Local Ensemble Transform Kalman Filter

Brian Hunt

11 June 2013

Review of Notation

- Forecast model: a known function M on a vector space of model states.
- Truth: an unknown sequence $\{x_n\}$ of model states to be estimated.
- Observations: a sequence $\{y_n^o\}$ of vectors in observation space.
- Forward operator: a known function H_n from model space to observation space.
- Observation error: $\varepsilon_n = y_n^o - H_n(x_n)$.
- Background (before data): superscript b.
- Analysis (after data): superscript a.

Kalman Filter Review

- The Kalman Filter tracks a mean $x_n^{b,a}$ and covariance $P_n^{b,a}$ representing a Gaussian distribution of model states.
- The analysis step assumes Gaussian observation errors and computes (exactly if H_n is linear) x_n^a and P_n^a from x_n^b, P_n^b, and the observation error covariance R_n.
- The forecast step sets $x_{n+1}^b = M(x_n^a)$.
- If M is nonlinear, there is no "right" way to propagate the covariance from P_n^a to P_{n+1}^b.

Practical Difficulties

- If x is m-dimensional, then P is an m-by-m matrix
- If m is large, the computational cost of storing and manipulating P may be prohibitive.
- Linearizing the model around the forecast trajectory (as required by the Extended Kalman Filter) can be expensive.
- Ensemble Kalman Filters make a low-rank approximation to P and avoid explicit linearization of the forecast model.

Ensemble Kalman Filters

- Introduced by Geir Evensen [1994].
- Use the sample mean and covariance of an ensemble of forecasts as the Kalman filter background.
- Form an ensemble of states that matches the Kalman Filter analysis mean and covariance and use it to initialize the next ensemble forecast.
- Assumption: the model is approximately linear within the range of the ensemble.

Ensemble Analysis

- Subscripts now index ensemble members and not time.
- Input: background ensemble $x_1^b, \ldots, x_k^b$ with sample mean $\bar{x}^b$ and covariance P^b; observation information y^o, H, R.
- Output: analysis ensemble $x_1^a, \ldots, x_k^a$ with sample mean $\bar{x}^a$ and covariance P^a determined as in the Kalman Filter.

Kalman Filter Equations

- Recall the Kalman filter analysis equations:

$$K = P^b H^T [H P^b H^T + R]^{-1}$$
$$\bar{x}^a = \bar{x}^b + K(y^o - H\bar{x}^b)$$
$$P^a = (I - KH)P^b$$

- Note: $\bar{x}^a - \bar{x}^b$ is called the analysis increment.
- What should we do if H is nonlinear?
- How do we determine the analysis ensemble?

Nonlinear Forward Operator H

- Write $P^b = X^b(X^b)^T$ where X^b is the matrix whose ith column is $(x_i^b - \bar{x}^b)/\sqrt{k-1}$.
- Let $Y^b = HX^b$ if H is linear.; then:

$$
\begin{aligned}
K &= X^b(Y^b)^T[Y^b(Y^b)^T + R]^{-1} \\
\bar{x}^a &= \bar{x}^b + K(y^o - H\bar{x}^b) \\
P^a &= (X^b - KY^b)(X^b)^T
\end{aligned}
$$

- For nonlinear H, let $y_i^b = H(x_i^b)$, let $\bar{y}^b$ be the mean of the y_i^b, and let Y^b be a matrix whose ith column is $(y_i^b - \bar{y}^b)/\sqrt{k-1}$.

Perturbed Observations

- There are many possible analysis ensembles whose mean and covariance are $\bar{x}^a$ and P^a.
- One can apply the Kalman Filter update to each ensemble member, perturbing the observed values differently for each:

$$x_i^a = x_i^b + K(y^o + \varepsilon_i - y_i^b)$$

  where $\varepsilon_i \sim N(0, R)$ [Burgers et al. 1998, Houtekamer & Mitchell 1998].
- Then if H is linear, one can show that the expected mean and covariance of the analysis ensemble are $\bar{x}^a$ and P^a.

Deterministic Approaches

- Square root filters track a matrix “square root” of the model state covariance matrix; in our case X^b for which $P^b = X^b(X^b)^T$.
- We seek a matrix X^a of (scaled) analysis ensemble perturbations such that $P^a = X^a(X^a)^T$.
- Add $\bar{x}^a$ to the (scaled) columns of X^a to get the analysis ensemble.
- Various approaches to determining X^a from X^b: EAKF [Anderson 2001], ETKF [Bishop et al. 2001, Wang et al. 2004], EnSRF [Whitaker & Hamill 2002]; see also [Tippett et al. 2003].

