Ensemble Kalman Filter potential

Former students (Shu-Chih Yang, Takemasa Miyoshi, Hong Li, Junjie Liu, Chris Danforth, Ji-Sun Kang, Matt Hoffman), and

Eugenia Kalnay

University of Maryland

Acknowledgements:

UMD Chaos-Weather Group: **Brian Hunt**, Istvan Szunyogh, Ed Ott, Jim Yorke, **Kayo Ide**, and students

Also: Malaquías Peña, Matteo Corazza, Pablo Grunman, DJ Patil, Debra Baker, Steve Greybush, Tamara Singleton, Steve Penny, Elana Fertig.

MOCA-09 J21- Advances in Data Assimilation - 23 July 2009

Conclusions from a clean comparison of GSI and EnKF (Whitaker, 2009)

- ✓ At T190/L64 resolution - 64 members - EnKF is better than the operational GSI (at T382/L64)
- ✓ It is computationally competitive with 3D-Var if the forecast costs are covered by the operational ensemble requirement

Conclusions from a clean comparison of JMA 4D-Var and LETKF (Miyoshi et al. 08)

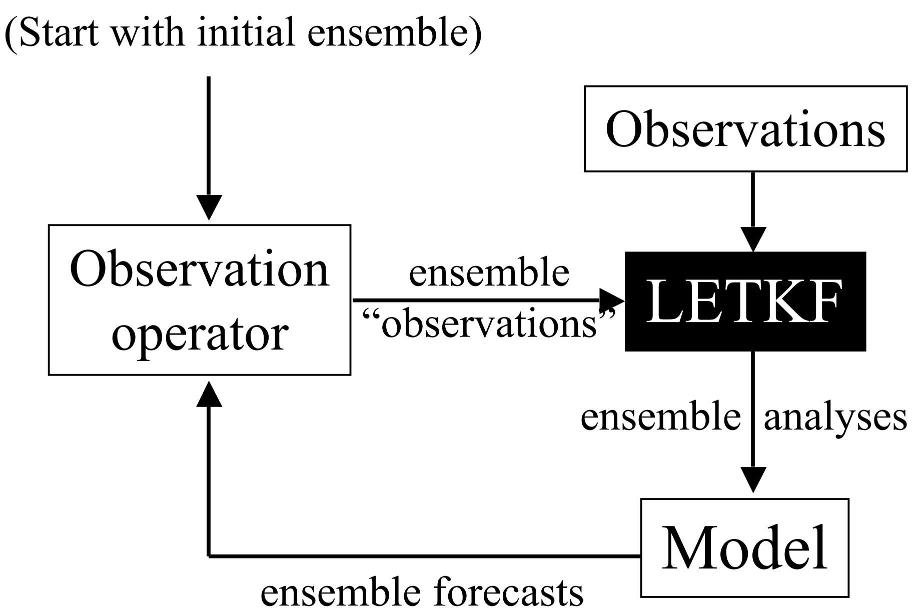
- ✓ At the same resolution LETKF is faster than the operational 4D-Var, better in the tropics and NH, worse in SH due to a model bias
- ✓ Plan to test simple low-dim method to correct model bias

Tools that improve LETKF/EnKF

We adapted ideas inspired by 4D-Var:

- ✓ **No-cost smoother** (Kalnay et al, Tellus 2007)
- ✓ "**Outer loop**", nonlinearities and long windows (Yang and Kalnay)
- ✓ Accelerating the **spin-up** (Kalnay and Yang, 2008)
- ✓ **Forecast sensitivity** to observations (Liu and Kalnay, QJ, 2008)
- ✓ **Analysis sensitivity** to observations and exact **cross-validation** (Liu et al., QJ, 2009)
- ✓ **Coarse** analysis resolution **without degradation** (Yang et al., QJ, 2009)
- ✓ Low-dimensional **model bias correction** (Li et al., MWR, 2009)
- ✓ Simultaneous estimation of **optimal inflation** and **observation errors** (Li et al., QJ, 2009).

Local Ensemble Transform Kalman Filter (Ott et al, 2004, Hunt et al, 2004, 2007) (a square root filter)

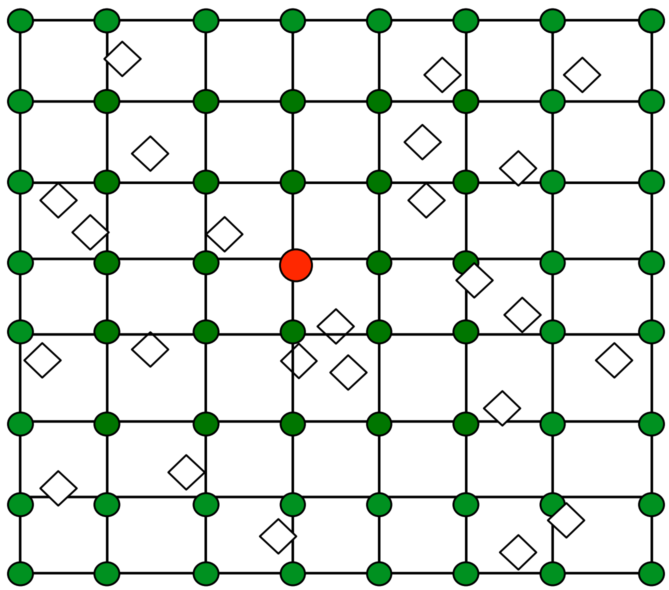


- Model independent (black box)
- Obs. assimilated simultaneously at each grid point
- 100% parallel: fast
- No **adjoint** needed
- **4D LETKF extension**

Localization based on observations

Perform data assimilation in a local volume, choosing observations

The state estimate is updated at the central grid red dot

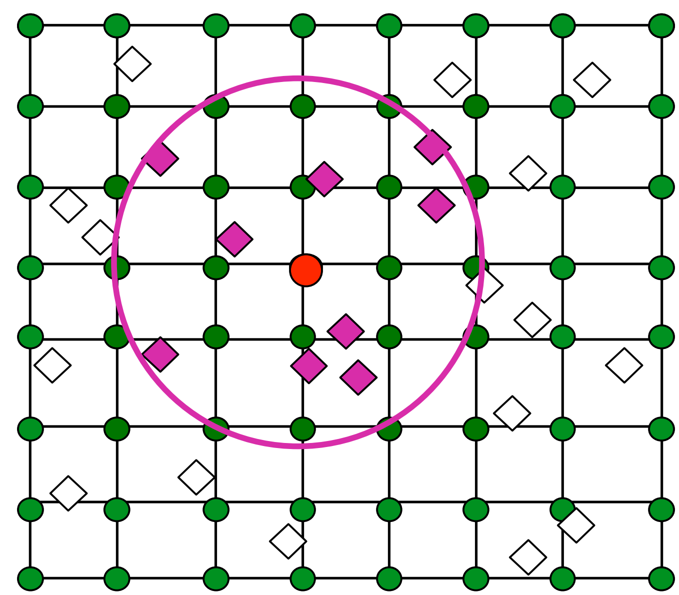


Localization based on observations

Perform data assimilation in a local volume, choosing observations

The state estimate is updated at the central grid red dot

All observations (purple diamonds) within the local region are assimilated



The LETKF algorithm can be described **in a single slide**!

Local Ensemble Transform Kalman Filter (LETKF)

Globally:

Forecast step: $\mathbf{x}_{n,k}^{b} = M_n\left(\mathbf{x}_{n-1,k}^{a}\right)$

Analysis step: construct $\mathbf{X}^b = \left[\mathbf{x}_1^b - \overline{\mathbf{x}}^b \mid \ldots \mid \mathbf{x}_K^b - \overline{\mathbf{x}}^b\right]$;

$$\mathbf{y}_i^b = H(\mathbf{x}_i^b);\ \mathbf{Y}_n^b = \left[\mathbf{y}_1^b - \overline{\mathbf{y}}^b \mid \ldots \mid \mathbf{y}_K^b - \overline{\mathbf{y}}^b\right]$$

Locally: Choose for each grid point the observations to be used, and compute the local analysis error covariance and perturbations in ensemble space:

