008), Snyder et al. (2008) for details
- thus, weights collapse ($\max w^i \to 1$) unless N_e scales as $\exp(\tau^2/2)$

Refinements of PF

Resampling

- "refresh" ensemble by resampling from approximate posterior pdf; members with small weights are dropped, while additional members are added near members with large weights (e.g. Xiong et al. 2006, Nakano et al. 2007)
- Does not overcome difficulties with PF update but reduces tendency for collapse over time

Sequential importance sampling

- generate $\mathbf{x}_k^i$ using information beyond system dynamics and $\mathbf{x}_{k-1}^i$

Importance Sampling

Basic idea

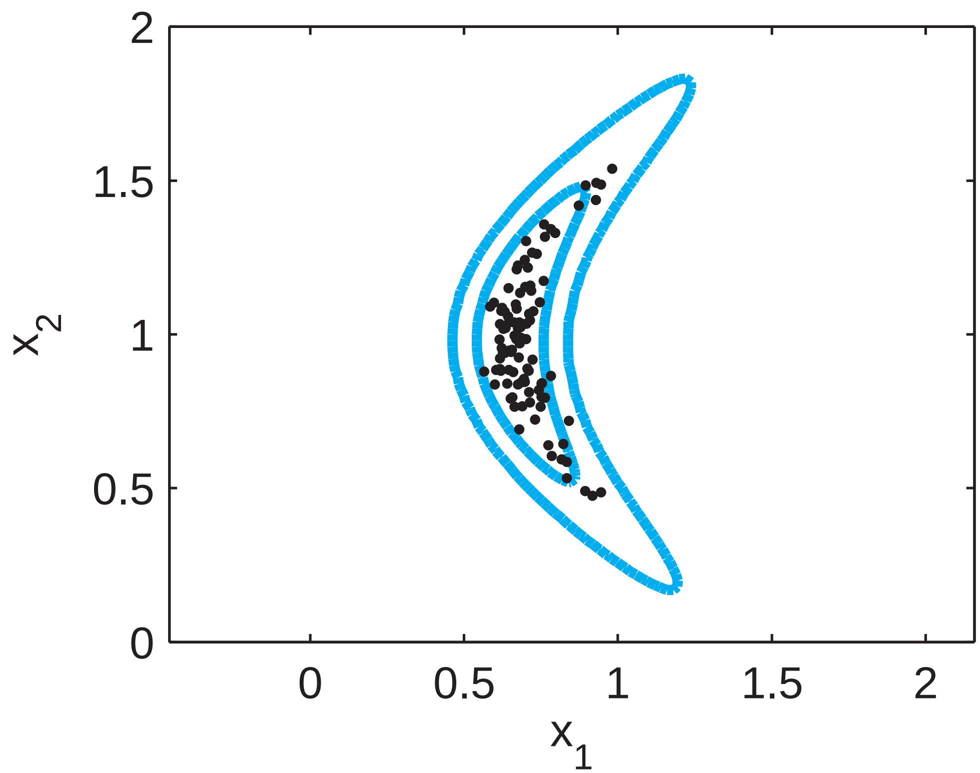
- Suppose $\pi(\mathbf{x})$ is hard to sample from, but $q(\mathbf{x})$ is not.
- draw $\{\mathbf{x}^i\}$ from $q(\mathbf{x})$ and approximate

$$\pi(\mathbf{x}) \approx \sum_{i=1}^{N_e} w^i \delta(\mathbf{x} - \mathbf{x}^i), \quad \text{where } w^i = \pi(\mathbf{x}^i)/q(\mathbf{x}^i)$$

- call $q(\mathbf{x})$ the proposal density

Importance Sampling (cont.)

