

**“What is data assimilation really solving?
What can it solve?
How is the calculation actually done?”**

Ichiro Fukumori (JPL)

GODAE Summer School
September, 2004

What are we really solving? How can we solve it?

OUTLINE

1. What is assimilation really solving?
 - a) Filtering vs Smoothing
 - b) Observability & Controllability
 - c) Data Error and Model Error

2. Examples and Practical Issues; e.g., ECCO
 - a) A hierarchical approach
 - b) Practical derivation and implementation
 - c) Assessment

What is assimilation really solving?

Filtering vs Smoothing

1. Kalman filtering and recursive smoothing are both **least-squares** inversions, but **solve different parts** of the assimilation problem,
2. **Filtered** solutions are **physically inconsistent**,
3. **Smoothed** solutions can be **physically consistent**,
4. Estimating **model error sources** is fundamental.

The Mathematical Problem of Assimilation

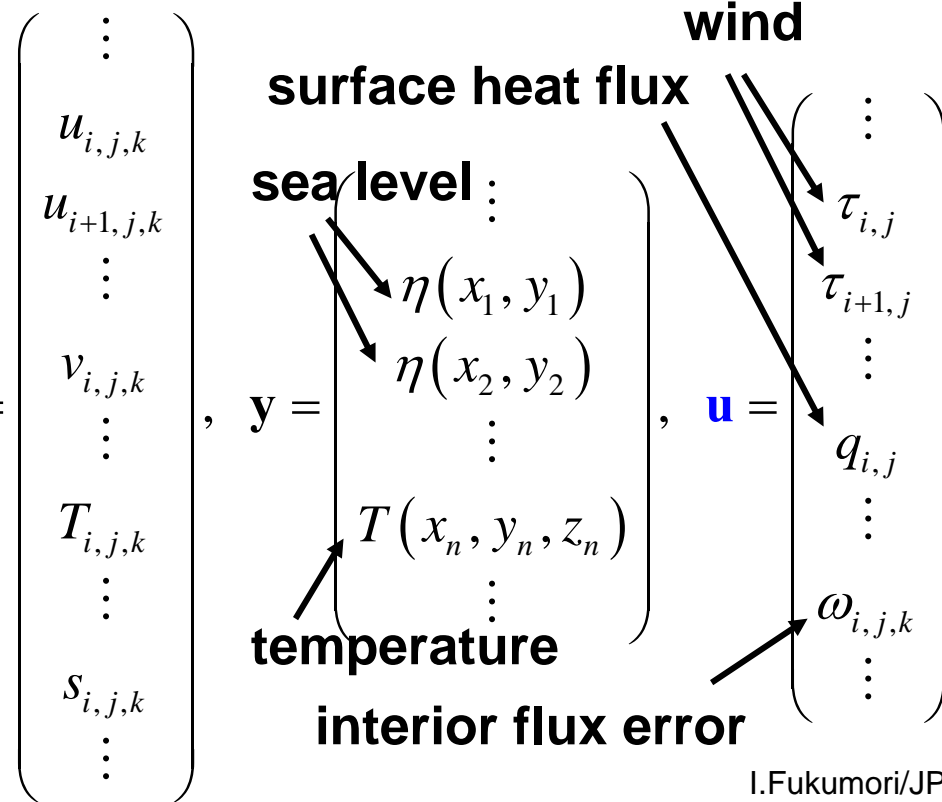
Different times

$$\begin{array}{l}
 \text{Observation Eqs} \\
 \vdots \\
 \mathbf{H}\mathbf{x}_t \\
 \mathbf{H}\mathbf{x}_{t+1} \\
 \vdots \\
 \text{Model Eqs} \\
 \vdots \\
 \mathbf{x}_{t+1} - \mathbf{A}\mathbf{x}_t - \mathbf{G}\mathbf{u}_t \\
 \mathbf{x}_{t+2} - \mathbf{A}\mathbf{x}_{t+1} - \mathbf{G}\mathbf{u}_{t+1} \\
 \vdots
 \end{array}
 \approx
 \begin{array}{l}
 \vdots \\
 \mathbf{y}_t \\
 \mathbf{y}_{t+1} \\
 \vdots \\
 \mathbf{0} \\
 \mathbf{0} \\
 \vdots
 \end{array}$$

Assimilation problem: Given observations, y , and a model (A, G, H), what are the model states (x) and their controls (u)?

