Data Assimilation Systems: Focus on EnKF diagnostics

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

and Eugenia Kalnay

UMCP

Acknowledgements:

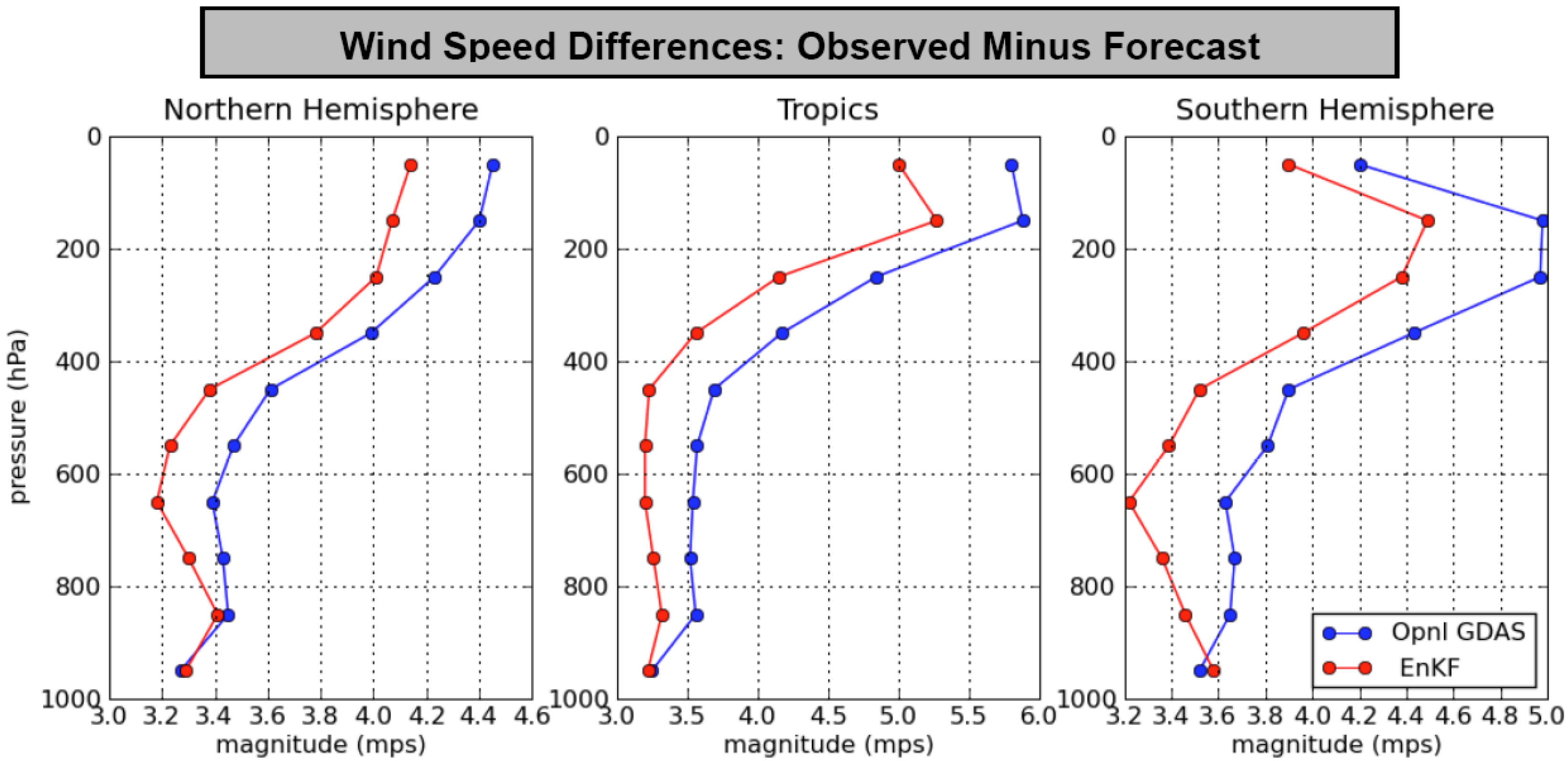
UMD Chaos-Weather Group: **Brian Hunt**, Istvan Szunyogh, Ed Ott, Jim Yorke, **Kayo Ide**, and students

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

Ensemble Kalman Filter: status and new ideas

- EnKF and 4D-Var are in a friendly competition:
- Jeff Whitaker results: EnKF better than GSI (3D-Var)
- Canada (Buehner): 4D-Var & EnKF the same in the NH and EnKF is better in the SH. Hybrid best.
- JMA (Miyoshi): at JMA, EnKF faster than 4D-Var, better in tropics and NH, worse in SH due to model bias.
- EnKF needs no adjoint model, priors, it adapts to changes in obs, it can even estimate ob errors.
- We "plagiarize" ideas and methods developed for 4D-Var and adapt them to the LETKF (Hunt et al., 2007)

Whitaker: Comparison of T190, 64 members EnKF with T382 operational GSI, same observations (JCSDA, 2009)



Vertical profiles of the RMS difference between six hour forecasts and in-situ observations for the period 2007120700 – 2008010718. Observations are aggregated in 100 hPa layers. The red curve is for the ensemble mean of the experimental 64-member T190 EnKF system, and the blue curve is for the T382 GSI-based GDAS system operational in December 2007.

There are several types of EnKF

1. Perturbed obs (e.g., Houtekamer and Mitchell)
2. Square root filters (e.g., Whitaker and Hamill)

Most filters get their speed from assimilating one observation at a time

The LETKF (Hunt 2005) assimilates all obs simultaneously and get its speed from local processing of each grid point

Because it is a Transform Square root filter, the LETKF analysis ensemble is explicitly expressed as a linear combination of the forecast ensemble

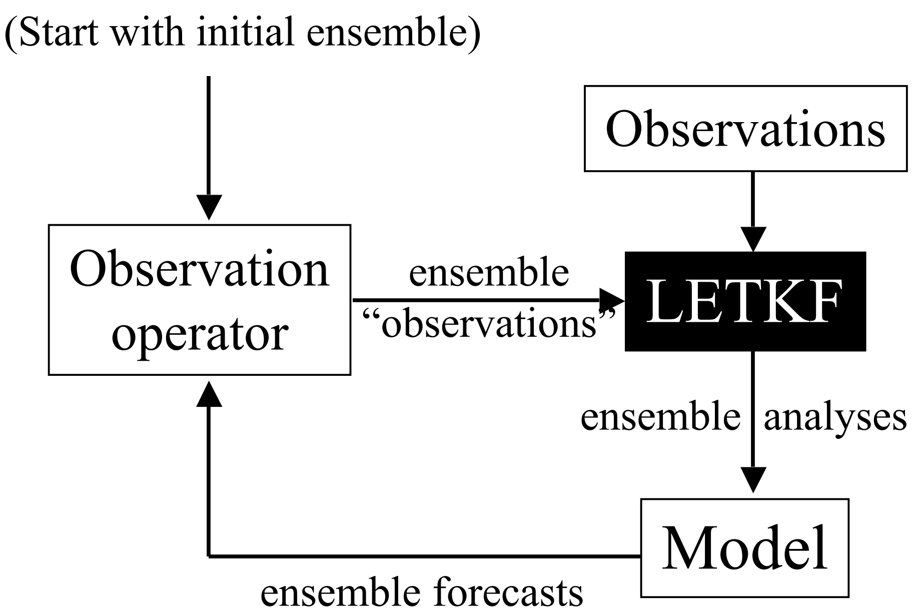
This has a number of nice properties, so here we will focus on the LETKF

Diagnostic tools that improve LETKF/EnKF

We adapted ideas that were 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 **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)

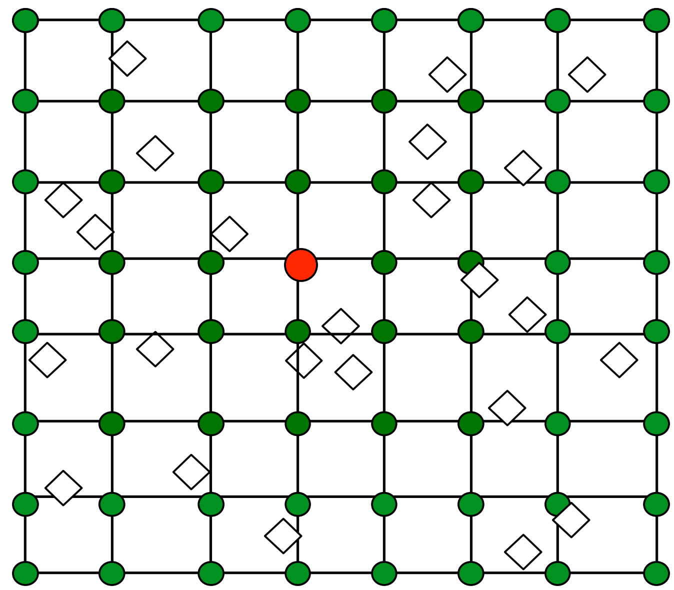


- Model independent (black box)
- Obs. assimilated simultaneously at each grid point
- 100% parallel: fast
- No **adjoint** needed
- **4D LETKF extension**

Localization based on observations

Perform data assimilation in a local volume, choosing observations

The state estimate is updated at the central grid red dot

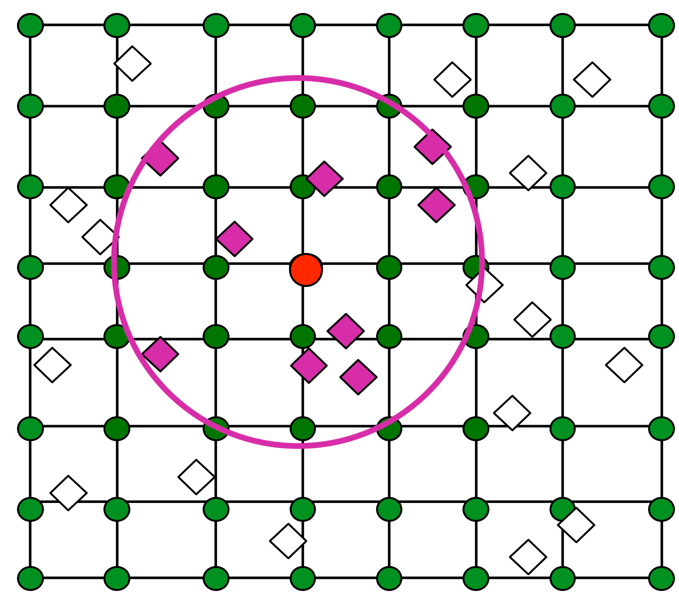


Localization based on observations

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 $\mathbf{X}^b = \left[\mathbf{x}_1^b - \overline{\mathbf{x}}^b \,|\, \ldots \,|\, \mathbf{x}_K^b - \overline{\mathbf{x}}^b\right];$

