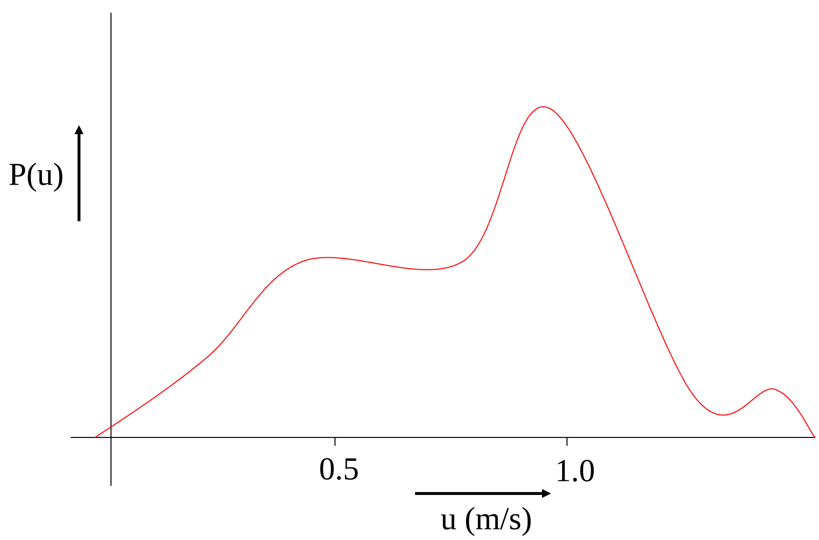
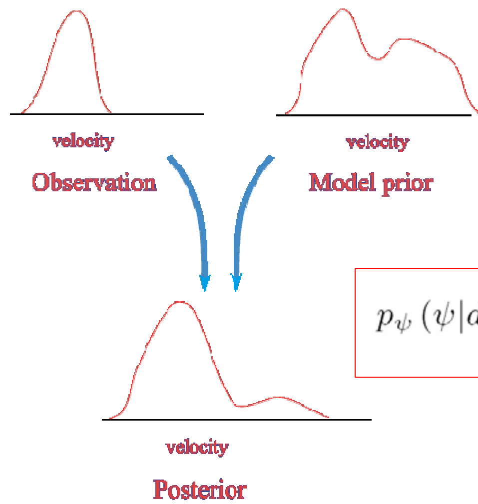
Particle filtering in geophysical systemes: Problems and potential solutions

Peter Jan van Leeuwen
IMAU

The basics: probability density functions



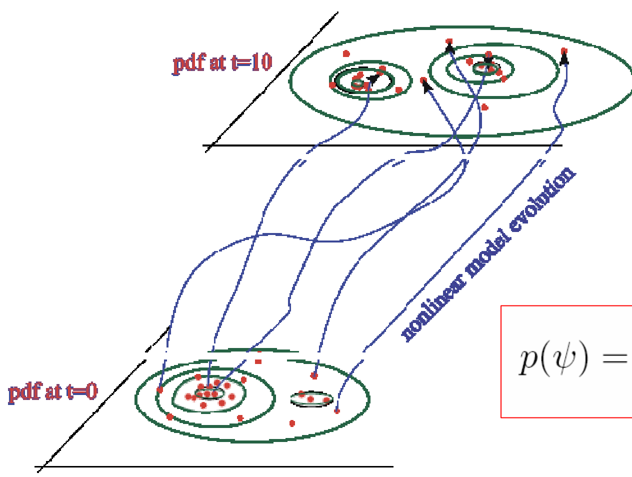
Data assimilation: general formulation



$$p_\psi\left(\psi|d\right) = \frac{p_d\left(d|\psi\right)p_\psi\left(\psi\right)}{\int p_d\left(d|\psi\right)p_\psi\left(\psi\right)\ d\psi}$$

NO INVERSION !!!

Propagation of pdf: Ensemble methods
‘efficient’ propagation for nonlinear models

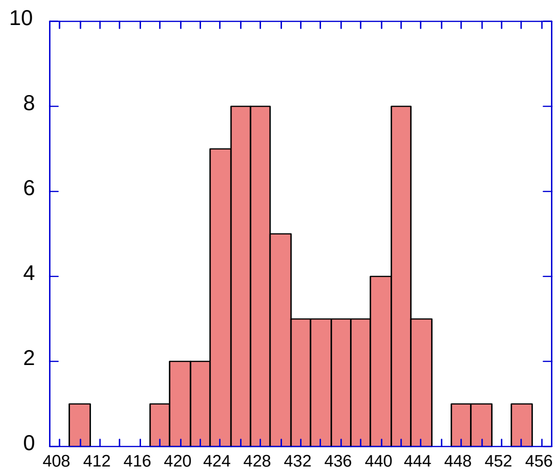


$$p(\psi) = \sum_i \frac{1}{N}\delta(\psi - \psi_i)$$

Model pdf's are non-Gaussian

Probability density function of layer thickness of first layer at day 41 during data-assimilation

Kalman filter ?
Variational methods ?