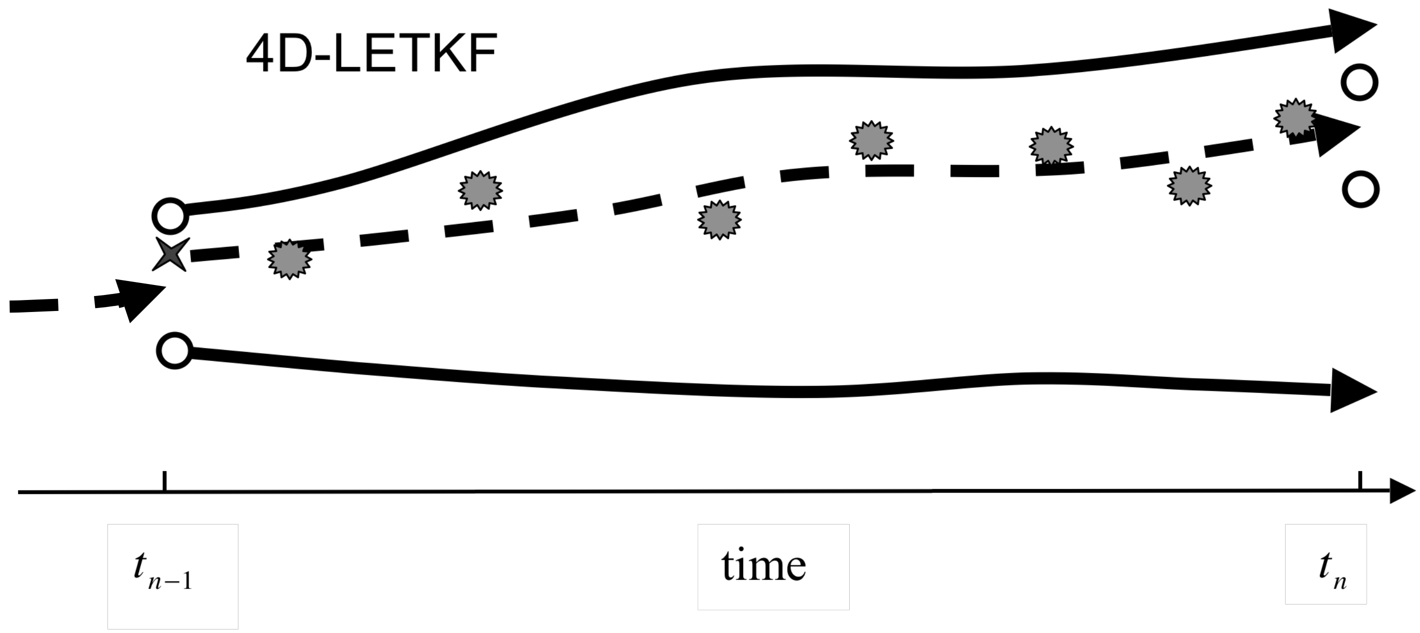
$$\tilde{\mathbf{P}}^a = \left[\left(K-1\right)\mathbf{I} + \mathbf{Y}^{bT}\mathbf{R}^{-1}\mathbf{Y}^b\right]^{-1};\ \mathbf{W}^a = [(K-1)\tilde{\mathbf{P}}^a]^{1/2}$$

Analysis mean in ensemble space: $\overline{\mathbf{w}}^a = \tilde{\mathbf{P}}^a\mathbf{Y}^{bT}\mathbf{R}^{-1}(\mathbf{y}^o - \overline{\mathbf{y}}^b)$

and add to $\mathbf{W}^a$ to get the analysis ensemble in ensemble space.

The new ensemble analyses in model space are the columns of $\mathbf{X}_n^a = \mathbf{X}_n^b\mathbf{W}^a + \overline{\mathbf{x}}^b$. Gathering the grid point analyses forms the new global analyses. Note that the the output of the LETKF are analysis weights $\overline{\mathbf{w}}^a$ and perturbation analysis matrices of weights $\mathbf{W}^a$. **These weights multiply the ensemble forecasts.**

No-cost LETKF smoother (✗): apply at t_{n-1} the same weights found optimal at t_n. It works for 3D- or 4D-LETKF



The no-cost smoother makes possible:

- ✓ Outer loop (like in 4D-Var)
- ✓ "Running in place" (faster spin-up)
- ✓ Use of future data in reanalysis
- ✓ Ability to use longer windows
- ✓ Coarse resolution analysis without degradation

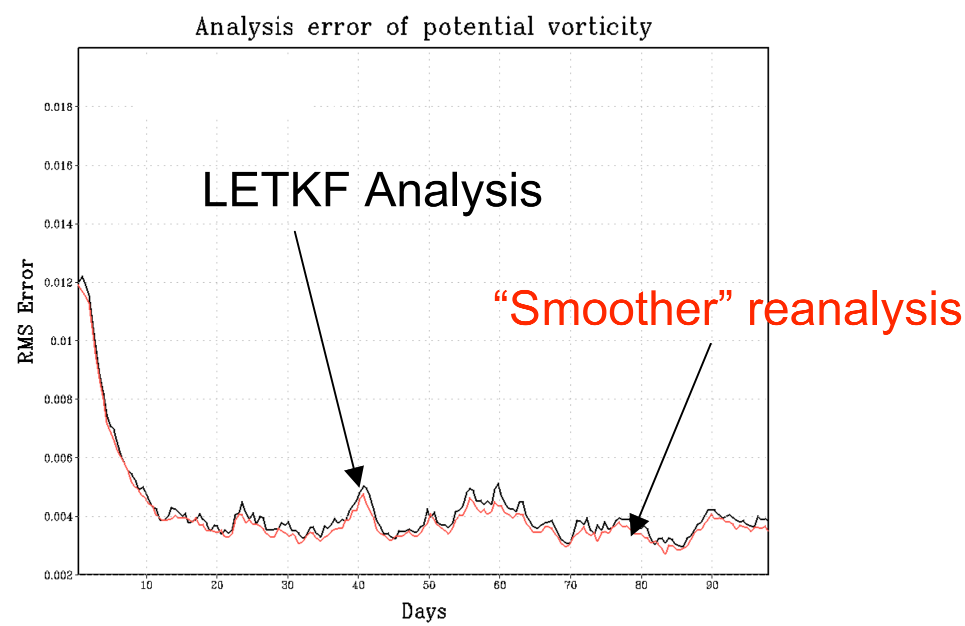
No-cost LETKF smoother tested on a QG model: it works…

LETKF analysis
at time n

$$\overline{\mathbf{x}}_n^a = \overline{\mathbf{x}}_n^f + \mathbf{X}_n^f \overline{\mathbf{w}}_n^a$$

Smoother analysis
at time n-1

$$\tilde{\mathbf{x}}_{n-1}^a = \overline{\mathbf{x}}_{n-1}^f + \mathbf{X}_{n-1}^f \overline{\mathbf{w}}_n^a$$



This very simple smoother allows us to go back and forth in time within an assimilation window: it allows assimilation of **future** data in reanalysis

Nonlinearities and “outer loop”

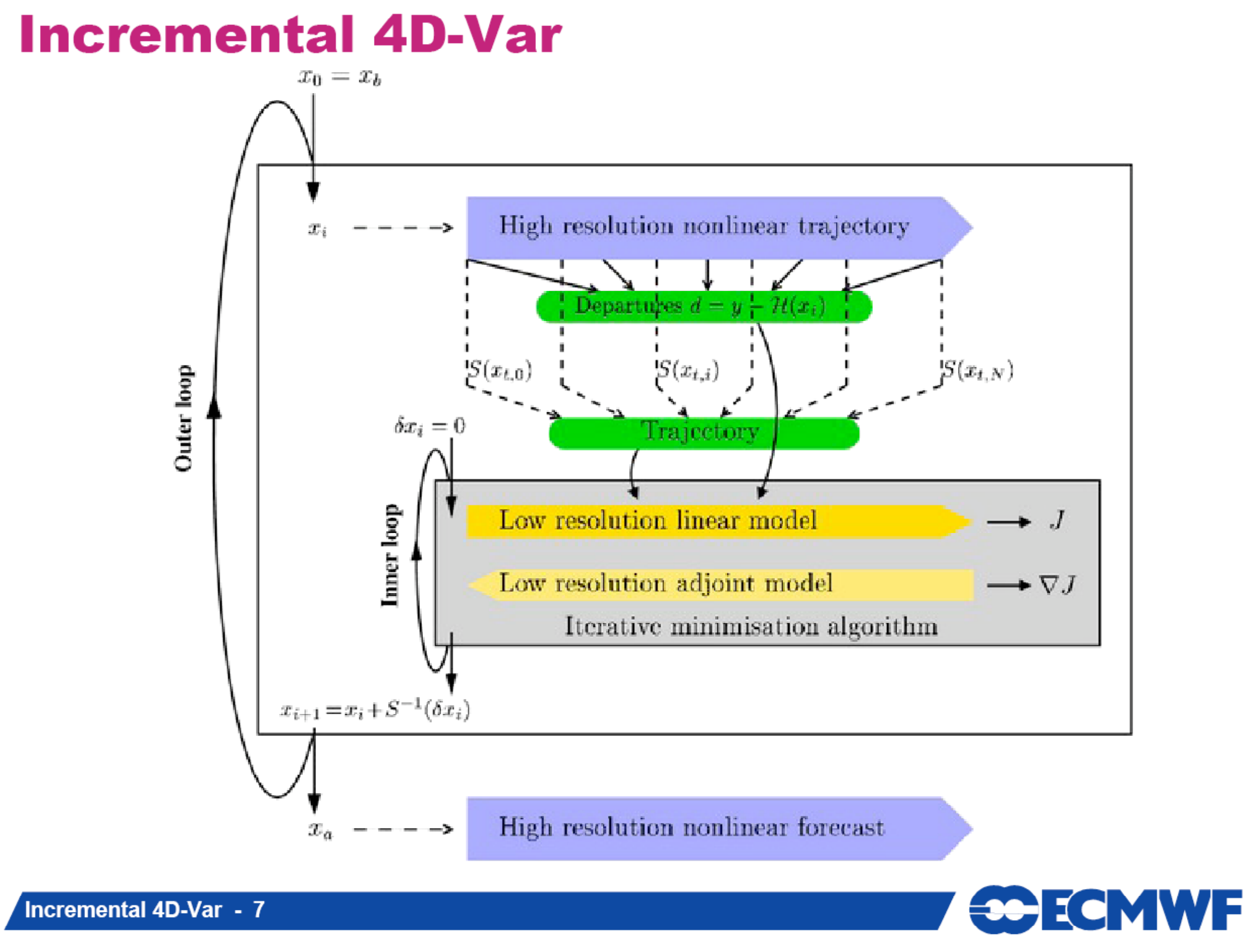
- The main disadvantage of EnKF is that it cannot handle nonlinear (non-Gaussian) perturbations and therefore needs short assimilation windows.
- It doesn’t have the **outer loop** so important in 3D-Var and 4D-Var (DaSilva, pers. comm. 2006)

Lorenz -3 variable model (Kalnay et al. 2007a Tellus), RMS analysis error

	4D-Var	LETKF
Window=8 steps	0.31	**0.30** (linear window)
Window=25 steps	**0.53**	0.66 (nonlinear window)

With long windows + Pires et al. => 4D-Var clearly wins!