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

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

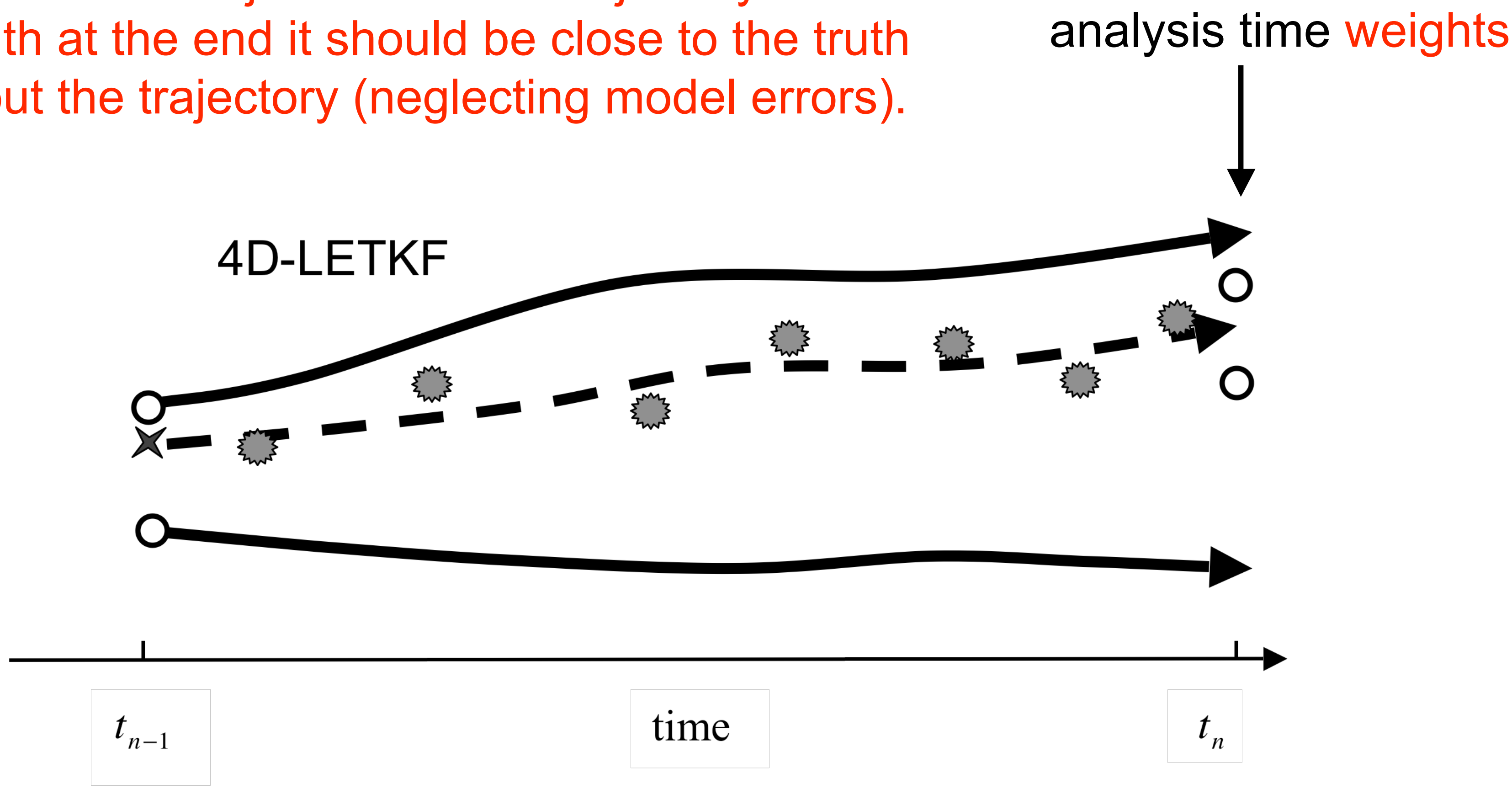
$$\tilde{\mathbf{P}}^a = \left[\left(K-1\right)\mathbf{I} + \mathbf{Y}^{bT}\mathbf{R}^{-1}\mathbf{Y}^b\right]^{-1};\ \mathbf{W}^a = [(K-1)\tilde{\mathbf{P}}^a]^{1/2}$$

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

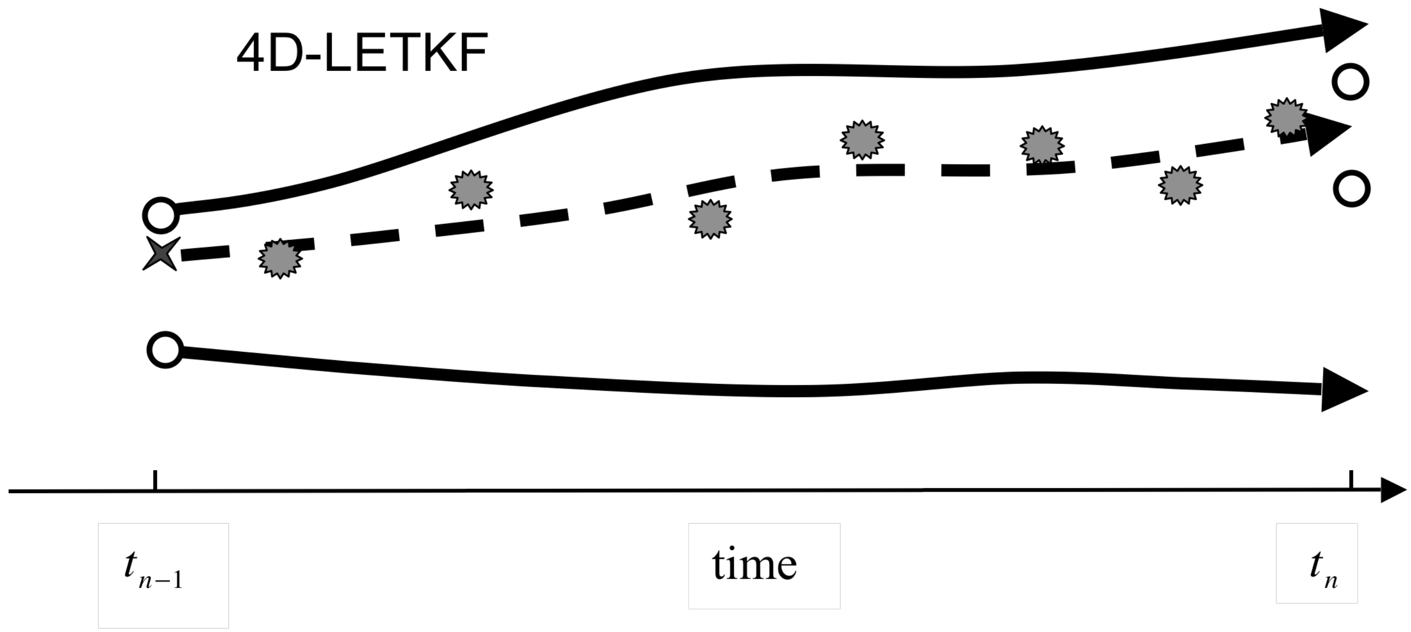
The new ensemble analyses in model space are the columns of $\mathbf{X}_n^a = \mathbf{X}_n^b\mathbf{W}^a + \overline{\mathbf{x}}^b$. Gathering the grid point analyses forms the new global analyses. Note that the the output of the LETKF are analysis weights $\overline{\mathbf{w}}^a$ and perturbation analysis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

A linear comb. of trajectories is ~a trajectory. If it is close to the truth at the end it should be close to the truth throughout the trajectory (neglecting model errors).



The 4D-LETKF produces an analysis in terms of **weights** of the ensemble forecast members at the analysis time t_n, giving the **trajectory** that best fits **all the observations** in the assimilation window.

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

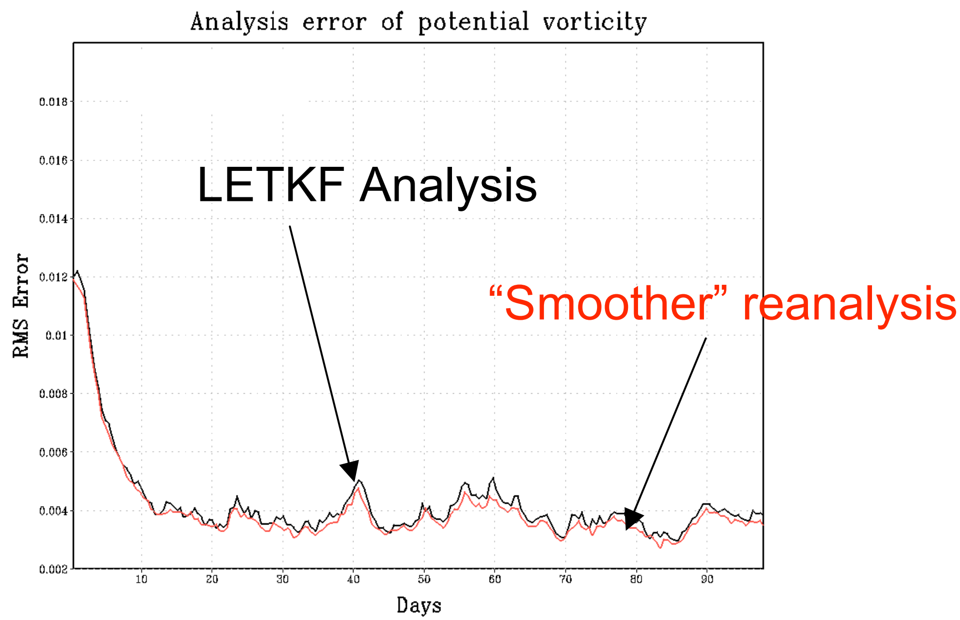
No-cost LETKF smoother tested on a QG model: It works!