Sequential Importance Sampling SIS

$$p_\psi(\psi|d) = \frac{p_d(d|\psi)\,p_\psi(\psi)}{\int p_d(d|\psi)\,p_\psi(\psi)\;d\psi}$$

Use ensemble $p(\psi) = \sum_i \frac{1}{N}\delta(\psi - \psi_i)$

$$p_\psi(\psi|d) = \sum_i w_i \delta(\psi - \psi_i)$$

with $$w_i = \frac{p_d(d|\psi_i)}{\sum_i p_d(d|\psi_i)}$$

Specifics of Bayes I

We are interested in:

$$\overline{f(\psi)} = \int f(\psi) p(\psi|d)\ d\psi$$

Using Bayes:

$$= \frac{1}{A} \int f(\psi) p(d|\psi) p(\psi)\ d\psi$$

But also:

$$= \frac{1}{A} \int f(\psi) p(d|\psi) \frac{p(\psi)}{q(\psi)} q(\psi)\ d\psi$$

q is the proposal density, from which it is ‘easy to draw samples’, and which might be conditioned on the new observations!

Specifics of Bayes II

Introduce particles *drawn from q:*

$$= \frac{1}{A} \sum_i f(\psi_i) p(d|\psi_i) \frac{p(\psi_i)}{q(\psi_i)}$$

Hence, again:

$$= \sum_i w_i f(\psi_i)$$

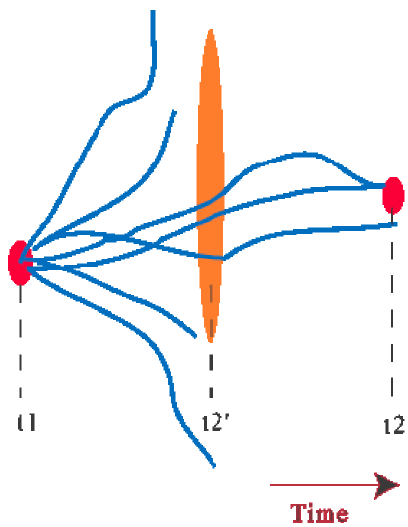
But now with weights:

$$w_i = \frac{1}{A} p(d|\psi_i) \frac{p(\psi_i)}{q(\psi_i)}$$

Specifics of Bayes III

It is possible to use particles from the prior that have been 'modified' to be closer to the observations.

Example: The Guided SIR



Do SIR at time t2'with observations from t2.
Increase observational Errors.
In this case *q* is represented by the SIR sample from t2'.
The extra weights from t2' (in the resampled ensemble) have to be compensated for:

$$w_i = \frac{1}{A} p(d|\psi_i) \frac{p(\psi_i)}{q(\psi_i)}$$

Filter degeneracy

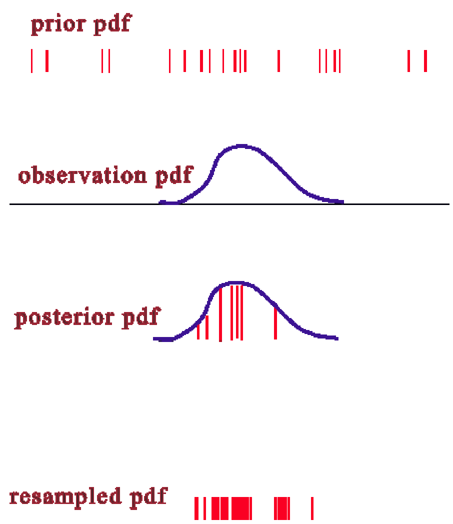
After a few updates the weights become more and more skewed:

$$w_i^n = A p(d^n|\psi_i^n) w_i^{n-1} = ... = A^n \Pi_{j=1}^n p(d^j|\psi_i^j)$$

Practice: after a few updates only one member has large weight, rest has weight zero…

Possible solution:
- Resampling, such that all particles have equal weight again -> SIR
- Integrate past times out -> MPF