- \mathbf{x} : model state
- \mathbf{y} : observations
- \mathbf{H} : observation operator
- \mathbf{A}, \mathbf{G} : model dynamics
- \mathbf{u} : control (e.g., forcing, and model errors)

e.g., $\mathbf{x} =$

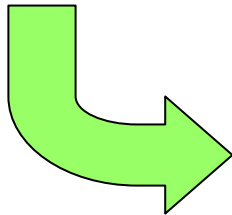
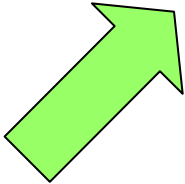


The Mathematical Problem of Assimilation

$$\begin{pmatrix} \vdots \\ \mathbf{H}\mathbf{x}_t \\ \mathbf{H}\mathbf{x}_{t+1} \\ \vdots \\ \mathbf{x}_{t+1} - \mathbf{A}\mathbf{x}_t - \mathbf{G}\mathbf{u}_t \\ \mathbf{x}_{t+2} - \mathbf{A}\mathbf{x}_{t+1} - \mathbf{G}\mathbf{u}_{t+1} \\ \vdots \end{pmatrix} \approx \begin{pmatrix} \vdots \\ \mathbf{y}_t \\ \mathbf{y}_{t+1} \\ \vdots \\ \mathbf{0} \\ \mathbf{0} \\ \vdots \end{pmatrix}$$

This problem is essentially an inverse problem

$$\mathbf{E}\mathbf{a} \approx \mathbf{b}$$



$$\begin{pmatrix} \ddots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \\ \dots & \mathbf{H} & & & & & \dots \\ \dots & & \mathbf{H} & & & & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ \dots & -\mathbf{A} & \mathbf{I} & & -\mathbf{G} & & \dots \\ \dots & & -\mathbf{A} & \mathbf{I} & & -\mathbf{G} & \dots \\ \ddots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{pmatrix} \begin{pmatrix} \vdots \\ \mathbf{x}_t \\ \mathbf{x}_{t+1} \\ \mathbf{x}_{t+2} \\ \vdots \\ \mathbf{u}_t \\ \mathbf{u}_{t+1} \\ \vdots \end{pmatrix} \approx \begin{pmatrix} \vdots \\ \mathbf{y}_t \\ \mathbf{y}_{t+1} \\ \vdots \\ \mathbf{0} \\ \mathbf{0} \\ \vdots \end{pmatrix}$$

Least-Squares Solution

Almost all direct assimilation methods can be recognized as a form of least-squares.

For a linear inverse problem

$$\mathbf{E}\mathbf{a} \approx \mathbf{b}$$

The least-squares solution is given by
(Gauss-Markov inversion)

$$\hat{\mathbf{a}} = \mathbf{a}_0 + \mathbf{R}_{aa} \mathbf{E}^T \left(\mathbf{E} \mathbf{R}_{aa} \mathbf{E}^T + \mathbf{R}_{bb} \right)^{-1} (\mathbf{b} - \mathbf{E} \mathbf{a}_0)$$

$$\mathbf{R}_{\hat{a}\hat{a}} = \mathbf{R}_{aa} - \mathbf{R}_{aa} \mathbf{E}^T \left(\mathbf{E} \mathbf{R}_{aa} \mathbf{E}^T + \mathbf{R}_{bb} \right)^{-1} \mathbf{E} \mathbf{R}_{aa}$$

$\hat{\mathbf{a}}$: least-squares solution

$\mathbf{a}_0, \mathbf{R}_{aa}$: first guess (prior solution) and its error covariance matrix