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)

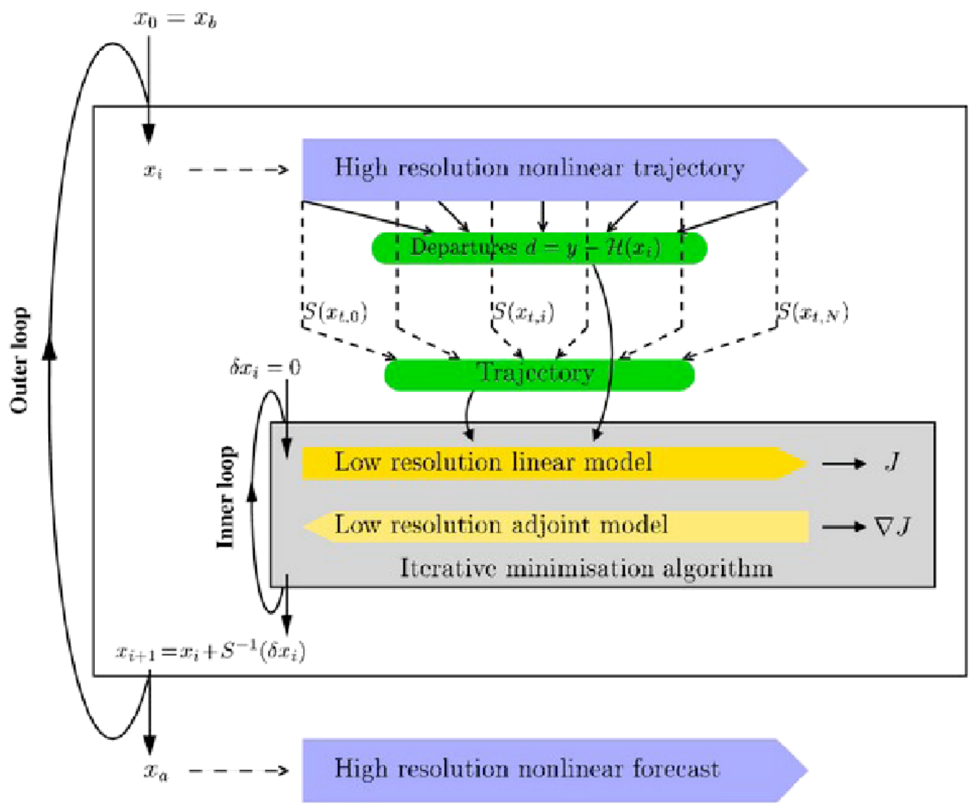
Lorenz -3 variable model (Kalnay et al. 2007a Tellus), RMS analysis error

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

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

“Outer loop” in 4D-Var

Incremental 4D-Var



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

Outer loop: similar to 4D-Var: use the final weights to correct only the <u>mean</u> 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…

Estimation of forecast sensitivity to observations without adjoint in an ensemble Kalman filter

Junjie Liu and Eugenia Kalnay

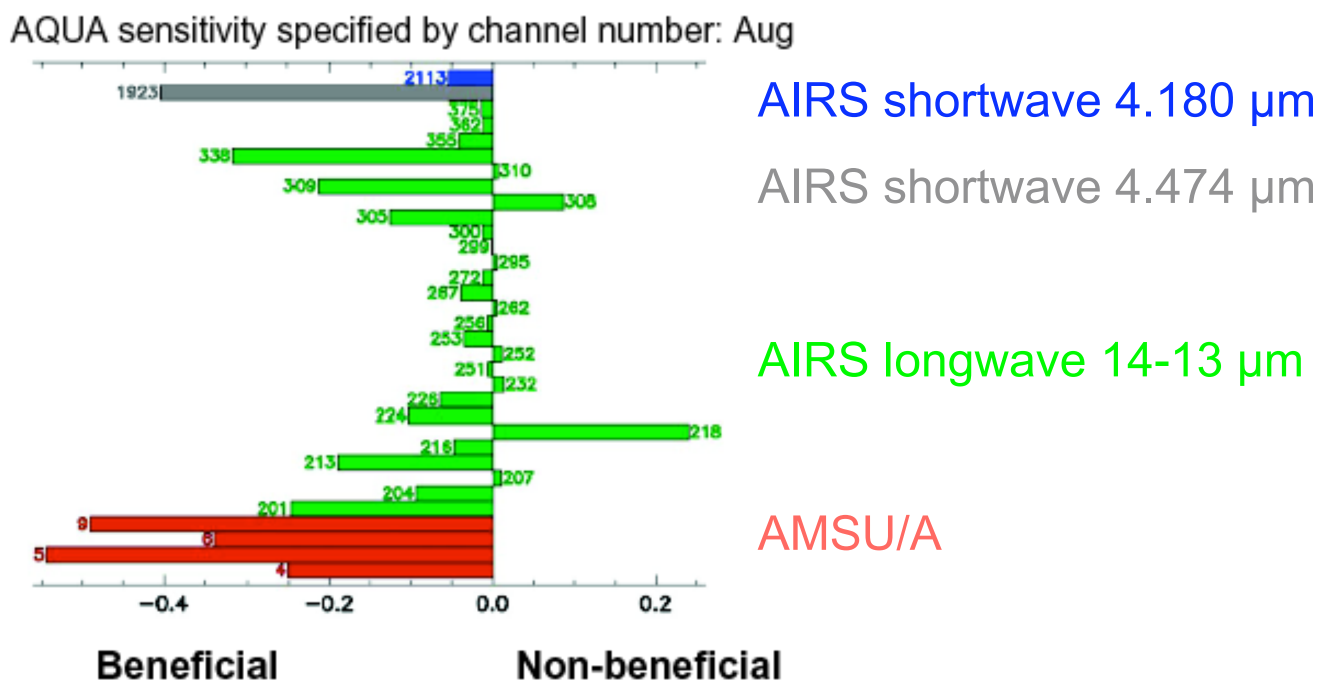
QJRMS October 2008

Inspired by Langland and Baker (2004)

and Zhu and Gelaro (2008)

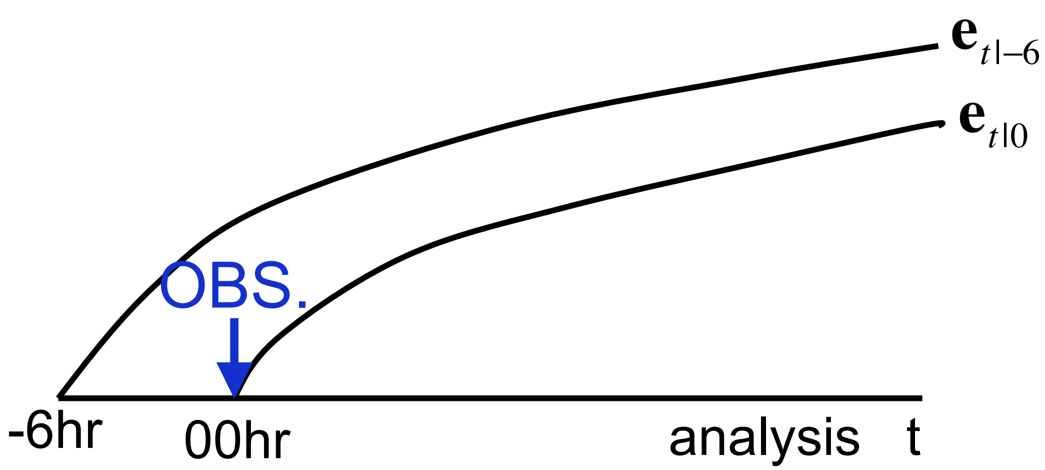
Ideal for diagnosing NCEP "5-day skill dropouts" because it remains valid for nonlinear perturbations

Motivation: Langland and Baker (2004)



- ➢ The adjoint method proposed by Langland and Baker (2004) and Zhu and Gelaro (2007) quantifies the reduction in forecast error for each individual observation source
- ➢ The adjoint method detects the observations which make the forecast worse.
- ➢ The adjoint method requires adjoint model which is difficult to get.

Schematic of the observation impact on the reduction of forecast error



$$\mathbf{e}_{t|-6} = \overline{\mathbf{x}}^{f}_{t|-6} - \overline{\mathbf{x}}^{a}_{t}$$

$$\mathbf{e}_{t|0} = \overline{\mathbf{x}}^{f}_{t|0} - \overline{\mathbf{x}}^{a}_{t}$$

(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.

- Observation impact on the reduction of forecast error: $J = \frac{1}{2}(\mathbf{e}^{T}_{t|0}\mathbf{e}_{t|0} - \mathbf{e}^{T}_{t|-6}\mathbf{e}_{t|-6})$

The ensemble forecast sensitivity method

Euclidian cost function: $J = \frac{1}{2}(\mathbf{e}_{t|0}^T \mathbf{e}_{t|0} - \mathbf{e}_{t|-6}^T \mathbf{e}_{t|-6})$ $\quad \mathbf{v}_0 = \mathbf{y}_0^o - h(\overline{\mathbf{x}}_{0|-6}^b)$

Cost function as function of obs. Increments: $J = \left\langle \mathbf{v}_0, \frac{\partial J}{\partial \mathbf{v}_0} \right\rangle$

The sensitivity of cost function with respect to the assimilated observations:

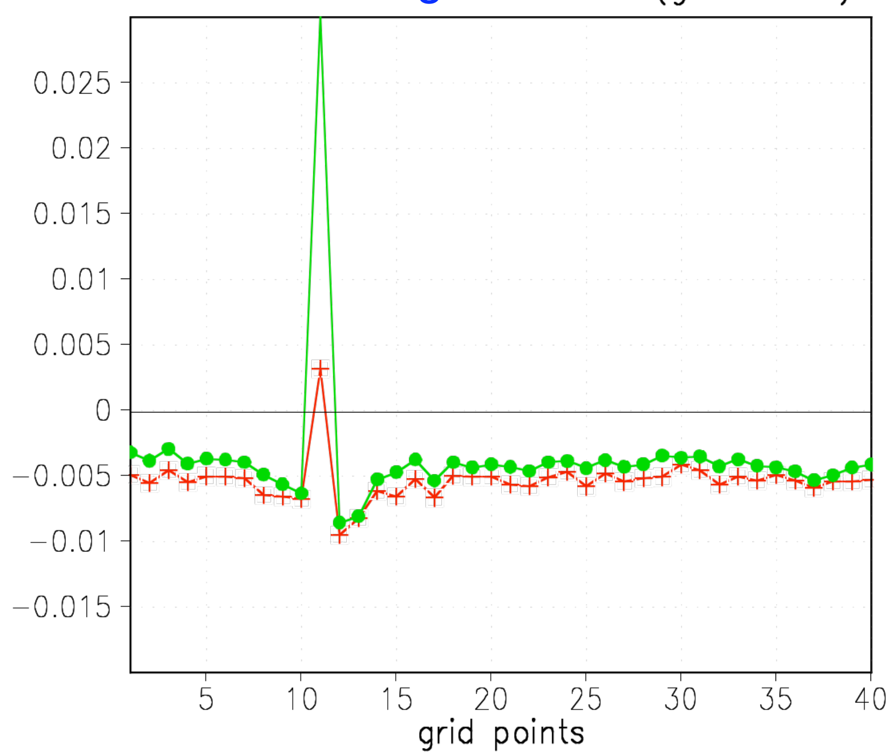
$$\frac{\partial J}{\partial \mathbf{v}_0} = \left[\tilde{\mathbf{K}}_0^T \mathbf{X}_{t|-6}^{fT} \right] \left[\mathbf{e}_{t|-6} + \mathbf{X}_{t|-6}^f \tilde{\mathbf{K}}_0 \mathbf{v}_0 \right]$$

With this formula we can predict the impact of observations on the forecasts!