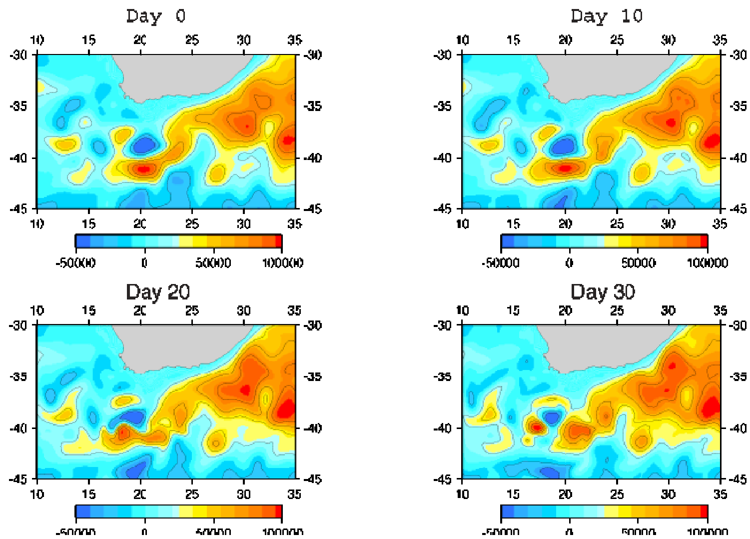
Sequential Importance Resampling SIR



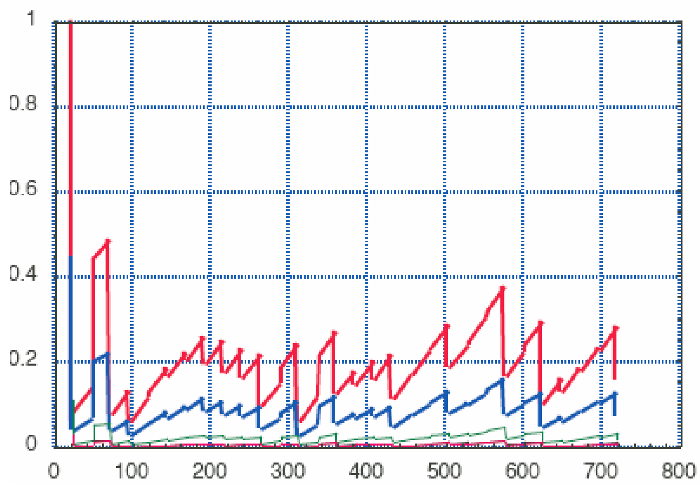
$$p_\psi(\psi|d) = \frac{p_d(d|\psi)\, p_\psi(\psi)}{\int p_d(d|\psi)\, p_\psi(\psi)\ d\psi}$$

$$p(\psi) = \sum_i w_i \delta(\psi - \psi_i)$$

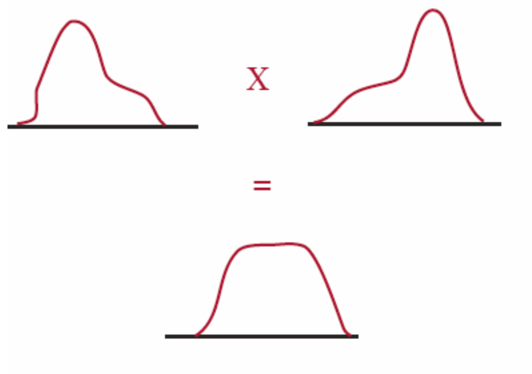
SIR-results for a quasi-geostrophic ocean model around South Africa with 512 members



Total variance in each layer



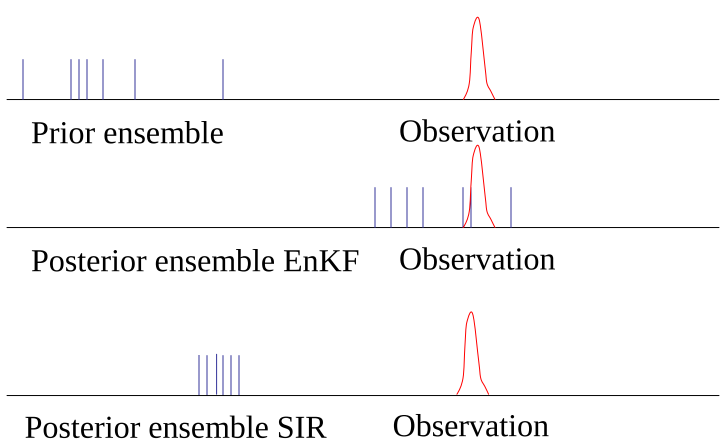
Variance-increase in non-Gaussian updates



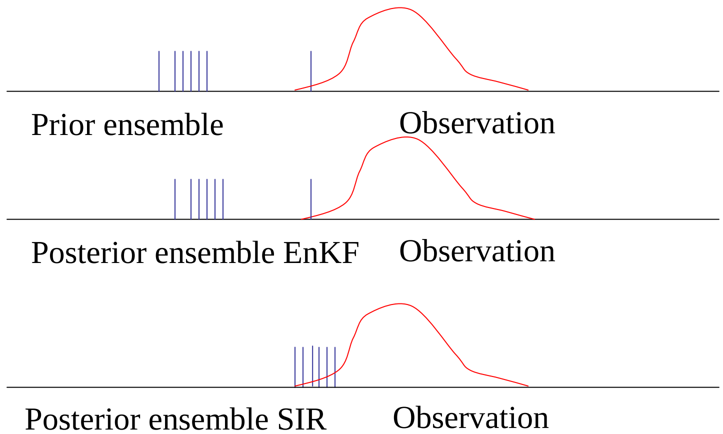
Other measure of uncertainty is ENTROPY :

$$E(\psi) = -\int p(\psi) \log \left(\frac{p(\psi)}{\sigma(\psi)} \right) \, d\psi$$

Filter degeneracy II: wide model pdf



Filter degeneracy III: narrow model pdf



Marginal Particle Filter I

The SIS updates the full joint (in time) pdf, so let's integrate the past out -> Marginal Particle Filter.

Prior: $$p(\psi^n|d^{1:n-1}) = \int p(\psi^n, \psi^{n-1}|d^{1:n-1})\ d\psi^{n-1}$$

$$= \int p(\psi^n||\psi^{n-1})p(\psi^{n-1}|d^{1:n-1})\ d\psi^{n-1}$$

Hence: $$p(\psi^n|d^{1:n}) = \frac{1}{A}p(d^n|\psi^n)p(\psi^n|d^{1:n-1})$$

$$= \frac{1}{A}p(d^n|\psi^n)\int p(\psi^n|\psi^{n-1})p(\psi^{n-1}|d^{1:n-1})\ d\psi^{n-1}$$

Marginal Particle Filter II

Use proposal density of similar form:

Draw ψ_{ji}^{n} from $\sum_i w_i^{n-1} q(\psi^n | d^n \psi_i^{n-1})$

(i.e. run the proposal model N times with different forcing from the prior ensemble ψ^{n-1} , for each member j)

Calculate importance weights:

$$w_i^n = p(d^n|\psi_i^n)\frac{\sum_i w_i^{n-1} p(\psi_i^n|\psi_i^{n-1})}{\sum_i w_i^{n-1} q(\psi_i^n|d^n\psi_i^{n-1})}$$

And normalize them.