\mathbf{R}_{bb} : error covariance matrix of \mathbf{b}

$\mathbf{R}_{\hat{a}\hat{a}}$: Error covariance of $\hat{\mathbf{a}}$

Kalman filter as Least-Squares

Linear inverse problem and its least-squares solution;

$$\mathbf{E}\mathbf{a} \approx \mathbf{b}$$



$$\hat{\mathbf{a}} = \mathbf{a}_0 + \mathbf{R}_{aa} \mathbf{E}^T \left(\mathbf{E} \mathbf{R}_{aa} \mathbf{E}^T + \mathbf{R}_{bb} \right)^{-1} \left(\mathbf{b} - \mathbf{E} \mathbf{a}_0 \right)$$

$$\mathbf{R}_{\hat{\mathbf{a}}\hat{\mathbf{a}}} = \mathbf{R}_{aa} - \mathbf{R}_{aa} \mathbf{E}^T \left(\mathbf{E} \mathbf{R}_{aa} \mathbf{E}^T + \mathbf{R}_{bb} \right)^{-1} \mathbf{E} \mathbf{R}_{aa}$$

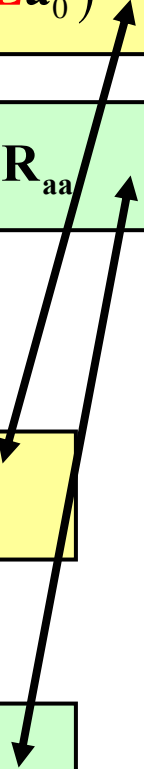
Kalman filter estimate :

“forecast” → $\hat{\mathbf{x}}_t^f = \mathbf{A} \hat{\mathbf{x}}_{t-1}^a + \mathbf{G} \mathbf{u}_{t-1}$

“analysis” → $\hat{\mathbf{x}}_t^a = \hat{\mathbf{x}}_t^f + \mathbf{P}_t^f \mathbf{H}^T \left(\mathbf{H} \mathbf{P}_t^f \mathbf{H}^T + \mathbf{R}_t \right)^{-1} \left(\mathbf{y}_t - \mathbf{H} \hat{\mathbf{x}}_t^f \right)$

$$\mathbf{P}_t^f = \mathbf{A} \hat{\mathbf{P}}_{t-1}^a \mathbf{A}^T + \mathbf{G} \mathbf{Q} \mathbf{G}^T$$

$$\hat{\mathbf{P}}_t^a = \mathbf{P}_t^f - \mathbf{P}_t^f \mathbf{H}^T \left(\mathbf{H} \mathbf{P}_t^f \mathbf{H}^T + \mathbf{R}_t \right)^{-1} \mathbf{H} \mathbf{P}_t^f$$



There is a direct correspondence between the Kalman filtering algorithm and least-squares; i.e., Kalman filtering is nothing more than an inversion of the observation equation/operator \mathbf{H} .

Inverse Solution

$$\hat{\mathbf{a}} = \mathbf{a}_0 + \mathbf{R}_{aa} \mathbf{E}^T \left(\mathbf{E} \mathbf{R}_{aa} \mathbf{E}^T + \mathbf{R}_{bb} \right)^{-1} (\mathbf{b} - \mathbf{E} \mathbf{a}_0)$$

is the linear inverse solution of $\mathbf{E} \mathbf{a} \approx \mathbf{b}$ for any of following;

1. **Minimum (expected error) variance:** $\min \left(\text{diag} \left\langle (\mathbf{a} - \bar{\mathbf{a}})(\mathbf{a} - \bar{\mathbf{a}})^T \right\rangle \right)$

if $\mathbf{R}_{aa} = \left\langle (\mathbf{a}_0 - \bar{\mathbf{a}})(\mathbf{a}_0 - \bar{\mathbf{a}})^T \right\rangle$, $\mathbf{R}_{bb} = \left\langle (\mathbf{b} - \mathbf{E} \bar{\mathbf{a}})(\mathbf{b} - \mathbf{E} \bar{\mathbf{a}})^T \right\rangle$ True a