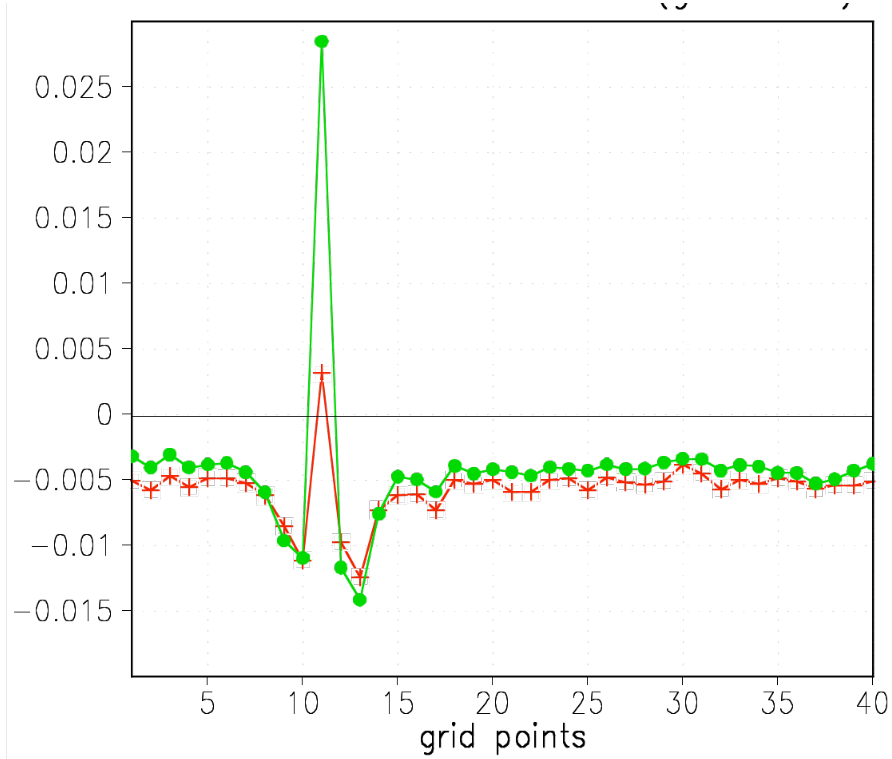
Test ability to detect the poor quality observation on the Lorenz 40 variable model

Observation impact from LB (red) and from ensemble sensitivity method (green)

Larger random error



Biased observation case



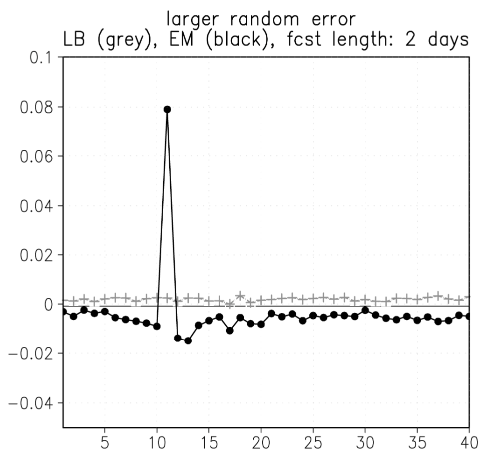
✓ Like adjoint method, ensemble sensitivity method can detect the observation poor quality (11th observation location)

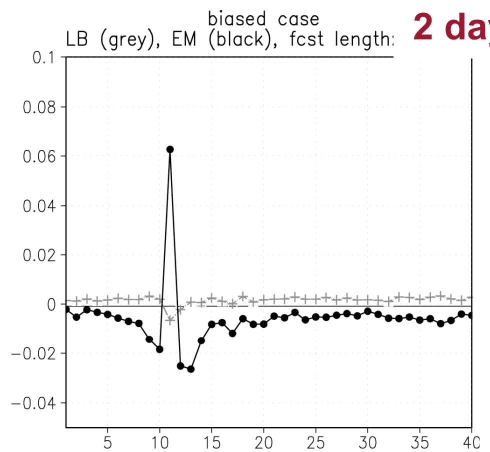
✓ The ensemble sensitivity method has a stronger signal when the observation has negative impact on the forecast.

Test ability to detect poor quality observation for different forecast lengths

Larger random error   Biased observation case

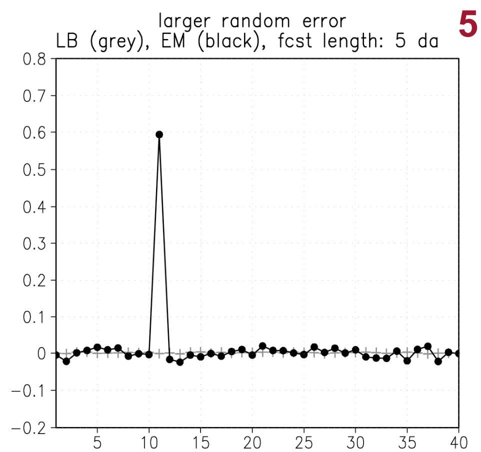
2 days

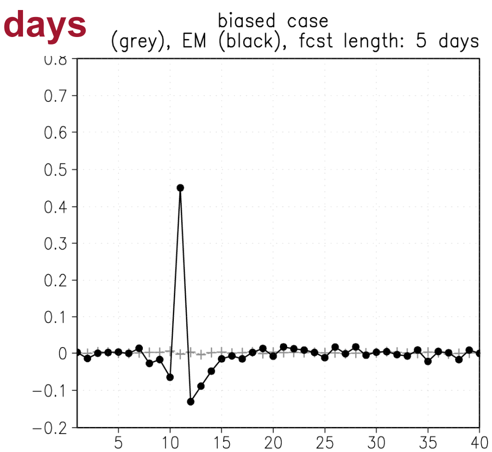




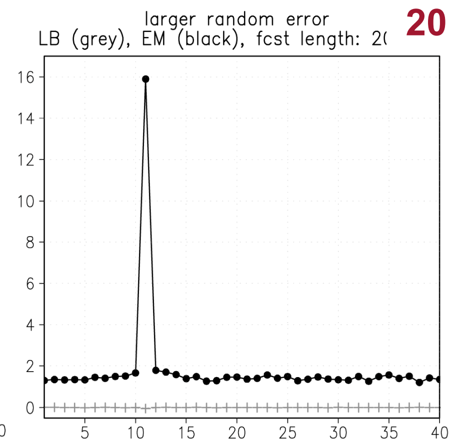
- ✓ After 2-days the adjoint has the wrong sensitivity sign!
- ✓ The ensemble sensitivity method has a strong signal even after forecast error has saturated!

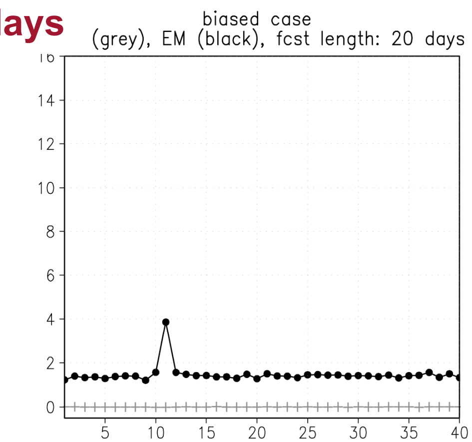
5 days





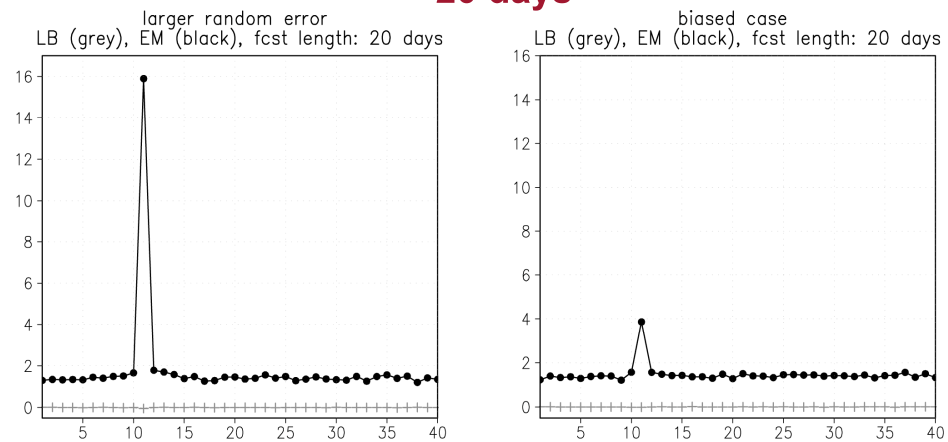
20 days

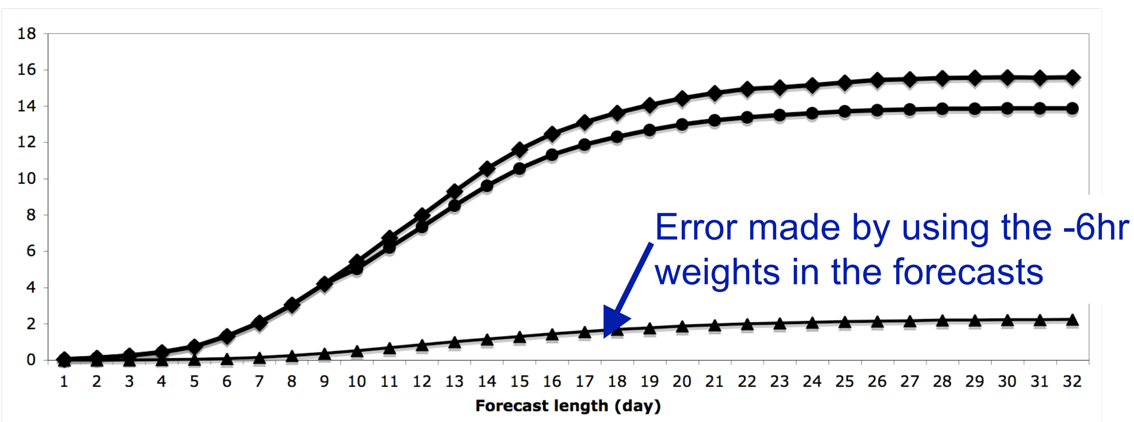




How can we possibly detect bad observations even after all skill is lost??? (Liu and Kalnay, 2009)

20 days





Mean Square Error of the -6hr weighted forecasts (diamonds), MSE of the 0hr ensemble mean (circles) and MS Difference between ensemble mean and weighted forecasts (triangles).