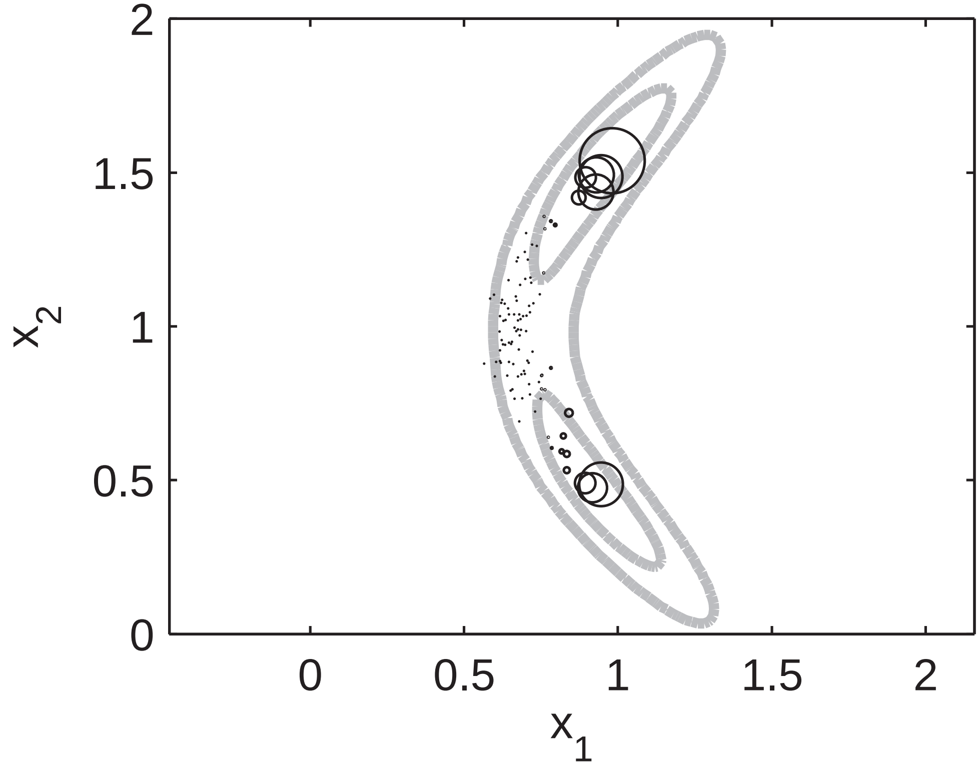
- $p(\mathbf{x})$, as before, and prior ensemble



- Want to sample from $p(\mathbf{x}|\mathbf{y})$
- IS says we should weight sample from $p(\mathbf{x})$ by $p(\mathbf{x}|\mathbf{y})/p(\mathbf{x}) = p(\mathbf{y}|\mathbf{x})$

Importance Sampling (cont.)

- $p(\mathbf{x}|\mathbf{y})$ and "weighted" ensemble (size $\propto$ weight)



Sequential Importance Sampling

Perform IS sequentially in time

- Given $\{\mathbf{x}_0^i\}$ from $q(\mathbf{x}_0)$, wish to sample from $p(\mathbf{x}_1, \mathbf{x}_0|\mathbf{y}_1)$
- Note factorization:
  $p(\mathbf{x}_1, \mathbf{x}_0|\mathbf{y}_1) \propto p(\mathbf{y}_1|\mathbf{x}_1, \mathbf{x}_0)p(\mathbf{x}_1, \mathbf{x}_0) = p(\mathbf{y}_1|\mathbf{x}_1)p(\mathbf{x}_1|\mathbf{x}_0)p(\mathbf{x}_0)$
- choose proposal of the form

$$q(\mathbf{x}_1, \mathbf{x}_0|\mathbf{y}_1) = q(\mathbf{x}_1|\mathbf{x}_0, \mathbf{y}_1)q(\mathbf{x}_0)$$

- update weights using

$$w_1^i \propto \frac{p(\mathbf{x}_1^i, \mathbf{x}_0^i|\mathbf{y}_1)}{q(\mathbf{x}_1^i, \mathbf{x}_0^i|\mathbf{y}_1)} = \frac{p(\mathbf{y}_1|\mathbf{x}_1^i)p(\mathbf{x}_1^i|\mathbf{x}_0^i)}{q(\mathbf{x}_1^i|\mathbf{x}_0^i, \mathbf{y}_1)}w_0^i$$

Sequential Importance Sampling (cont.)