“Outer loop” in 4D-Var



Nonlinearities: "Outer Loop"

Outer loop: similar to 4D-Var: use the final weights to correct only the mean initial analysis, keeping the initial perturbations. Repeat the analysis once or twice. It re-centers the ensemble on a more accurate nonlinear solution.

Lorenz -3 variable model RMS analysis error

	4D-Var	LETKF	LETKF +outer loop	LETKF +RIP
Window=8 steps	0.31	0.30	**0.27**	0.27
Window=25 steps	0.53	0.66	**0.48**	**0.39**

"Running in place" smoothes both the **analysis** and the **analysis error covariance** and iterates a few times…

Nonlinearities, “Outer Loop” and “Running in Place”

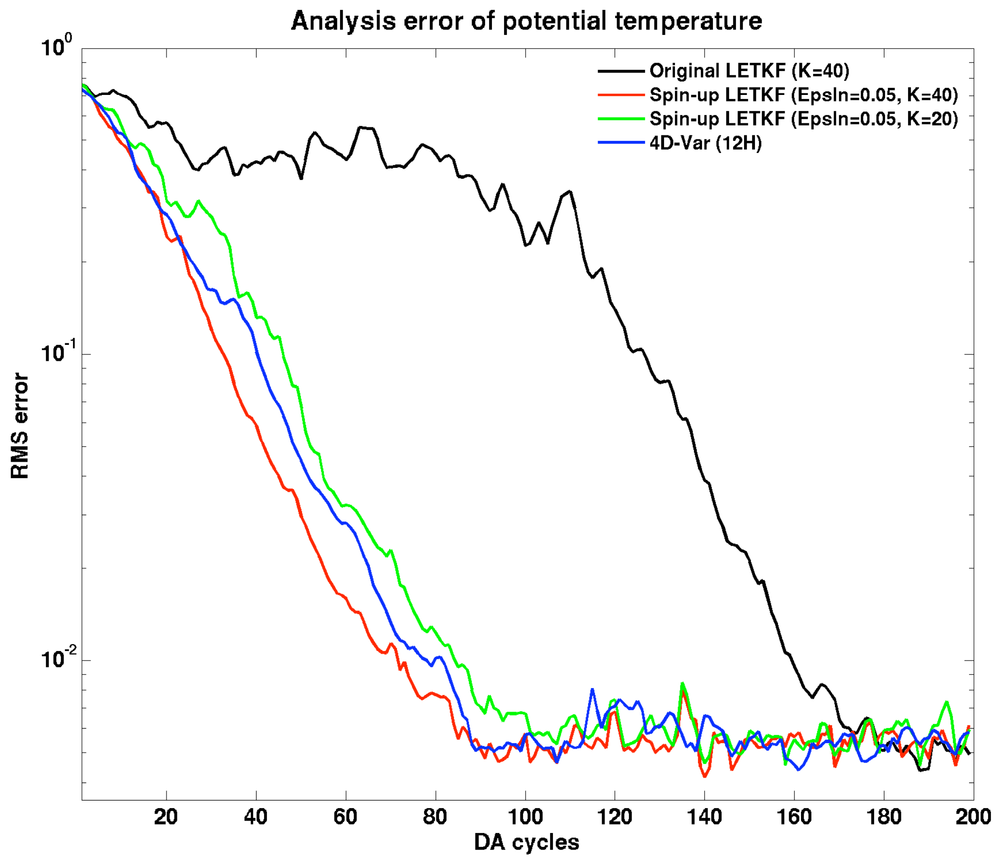
Outer loop: similar to 4D-Var: use the final weights to correct only the mean initial analysis, keeping the initial perturbations. Repeat the analysis once or twice. It centers the ensemble on a more accurate nonlinear solution.

Lorenz -3 variable model RMS analysis error

	4D-Var	LETKF	LETKF +outer loop	LETKF +RIP
Window=8 steps	0.31	0.30	**0.27**	0.27
Window=25 steps	0.53	0.66	**0.48**	**0.39**

“Running in place” smoothes both the **analysis** and the **analysis error covariance** and iterates a few times…

Running in Place: Results with a QG model



Spin-up depends on initial perturbations, but RIP works well even with random perturbations. It becomes as fast as 4D-Var (blue). RIP takes only 2-6 iterations.

Running in Place: Results with a QG model

	LETKF Random initial ensemble		LETKF B3DV initial ensemble		LETKF, Random initial ensemble	Variational	
	No RIP	With RIP	No RIP	With RIP	Fixed 10 iterations RIP	3D-Var B3DV	4D-Var 0.05B3DV
Spin-up: DA cycles to reach 5% error	**141**	**46**	**54**	**37**	**37**	**44**	**54**
RMS error ($x10^{-2}$)	0.5	0.54	0.5	0.52	1.16	1.24	0.54

LETKF spin-up from random perturbations: 141 cycles. With RIP: 46 cycles

LETKF spin-up from 3D-Var perts. 54 cycles. With RIP: 37 cycles

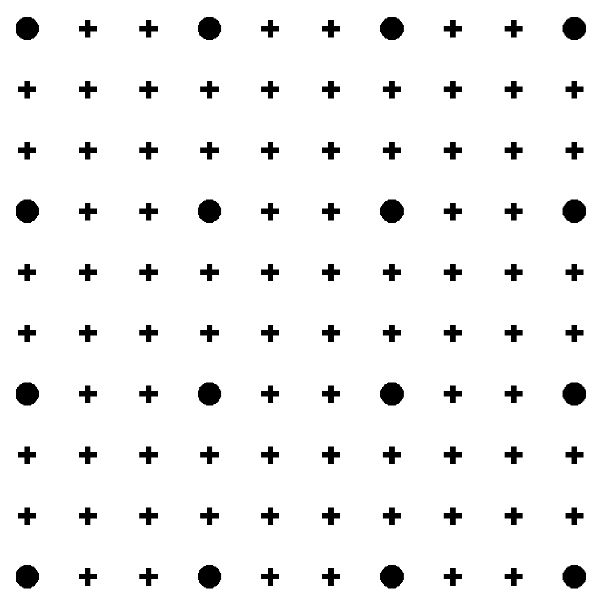
4D-Var spin-up using 3D-Var prior: 54 cycles.

RIP is robust, with or without prior information

Coarse analysis with interpolated weights

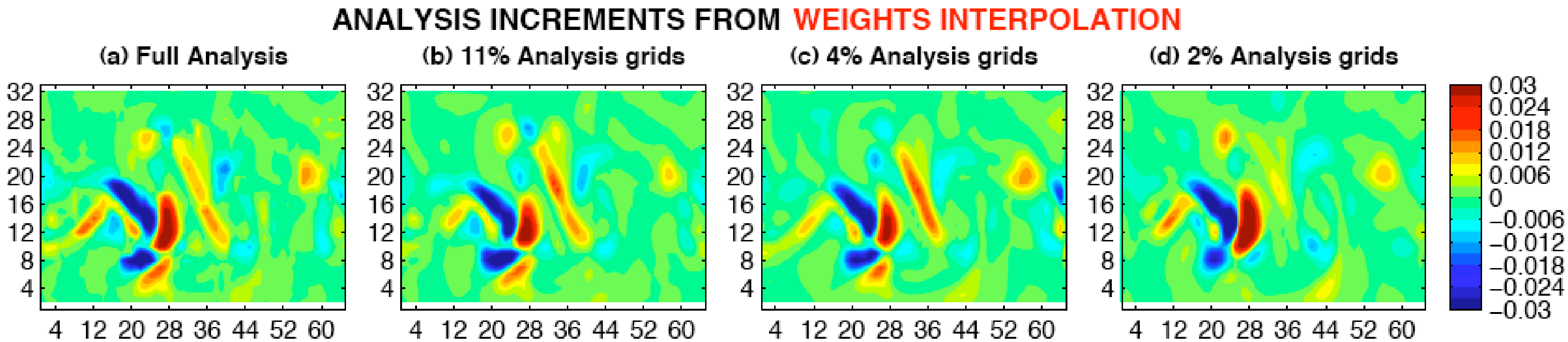
Yang et al (2008)

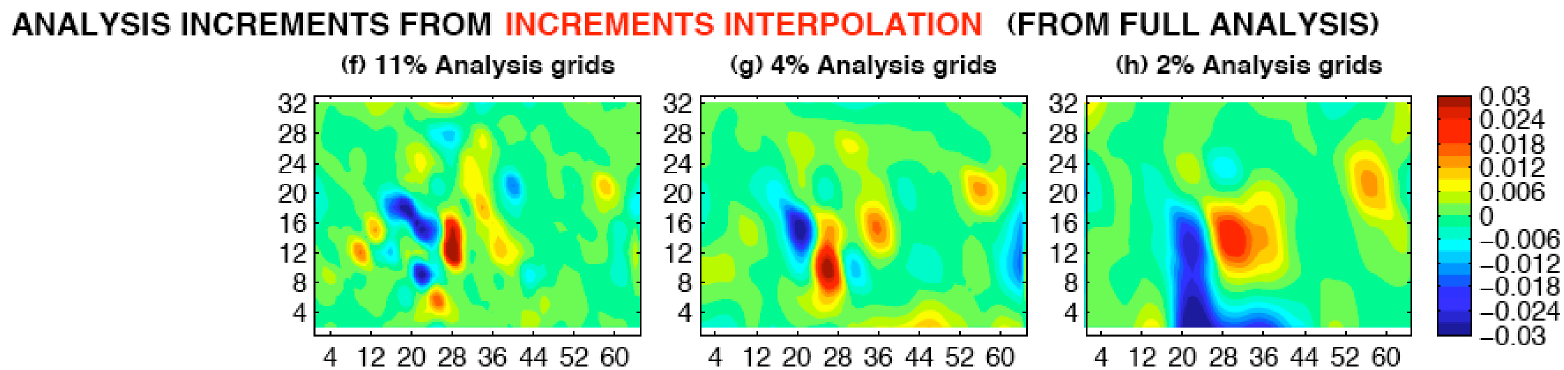
- In EnKF the analysis is a weighted average of the forecast ensemble
- We performed experiments with a QG model **interpolating** **<u>weights</u>** compared to **analysis** **<u>increments</u>**.
- Coarse grids of 11%, 4% and 2% interpolated analysis points.
- **Weight fields vary on large scales: they interpolate very well**