- ✓ After 20-days there is no forecast skill but the ensemble sensitivity still detects the wrong observation.
- ✓ The ensemble sensitivity is based on the assumption that the analysis weights can be used in the forecasts. This is accurate even after forecast error has saturated (triangles).
- ✓ As a result we can identify a bad observation even after forecast skill is lost.

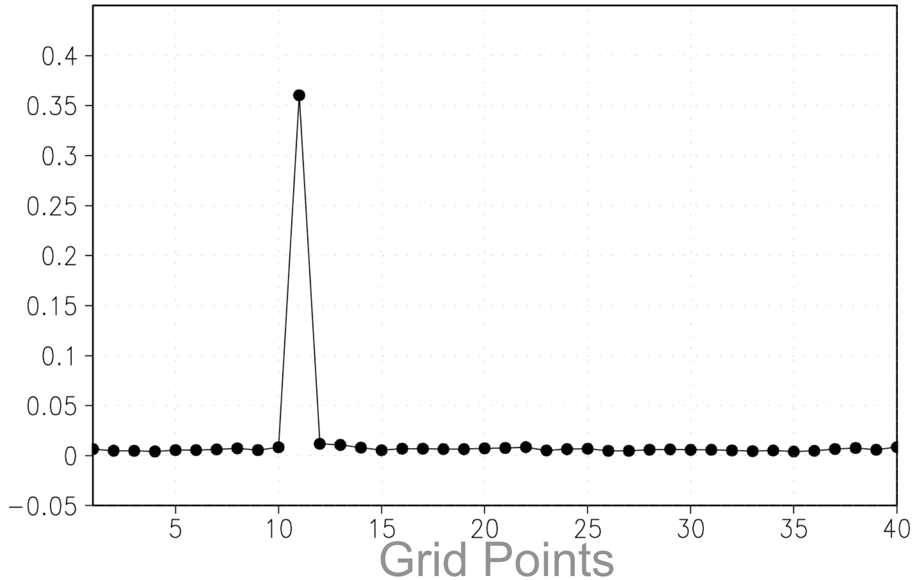
Analysis sensitivity to observations and exact Cross-Validation in an EnKF

Junjie Liu, E Kalnay, T Miyoshi and C Cardinali
QJRMS 2009

Inspired in Cardinali et al., 2004

Observation quality control using cross-validation

$$\frac{1}{T}\sum_{t=1}^{T}(y_i^o - y_i^{a(-i)})^2 = \frac{1}{T}\sum_{t=1}^{T}\frac{(y_{i,t}^o - y_{i,t}^a)^2}{(1-S_{ii,t}^o)^2}, \quad i = 1,\cdots,m$$



Experimental design: the observation error standard deviation at the 11th point is 4 times larger than the others.

- The difference between the predicted observation $y_i^{a(-i)}$ and the actual obs y_i^o is larger when the ith observation has larger error (11th point).
- Does not need much computational time.

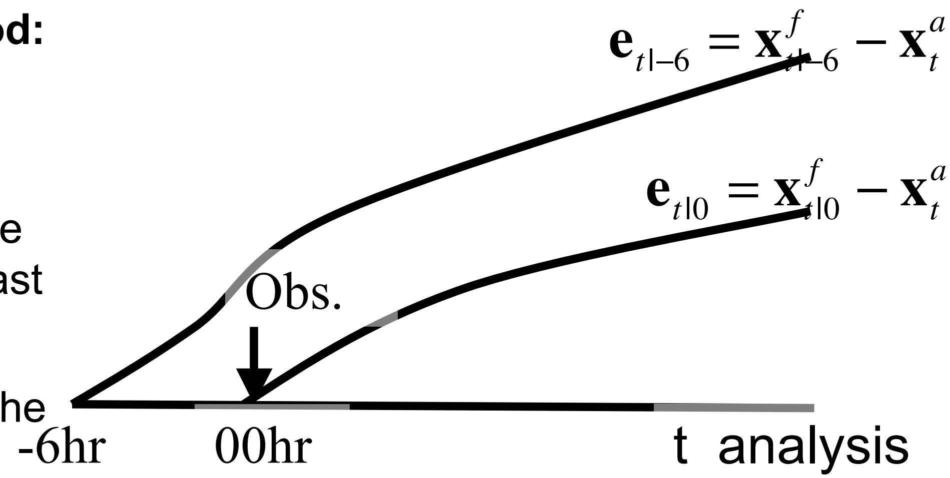
Observation impact based on self-sensitivity & the observation impact from adjoint and ensemble sensitivity method

The adjoint and ensemble sensitivity method:

$$J = [(\mathbf{x}^f_{t|0} - \mathbf{x}^a_t)^2 - (\mathbf{x}^f_{t|-6} - \mathbf{x}^a_t)^2]$$

* The cost function reflects the impact of all the observations assimilated at 00hr on the forecast error difference (model space) at time t.

* The cost function is rewritten as function of the observations assimilated at 00hr.

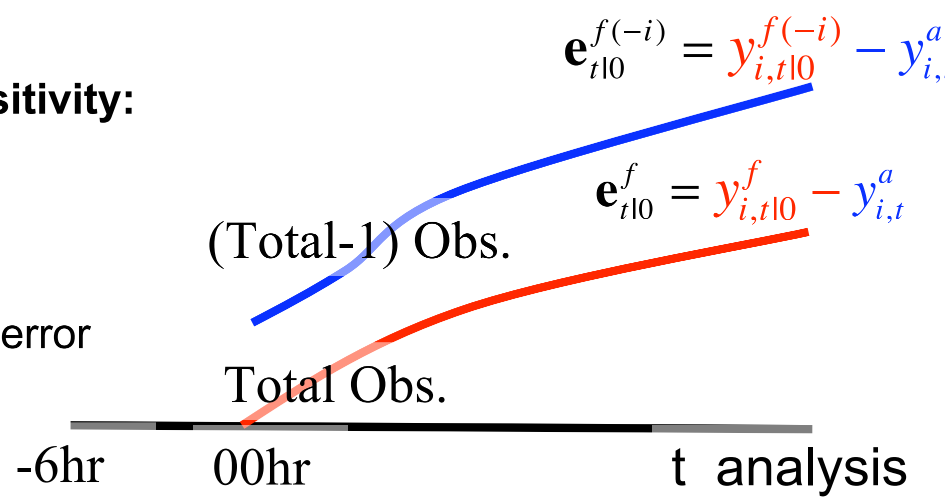


The observation impact based on self-sensitivity:

$$J_i = [(y^f_{i,t|0} - y^a_{i,t})^2 - (y^{f(-i)}_{i,t|0} - y^a_{i,t})^2]$$

* The cost function reflects the impact of the i^th^ observation assimilated at 00hr on the forecast error difference (observation space) at time t.

* There is no need to rewrite the cost function.

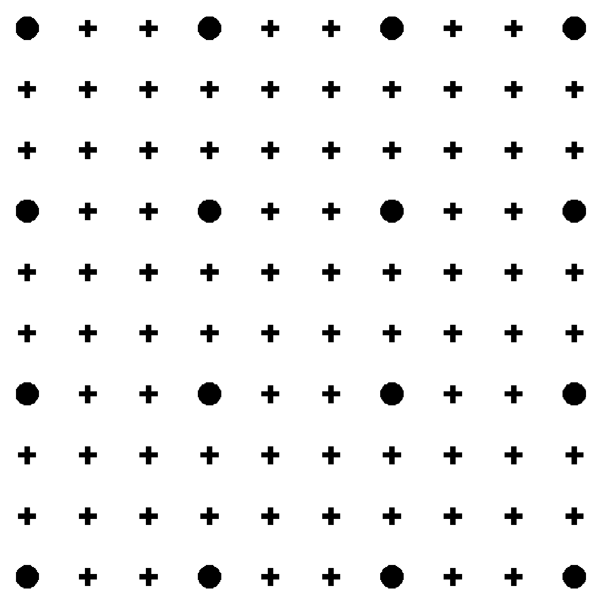


(Langland and Baker, 2004; Liu and Kalnay, 2008)