Choice of proposal is known to be crucial

Simplest: transition density as proposal

- take $q(\mathbf{x}_k|\mathbf{x}_{k-1}, \mathbf{y}_k) = p(\mathbf{x}_k|\mathbf{x}_{k-1})$; i.e. evolve particles from t_{k-1} under system dynamics
- weights updated by $w_k^i \propto w_{k-1}^i p(\mathbf{y}_k|\mathbf{x}_k^i)$

Sequential Importance Sampling (cont.)

An "optimal" proposal (e.g. Doucet et al. 2000)

- $q(\mathbf{x}_k|\mathbf{x}_{k-1},\mathbf{y}_k) = p(\mathbf{x}_k|\mathbf{x}_{k-1},\mathbf{y}_k)$; use obs at t_k in proposal at t_k
- Papadakis et al. (2010) use this; van Leeuwen (2010) is similar
- weights updated by $w_k^i \propto w_{k-1}^i p(\mathbf{y}_k|\mathbf{x}_{k-1}^i)$
- for linear, Gaussian systems, easy to show that w_k^i behaves like case with prior as proposal, but $\mathrm{var}\left(\log\left(p(\mathbf{y}_k|\mathbf{x}_{k-1}^i)\right)\right)$ is quantitatively smaller, by amount depending on $\mathbf{Q}$.

N_e still grows exponentially, but w/ reduced exponent

- For fixed problem, benefits can be substantial, e.g.,

$$\mathrm{var}(\log(p(\mathbf{y}_k|\mathbf{x}_{k-1}^i))) = \alpha\mathrm{var}(\log(p(\mathbf{y}_k|\mathbf{x}_k^i))) \quad \Rightarrow$$
$$\text{ensemble size for } p(\mathbf{y}_k|\mathbf{x}_{k-1}^i) \sim \left[\text{ensemble size for } p(\mathbf{y}_k|\mathbf{x}_k^i)\right]^\alpha$$

Mixture (or Gaussian-Sum) Filters

Approximate pdfs as sums of Gaussians

- Start with $\{\mathbf{x}^i, \mathbf{P}^i\}$. Approximate prior pdf as

$$p(\mathbf{x}) = \sum_{i=1}^{N_e} w^i N(\mathbf{x}; \mathbf{x}^i, \mathbf{P}^i)$$

- To compute $p(\mathbf{x}|\mathbf{y})$ must update w^i (via PF-like eqns) and $\mathbf{x}^i$, $\mathbf{P}^i$ (via KF-like eqns); see Alspach and Sorenson (1972)
- Geophysical interest: Anderson and Anderson (1999), Bengtsson et al. (2003), Smith (2007), Hoteit et al. (2011)

Limitations

- Update of weights subject to collapse, as in PF; closely related to optimal proposal if we choose $\mathbf{P}^i = \mathbf{Q}$
- Must update $\{\mathbf{x}^i, \mathbf{P}^i\}$ in addition to weights

Summary

EnKF as approximation to BLUE

- EnKF $\neq$ assume everything is Gaussian
- Non-Gaussian aspects depend on specific EnKF scheme

Iterated ensemble smoother

- Mimics incremental 4DVar but not equivalent
  (except in linear, Gaussian case!)
- Innovation fixed, gain changes at each iteration

Particle filters

- For naive particle filter, N_e increases exponentially with problem size
- Potential for PF using more clever proposal distributions
- Evidence that these lead to N_e that still increases exponentially, but with smaller exponent

Comments

How important is non-Gaussianity for our applications?

A key idea missing from PFs (so far) is localization

Selected References

Nonlinear modifications of the EnKF:

Anderson, J. L., 2001: An ensemble adjustment filter for data assimilation. *Monthly Wea. Rev.*, **129**, 2884–2903.