SIR versus Marginal Particle Filter

- MPF is an O(N^2) method
- SIR suffers from sample noise due to resampling
- Several resampling methods posible:

  - sample directly from weighted ensemble
  - residual sampling
  - universal sampling

SIR on large-scale problem:

- SIR still degenerate:
  - weights differ too much (variance too high)
  - hence very small ensemble to resample from

- Larger ensemble not realistic

Possible solutions

- Explore proposal density
- Approximations in formalism
- Follow solutions used in Ensemble Kalman Filter (EnKF)

Exploring the proposal density: Auxilary Particle Filter I
(Adaptive Particle Filter)

One of the reasons that the SIR fails is that the likelihood

$$p(d^n|\psi^n)$$

in geophysical problems is very narrow: the prior ensemble is much wider than the pdf of the observations.

Hence, the majority of the particles gets very low weight.

Use the observations to guide the ensemble, i.e. determine the weights of the posterior ensemble at time *n-1* with respect to the new observations at time *n*.

Auxilary Particle Filter II

- Generate representation of each ψ_i^{n-1} at time *n*.

  (Use e.g. the model with zero random forcing.)
- Calculate weights (as normal, from likelihood)
- Resample particles ψ_i^{n-1} at time *n-1*
- Run SIS (or SIR) with the new prior ensemble from time *n-1*

Approximate Particle Filters

Merging Particle Filter: Generate linear combinations from prior ensemble that preserve mean and variance. Hence, the scheme is still variance minimizing (unlike EnKF and its variants..)

Kernel dressing: 'Dress' each particle with a continuous pdf (usually a Gaussian) to obtain a global continuous pdf. Update both particles (mean of pdf's) and covariances (using KF-like update). (Related to Gaussian mixture models)

Maximum Entropy Particle Filter:

Maximum Entropy Particle Filter I

Without observations the pdf is the 'background pdf' Q.
The model pdf p relaxes to this pdf in absence of observations.
Q usually taken as a Gaussian mixture, with coefficients found from 'model climatology'.

When observations are present the model pdf p will be as close as possible to Q, and follows the observations as constraints.

The closeness to Q is expressed as the relative entropy:

$$H(p, Q) = \int p \log\left(\frac{p}{Q}\right) \, d\psi$$

Maximum Entropy Particle Filter II

H is maximum when p is given by:

$$p(\psi, \lambda, \Lambda) = \frac{1}{Z(\lambda, \Lambda)} \exp\left[\lambda h(\psi) + 1/2 h(\psi) \Lambda h^T(\psi)\right] Q(\psi)$$

With λ and Λ the lagrange multipliers, found from

$$H = max\left[\eta\ \lambda + 1/2\ P\ \Lambda - \log(Z)\right]$$

In which

$$\eta = \frac{1}{N} \sum_i h(\psi_i) \qquad P = \frac{1}{N} \sum_i h(\psi_i) h(\psi_i)$$

Maximum Entropy Particle Filter III

- Run ensemble to observation time
- Calculate η and P from the ensemble
- Determine λ and Λ
- Use Bayes with Gaussian statistics to update λ and Λ:

$$\lambda^{new} = \lambda + R^{-1} d$$

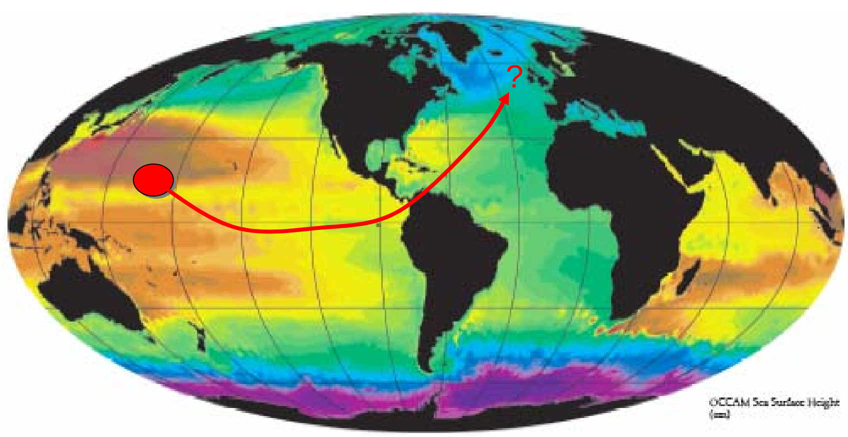
$$\Lambda^{new} = \Lambda - R^{-1}$$

- Sample new ensemble from new p

Costs of SIR, EnKF, MEPF

- All need integration of particles
- SIR analysis: O(Nnm)
- EnKF analysis O(Nnm)
- MEPF analysis O(Mn^2 n_max)
  with M number of mixtures in Q and
  n_max number of EOF's of C