Coarse analysis with interpolated weights

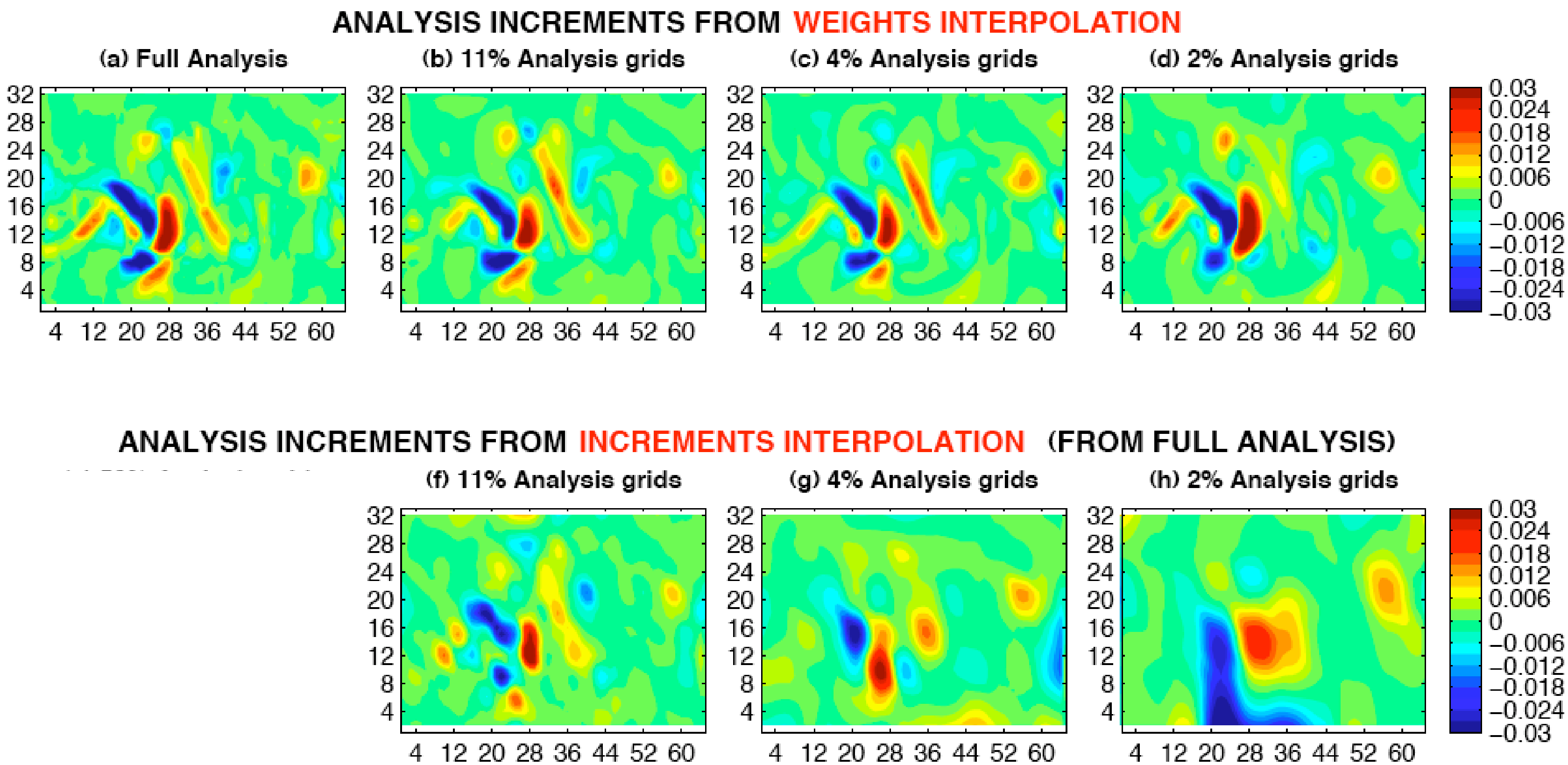
Yang et al (2008)

- In EnKF the analysis is a weighted average of the forecast ensemble
- We performed experiments with a QG model **interpolating** **weights** compared to **analysis** **increments**.
- 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

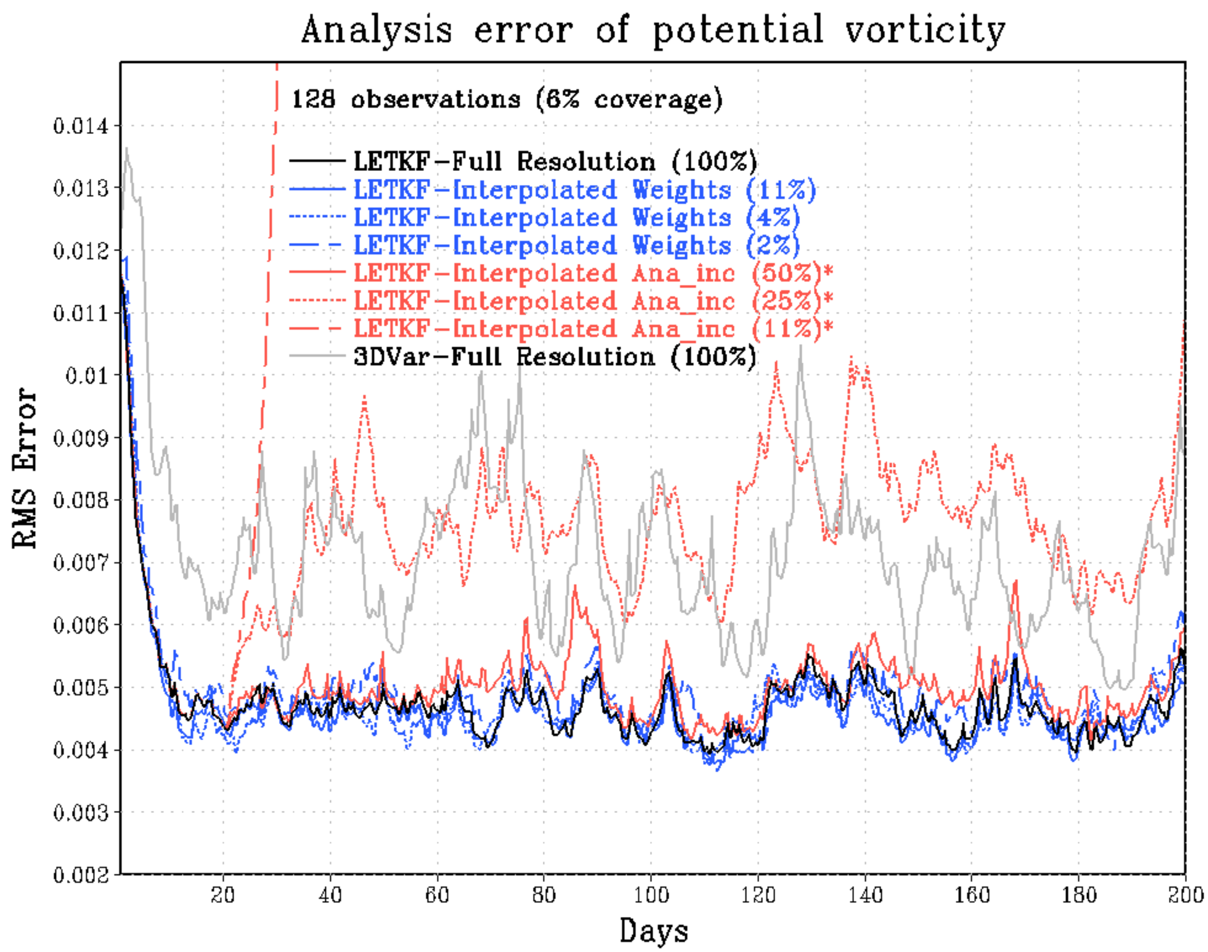


With increment interpolation, the analysis degrades quickly…

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

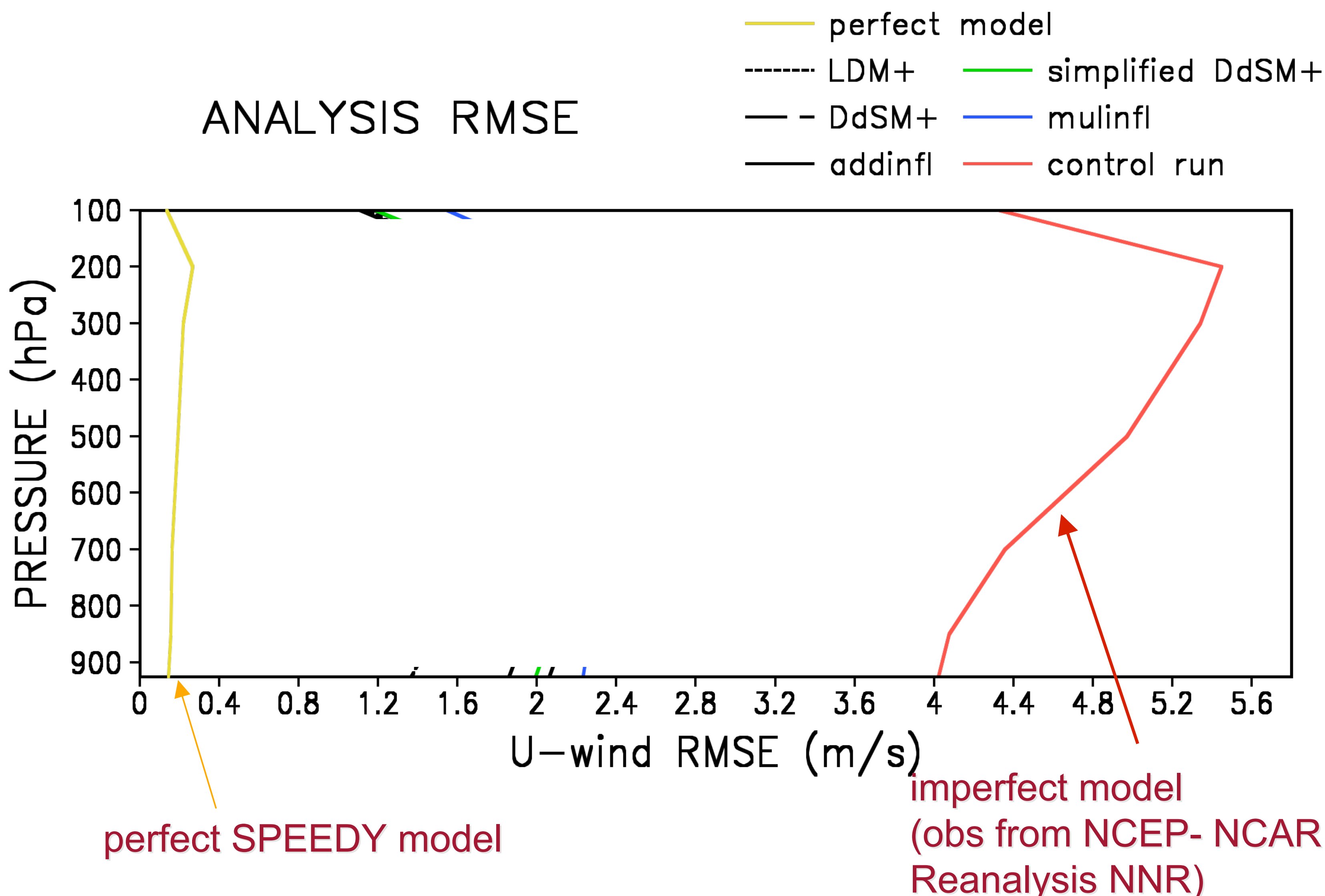
With weight interpolation, there is no degradation, the analysis is actually slightly better!