Anderson, J. L., 2010: A non-Gaussian ensemble filter update for data assimilation. *Monthly Wea. Rev.*, **138**, 4186-4198.

Annan, J. D., D. J. Lunt, J. C. Hargreaves and P. J. Valdes, 2005: Parameter estimation in an atmospheric GCM using the ensemble Kalman filter. *Nonlin. Processes Geophys.*, **12**, 363–371.

Bengtsson T., C. Snyder, and D. Nychka, 2003: Toward a nonlinear ensemble filter for high-dimensional systems. *J. Geophys. Research*, **108(D24)**, 8775–8785.

Harlim, J., and B. R. Hunt, 2007: A non-Gaussian ensemble filter for assimilating infrequent noisy observations *Tellus*, **59A**, 225–237.

Kalnay, E. and S.-C. Yang, 2010: Accelerating the spin-up of ensemble Kalman filtering. *Q. J. R. Met. Soc.*, **136**, 1644–1651.

Zupanski, M., 2005: Maximum likelihood ensemble filter: Theoretical aspects. *Monthly Wea. Rev.*, **133**, 1710-1726.

Particle filters:

Arulampalam, M. S., S. Maskell, N. Gordon and T. Clapp, 2002: A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking. *IEEE Trans. Signal Processing*, **50**, 174–188.

Bengtsson, T., P. Bickel and B. Li, 2008: Curse-of-dimensionality revisited: Collapse of the particle filter in very large scale systems. *IMS Collections*, **2**, 316–334. doi: 10.1214/193940307000000518.

Doucet, A., S. Godsill, and C. Andrieu, 2000: On sequential Monte Carlo sampling methods for Bayesian filtering. *Statist. Comput.*, **10**, 197-208.

Gordon, N. J., D. J. Salmond and A. F. M. Smith, 1993: Novel approach to nonlinear/non-Gaussian Bayesian state estimation. *IEEE Proceedings-F*, **140**, 107–113.

van Leeuwen, P. J., 2010: Nonlinear data assimilation in geosciences: an extremely efficient particle filter. *Q. J. R. Met. Soc.*, **136**, 1991-1999.

van Leeuwen, P. J., 2003: A variance-minimizing filter for large-scale applications. *Monthly Wea. Rev.*, **131**, 2071–2084.

Nakano, S., G. Ueno and T. Higuchi, 2007: Merging particle filter for sequential data assimilation. *Nonlin. Processes Geophys.*, **14**, 395–408.

Papadakis, N., E. Mémin, A. Cuzol and N. Gengembre, 2010: Data assimilation with the weighted ensemble Kalman filter. *Tellus*, **62A**, 673–697.

Snyder, C., T. Bengtsson, P. Bickel and J. Anderson, 2008: Obstacles to high-dimensional particle filtering. *Monthly Wea. Rev.*, **136**, 4629–4640.

Xiong, X., I. M. Navon and B. Uzunoglu, 2006: A note on the particle filter with posterior Gaussian resampling. *Tellus*, **58A**, 456–460.

Zhou, Y., D. McLaughlin and D. Entekhabi, 2006: Assessing the performance of the ensemble Kalman filter for land surface data assimilation. *Monthly Wea. Rev.*, **134**, 2128–2142.

Gaussian-sum filters (like particle filters, but approximate prior distribution as a sum of Gaussians centered on ensemble members):

Alspach, D. L., and H. W. Sorensen, 1972: Nonlinear Bayesian estimation using Gaussian sum approximation, *IEEE Trans. Autom. Control*, **17**, 439–448.

Anderson, J. L., and S. L. Anderson, 1999: A Monte-Carlo implementation of the nonlinear filtering problem to produce ensemble assimilations and forecasts. *Monthly Wea. Rev.*, **127**, 2741–2758.

Bengtsson T., C. Snyder, and D. Nychka, 2003: Toward a nonlinear ensemble filter for high-dimensional systems. *J. Geophys. Research*, **108(D24)**, 8775–8785.

