Building State-of-the-Art Forecast Systems with the Ensemble Kalman Filter

Jeff Anderson representing the NCAR Data Assimilation Research Testbed
Want to predict where the ball will land.
Want to predict where the ball will land.
Prediction Model

For the ball this is simple:

\[ x = x_{\text{initial}} + u_{\text{initial}}t \]
\[ y = y_{\text{initial}} + v_{\text{initial}}t - \frac{1}{2} gt^2 \]
Unsure about release point, velocity, angle…
Sample this with an ‘ensemble’ of blue balls.
Unsure about release point, velocity, angle… Sample this with an ‘ensemble’ of blue balls.
Need observations (measurements) of the red ball.

All observations have errors.

Observe position of ball every half second after throw.
Observations of the Red Ball

Time 0.5 secs
Observations of the Red Ball

Time 1.5 secs
Building a Forecast System

Prediction Model

Observing System

Data Assimilation

Forecasts

Observations
Building a Forecast System

- Prediction Model
- Observing System
- Data Assimilation
- Forecasts
- Observations
- Analysis
Make large ensemble of forecasts. Closer to observation => more likely. Fifty likely forecasts are shown (darker blue => more likely).
Fifty balls at time 0.5 are an ensemble analysis. Show uncertainty of best estimate of red ball’s location.
Analysis ensemble are initial conditions for 50 forecasts. Green is weighted mean of ensemble forecast at time 2.0. This is best single forecast given observations at time 0.5.
Start with forecast at time 1.0 that used observations at time 0.5.
Add information from observation at time 1.0.
New ensemble analysis is initial conditions for 50 forecasts. Green is best single forecast of red ball at time 2.0 given observations at time 0.5 and 1.0.
Next ensemble analysis is initial conditions for 50 forecasts.
Next ensemble analysis is initial conditions for 50 forecasts. Green is best single forecast of red ball at time 2.0 given observations at time 0.5, 1.0 and 1.5.
This thrown ball example is in a 2-dimensional space. Really a 4-dimensional ‘phase’ space including velocity.

Atmosphere, ocean, land, coupled models are BIG.

But they’re still just a ‘ball’ moving in a HUGE phase space.

As many as 100 million dimensions at present.
DART provides data assimilation ‘glue’ to build state-of-the-art ensemble forecast systems for even the largest models.
Provide State-of-the-Art Data Assimilation capability to:

- Prediction research scientists,
- Model developers,
- Observation system developers,

Who may not have any assimilation expertise.
DART Design Constraints

- Models small to huge.
- Few or many observations.
- Tiny to huge computational resources.
- Entry cost must be low.
- Competitive with existing methods for weather prediction:
  Scientific quality of results,
  Total computational effort.
A General Description of the Forecast Problem

A system governed by (stochastic) Difference Equation:

\[ dx_t = f(x_t, t) + G(x_t, t)d\beta_t, \quad t \geq 0 \]  \hspace{1cm} (1)

Observations at discrete times:

\[ y_k = h(x_k, t_k) + v_k; \quad k = 1, 2, \ldots; \quad t_{k+1} > t_k \geq t_0 \]  \hspace{1cm} (2)

Observational error white in time and Gaussian (nice, not essential).

\[ v_k \rightarrow N(0, R_k) \]  \hspace{1cm} (3)

Complete history of observations is:

\[ Y_\tau = \{ y_l; t_l \leq \tau \} \]  \hspace{1cm} (4)

**Goal:** Find probability distribution for state:

\[ p(x, t \mid Y_t) \quad \text{Analysis} \quad p(x, t^+ \mid Y_t) \quad \text{Forecast} \]  \hspace{1cm} (5)
A General Description of the Forecast Problem

State between observation times obtained from Difference Equation. Need to update state given new observations:

\[ p(x,t_k | Y_{t_k}) = p(x,t_k | y_k, Y_{t_k-1}) \quad (6) \]

Apply Bayes’ rule:

\[ p(x,t_k | Y_{t_k}) = \frac{p(y_k | x_k, Y_{t_k-1}) p(x,t_k | Y_{t_k-1})}{p(y_k | Y_{t_k-1})} \quad (7) \]

Noise is white in time (3), so:

\[ p(y_k | x_k, Y_{t_k-1}) = p(y_k | x_k) \quad (8) \]

Integrate numerator to get normalizing denominator:

\[ p(y_k | Y_{t_k-1}) = \int p(y_k | x) p(x,t_k | Y_{t_k-1}) dx \quad (9) \]
Probability after new observation:

\[ p(x, t_k | Y_{t_k}) = \frac{p(y_k | x) p(x, t_k | Y_{t_{k-1}})}{\int p(y_k | \xi) p(\xi, t_k | Y_{t_{k-1}}) d\xi} \] (10)

Likelihood

Prior (forecast)

Posterior (analysis).