2. **Least-squares:** $\min \left((\mathbf{a} - \mathbf{a}_0)^T \mathbf{W}_a (\mathbf{a} - \mathbf{a}_0) + (\mathbf{b} - \mathbf{E} \mathbf{a})^T \mathbf{W}_b (\mathbf{b} - \mathbf{E} \mathbf{a}) \right)$

if $\mathbf{W}_a = \mathbf{R}_{aa}^{-1}$, $\mathbf{W}_b = \mathbf{R}_{bb}^{-1}$

3. **Maximum likelihood:** $\max \left[\wp(\mathbf{a}) \right]$

if Gaussian probability distribution

$$\wp(\mathbf{a}) = \exp \left(-(\mathbf{a} - \mathbf{a}_0)^T \mathbf{R}_{aa}^{-1} (\mathbf{a} - \mathbf{a}_0) - (\mathbf{b} - \mathbf{E} \mathbf{a})^T \mathbf{R}_{bb}^{-1} (\mathbf{b} - \mathbf{E} \mathbf{a}) \right)$$

Least-squares does not necessarily assume Gaussian statistics.

Kalman filter as Least-Squares

$$\begin{pmatrix} \vdots \\ \mathbf{H}\mathbf{x}_t \\ \mathbf{H}\mathbf{x}_{t+1} \\ \vdots \\ \mathbf{x}_{t+1} - \mathbf{A}\mathbf{x}_t - \mathbf{G}\mathbf{u}_t \\ \mathbf{x}_{t+2} - \mathbf{A}\mathbf{x}_{t+1} - \mathbf{G}\mathbf{u}_{t+1} \\ \vdots \end{pmatrix} \approx \begin{pmatrix} \vdots \\ \mathbf{y}_t \\ \mathbf{y}_{t+1} \\ \vdots \\ \mathbf{0} \\ \mathbf{0} \\ \vdots \end{pmatrix}$$

$$\begin{pmatrix} \vdots \\ \mathbf{H}\mathbf{x}_{t-1} \\ \mathbf{H}\mathbf{x}_t \\ \mathbf{x}_1 - \mathbf{A}\mathbf{x}_0 - \mathbf{G}\mathbf{u}_0 \\ \vdots \\ \mathbf{x}_{t+1} - \mathbf{A}\mathbf{x}_t - \mathbf{G}\mathbf{u}_t \\ \mathbf{x}_t - \mathbf{A}\mathbf{x}_{t-1} - \mathbf{G}\mathbf{u}_{t-1} \end{pmatrix} \approx \begin{pmatrix} \vdots \\ \mathbf{y}_{t-1} \\ \mathbf{y}_t \\ \mathbf{0} \\ \vdots \\ \mathbf{0} \\ \mathbf{0} \end{pmatrix}$$

$$\hat{\mathbf{x}}_t^a = \hat{\mathbf{x}}_t^f + \mathbf{P}_t^f \mathbf{H}^T (\mathbf{H}\mathbf{P}_t^f \mathbf{H}^T + \mathbf{R}_t)^{-1} (\mathbf{y}_t - \mathbf{H}\hat{\mathbf{x}}_t^f)$$

- This is an **inversion of the observation equation** but **NOT** an inversion of the model equations.
- The estimate is the **least-squares solution for the end state**, but not at intermediate times.
- The **virtue** of Kalman filtering is that these inverted **end states do not have to be recomputed over the entire period** every time new observations are obtained.

Kalman filter as Least-Squares

The temporal evolution of filtered estimates are physically inconsistent (e.g., budgets do not close).

Kalman filter estimate : $\hat{\mathbf{x}}_t^f = \mathbf{A}\hat{\mathbf{x}}_{t-1}^a + \mathbf{G}\mathbf{u}_{t-1}$

$$\hat{\mathbf{x}}_t^a = \hat{\mathbf{x}}_t^f + \mathbf{P}_t^f \mathbf{H}^T \left(\mathbf{H}\mathbf{P}_t^f \mathbf{H}^T + \mathbf{R}_t \right)^{-1} \left(\mathbf{y}_t - \mathbf{H}\hat{\mathbf{x}}_t^f \right)$$