Solution used in EnKF (Ensemble Kalman Filter)



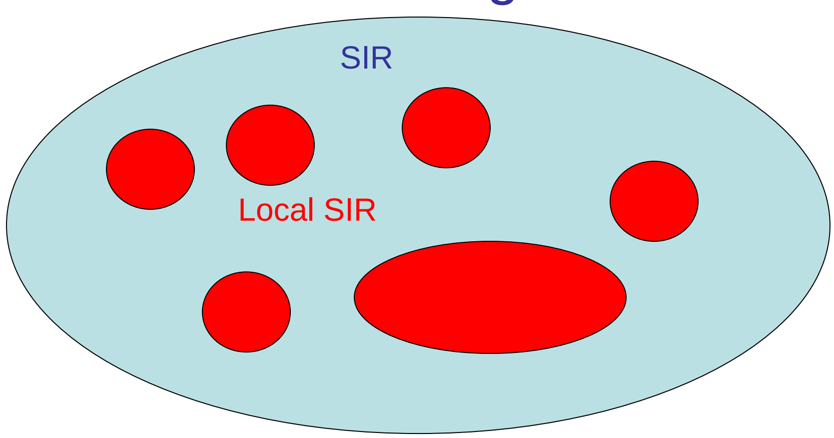
Local updating

Local updating

- Reduces spurious covariances due to small ensemble size
- Decouples ensemble members in different areas -->

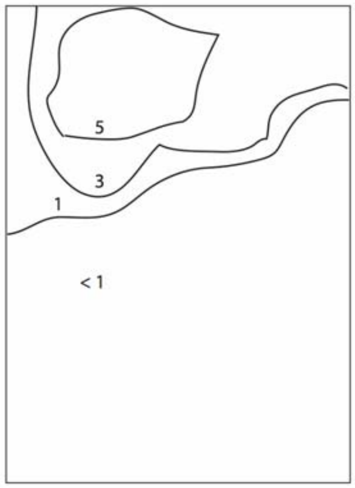
  Increase of effective ensemble size
- Brings ‘new blood’ in the ensemble

Local SIR: use only local obs in the weights



Easy to implement, but for the resampling step

How to do the resampling?



Relative weight of member 1

- In SIR all particles are always balanced
- In Local SIR they are not. How do we glue different particles together?

Resampling in Local SIR

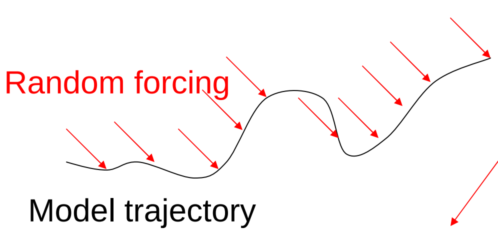
- Use the pdf: $p_\psi\left(\psi(x)|\psi(x-\Delta x)\right)$
- Adapt the forcing locally (Local smoother)
- Use EnKF solution as ‘background field’

Resampling using pdf

- Start at some model point and choose randomly from weighted ensemble
- Run along the grid and use this member as long as weight > 1
- When at some point x the weight < 1 choose *randomly* among those members that:
    1. Have high weight at x
    2. Resemble member at x-dx

  (i.e. distance smaller than standard deviation)
- In this way the probabilistic features are kept!

Local smoother

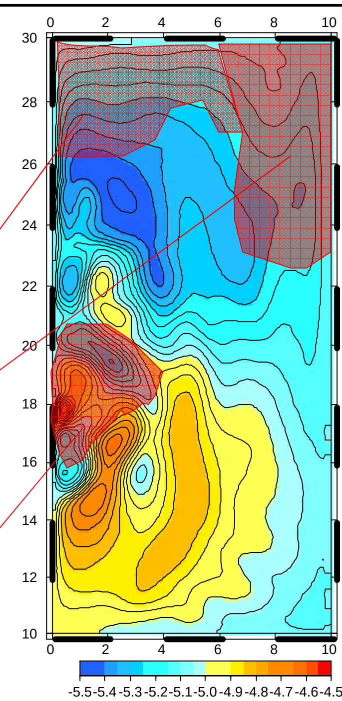


Particle 17

Particle 3

Particle 7

Apply weights on forcing, but locally !!!



Practical implementation

1 Run particles to observation time.
2 Determine local weights
3 Generate equal-weight forcing ensemble from locally weighted particles
4 Re-run the particles with this forcing ensemble
5 Either: Back to 1 OR Apply SIR with new prior weights

For 3 use repeatable random field generator

Use EnKF as background ensemble

- Perform Local SIS
- Resample such that member 1 is as close as possible to EnKF-member 1

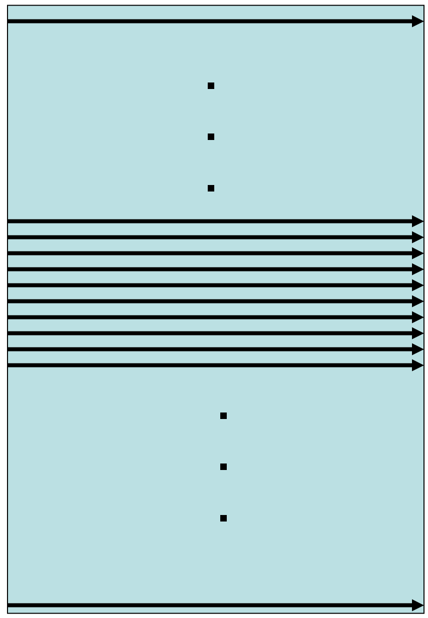
Note: allows for use of EnKF-solution in areas when all members are far from observations

Conclusions

- ‘Standard’ Particle Filters (SIS, SIR) do not work on large-scale problems
- Maybe smart proposal pdf helps (use of Dyn. Sys. Theory?)
- Maybe localization helps
- Approximate Particle Filters might be needed

Remarks

- What do we want from particle filtering?
- Mean?
- Mode?
- Modal information?
- ….