Denominator just normalization.
Independent evolving estimates,
Associate probability with each estimate given observations,
Eliminate unlikely estimates,
Duplicate likely estimates,
Can represent arbitrary probability distribution,
Scales very poorly for large problems.
Four-Dimensional Variational Method:

Minimize a cost function motivated by Eq. 10,
Find optimal fit of evolving model to observations,
Use variational calculus (adjoint) to compute gradient,
State-of-the-art for weather prediction until recently.

Creating model adjoints requires huge effort.
Inconsistent with requirement for easy entry.
Only provides estimate of mean state.
Methods for Solving the Forecast Problem: Kalman Filter

Assumes:

- Linear model
- Gaussian noise
- Gaussian state
- Linear forward operator,

\[
dx_t = f(x_t, t) + G(x_t, t) d\beta_t, \quad t \geq 0
\]

\[
y_k = h(x_k, t_k) + v_k; \quad k = 1, 2, \ldots; \quad t_{k+1} > t_k \geq t_0
\]

Gaussian observation error
Product of Two Gaussians

Product of d-dimensional normals with means $\mu_1$ and $\mu_2$ and covariance matrices $\Sigma_1$ and $\Sigma_2$ is normal.

$$N(\mu_1, \Sigma_1)N(\mu_2, \Sigma_2) = cN(\mu, \Sigma)$$
Product of Two Gaussians

Product of d-dimensional normals with means $\mu_1$ and $\mu_2$ and covariance matrices $\Sigma_1$ and $\Sigma_2$ is normal.

$$N(\mu_1, \Sigma_1)N(\mu_2, \Sigma_2) = cN(\mu, \Sigma)$$

Covariance: $\Sigma = (\Sigma_1^{-1} + \Sigma_2^{-1})^{-1}$

Mean: $u = (\Sigma_1^{-1} + \Sigma_2^{-1})^{-1}(\Sigma_1^{-1}\mu_1 + \Sigma_2^{-1}\mu_2)$
Product of Two Gaussians

Product of $d$-dimensional normals with means $\mu_1$ and $\mu_2$ and covariance matrices $\Sigma_1$ and $\Sigma_2$ is normal.

$$N(\mu_1, \Sigma_1)N(\mu_2, \Sigma_2) = cN(\mu, \Sigma)$$

Covariance: 
$$\Sigma = (\Sigma_1^{-1} + \Sigma_2^{-1})^{-1}$$

Mean: 
$$u = (\Sigma_1^{-1} + \Sigma_2^{-1})^{-1} (\Sigma_1^{-1} \mu_1 + \Sigma_2^{-1} \mu_2)$$

Weight: 
$$c = \frac{1}{(2 \pi)^{d/2} |\Sigma_1 + \Sigma_2|^{1/2}} \exp \left\{ -\frac{1}{2} \left[ (\mu_2 - \mu_1)^T (\Sigma_1 + \Sigma_2)^{-1} (\mu_2 - \mu_1) \right] \right\}$$

We’ll ignore the weight since we immediately normalize products to be PDFs.
The Kalman Filter

\[
p(x_t,t_k \mid Y_{t_k}) = \frac{p(y_k \mid x) p(x,t_k \mid Y_{t_{k-1}})}{\int p(y_k \mid \xi) p(\xi,t_k \mid Y_{t_{k-1}}) d\xi}
\]  

(10)

Numerator is just product of two gaussians.