Combining the two steps give,

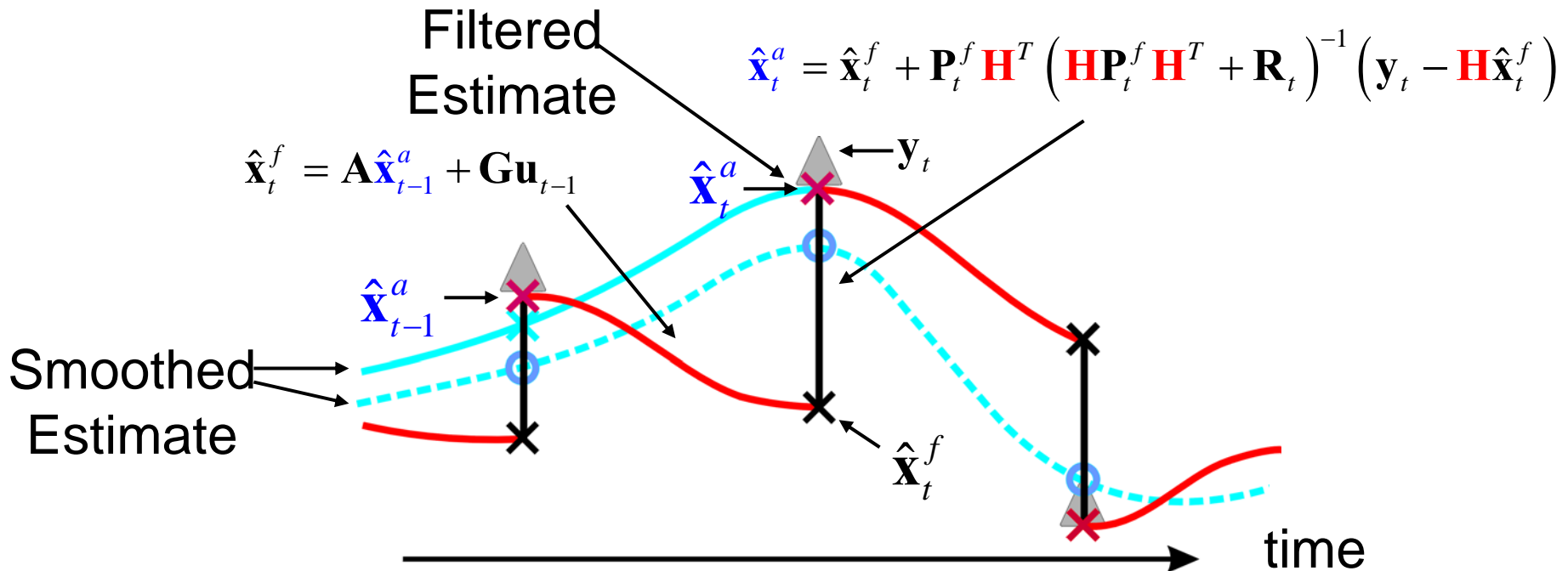
$$\hat{\mathbf{x}}_t^a = \mathbf{A}\hat{\mathbf{x}}_{t-1}^a + \mathbf{G}\mathbf{u}_{t-1} + \underbrace{\mathbf{P}_t^f \mathbf{H}^T \left(\mathbf{H}\mathbf{P}_t^f \mathbf{H}^T + \mathbf{R}_t \right)^{-1} \left(\mathbf{y}_t - \mathbf{H}\hat{\mathbf{x}}_t^f \right)}$$

Not associated with any particular physics (e.g., advection, mixing, external forcing) embodied in \mathbf{A} and \mathbf{G} .

Consistency of Temporal Evolution

$$\hat{\mathbf{x}}_t^a = \mathbf{A}\hat{\mathbf{x}}_{t-1}^a + \mathbf{G}\mathbf{u}_{t-1} + \mathbf{P}_t^f \mathbf{H}^T \left(\mathbf{H}\mathbf{P}_t^f \mathbf{H}^T + \mathbf{R}_t \right)^{-1} \left(\mathbf{y}_t - \mathbf{H}\hat{\mathbf{x}}_t^f \right)$$