Smith, K. W., 2007: Cluster ensemble Kalman filter. *Tellus*, **59**, 749–757.

Hoteit, I., X. Luo and D.-T. Pham, 2011: Particle Kalman filtering: A nonlinear Bayesian framework for ensemble Kalman filters. *Monthly Wea. Rev.*,accepted.

4DVar and an Iterated Kalman Smoother

Recall 4DVar

- Consider perfect model/strong constraint for simplicity here. $\mathbf{x}_0$ determines $\mathbf{x}_{1:N_t}$ through $\mathbf{x}_k = M(\mathbf{x}_{k-1})$.
- Full cost function from $\log(p(\mathbf{x}_0|\mathbf{y}_{1:k}))$:

$$J(\mathbf{x}_0) = (\mathbf{x}_0 - \mathbf{x}^f{}_0)^T (\mathbf{P}^f_0)^{-1} (\mathbf{x}_0 - \mathbf{x}^f{}_0) \\ + (\mathbf{y}_{1:N_t} - H(\mathbf{x}_{1:N_t}))^T \mathbf{R}^{-1}_{1:N_t} (\mathbf{y}_{1:N_t} - H(\mathbf{x}_{1:N_t})),$$

4DVar and an Iterated Kalman Smoother

Recall incremental 4DVar

- Linearize about latest guess, $\mathbf{x}^n_{0:N_t}$; e.g., $H(\mathbf{x}_k) \approx H(\mathbf{x}^n_k) + \mathbf{H}\delta\mathbf{x}_k$ and $\delta\mathbf{x}_k = \mathbf{M}_{k-1}\delta\mathbf{x}_{k-1}$
- Yields quadratic cost function for increments:

$$\hat{J}(\delta\mathbf{x}_0) = (\delta\mathbf{x}_0 - \delta\mathbf{x}^f_0)^T(\mathbf{P}^f_0)^{-1}(\delta\mathbf{x}_0 - \delta\mathbf{x}^f_0) \\ + (\delta\mathbf{y}_{1:N_t} - \mathbf{H}\delta\mathbf{x}_{1:N_t})^T\mathbf{R}^{-1}_{1:N_t}(\delta\mathbf{y}_{1:N_t} - \mathbf{H}\delta\mathbf{x}_{1:N_t}),$$

- Iteration: Compute $\delta\mathbf{x}^a_0$ as minimizer of $\hat{J}$; set $\mathbf{x}^{n+1}_0 = \mathbf{x}^n_0 + \delta\mathbf{x}^a_0$; compute $\mathbf{x}^{n+1}_{1:N_t}$ and linearize again

Incremental 4DVar = Iterated KS

Equivalent linear, Gaussian system

- Consider:

$$
\begin{aligned}
\delta\mathbf{x}_0 &\sim N(\delta\mathbf{x}_0^f, \mathbf{P}_0^f) \\
\delta\mathbf{x}_k &= \mathbf{M}_{k-1}\delta\mathbf{x}_{k-1} \\
\delta\mathbf{y}_k &= \mathbf{H}\delta\mathbf{x}_k + \epsilon_k, \quad \epsilon_k \sim N(0, \mathbf{R}_k)
\end{aligned}
$$

- Cost fn from this system is $\hat{J}(\delta\mathbf{x}_0)$ from incremental 4DVar

Iterated Kalman smoother

- $\delta\mathbf{x}_0^a = \arg\min \hat{J}$ can also be computed with Kalman smoother:

$$
\delta\mathbf{x}_0^a = \delta\mathbf{x}_0^f + \mathbf{K}_{0|1:N_t}(\delta\mathbf{y}_{1:N_t} - \mathbf{H}\delta\mathbf{x}_{1:N_t}^f)
$$