Denominator just normalizes posterior to be a PDF.
The Kalman Filter

\[ p(x, t_k | Y_{t_k}) = \frac{p(y_k | x) p(x, t_k | Y_{t_k-1})}{\int p(y_k | \xi) p(\xi, t_k | Y_{t_k-1}) d\xi} \] (10)
The Kalman Filter

\[ p(x, t_k | Y_{t_k}) = \frac{p(y_k | x) p(x, t_k | Y_{t_{k-1}})}{\int p(y_k | \xi) p(\xi, t_k | Y_{t_{k-1}}) d\xi} \]  

(10)
The Kalman Filter

\[ p(x,t_k|Y_{t_k}) = \frac{p(y_k|x)p(x,t_k|Y_{t_k-1})}{\int p(y_k|\xi)p(\xi,t_k|Y_{t_k-1})d\xi} \]
Product of d-dimensional normals with means $\mu_1$ and $\mu_2$ and covariance matrices $\Sigma_1$ and $\Sigma_2$ is normal.

\[ N(\mu_1, \Sigma_1)N(\mu_2, \Sigma_2) = cN(\mu, \Sigma) \]

Covariance:

\[ \Sigma = (\Sigma_1^{-1} + \Sigma_2^{-1})^{-1} \]

Mean:

\[ u = (\Sigma_1^{-1} + \Sigma_2^{-1})^{-1}(\Sigma_1^{-1} \mu_1 + \Sigma_2^{-1} \mu_2) \]

Must store and invert covariance matrices. 
**Too big** to store for large problems. 
**Too costly** to invert, $> O(n^2)$.
Represent a prior pdf by a sample (ensemble) of N values:

Example: Predict temperature on the Albany campus.
Represent a prior pdf by a sample (ensemble) of N values:

\[
T = \frac{1}{N} \sum_{n=1}^{N} T_n
\]

Use sample mean and sample standard deviation to determine a corresponding continuous distribution:

\[
\sigma_T = \sqrt{\frac{1}{N-1} \sum_{n=1}^{N} (T_n - \bar{T})^2}
\]

\[
\text{Normal}(\bar{T}, \sigma_T)
\]
If posterior ensemble at time $t_1$ is $T_{1,n}$, $n = 1, \ldots, N$
If posterior ensemble at time $t_1$ is $T_{1,n}, n = 1, \ldots, N$, advance each member to time $t_2$ with model, $T_{2,n} = L(T_{1,n})$ $n = 1, \ldots, N$. 

![Diagram showing temperature distribution with prior and posterior ensemble.](image-url)
Same as advancing continuous pdf at time $t_1$ ...
Same as advancing continuous pdf at time $t_1$ to time $t_2$ with model L.
Fit a Gaussian to the sample.
Get the observation likelihood.
Compute the continuous posterior PDF.
Use a deterministic algorithm to ‘adjust’ the ensemble.
First, ‘shift’ the ensemble to have the exact mean of the posterior.
First, ‘shift’ the ensemble to have the exact mean of the posterior. Second, linearly contract to have the exact variance of the posterior. Sample statistics are identical to Kalman filter.
So far, we have a known observation likelihood for a single variable.

Now, suppose the model state has an additional variable, temperature at Troy.

How should ensemble members update the additional variable?
Assume that all we know is the prior joint distribution.

One variable is observed.

What should happen to the unobserved variable?
Assume that all we know is the prior joint distribution.

One variable is observed.

Update observed variable with ensemble Kalman filter.
Ensemble filters: Updating additional prior state variables

Assume that all we know is the prior joint distribution.

One variable is observed.

Update observed variable with ensemble Kalman filter.
Ensemble filters: Updating additional prior state variables

Assume that all we know is the prior joint distribution.

One variable is observed.

Update observed variable with ensemble Kalman filter.
Ensemble filters: Updating additional prior state variables

Assume that all we know is the prior joint distribution.

One variable is observed.

Compute increments for prior ensemble members of observed variable.
Ensemble filters: Updating additional prior state variables

Assume that all we know is the prior joint distribution.

One variable is observed.

Compute increments for prior ensemble members of observed variable.
Ensemble filters: Updating additional prior state variables

Assume that all we know is the prior joint distribution.

One variable is observed.

Compute increments for prior ensemble members of observed variable.
Ensemble filters: Updating additional prior state variables

Assume that all we know is the prior joint distribution.

One variable is observed.

Compute increments for prior ensemble members of observed variable.
Ensemble filters: Updating additional prior state variables

Assume that all we know is the prior joint distribution.

One variable is observed.