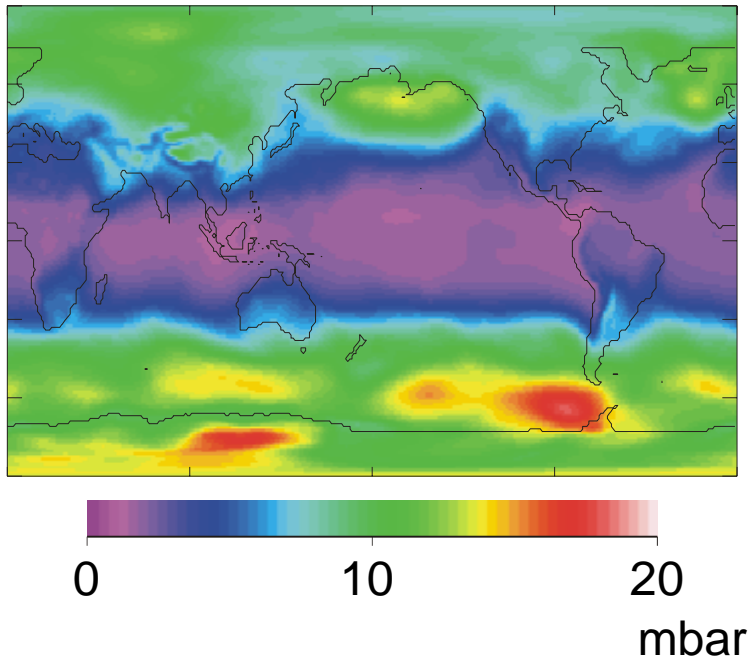
Because of the filter's data increments, the **temporal evolution** of the filtered state from $\hat{\mathbf{x}}_{t-1}^a$ to $\hat{\mathbf{x}}_t^a$, etc, **cannot be physically accounted for**; e.g., budgets cannot be closed.



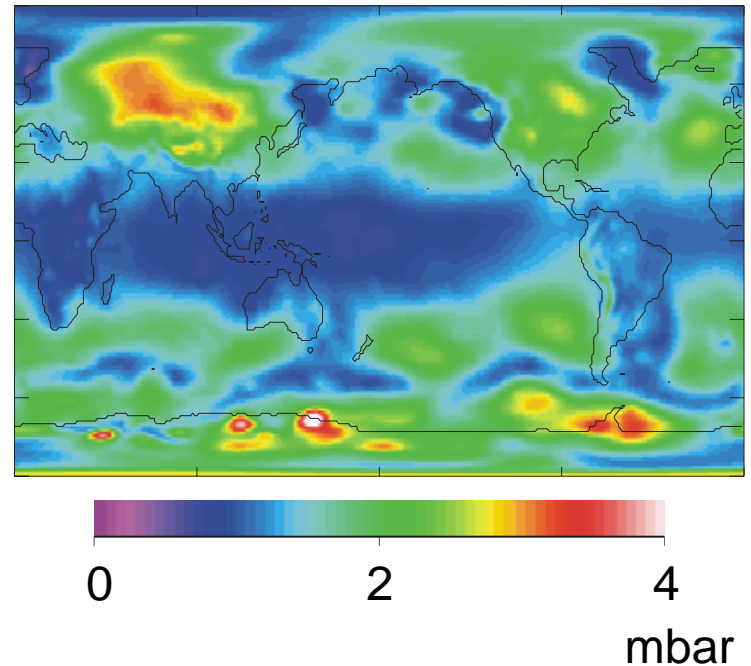
Example: Inconsistencies in Atmospheric Analyses

24% of mass change in NCEP's operational analysis is physically **unaccounted for**.

change over 6-hours



assimilation increment



Smother as Least-Squares

Kalman filter estimate:

$$\hat{\mathbf{x}}_t^a = \mathbf{A}\hat{\mathbf{x}}_{t-1}^a + \mathbf{G}\mathbf{u}_{t-1} + \mathbf{P}_t^f \mathbf{H}^T \left(\mathbf{H}\mathbf{P}_t^f \mathbf{H}^T + \mathbf{R}_t \right)^{-1} \left(\mathbf{y}_t - \mathbf{H}\hat{\mathbf{x}}_t^f \right)$$