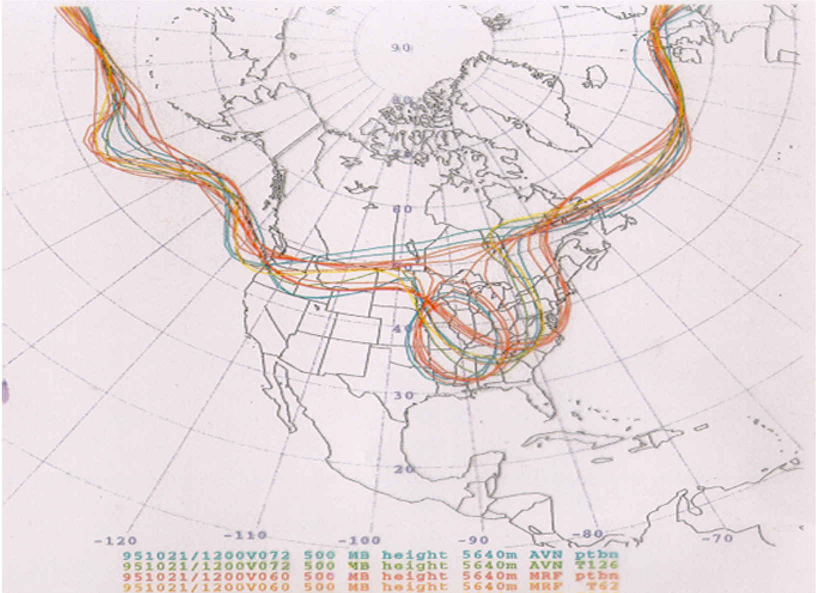
Reduced Rank: Pros and Cons

- The background covariance allows analysis increments only in the space spanned by the ensemble perturbations.
- Indeed, in all the approaches on the last two slides, each analysis ensemble member can be written $x_i^a = \bar{x}^b + X^b w_i$.
- If the ensemble is too small, the filter may fail.
- If an ensemble of reasonable size is successful, analysis computations can be done in the ensemble space spanned by the background ensemble perturbations.

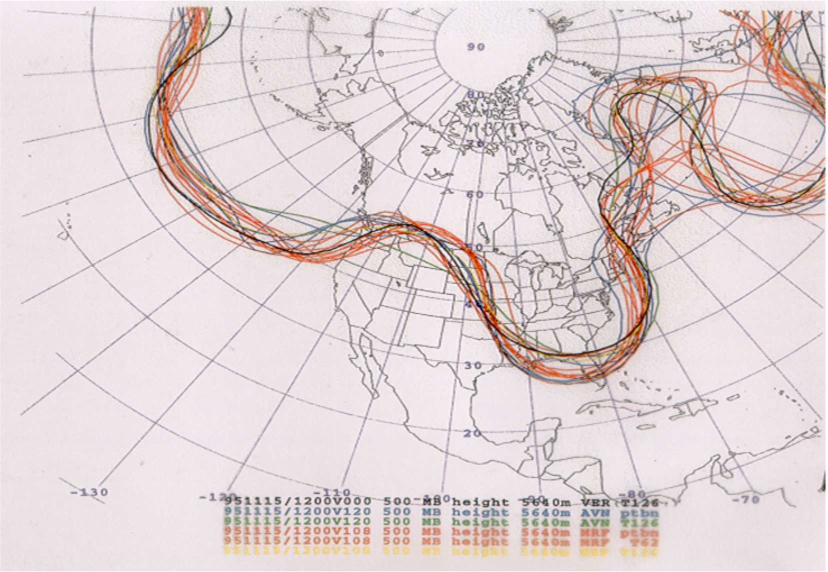
Underlying Assumptions

- A linear combination of background ensemble states is a plausible state.
- The truth lies reasonably close to the ensemble space.
- More precisely, most of the uncertainty in the background state lies in the ensemble space.
- In particular, forecast uncertainty lies in a relatively low-dimensional space.

A 2.5 Day Ensemble Forecast



A 4.5 Day Ensemble Forecast



Spatial Localization

- For a spatially extended system, it may not be feasible to forecast an ensemble large enough to span the possible global states.
- If long-distance correlations are weak, the ensemble covariance will represent spurious long-distance correlations.
- Localization reduces the effect of spurious correlations and allows analysis increments from a higher dimensional space.