Compute increments for prior ensemble members of observed variable.
Ensemble filters: Updating additional prior state variables

Using only increments guarantees that if observation had no impact on observed variable, the unobserved variable is unchanged.

Highly desirable!
Ensemble filters: Updating additional prior state variables

Assume that all we know is the prior joint distribution.

How should the unobserved variable be impacted?

1st choice: least squares

Equivalent to linear regression.

Same as assuming binormal prior.
Ensemble filters: Updating additional prior state variables

Have joint prior distribution of two variables.

How should the unobserved variable be impacted?

1\textsuperscript{st} choice: least squares

Begin by finding least squares fit.
Ensemble filters: Updating additional prior state variables

Have joint prior distribution of two variables.

Next, regress the observed variable increments onto increments for the unobserved variable.

Equivalent to first finding image of increment in joint space.
Ensemble filters: Updating additional prior state variables

Have joint prior distribution of two variables.

Next, regress the observed variable increments onto increments for the unobserved variable.

Equivalent to first finding image of increment in joint space.
Ensemble filters: Updating additional prior state variables

Have joint prior distribution of two variables.

Next, regress the observed variable increments onto increments for the unobserved variable.

Equivalent to first finding image of increment in joint space.
Ensemble filters: Updating additional prior state variables

Have joint prior distribution of two variables.

Next, regress the observed variable increments onto increments for the unobserved variable.

Equivalent to first finding image of increment in joint space.
Ensemble filters: Updating additional prior state variables

Have joint prior distribution of two variables.

Next, regress the observed variable increments onto increments for the unobserved variable.

Equivalent to first finding image of increment in joint space.
Ensemble filters: Updating additional prior state variables

Have joint prior distribution of two variables.

Regression: Equivalent to first finding image of increment in joint space.

Then projecting from joint space onto unobserved priors.
Ensemble filters: Updating additional prior state variables

Have joint prior distribution of two variables.

Regression: Equivalent to first finding image of increment in joint space.

Then projecting from joint space onto unobserved priors.
Ensemble filters: Updating additional prior state variables

Have joint prior distribution of two variables.

Regression: Equivalent to first finding image of increment in joint space.

Then projecting from joint space onto unobserved priors.
Ensemble filters: Updating additional prior state variables

Have joint prior distribution of two variables.

Regression: Equivalent to first finding image of increment in joint space.

Then projecting from joint space onto unobserved priors.
Ensemble filters: Updating additional prior state variables

Have joint prior distribution of two variables.

Regression: Equivalent to first finding image of increment in joint space.

Then projecting from joint space onto unobserved priors.
Ensemble filters: Updating additional prior state variables

Now have an updated (posterior) ensemble for the unobserved variable.

We’ve expanded this plot. Same information as previous slides.

Compressed these two.
Ensemble filters: Updating additional prior state variables

Now have an updated (posterior) ensemble for the unobserved variable.

Fitting Gaussians shows that mean and variance have changed.
Ensemble filters: Updating additional prior state variables

Now have an updated (posterior) ensemble for the unobserved variable.

Fitting Gaussians shows that mean and variance have changed.

Other features of the prior distribution may also have changed.
Properties of Ensemble Kalman Filter

For linear, gaussian problem:

If, ensemble size $N > N_{\text{crit}}$

Mean and covariance are identical to Kalman Filter,

Else

Diverges.

$N_{\text{crit}}$: Number of positive singular values in SVD of covariance matrix.
1. Use model to advance ensemble (3 members here) to time at which next observation becomes available.

Ensemble state estimate after using previous observation (analysis)

Ensemble state at time of next observation (prior)
2. Get prior ensemble sample of observation, $y = h(x)$, by applying forward operator $h$ to each ensemble member.

Theory: observations from instruments with uncorrelated errors can be done sequentially.
3. Get **observed value** and **observational error distribution** from observing system.
4. Find the **increments** for the prior observation ensemble (this is a scalar problem for uncorrelated observation errors).
4. Find the increments for the prior observation ensemble (this is a scalar problem for uncorrelated observation errors).

Note: Difference between various ensemble filter methods is primarily in observation increment calculation.
5. Use ensemble samples of $y$ and each state variable to linearly regress observation increments onto state variable increments.
5. Use ensemble samples of $y$ and each state variable to linearly regress observation increments onto state variable increments.