1/(3x3)=11% analysis grid

Weight interpolation versus **Increment** interpolation



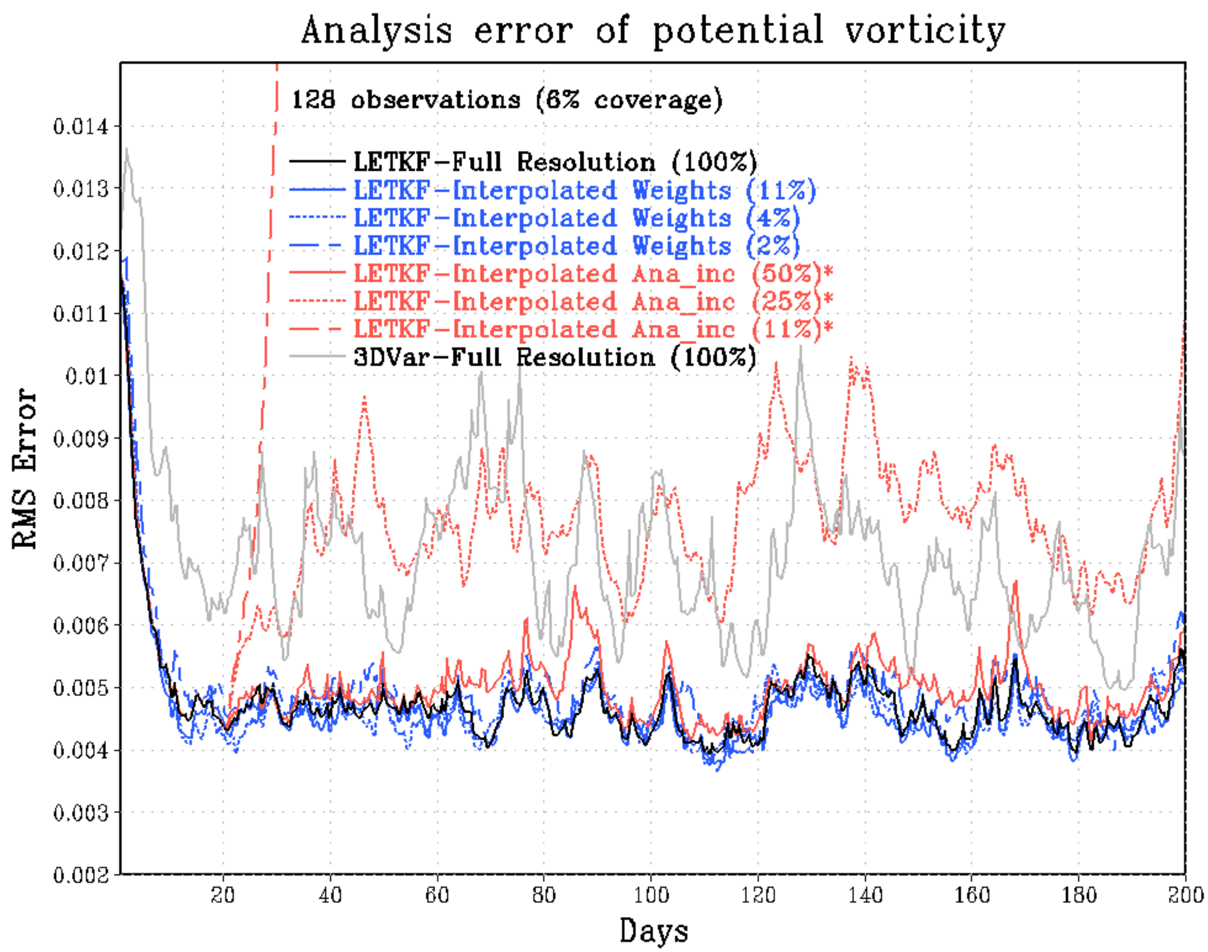


With increment interpolation, the analysis degrades quickly…

With weight interpolation, there is almost no degradation!

LETKF maintains balance and conservation properties

Impact of coarse analysis on accuracy

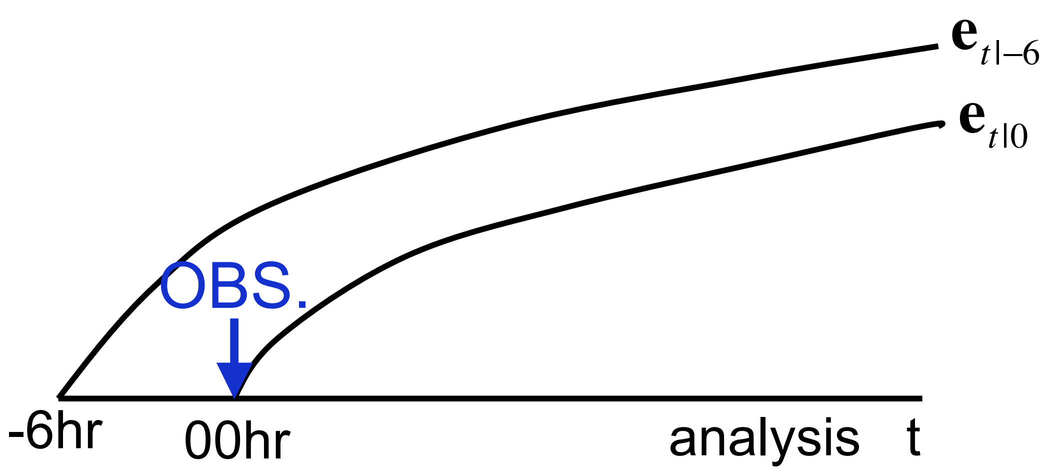


With increment interpolation, the analysis degrades

With weight interpolation, there is no degradation, the analysis is actually slightly better!

Forecast sensitivity to observations

Liu & Kalnay' 08; Li, Liu, and Kalnay, in preparation



$$\mathbf{e}_{t|0} = \overline{\mathbf{x}}^f_{t|0} - \overline{\mathbf{x}}^a_t$$

(Adapted from Langland and Baker, 2004)

The only difference between $\mathbf{e}_{t|0}$ and $\mathbf{e}_{t|-6}$ is the assimilation of observations at 00hr:

$$(\overline{\mathbf{x}}^a_0 - \overline{\mathbf{x}}^b_{0|-6}) = \mathbf{K}(\mathbf{y} - H(\mathbf{x}^b_{0|-6}))$$

- Observation impact on the reduction of forecast error:

$$\Delta\mathbf{e}^2 = (\mathbf{e}^T_{t|0}\mathbf{e}_{t|0} - \mathbf{e}^T_{t|-6}\mathbf{e}_{t|-6}) = (\mathbf{e}^T_{t|0} - \mathbf{e}^T_{t|-6})(\mathbf{e}_{t|0} + \mathbf{e}_{t|-6})$$

Forecast sensitivity to observations

$$\Delta \mathbf{e}^2 = (\mathbf{e}_{t|0}^T \mathbf{e}_{t|0} - \mathbf{e}_{t|-6}^T \mathbf{e}_{t|-6}) = (\mathbf{e}_{t|0}^T - \mathbf{e}_{t|-6}^T)(\mathbf{e}_{t|0} + \mathbf{e}_{t|-6})$$

$$= (\overline{\mathbf{x}}_{t|0}^f - \overline{\mathbf{x}}_{t|-6}^f)^T (\mathbf{e}_{t|0} + \mathbf{e}_{t|-6})$$

$$= \left[\mathbf{M}(\overline{\mathbf{x}}_0^a - \overline{\mathbf{x}}_{0|-6}^b) \right]^T (\mathbf{e}_{t|0} + \mathbf{e}_{t|-6}), \text{ so that}$$

$$\Delta \mathbf{e}^2 = \left[\mathbf{MK}(\mathbf{y} - H(\mathbf{x}_{0|-6}^b)) \right]^T (\mathbf{e}_{t|0} + \mathbf{e}_{t|-6})$$

Langland and Baker (2004) solve this with the adjoint:

$$\Delta \mathbf{e}^2 = \left[(\mathbf{y} - H(\mathbf{x}_{0|-6}^b)) \right]^T \mathbf{K}^T \mathbf{M}^T (\mathbf{e}_{t|0} + \mathbf{e}_{t|-6})$$

This requires the adjoint of the model $\mathbf{M}^T$ and of the data assimilation system $\mathbf{K}^T$ (Langland and Baker, 2004)

Forecast sensitivity to observations

Langland and Baker (2004):

$$\Delta \mathbf{e}^2 = \left[\mathbf{MK}(\mathbf{y} - H(\mathbf{x}^b_{0|-6})\right]^T (\mathbf{e}_{t|0} + \mathbf{e}_{t|-6})$$
$$= \left[(\mathbf{y} - H(\mathbf{x}^b_{0|-6})\right]^T \mathbf{K}^T \mathbf{M}^T (\mathbf{e}_{t|0} + \mathbf{e}_{t|-6})$$