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

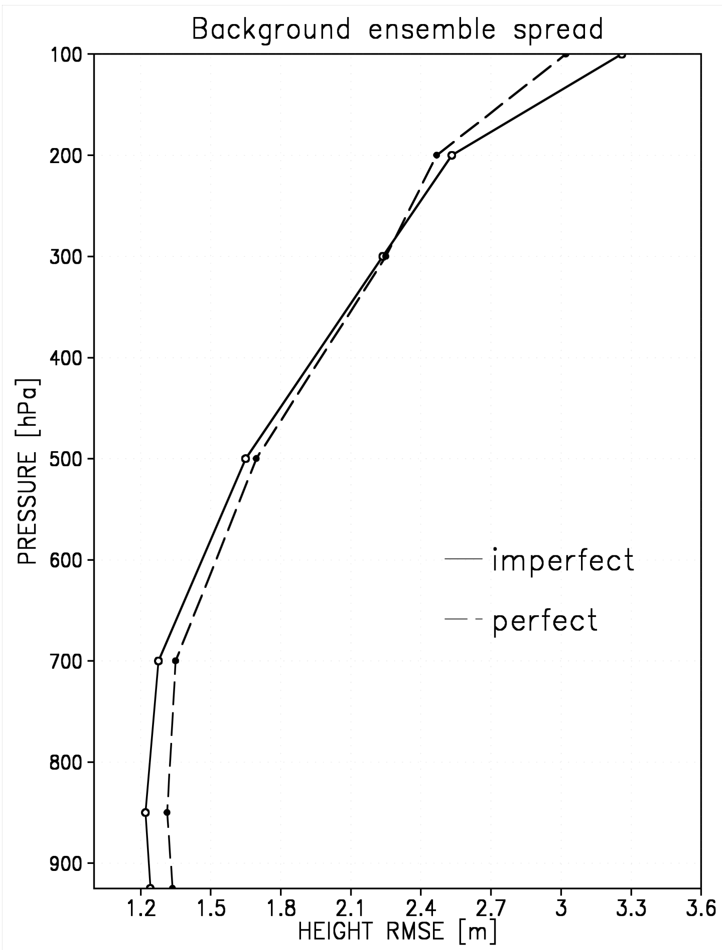
Hong Li, Chris Danforth, Takemasa Miyoshi, and Eugenia Kalnay, MWR (2009)

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

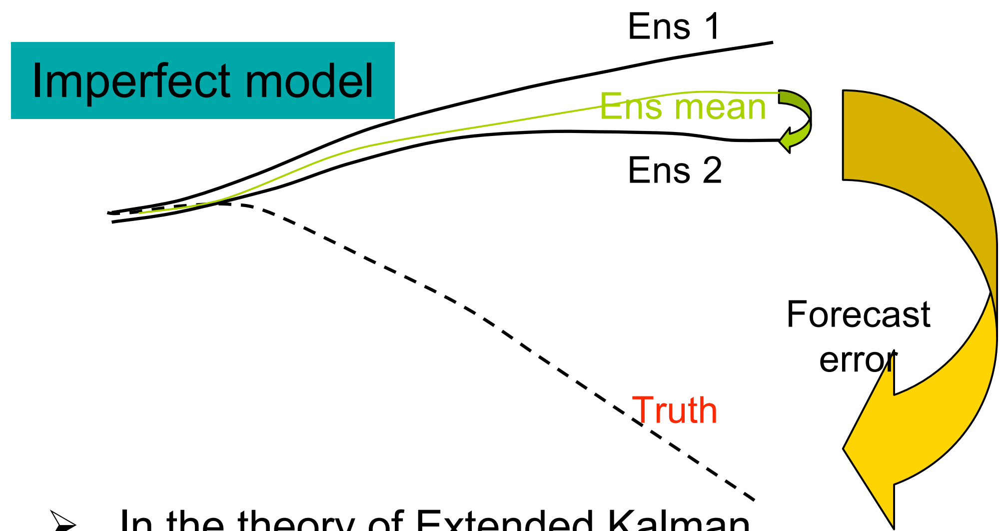
Model error: If we assume a perfect model in EnKF, we underestimate the analysis errors (Li, 2007)



— Why is EnKF vulnerable to model errors ?



The ensemble spread is 'blind' to model errors



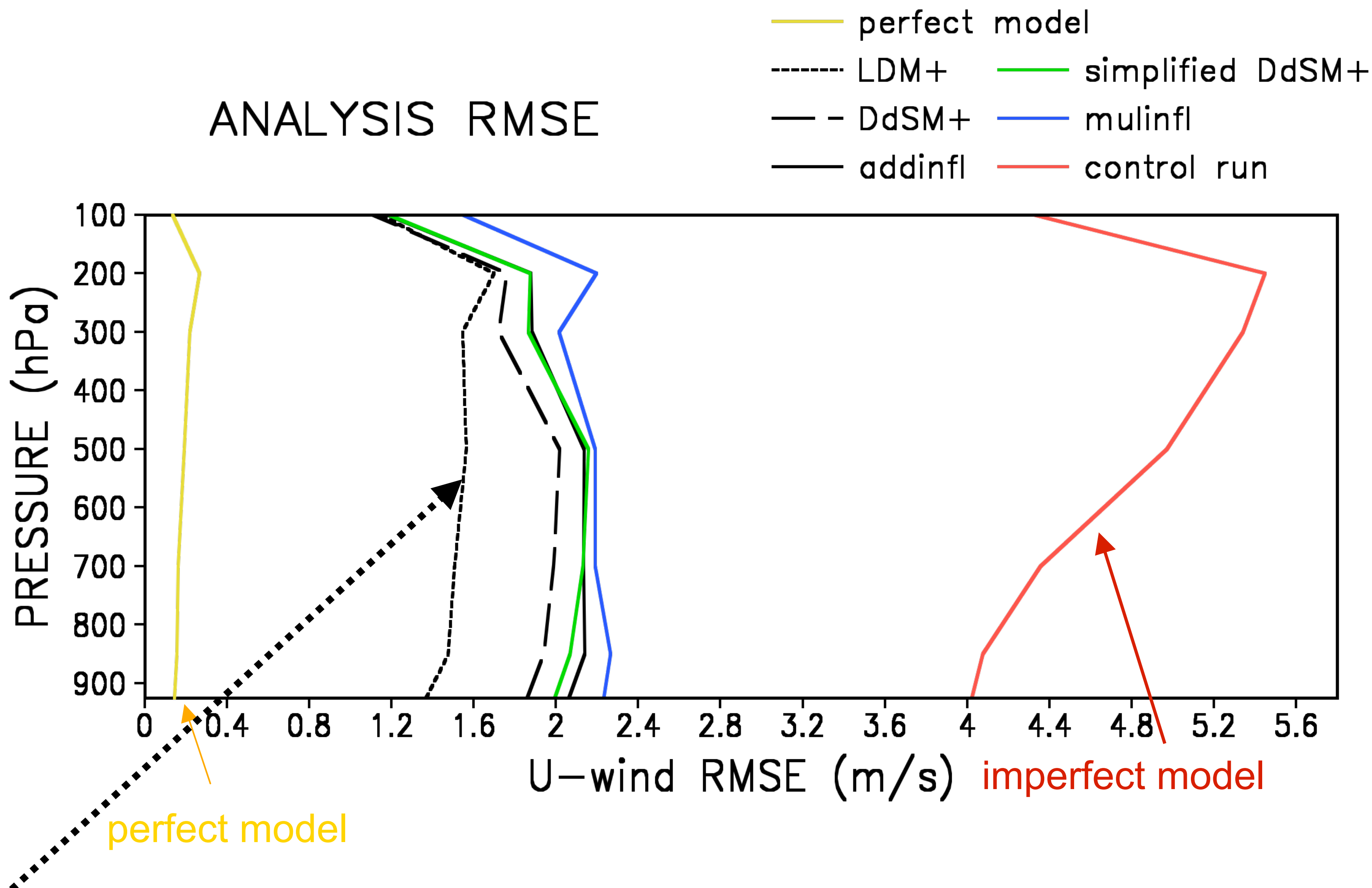
- In the theory of Extended Kalman filter, forecast error is represented by the growth of errors in IC and the model errors.

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

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

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

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
- Wang and Bishop (2003) and Miyoshi (2005) proposed a technique to estimate the covariance inflation parameter online. It works well if ob errors are accurate.
- 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

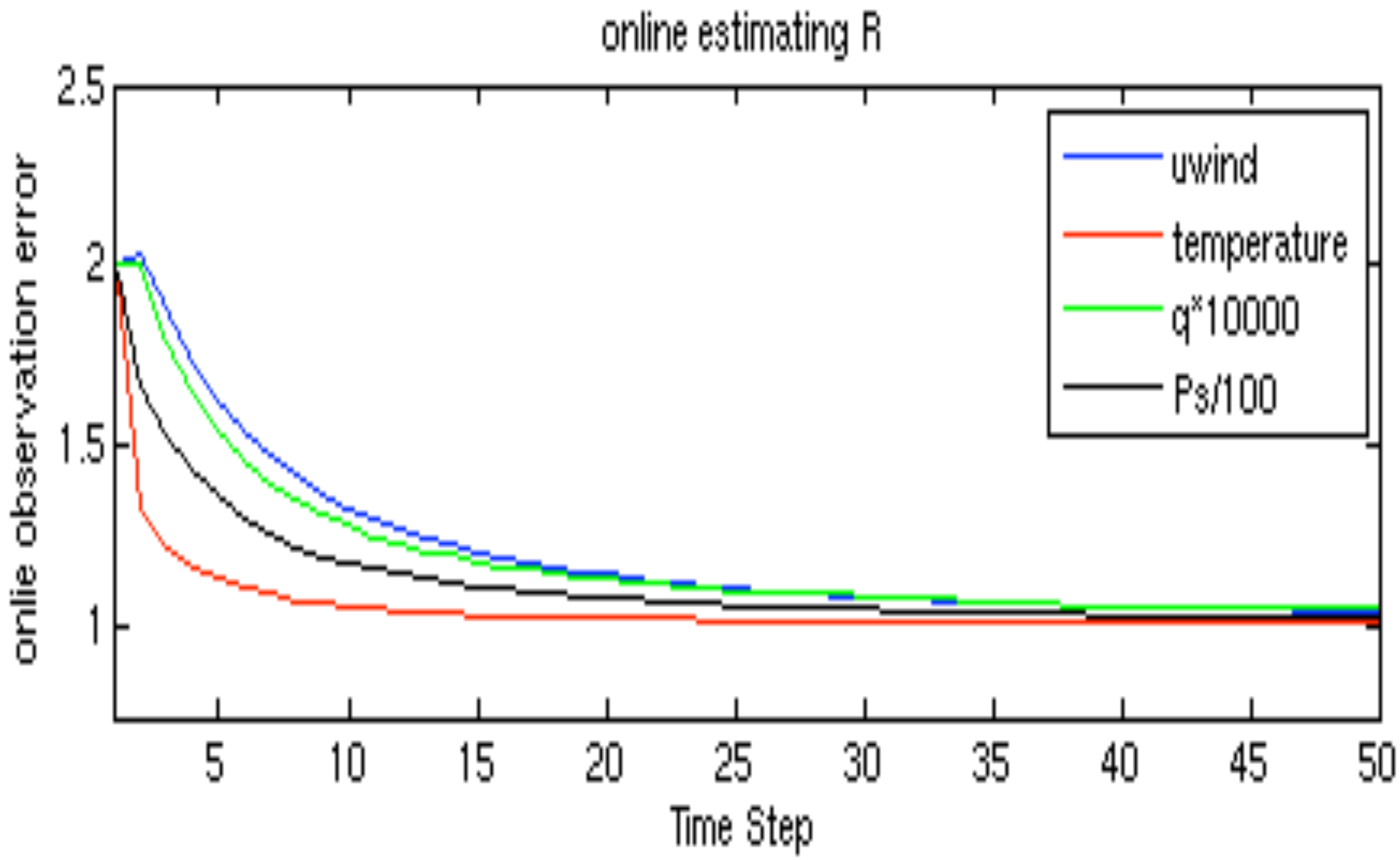
$$\Delta^o = \frac{(\mathbf{d}_{o-b}^T \mathbf{d}_{o-b}) - Tr(\mathbf{R})}{Tr(\mathbf{HP}^b\mathbf{H}^\mathrm{T})}$$ OMB[2]