Theory: impact of observation increments on each state variable can be handled independently!
5. Use ensemble samples of $y$ and each state variable to linearly regress observation increments onto state variable increments.
6. When all ensemble members for each state variable are updated, integrate to time of next observation ...
40 state variables: $X_1, X_2, \ldots, X_{40}$.

$$\frac{dX_i}{dt} = (X_{i+1} - X_{i-2})X_{i-1} - X_i + F.$$ 

Acts ‘something’ like weather around a latitude band.
Lorenz-96 is sensitive to small perturbations

Introduce 20 ‘ensemble’ state estimates. Each is perturbed for each of the 40-variables at time 0. Refer to unperturbed control integration as ‘truth’.
Assimilate ‘observations’ from 40 random locations.

Interpolate truth to station location.
Simulate observational error:
    Add random draw from $N(0, 16)$ to each.
Start from ‘climatological’ 20-member ensemble.

Lorenz96 20-member assimilation; no localization or inflation
Some Error Sources in Ensemble Filters

1. Model error

2. Obs. operator error; Representativeness

3. Observation error

4. Sampling Error; Gaussian Assumption

5. Sampling Error; Assuming Linear Statistical Relation
Sampling Error: Observations Impact Unrelated State Variables

Plot shows expected absolute value of sample correlation vs. true correlation.

Unrelated obs. reduce spread, increase error.

Attack with localization.

Reduce impact of observation on weakly correlated state variables.

Let weight go to zero for many ‘unrelated’ variables to save on computing.
Lorenz-96 Assimilation with localization of observation impact

Localization from Hierarchical Filter

No Localization
Lorenz-96 Assimilation with localization of observation impact

Localization from Hierarchical Filter

3 Sample Observation Localizations

State Variable
Some Error Sources in Ensemble Filters

1. Model error

2. Obs. operator error; Representativeness

3. Observation error

4. Sampling Error; Gaussian Assumption

5. Sampling Error; Assuming Linear Statistical Relation
dXi / dt = (Xi+1 - Xi-2)Xi-1 - Xi + F.
For truth, use F = 8.
In assimilating model, use F = 6.

Time evolution for first state variable shown. Assimilating model quickly diverges from ‘true’ model.
\[
\frac{dX_i}{dt} = (X_{i+1} - X_{i-2})X_{i-1} - X_i + F.
\]
For truth, use \( F = 8 \).
In assimilating model, use \( F = 6 \).
Use inflation.
Simply increase prior ensemble variance for each state variable. Adaptive algorithms use observations to guide this.
Inflation is a function of state variable and time. Automatically selected by adaptive inflation algorithm.
(Ensemble) KF optimal for linear model, gaussian likelihood, perfect model.

In KF, only mean and covariance have meaning.

Ensemble allows computation of many other statistics.
What do they mean? Not entirely clear.

What do they mean when there are all sorts of error? Even less clear.

Must Calibrate and Validate results.
DART provides data assimilation ‘glue’ to build ensemble forecast systems for the atmosphere, ocean, land, …
Science: A global atmospheric ensemble reanalysis.

Collaborators: Model Developers at NCAR

O(1 million) atmospheric obs are assimilated every day.

Assimilation uses 80 members of 2° FV CAM forced by a single ocean (Hadley+ NCEP-OI2) and produces a very competitive reanalysis.

500 hPa GPH
Feb 17 2003

1998-2010
4x daily is available.
Science: *Do new satellite observations of cloud motion improve hurricane forecasts?*

Atmospheric motion vectors from CIMMS at University of Wisconsin.

Collaborator: Ting-Chi Wu,  
Graduate Student,  
University of Miami.
Wu et al., 2014, MWR, 142, 49–71.
Science: \textit{Where should more observations be taken to improve landfall forecasts?}

Ensemble sensitivity analysis for Katrina.

Collaborator: Ryan Torn, University at Albany.
Hurricane Katrina Sensitivity Analysis

Contours are ensemble mean 48h forecast of deep-layer mean wind. Color shows where observations could help.
Identification of Model Systematic Errors

- Prediction Model
- Observing System
- Data Assimilation
- Analysis
- Diagnostics

Identify Systematic Errors
Science: Diagnosing and correcting errors in the CAM FV core.
Collaborator: Peter Lauritzen, CGD.
Gridpoint noise detected in CAM/DART analysis

Ensemble Mean V at 266 hPa at 6 hours

CAM FV core - 80 member mean - 00Z 25 September 2006
Suspicions turned to the polar filter (DPF)

Ensemble Mean V at 266 hPa at 6 hours

Fourier

Algebraic

CAM FV core - 80 member mean - 00Z 25 September 2006
Continuous polar filter (alt-pft) eliminated noise.
Differences mostly in transition region of default filter.
The use of DART diagnosed a problem that had been unrecognized (or at least undocumented).