Example

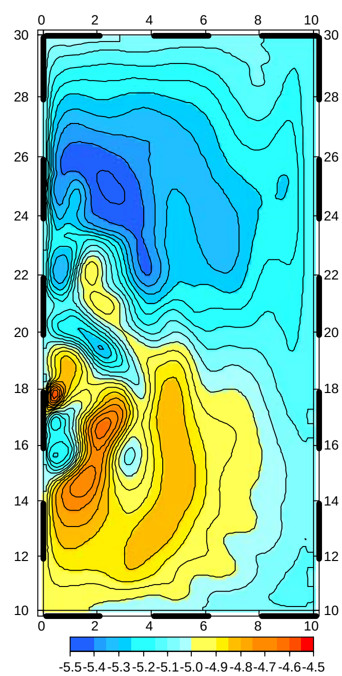
Two-layer primitive equation model of a double gyre.

Lx=2000 km, Ly = 4000 km
Δx, Δy = 20 km

H1=1000 m, H2=4000 m

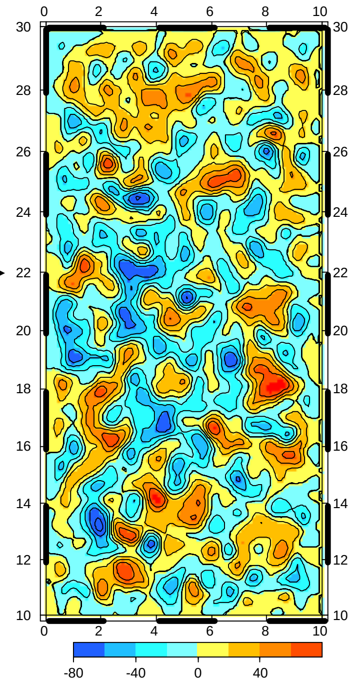
Wind profile 0.6 cos (y/L)

Observations
sea-surface height
Δx = 40 km,
σ= 2 cm
Interval: 10 days (others..)

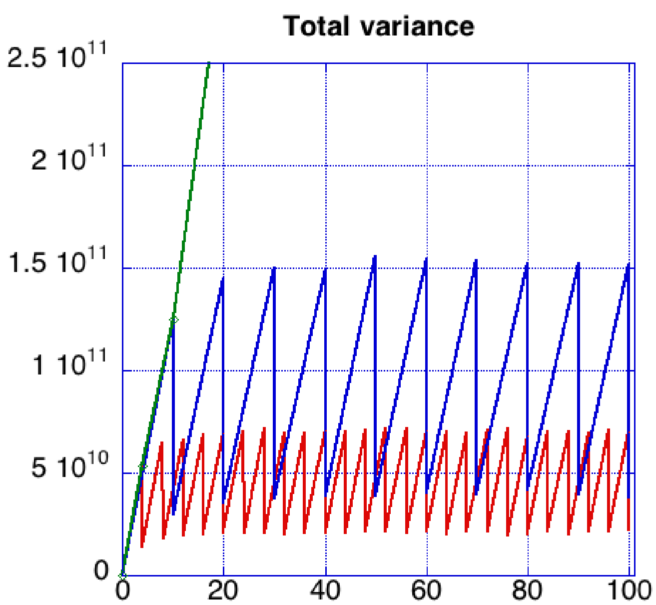


Statistics

Red noise random fields added to layer thicknesses.