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

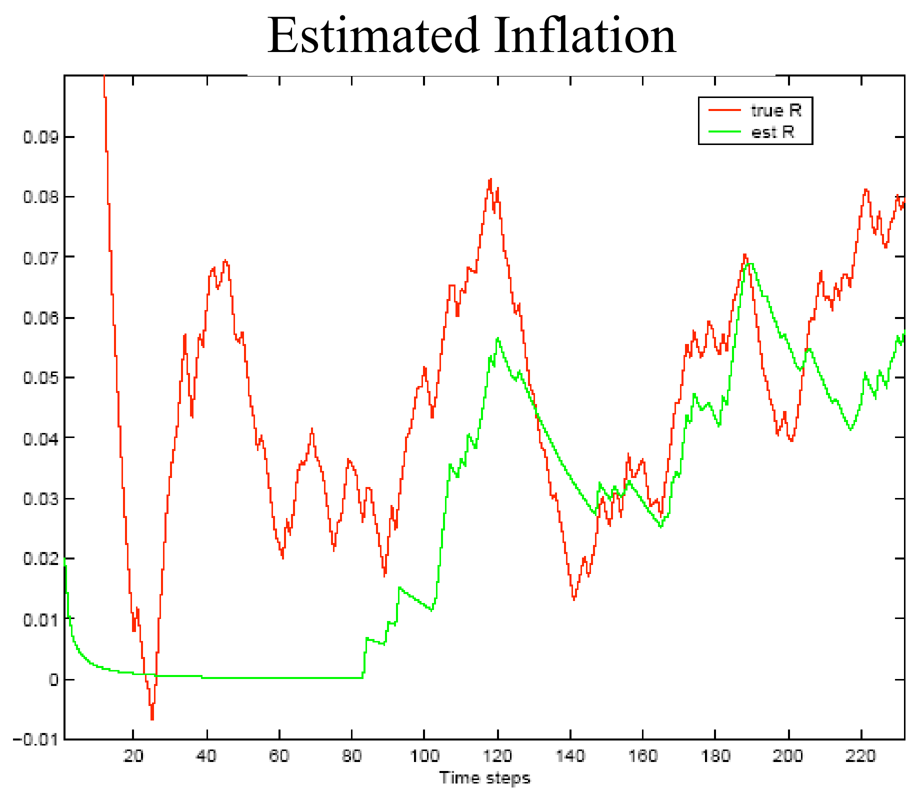
Here we use a simple KF to estimate both Δ and σ_o^2 online.

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)

Estimation of the 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

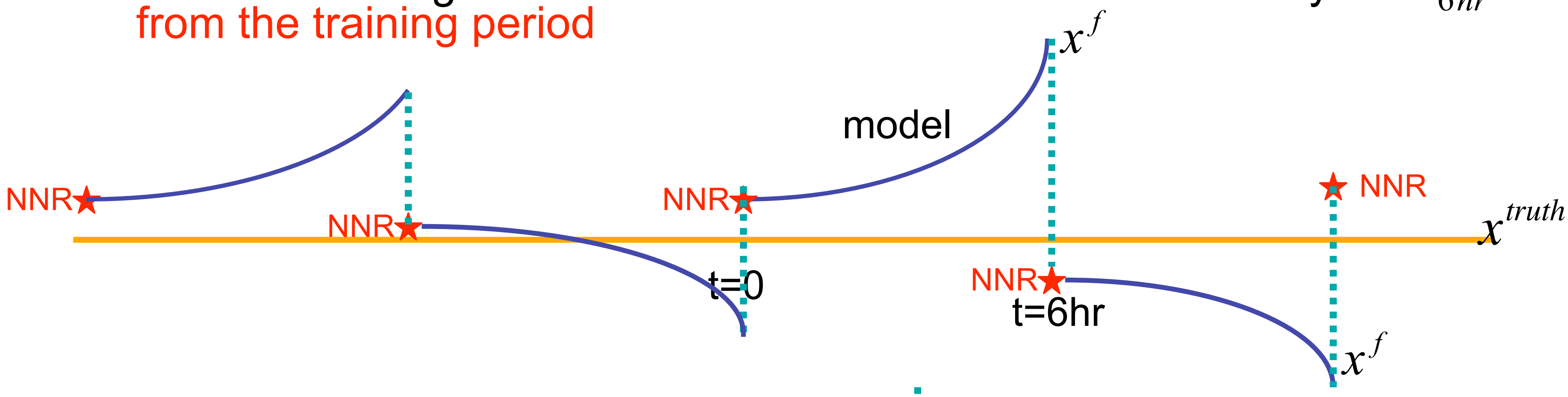
- EnKF and 4D-Var give similar results in Canada and in JMA, except for model bias. (Buehner et al, Miyoshi et al)
- EnKF is better than GSI with half resolution model, 64 members. Computationally competitive (Whitaker)
- Many ideas to further improve EnKF were inspired in 4D-Var:
  - No-cost smoothing 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 obs. errors

Extra Slides on Low Dim Method

Bias removal schemes (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.

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

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:

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

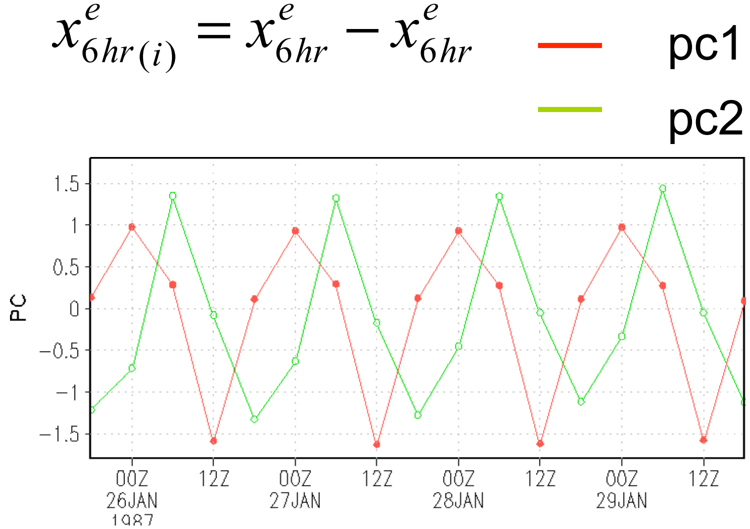
Having a large number of estimated errors allows to estimate the global model error beyond the bias

SPEEDY 6 hr model errors against NNR (diurnal cycle)

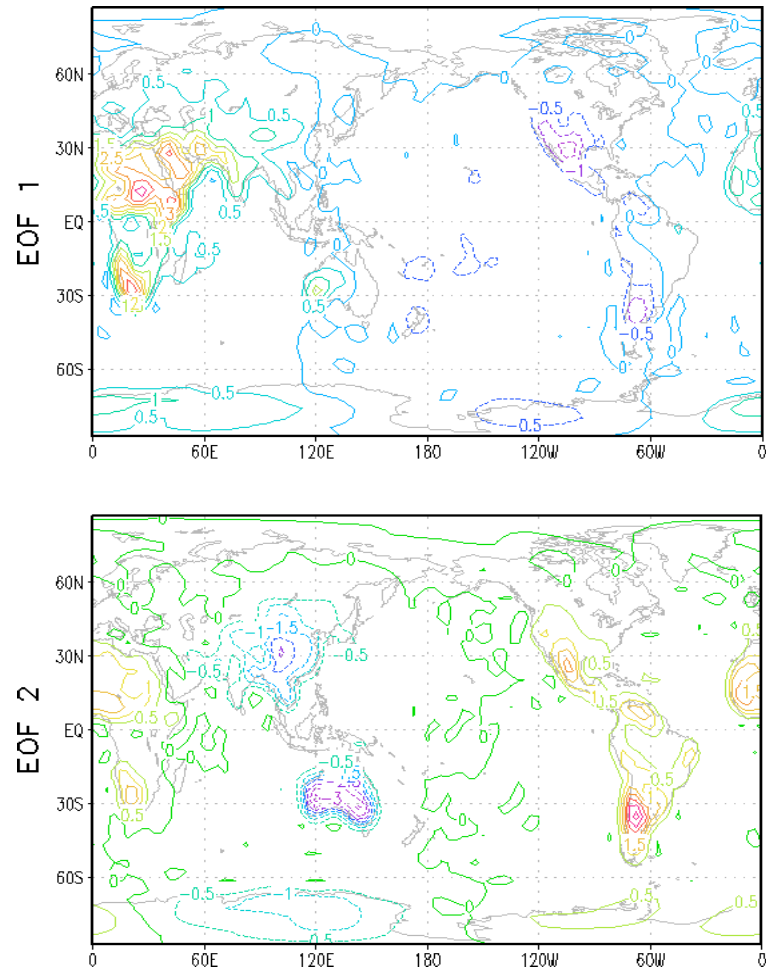
1987 Jan 1~ Feb 15

Error anomalies

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



Leading EOFs for 925 mb TEMP



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