With EnKF we can use the original equation without “adjointing”:

Recall that $\mathbf{K} = \mathbf{P}^a \mathbf{H}^T \mathbf{R}^{-1} = 1/(K-1)\mathbf{X}^a \mathbf{X}^{aT} \mathbf{H}^T \mathbf{R}^{-1}$ so that

$$\mathbf{MK} = 1/(K-1)\mathbf{MX}^a(\mathbf{X}^{aT}\mathbf{H}^T)\mathbf{R}^{-1} = 1/(K-1)\mathbf{X}^f_{t|0}\mathbf{Y}^{aT}\mathbf{R}^{-1}$$

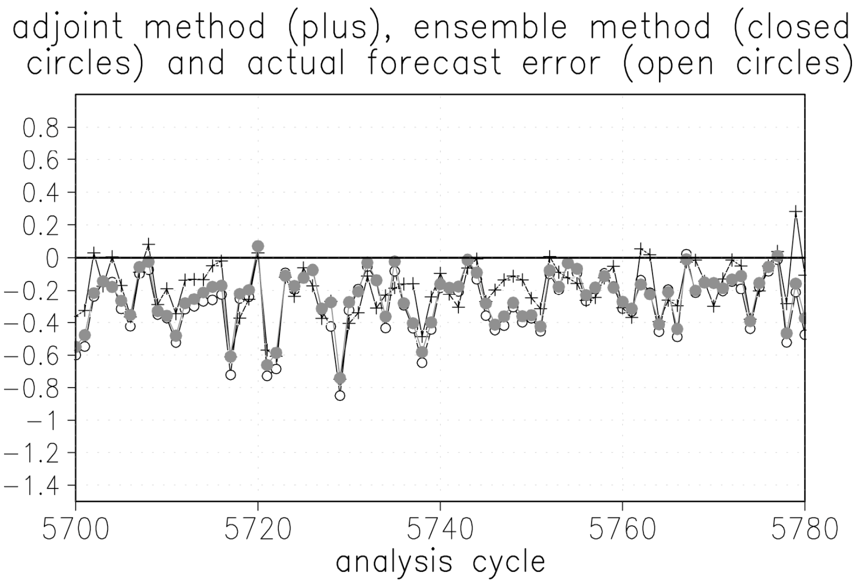
Thus,

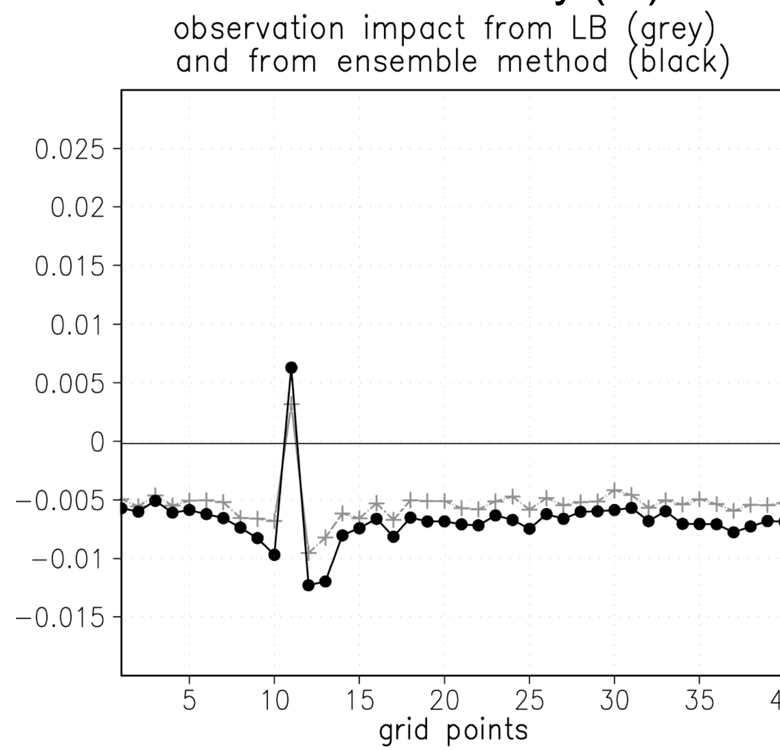
$$\Delta \mathbf{e}^2 = \left[\mathbf{MK}(\mathbf{y} - H(\mathbf{x}^b_{0|-6})\right]^T (\mathbf{e}_{t|0} + \mathbf{e}_{t|-6})$$
$$= \left[(\mathbf{y} - H(\mathbf{x}^b_{0|-6})\right]^T / (K-1)\, \mathbf{R}^{-1}\mathbf{Y}^a_0 \mathbf{X}^{fT}_{t|0}(\mathbf{e}_{t|0} + \mathbf{e}_{t|-6})$$

This is a product using the available **nonlinear** forecast ensemble $\mathbf{X}^{fT}_{t|0}$ and $\mathbf{Y}^a_0 = (\mathbf{HX}^a)$

Test ability to detect impacts and a poor quality observation in the Lorenz 40 variable model

Observation impact from LB(+) and from ensemble sensitivity (●)





✓Both the adjoint and the ensemble sensitivity detect the day-to-day impact of the observations on the 24 hr forecast.

✓Like the adjoint method, the ensemble sensitivity method can detect the observation poor quality (11th observation location)

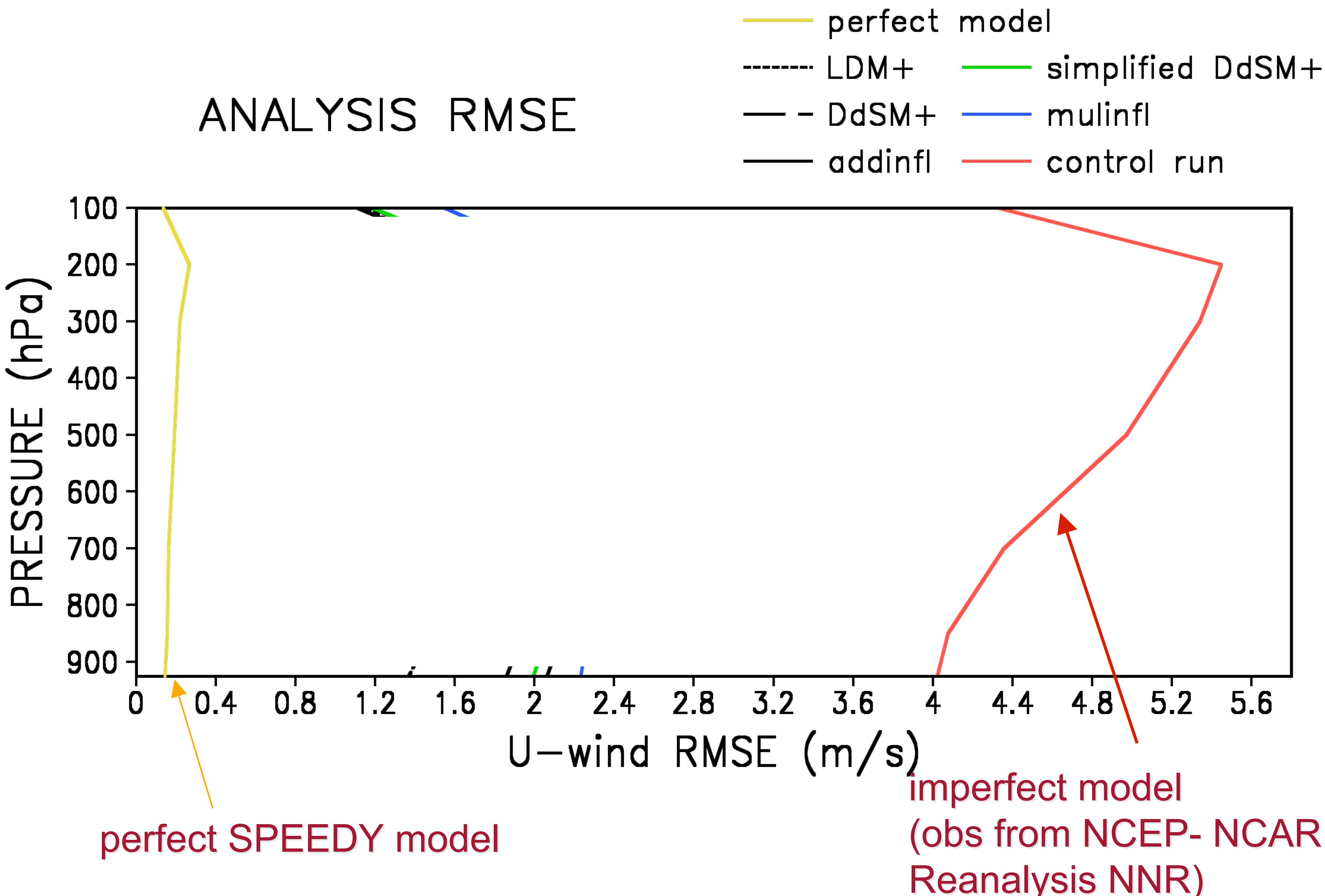
✓ The ensemble sensitivity is nonlinear and remains valid for longer forecasts