Localization Approaches

- A standard approach is to localize the background covariance P^b by multiplying each element by a factor between 1 (for variables at the same grid point) and 0 (for variables farther apart than a chosen localization distance.
- Our approach [LEKF, Ott et al. 2004; LETKF, Hunt et al. 2007] is do a separate analysis for each grid point using only observations from within a local region that we choose.
- These analyses can be done in parallel.

Inflation does "Time Localization"

- Many filters use multiplicative covariance inflation, which at each step multiplies the forecast (background) covariance by ρ for an ad hoc parameter $\rho > 1$.
- This is equivalent to multiplying the term corresponding to observations from t analysis cycles ago in the least squares cost function by a weight ρ^{-t}.
- Covariance inflation compensates for effects of model error, model nonlinearity, etc.

LETKF Formalism

- LETKF stands for Local Ensemble Transform Kalman Filter; equivalent to LEKF but formulated more like ETKF.
- Recall $P^b = X^b(X^b)^T$ where X^b is a matrix whose ith column is $(x_i^b - \bar{x}^b)/\sqrt{k-1}$.
- Consider model states $x = \bar{x}^b + X^b w$ where w is a k-dimensional vector.
- If w has mean 0 and covariance I, then x has mean $\bar{x}^b$ and covariance P^b.

Analysis in Ensemble Space

- Let $y_i^b = H(x_i^b)$, let $\bar{y}^b$ be their mean, and form the matrix Y^b of perturbations like X^b.
- Make the linear approximation:

$$H(\bar{x}^b + X^b w) \approx \bar{y}^b + Y^b w$$

- Minimize the cost function:

$$\begin{aligned} J(w) =& \rho^{-1} w^T w \\ & + (\bar{y}^b + Y^b w - y^o)^T R^{-1} (\bar{y}^b + Y^b w - y^o) \end{aligned}$$

- Compare to:

$$\begin{aligned} J(x) =& (x - \bar{x}^b)^T (\rho P^b)^{-1} (x - \bar{x}^b) \\ & + (H(x) - y^o)^T R^{-1} (H(x) - y^o) \end{aligned}$$

Analysis in Ensemble Space

- The analysis mean $\bar{w}^a$ and covariance A are:

$$A = [I + (Y^b)^T R^{-1} Y^b]^{-1}$$
$$\bar{w}^a = A(Y^b) R^{-1} (y^o - \bar{y}^b)$$

- Then $\bar{x}^a = \bar{x}^b + X^b \bar{w}^a$ and $P^a = X^b A (X^b)^T$.
- Notice that the analysis equations in w coordinates depend only on the background ensemble $\{y_i^b\}$ in observation space and the observation data y^o and R.
- The matrix that is inverted to find A is k-by-k and has no small eigenvalues.

Asynchronous Observations

- With this formulation, the filter easily becomes “4D” (properly takes into account temporal information) when observations are asynchronous.
- When mapping the background ensemble members x_i^b into observation space, use the background state at the appropriate time for each observation.
- Assumption: a linear combination of ensemble trajectories is an approximate model trajectory.

Choice of Analysis Ensemble

- To form the analysis ensemble vectors w_i^a, add to $\bar{w}^a$ the columns of the symmetric square root $W^a = [(k-1)A]^{1/2}$.
- The analysis ensemble $x_i^a = \bar{x}^b + X^b w_i^a$ then has the correct mean and covariance.
- Other choices are possible, but this choice minimizes the change (in w, or in x with P^b-norm) between the background and analysis ensembles.
- Recall that we determine a different w_i^a for each grid point, using only observations from a local region near that grid point.

Weight Interpolation

- If the localization distance is large relative to the grid spacing, overlapping local regions make nearby analyses consistent.
- For a high-resolution model, the overlap causes a lot of computational redundancy.
- Solution: Compute the weights w_i^a on a coarse grid of analysis points and interpolate them to the other grid points.
- Can improve balance [Yang et al. 2009].

Tapering Observation Influence

- At a given grid point, our local analysis (as described so far) uses nearby observations at "full" influence, while distant observations have zero influence.
- The influence can be tapered more smoothly from "1" to 0 by multiplying each diagonal entry σ_j^{-2} in R^{-1} by a factor $0 \leq \alpha_j \leq 1$ that depends on the distance from the corresponding observation location to the analysis grid point.
- If R is not diagonal, replace R^{-1} by $DR^{-1}D$, where D is diagonal with entries $\sqrt{\alpha_j}$.