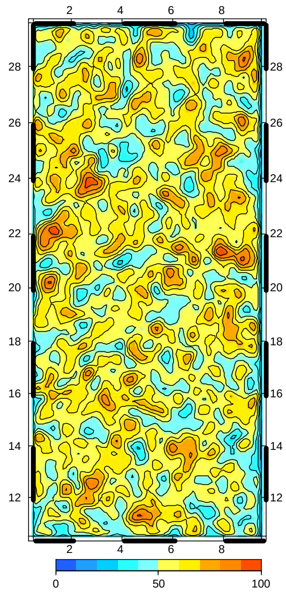


64 members,
80% local,
20% global

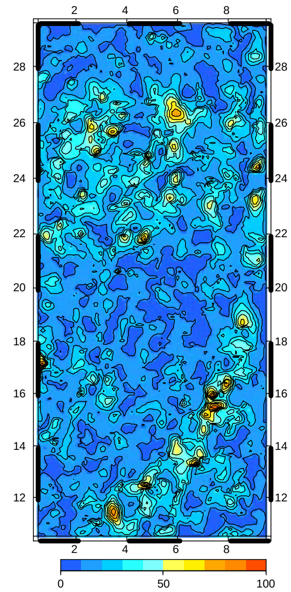


Variance reduction upper layer

Variance upper layer before analysis

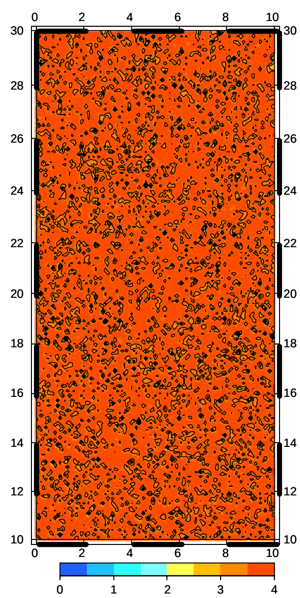


Variance upper layer after analysis

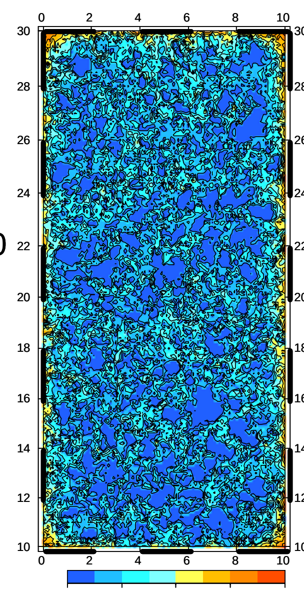


Entropy upper layer thickness

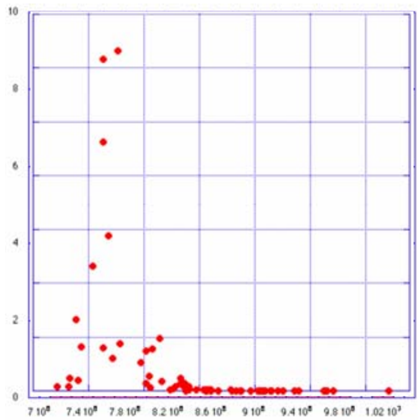
Entropy before analysis at day 50



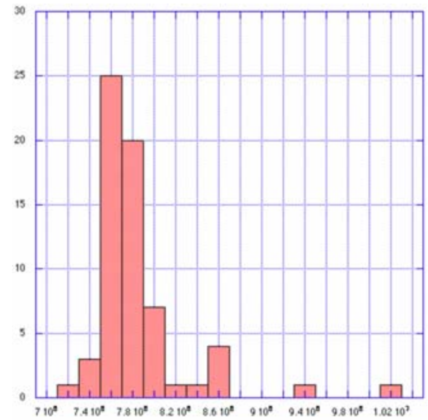
Entropy after analysis at day 50



Multi-modal pdf I

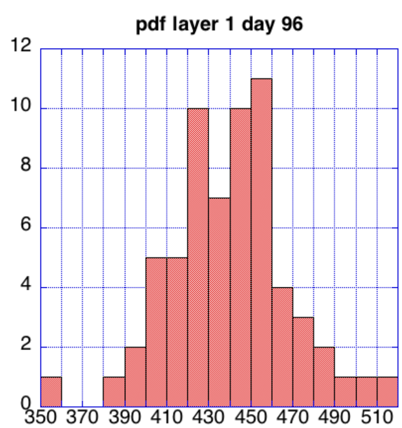


Before analysis

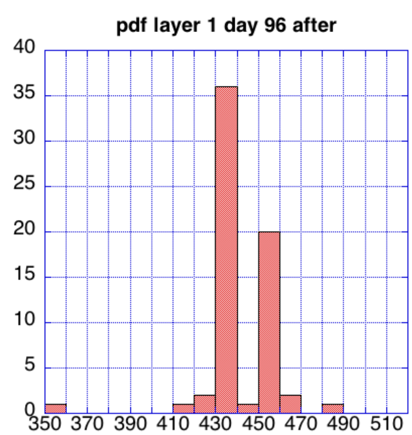


After analysis

Multi-modal pdf II

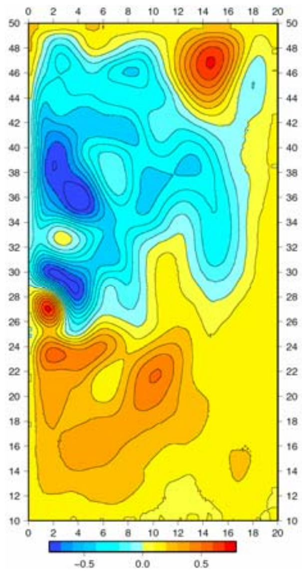


Before assimilation

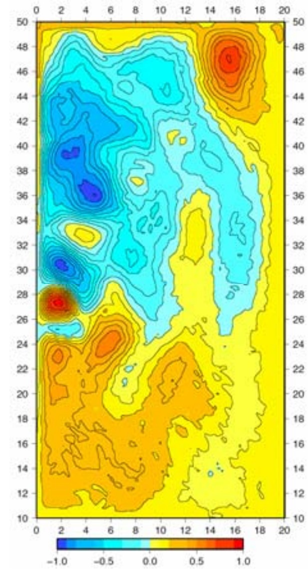


After assimilation

Smoothness



Mean sea-surface height field



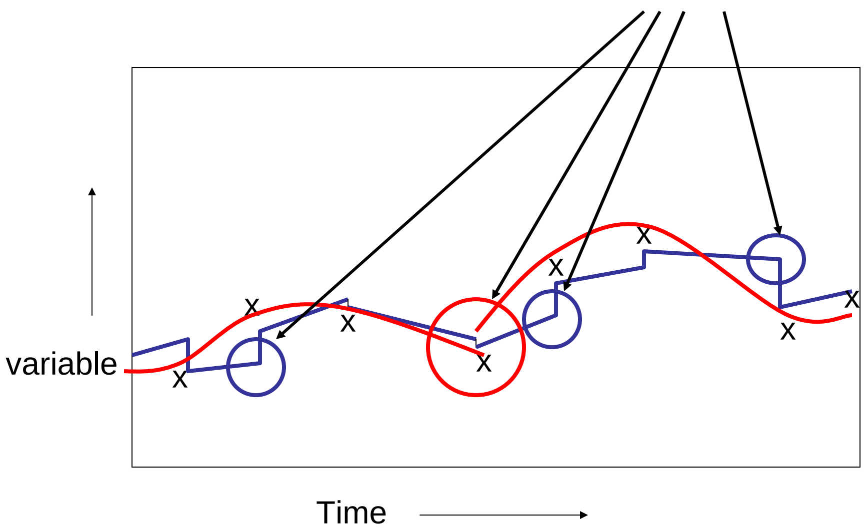
Ensemble member 15

Conclusions