Model error: comparison of methods to correct model bias and inflation

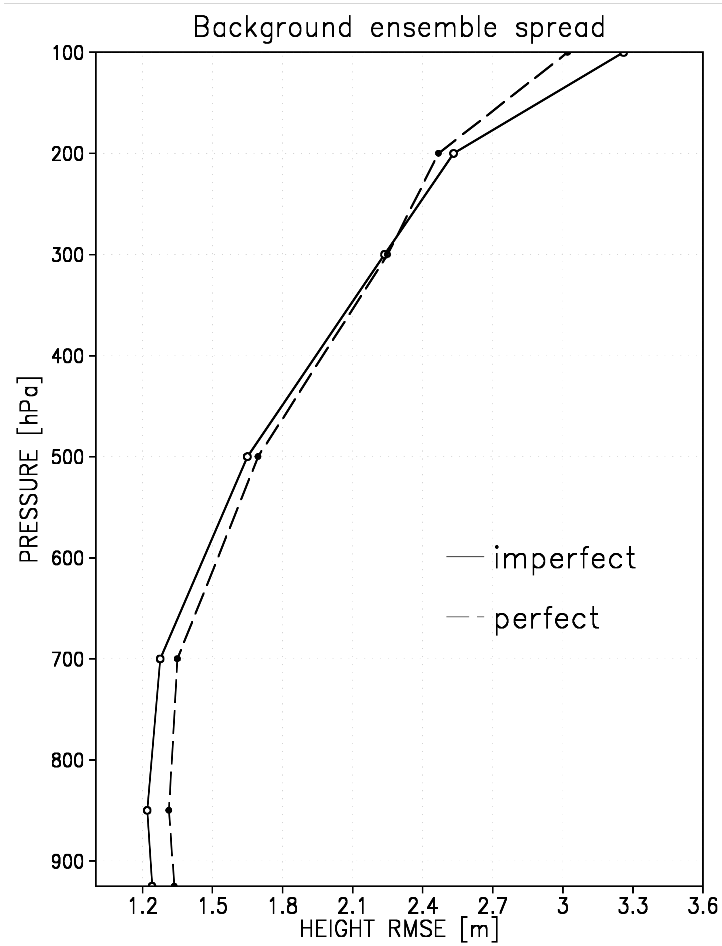
Hong Li, Chris Danforth, Takemasa Miyoshi,
and Eugenia Kalnay, MWR (2009)

Inspired by the work of Dick Dee, but with model errors estimated in model space, not in obs space

Model error: If we assume a perfect model in EnKF, we underestimate the analysis errors (Li, 2007)

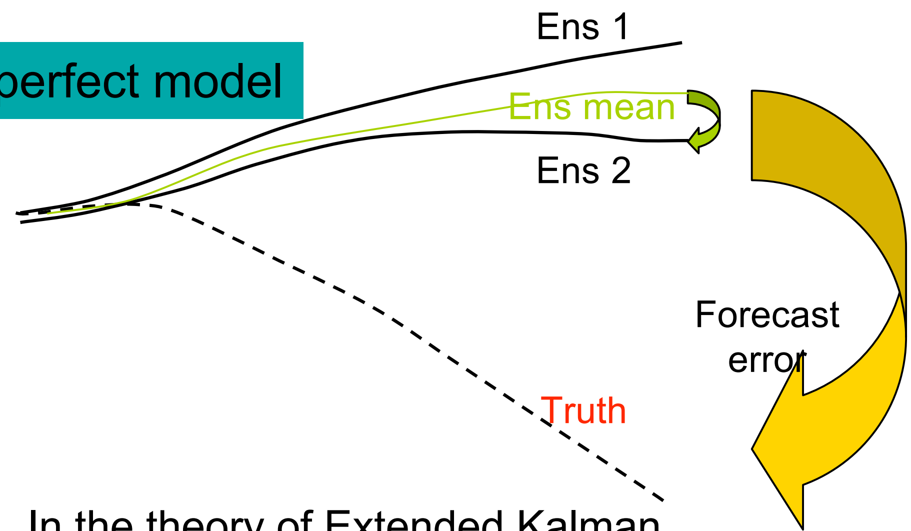


— Why is EnKF vulnerable to model errors ?



The ensemble spread is 'blind' to model errors

Imperfect model



- In the theory of Extended Kalman filter, forecast error is represented by the growth of errors in IC and the model errors.

$$\mathbf{P}_i^f = \mathbf{M}_{\mathbf{x}_{i-1}^a} \mathbf{P}_{i-1}^a \mathbf{M}_{\mathbf{x}_{i-1}^a}^T + \mathbf{Q}$$

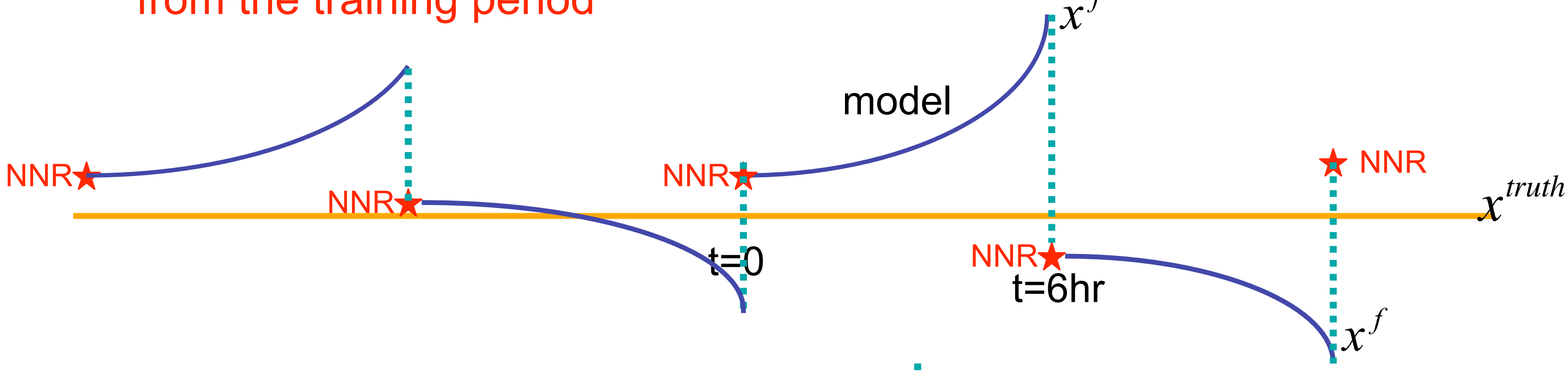
- However, in ensemble Kalman filter, error estimated by the ensemble spread can only represent the first type of errors.

$$\mathbf{P}_i^f \approx \frac{1}{k-1} \sum_{i=1}^{K} (x_i^f - \overline{x^f})(x_i^f - \overline{x^f})^T$$

Bias removal scheme: Low Dimensional Method

2.3 Low-dim method (Danforth et al, 2007: Estimating and correcting global weather model error. *Mon. Wea. Rev, J. Atmos. Sci., 2007*)

- Generate a long time series of model forecast minus reanalysis x^e_{6hr} from the training period



We collect a large number of estimated errors and estimate from them bias, etc.

$$\varepsilon^f_{n+1} = \mathbf{x}^f_{n+1} - \mathbf{x}^t_{n+1} = \boxed{M(\mathbf{x}^a_n) - M(\mathbf{x}^t_n)} + \boxed{\mathbf{b} + \sum_{l=1}^{L} \beta_{n,l}\mathbf{e}_l + \sum_{m=1}^{M} \gamma_{n,m}\mathbf{f}_m}$$

Forecast error due to error in IC

Time-mean model bias

Diurnal model error

State dependent model error

Low-dimensional method

Include Bias, Diurnal and State-Dependent model errors:

$$\text{model error} = \mathbf{b} + \sum_{l=1}^{2} \beta_{n,l}\mathbf{e}_l + \sum_{m=1}^{10} \gamma_{n,m}\mathbf{f}_m$$

Having a large number of estimated errors allows to estimate the global model error beyond the bias

SPEEDY 6 hr model errors against NNR (diurnal cycle)