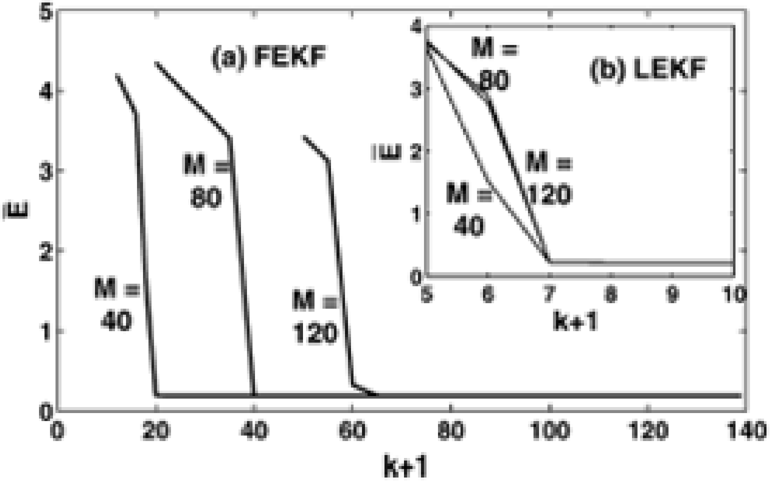
Toy Model Results

- Perfect model test with Lorenz-96 system of coupled ODEs on a circular lattice:

$$dx_m/dt = (x_{m+1} - x_{m-2})x_{m-1}x_m + 8.$$

- We used a model trajectory as the truth and add noise to simulate observations.
- We compared our LEKF, using only observations from within 6 grid points, with a global ensemble Kalman filter (FEKF) [Ott et al. 2004].

Global vs. Local Filter



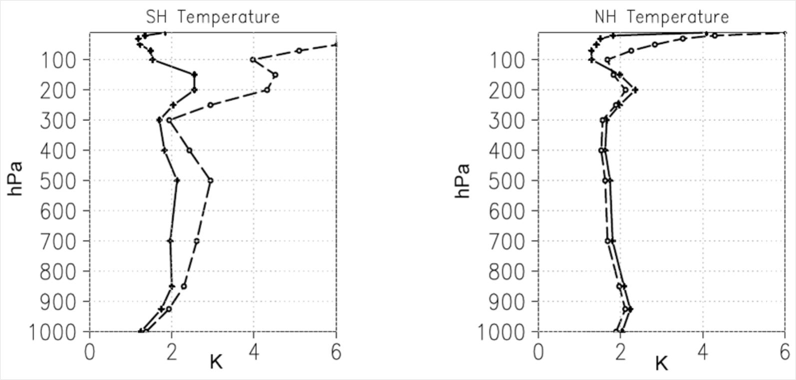
M = model size, k+1 = ensemble size, E = error

Comparison to NCEP (2004) 3D-Var

- We ran the U.S. National Centers for Environmental Prediction global forecast model (GSM) at reduced T62 resolution (about $500,000$ grid points).
- We compared our LETKF analyses with analyses from NCEP's 3D-Var system (SSI) using actual Winter 2004 non-satellite observations (about $300,000$ per 6 hours).
- We used a 60-member ensemble, 800 km radius local region, spatially varying inflation $\rho \sim 1.25$ [Szunyogh et al. 2008].

48-hour Forecast Error

(compared to radiosonde observations)



+ = our LETKF   o = NWS 3D-Var

Computational Speed

- At NCEP, less than 10 minutes of every 6 hour cycle is used for data assimilation.
- Our implementation took about 15 minutes on a 40-processor Linux cluster.
- The computation time is approximately:
  - linear in the number of observations;
  - linear in the number of model grid points;
  - quadratic in the number of ensemble members.

Extensions

- Parameter/Bias Estimation: treat parameters of M or H as state variables with time derivative zero (Baek et al. 2006, 2009; Cornick et al. 2009; Fertig et al. 2009).
- Nongaussian Filter: minimize nonquadratic cost function numerically in ensemble space (Harlim & Hunt 2007), as in MLEF (Zupanski 2006).

Conclusions

- LETKF is a relatively simple, efficient, and flexible framework for reduced-rank data assimilation that works in practice.
- The method is largely model-independent.
- It scales well to large systems.
- Full citations for references to our group's work are on the publications page at: `http://www.weatherchaos.umd.edu`
- Motto: Forecast globally, assimilate locally.