Could have an important effect on any physics in which meridional mixing is important.

The problem can be seen in ‘free runs’ - it is not a data assimilation artifact.

Without assimilation, can’t get reproducing occurrences to diagnose.
• Climate change over time scales of 1 to several decades has been identified as very important for mitigation and infrastructure planning.
• Need ocean initial conditions for the IPCC decadal prediction program (and maybe a crystal ball, too!).
World Ocean Database T, S observation counts.

These counts are for 1998 & 1999 and are representative.

<table>
<thead>
<tr>
<th>Variable</th>
<th>Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>FLOAT_SALINITY</td>
<td>68200</td>
</tr>
<tr>
<td>FLOAT_TEMPERATURE</td>
<td>395032</td>
</tr>
<tr>
<td>DRIFTER_TEMPERATURE</td>
<td>33963</td>
</tr>
<tr>
<td>MOORING_SALINITY</td>
<td>27476</td>
</tr>
<tr>
<td>MOORING_TEMPERATURE</td>
<td>623967</td>
</tr>
<tr>
<td>BOTTLE_SALINITY</td>
<td>79855</td>
</tr>
<tr>
<td>BOTTLE_TEMPERATURE</td>
<td>81488</td>
</tr>
<tr>
<td>CTD_SALINITY</td>
<td>328812</td>
</tr>
<tr>
<td>CTD_TEMPERATURE</td>
<td>368715</td>
</tr>
<tr>
<td>STD_SALINITY</td>
<td>674</td>
</tr>
<tr>
<td>STD_TEMPERATURE</td>
<td>677</td>
</tr>
<tr>
<td>XCTD_SALINITY</td>
<td>3328</td>
</tr>
<tr>
<td>XCTD_TEMPERATURE</td>
<td>5790</td>
</tr>
<tr>
<td>MBT_TEMPERATURE</td>
<td>58206</td>
</tr>
<tr>
<td>XBT_TEMPERATURE</td>
<td>1093330</td>
</tr>
<tr>
<td>APB_TEMPERATURE</td>
<td>580111</td>
</tr>
</tbody>
</table>
Physical Space: 1998/1999 SST Anomaly from HadOII-SST

Coupled Free Run

- POP 1 DATM
- RMSE = 2.289°C

POP forced by observed atmosphere (hindcast)

- POP 48 CAM
- RMSE = 1.317°C

- DARTPOP_23
- RMSE = 1.042°C

- DARTPOP_48
- RMSE = 0.908°C

University at Albany, 22 Oct. 2015
Science: Land surface analysis with DART/CLM.
Collaborator: Yongfei Zhang, UT Austin.

Land surface analysis with DART/CLM:
Estimate snow water equivalent with observations of snow cover fraction from satellites (MODIS).
• 80 member ensemble for onset of NH winter
• Assimilate once per day
• Level 3 MODIS product – regridded to a daily 1 degree grid
• Observation error variance is 0.1 (for lack of a better value)
• Observations can impact state variables within 200km
• CLM variable to be updated is the snow water equivalent “H2OSNO”
An early result: assimilation of MODIS snowcover fraction on total snow water equivalent in CLM.

Prior for Nov 30, 2002

Focus on the non-zero increments

Increments (Prior – Posterior)

The model state is changing in reasonable places, by reasonable amounts. At this point, that’s all we’re looking for.

Thanks Yongfei!
Science: Regional Atmospheric Chemistry.
Collaborator: Arthur Mizzi, NCAR/ACD.
WRF/Chem Chemical Weather Forecast System

- **WRF-Chem** – Weather Research and Forecasting Model (WRF) with online chemistry.

- **Meteorological Observations** – NOAA PREPBUFR conventional observations.

- **Chemistry Observations** – MOPITT CO retrieval profiles (also IASI CO retrievals – results not shown).
WRF/Chem-DART cycling with conventional meteorological observations and MOPITT CO V5 retrieval profiles.