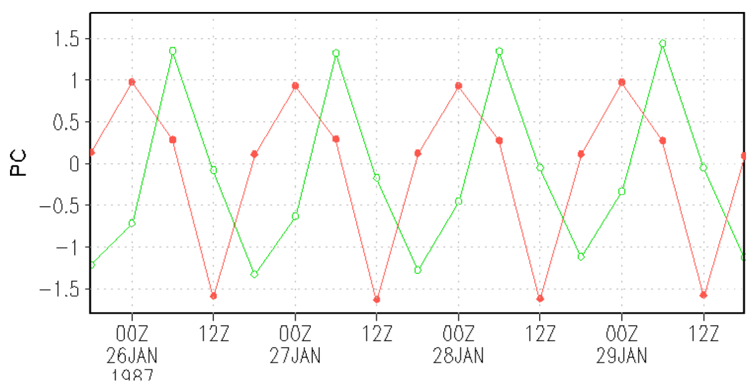
1987 Jan 1~ Feb 15

Error anomalies

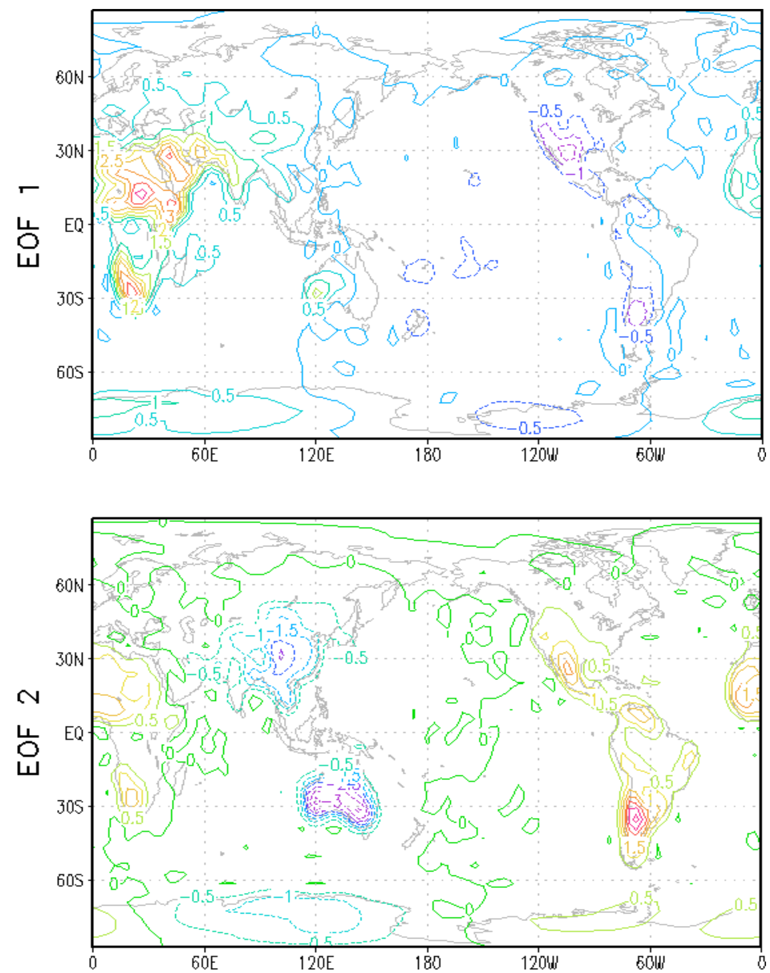
$$x^{e\,'}_{6hr(i)} = x^{e}_{6hr} - \overline{x^{e}_{6hr}}$$

pc1

pc2

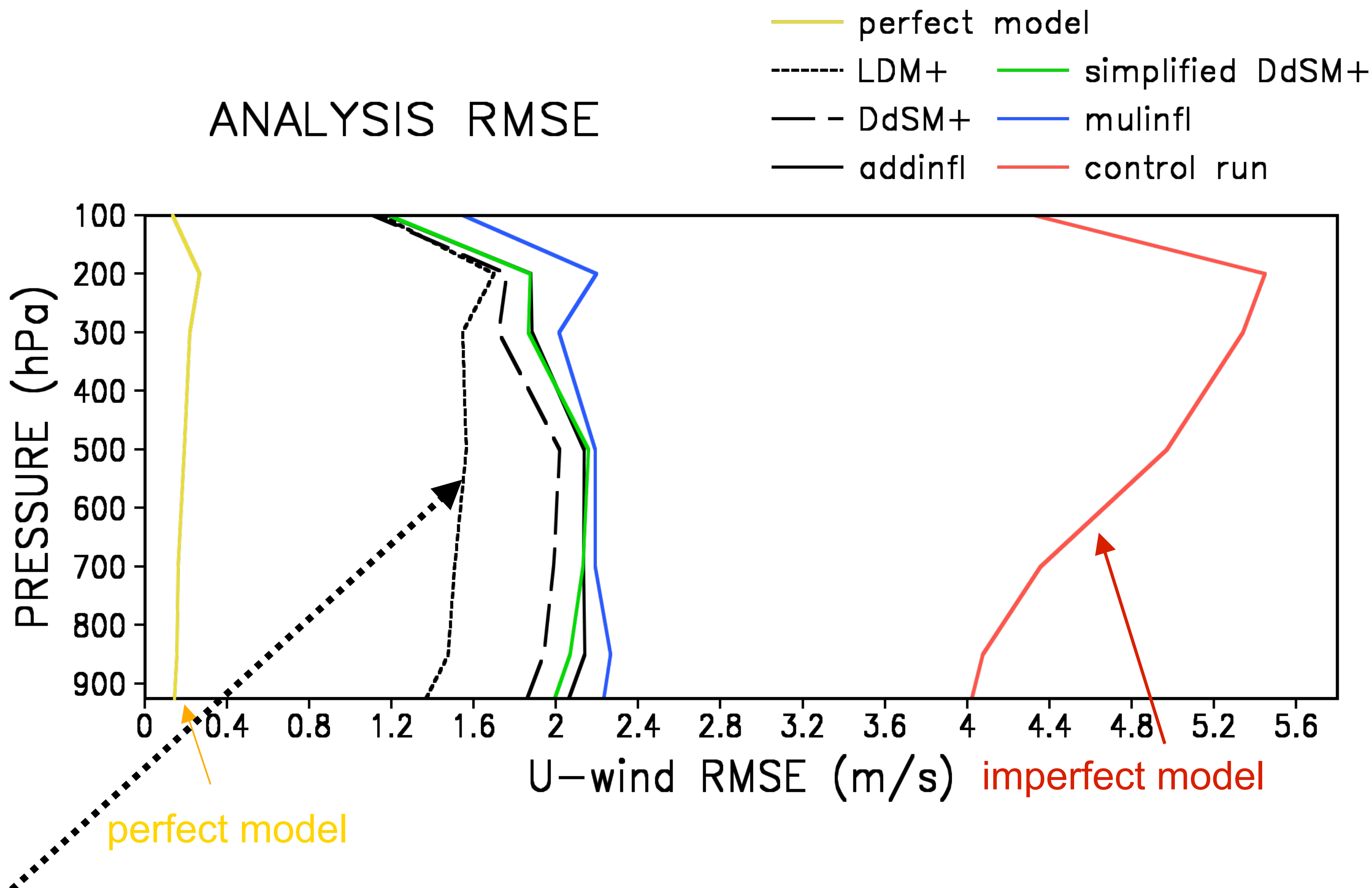


Leading EOFs for 925 mb TEMP



- For temperature at lower-levels, in addition to the time-independent bias, SPEEDY has strong diurnal cycle errors because it lacks diurnal radiation forcing

We compared several methods to handle bias and random model errors



Low Dimensional Method to correct the bias (Danforth et al, 2007) combined with additive inflation

Simultaneous estimation of EnKF inflation and obs errors in the presence of model errors

Hong Li, Miyoshi and Kalnay (QJ, 2009)

Inspired by Houtekamer et al. (2001) and Desroziers et al. (2005)

- Any data assimilation scheme requires accurate statistics for the observation and background errors (usually tuned or from gut feeling).

- EnKF needs inflation of the background error covariance: tuning is expensive

- We introduce a method to **simultaneously** estimate **ob errors** and **inflation**.

Diagnosis of observation error statistics

Houtekamer et al (2001) well known statistical relationship:

OMB*OMB $$< \mathbf{d}_{o-b} \mathbf{d}_{o-b}^T >= \mathbf{H}\mathbf{P}^b\mathbf{H}^T + \mathbf{R}$$

Desroziers et al, 2005, introduced two new statistical relationships:

OMA*OMB $$< \mathbf{d}_{o-a} \mathbf{d}_{o-b}^T >= \mathbf{R}$$

AMB*OMB $$< \mathbf{d}_{a-b} \mathbf{d}_{o-b}{}^T >= \mathbf{H}\mathbf{P}^b\mathbf{H}^T$$

These relationships are correct if the **R** and **B** statistics are correct and errors are uncorrelated!

With inflation: $\mathbf{H}\mathbf{P}^b\mathbf{H}^T \rightarrow \mathbf{H}\Delta\mathbf{P}^b\mathbf{H}^T$ with $\Delta > 1$

Diagnosis of observation error statistics

Transposing, we get "observations" of Δ and σ_o^2

$$\Delta^o = \frac{(\mathbf{d}_{o-b}^T \mathbf{d}_{o-b}) - Tr(\mathbf{R})}{Tr(\mathbf{HP}^b\mathbf{H}^\mathrm{T})}$$

OMB[2]

$$\Delta^o = \sum_{j=1}^{p} (y_j^a - y_j^b)(y_j^o - y_j^b) / Tr(\mathbf{HP}^b\mathbf{H}^T)$$