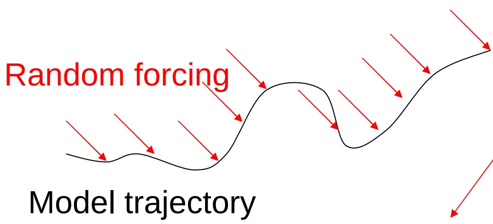
- SIR needs > 512 members for primitive equations (probably > 10000 …)
- Local SIR works here with 64 members, ‘robust’
- Local SIR brings ‘new blood’ in the ensemble
- Statistics glueing ‘solved’
- Smoothness remains issue

Nonlinear filtering with large-scale models possible
Opens possibilities for hybrid local methods: LSIR with EnKF

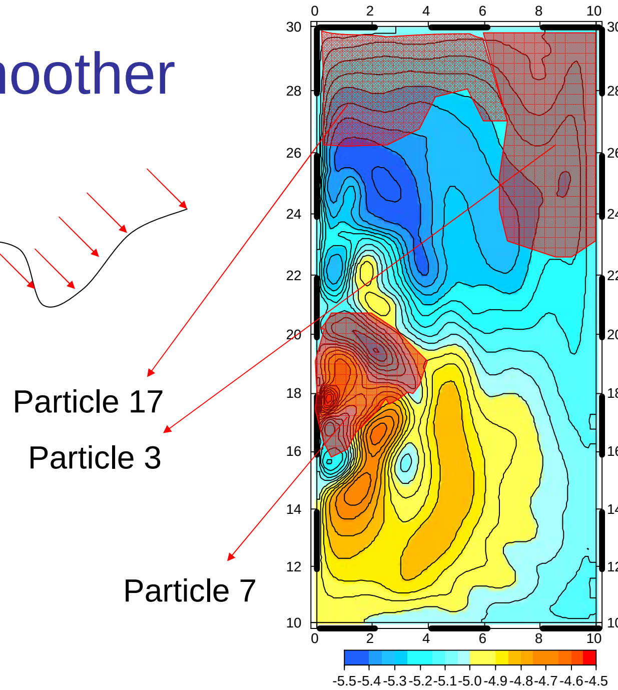
Filters and smoother are discontinuous



Local smoother



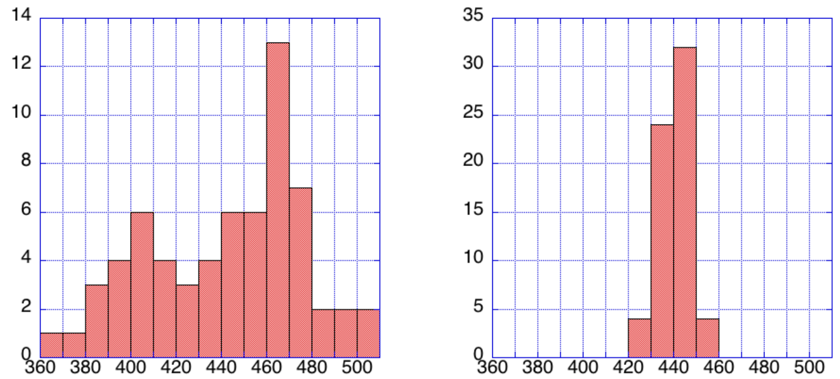
Apply weights
on forcing,
but locally !!!



Practical implementation

1 Run particles to observation time.
2 Determine local weights
3 Generate equal-weight forcing ensemble from locally weighted particles
4 Re-run the particles with this forcing ensemble
5 Back to 1

For 3 use repeatable random field generator



14
12
10
8
6
4
2
0
360 380 400 420 440 460 480 500
35
30
25
20
15
10
5
0
360 380 400 420 440 460 480 500