Continuous six-hr cycling (00Z, 06Z, 12Z, and 18Z).

CONUS grid with 101x41x34 grid points and 100 km resolution.

20-member ensemble.

June 1 - 30, 2008 (112 cycles) study period.

Full state variable/obs interaction.

Initial and lateral chemical boundary conditions from MOZART-4 simulation.

Two experiments:

- Exp 1: PREPBUFR conventional obs (CNTL DA).
- Exp 2: MOPITT CO retrieval profiles and PREPBUFR conventional obs (CHEM DA).
WRF/Chem Chemical Weather Forecast System

CO Time Series

Date

VMR (ppb)

CHM NL F
CHM NL A
CNL NL F
CNL NL A
MOPITT

Longitude (Degrees East)
Latitude (Degrees North)

CO Fcst CNL–NL
June 29, 2008 00Z

CO Fcst CHM–NL
June 29, 2008 00Z

CO Forecast Diff
June 29, 2008 00Z
Science: Global Atmospheric Chemistry.
Collaborators: Jerome Barre,
    Benjamin Gaubert, NCAR/ACD.

Uses global CAM/Chem model, 1 degree.

Have full meteorological assimilation capability already.
MOPITT CO:
On TERRA satellite
tropospheric profiles
Global coverage in 4 days
Multispectral retrievals
high sensitivity on surface land/day

IASI CO:
On MetOpA satellite
tropospheric profiles
Global coverage in 1 day
Only thermal infrared
Sensitivity on upper PBL & mid troposphere
Control run: Met Only assimilated

MOPITT run: Met + MOPITT assimilated

IASI run: Met + IASI assimilated

Combined run: Met + MOP+ IASI assimilated
A system governed by (stochastic) Difference Equation:

\[ dx_t = f(x_t, t) + G(x_t, t) \, d\beta_t, \quad t \geq 0 \]  

(1)

Observations at discrete times:

\[ y_k = h(x_k, t_k) + v_k; \quad k = 1, 2, \ldots; \quad t_{k+1} > t_k \geq t_0 \]  

(2)

Observational error white in time and Gaussian (nice, not essential).

\[ v_k \rightarrow N(0, R_k) \]  

(3)

Complete history of observations is:

\[ Y_\tau = \{ y_l; t_l \leq \tau \} \]  

(4)

Goal: Find probability distribution for state:

\[ p(x, t \mid Y_t) \quad \text{Analysis} \quad p(x, t^+ \mid Y_t) \quad \text{Forecast} \]  

(5)
A system governed by (stochastic) Difference Equation:

\[ dx_t = f(x_t, t) + G(x_t, t) \, d\beta_t , \quad t \geq 0 \]  \hspace{1cm} (1)
A system governed by (stochastic) Difference Equation:

\[ dx_t = f(x_t, t; \alpha) + G(x_t, t) d\beta_t, \quad t \geq 0 \]  

Most geophysical models have ‘tuning’ parameters.

Model prediction might also depend on ‘external forcing’.

Example: Sources of chemical tracers.
A system governed by (stochastic) Difference Equation:

\[ dx_t = f(x_t, t; \alpha) + G(x_t, t) \, d\beta_t, \quad t \geq 0 \]  \tag{1}

One solution: State augmentation.

Define extended state vector \( x^+ = (x, \alpha) \)

Prediction model becomes (just a change in notation):

\[ dx^+_t = f(x^+_t, t) + G(x_t, t) \, d\beta_t, \quad t \geq 0 \]
Define extended state vector \( x^+ = (x, \alpha) \)

Prediction model becomes:

\[
dx_t^+ = f(x_t^+, t) + G(x_t, t) d\beta_t, \quad t \geq 0
\]

Problem: In general, no time prediction model for parameters.

If we had a prediction model, they would just have been state.

Kalman filter prior covariance comes from prediction model.
Define extended state vector \( x^+ = (x, \alpha) \)

Prediction model becomes:
\[
dx^+_t = f(x^+_t, t) + G(x_t, t) d\beta_t, \quad t \geq 0
\]

Prior ensembles for parameters must be specified.

The prior sample covariance controls the impact of observations on parameters.

If prior covariance is not well-known, estimating parameters can be challenging.