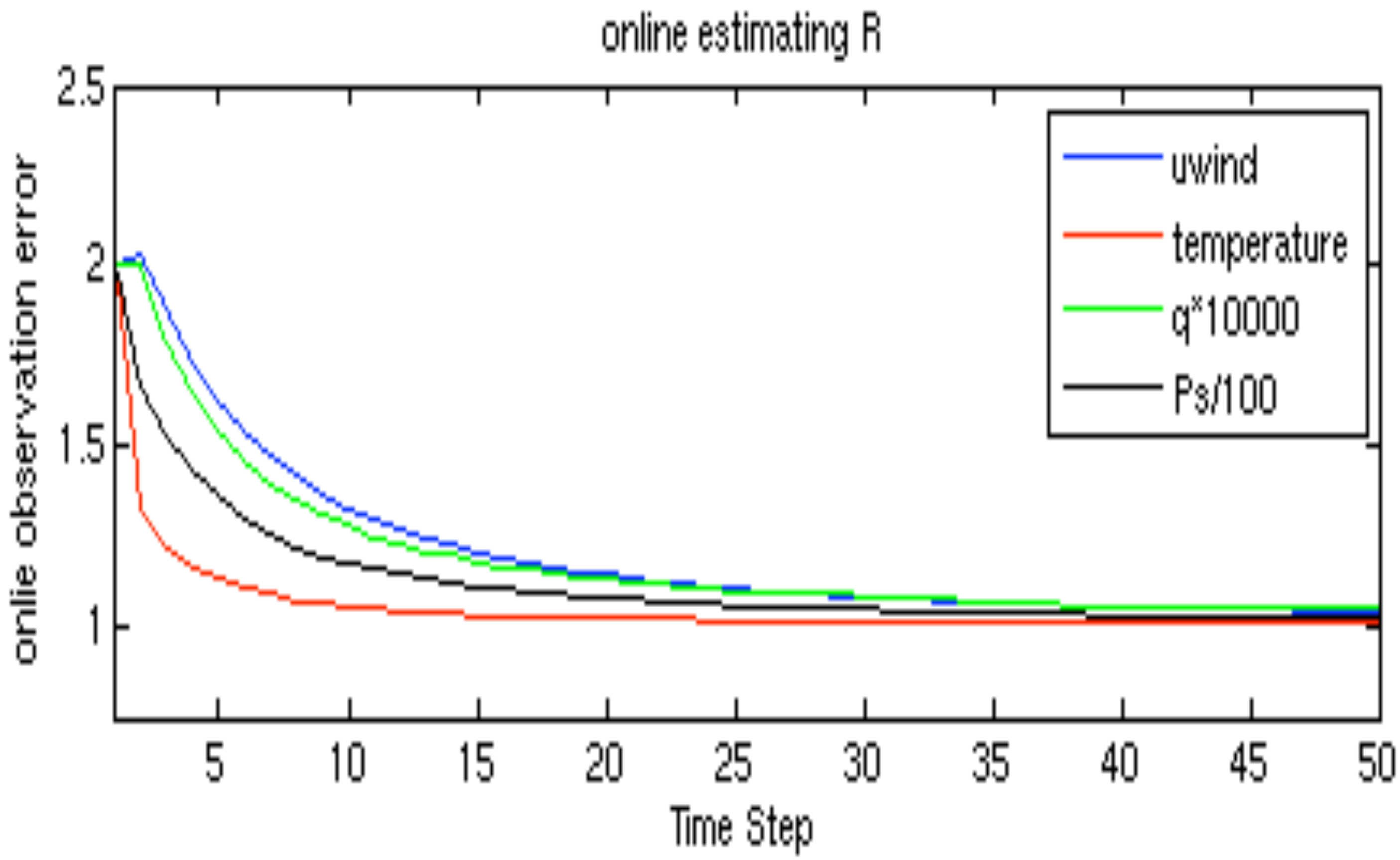
AMB*OMB

$$(\tilde{\sigma}_o)^2 = \mathbf{d}_{o-a}^T \mathbf{d}_{o-b} / p = \sum_{j=1}^{p} (y_j^o - y_j^a)(y_j^o - y_j^b) / p$$

OMA*OMB

Here we use a simple KF to estimate both Δ and σ_o^2 online.

SPEEDY model: online estimated observational errors, each variable started with 2 not 1

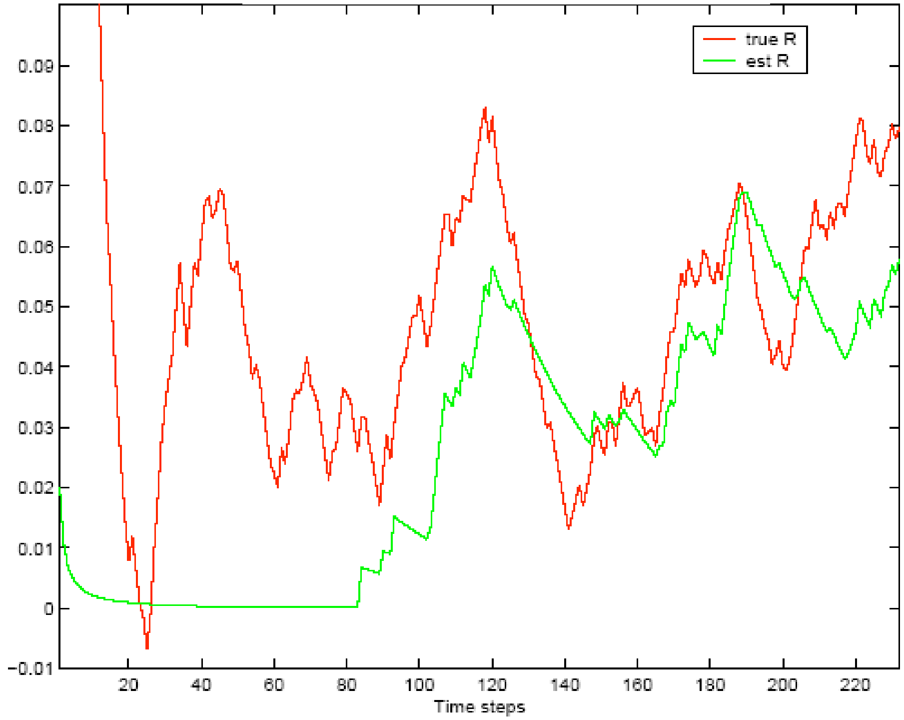


The original wrongly specified R quickly converges to the correct value of R (in about 5-10 days)

We are starting to test correlated ob errors

Estimation of the inflation

Estimated Inflation



Using an initially wrong R and Δ but estimating them adaptively

Using a perfect R and estimating Δ adaptively

After **R** converges, the time dependent inflation factors are quite similar

Tests with LETKF with imperfect L40 model: added random errors to the model

Error amplitude (random)	A: true σ_o^2=1.0 (tuned) constant Δ		B: true σ_o^2=1.0 adaptive Δ		C: adaptive σ_o^2 adaptive Δ		
a	Δ	RMSE	Δ	RMSE	Δ	RMSE	σ_o^2
4	0.25	0.36	0.27	0.36	0.39	0.38	0.93
20	0.45	0.47	0.41	0.47	0.38	0.48	1.02
100	1.00	0.64	0.87	0.64	0.80	0.64	1.05

The method works quite well even with very large random errors!

Tests with LETKF with imperfect L40 model: added biases to the model

Error amplitude (bias)	A: true σ_o^2=1.0 (tuned) constant Δ		B: true σ_o^2=1.0 adaptive Δ		C: adaptive σ_o^2 adaptive Δ		
α	Δ	RMSE	Δ	RMSE	Δ	RMSE	σ_o^2
1	0.35	0.40	0.31	0.42	0.35	0.41	0.96
4	1.00	0.59	0.78	0.61	0.77	0.61	1.01
7	1.50	0.68	1.11	0.71	0.81	0.80	1.36

The method works well for low biases, but less well for large biases: Model bias needs to be accounted by a separate bias correction

Summary

- EnKF and 4D-Var are now competitive (Buehner, Miyoshi)
- EnKF is better than GSI (3D-var) with half resolution model, 64 members. (Whitaker)
- Many ideas to further improve EnKF were inspired in 4D-Var:
  - No-cost smoothing and "running in place"
  - A simple outer loop to deal with nonlinearities
  - Adjoint forecast sensitivity without adjoint model
  - Analysis sensitivity and exact cross-validation
  - Coarse resolution analysis without degradation
  - Correction of model bias combined with additive inflation gives the best results
  - Can estimate simultaneously optimal inflation and ob. errors. Will test correlated ob errors.

Miyoshi's LETKF code

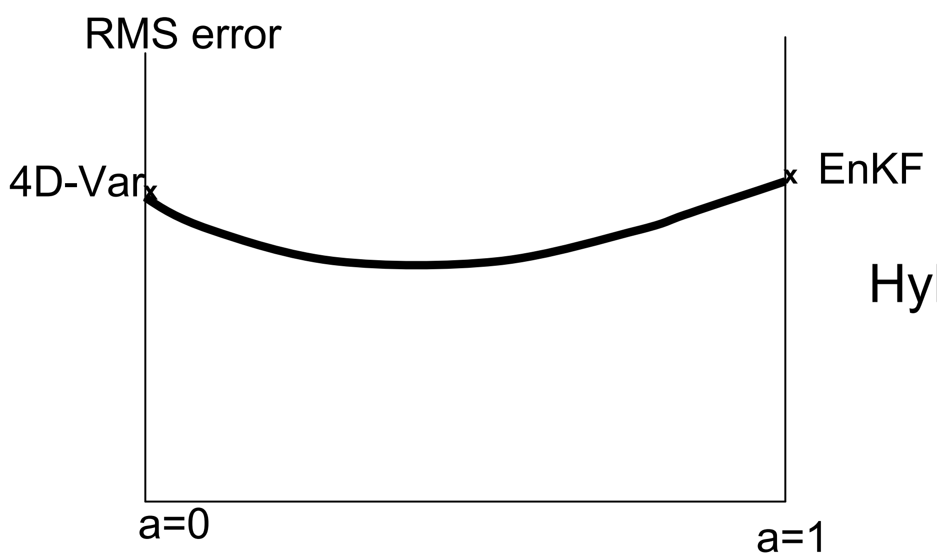
Takemasa Miyoshi has a wonderful new LETKF code based on the work he did at JMA. He has made it available to all at "Google code Miyoshi LETKF".

It is MPI (parallel) modular and very efficient, and the same code has been coupled to Lorenz (1996), SPEEDY and the Regional Ocean Modular System (ROMS) at high resolution.

Thoughts on hybrid

Dale Barker suggested that a fast path for NCEP to the use of hybrid would be to make first a GSI-EnKF hybrid, and then replace GSI with 4D-Var. Seems a very sensible idea.

As shown by Buehner et al., hybrid Var and EnKF may be the most accurate approach ("sweet spot").



Hybrid=a*EnKF+(1-a)*4D-Var

Final thoughts

- EnKF is relatively new, but it has been shown to be better than 3D-Var and comparable to 4D-Var.
- It is simple to program and maintain: no LTM, no adjoint, no background error covariance, adapts to changes in observing systems.
- Ideally the tuning parameters (inflation and obs. errors, and localization) will be estimated adaptively. Miyoshi developed a new adaptive localization.
- Applications and properties of 4D-Var can be easily adapted to EnKF.