- Thus, sequence of KS updates, with $\mathbf{M}_k$, $\mathbf{H}$ and $\mathbf{K}_{0|1:N_t}$ from re-linearization about $\mathbf{x}_{1:N_t}^n$ at each step, reproduces incremental 4DVar
- Note that initial cov of $\delta\mathbf{x}_0$ is $\mathbf{P}_0^f$; does not change during iteration
- see also Jazwinski (1970, section 9.7)

Iterated Ensemble KS

Approximate iterated KS using ensemble ideas

- Returning to full fields, KS update becomes

$$\mathbf{x}_0^{n+1} = \mathbf{x}^f{}_0 + \mathbf{K}_{0|1:N_t}(\mathbf{y}_{1:N_t} - (H(\mathbf{x}^n_{1:N_t}) + \mathbf{H}\delta\mathbf{x}^f_{1:N_t}))$$

- Now make usual replacements

$$\mathbf{H}\delta\mathbf{x}^f_k \approx H(\mathbf{x}^f{}_k) - H(\mathbf{x}^n_k),$$
$$\mathbf{K}_{0|1:N_t} \approx \hat{\mathbf{K}}_{0|1:N_t} = \mathrm{cov}(\mathbf{x}_0, H(\mathbf{x}_{1:N_t}))[\mathrm{cov}(H(\mathbf{x}_{1:N_t})) + \mathbf{R}_{1:N_t}]^{-1}$$

Iterated Ensemble KS

Approximate iterated KS using ensemble ideas

- Returning to full fields, KS update becomes

$$\mathbf{x}_0^{n+1} = \mathbf{x}^f{}_0 + \mathbf{K}_{0|1:N_t}(\mathbf{y}_{1:N_t} - (H(\mathbf{x}_{1:N_t}^n) + \mathbf{H}\delta\mathbf{x}_{1:N_t}^f))$$

- Now make usual replacements

$$\mathbf{H}\delta\mathbf{x}_k^f \approx H(\mathbf{x}^f{}_k) - H(\mathbf{x}_k^n),$$
$$\mathbf{K}_{0|1:N_t} \approx \hat{\mathbf{K}}_{0|1:N_t} = \mathrm{cov}(\mathbf{x}_0, H(\mathbf{x}_{1:N_t}))[\mathrm{cov}(H(\mathbf{x}_{1:N_t})) + \mathbf{R}_{1:N_t}]^{-1}$$

Iteration for ensemble smoother

- Ensemble ICs drawn from $N(\mathbf{x}_0^n, \mathbf{P}_0^f)$ to approximate linearization about $\mathbf{x}^n$ in H and M.
- Ensemble mean at iteration $n+1$ given by

$$\mathbf{x}_0^{n+1} = \mathbf{x}^f{}_0 + \hat{\mathbf{K}}_{0|1:N_t}^n(\mathbf{y}_{1:N_t} - \overline{H(\mathbf{x}_{1:N_t})})$$

- Same as usual update, but gain changes at each iteration

Kalnay-Yang Iteration for Ensemble KS

"Running in place" from Kalnay and Yang (2010)

- Ensemble mean at iteration $n+1$ given by

$$\mathbf{x}_0^{n+1} = \mathbf{x}_0^n + \hat{\mathbf{K}}_{0|1:N_t}^n(\mathbf{y}_{1:N_t} - \overline{H(\mathbf{x}_{1:N_t})})$$

- Innovation is recalculated using most recent guess and gain changes at each iteration
- Intended to speed spin up of EnKS when initial estimate of $\mathbf{P}_0^f$ is poor

Converges to observations when H and M are linear

- Let $\mathbf{L}^n = \mathbf{I} - \mathbf{H}^T\hat{\mathbf{K}}_{0|1:N_t}^n$. Easy to show

$$\mathbf{H}\mathbf{x}_0^{n+1} = \left(\prod_{m=1}^{n} \mathbf{L}^m\right)\mathbf{H}\mathbf{x}^f{}_0 + \left(\mathbf{I} - \prod_{m=1}^{n} \mathbf{L}^m\right)\mathbf{y}$$

- Properties in nonlinear case are unclear