Smoothing problem: Given the updated estimate $\hat{\mathbf{x}}_t^a$ invert the model to correct the prior state \mathbf{x}_{t-1} and control \mathbf{u}_{t-1} that;

$$\hat{\mathbf{x}}_t^a = \mathbf{A}\hat{\mathbf{x}}_{t-1}^s + \mathbf{G}\hat{\mathbf{u}}_{t-1}^s = \begin{pmatrix} \mathbf{A} & \mathbf{G} \end{pmatrix} \begin{pmatrix} \hat{\mathbf{x}}_{t-1}^s \\ \hat{\mathbf{u}}_{t-1}^s \end{pmatrix}$$

A least-squares solution that exactly satisfies the inverse problem $\mathbf{E}\mathbf{a} = \mathbf{b}$ can be obtained by setting $\mathbf{R}_{bb} = 0$ in the canonical least-squares solution;

By substitution, we have;

$$\begin{pmatrix} \hat{\mathbf{x}}_{t-1}^s \\ \hat{\mathbf{u}}_{t-1}^s \end{pmatrix} = \begin{pmatrix} \hat{\mathbf{x}}_{t-1}^a \\ \mathbf{u}_{t-1} \end{pmatrix} + \begin{pmatrix} \mathbf{P}_{t-1}^a \mathbf{A}^T \left(\mathbf{A}\mathbf{P}_{t-1}^a \mathbf{A}^T + \mathbf{G}\mathbf{Q}_{t-1} \mathbf{G}^T \right)^{-1} \\ \mathbf{Q}_{t-1} \mathbf{G}^T \left(\mathbf{G}\mathbf{Q}_{t-1} \mathbf{G}^T + \mathbf{A}\mathbf{P}_{t-1}^a \mathbf{A}^T \right)^{-1} \end{pmatrix} \left(\hat{\mathbf{x}}_t^a - \mathbf{A}\hat{\mathbf{x}}_{t-1}^a - \mathbf{G}\mathbf{u}_{t-1} \right)$$

Smother as Least-Squares

More generally,

$$\begin{pmatrix} \hat{\mathbf{x}}_t^s \\ \hat{\mathbf{u}}_t^s \end{pmatrix} = \begin{pmatrix} \hat{\mathbf{x}}_t^a \\ \mathbf{u}_t \end{pmatrix} + \begin{pmatrix} \mathbf{P}_t^a \mathbf{A}^T (\mathbf{A} \mathbf{P}_t^a \mathbf{A}^T + \mathbf{G} \mathbf{Q}_t \mathbf{G}^T)^{-1} \\ \mathbf{Q}_t \mathbf{G}^T (\mathbf{G} \mathbf{Q}_t \mathbf{G}^T + \mathbf{A} \mathbf{P}_t^a \mathbf{A}^T)^{-1} \end{pmatrix} (\hat{\mathbf{x}}_{t+1}^s - \mathbf{A} \hat{\mathbf{x}}_t^a - \mathbf{G} \mathbf{u}_t)$$

$$\begin{pmatrix} \mathbf{P}_t^s \\ \mathbf{Q}_t^s \end{pmatrix} = \begin{pmatrix} \mathbf{P}_t^a \\ \mathbf{Q}_t \end{pmatrix} - \begin{pmatrix} \mathbf{P}_t^a \mathbf{A}^T (\mathbf{A} \mathbf{P}_t^a \mathbf{A}^T + \mathbf{G} \mathbf{Q}_t \mathbf{G}^T)^{-1} \mathbf{A} \mathbf{P}_t^a \\ \mathbf{Q}_t \mathbf{G}^T (\mathbf{G} \mathbf{Q}_t \mathbf{G}^T + \mathbf{A} \mathbf{P}_t^a \mathbf{A}^T)^{-1} \mathbf{G} \mathbf{Q}_t \end{pmatrix} + \begin{pmatrix} \mathbf{L}_t^a \mathbf{P}_{t+1}^s \mathbf{L}_t^{aT} \\ \mathbf{M}_t^a \mathbf{P}_{t+1}^s \mathbf{M}_t^{aT} \end{pmatrix}$$

where, $\mathbf{L}_t^a \equiv \mathbf{P}_t^a \mathbf{A}^T (\mathbf{A} \mathbf{P}_t^a \mathbf{A}^T + \mathbf{G} \mathbf{Q}_t \mathbf{G}^T)^{-1}$ Extra term due to errors in $\hat{\mathbf{x}}_{t+1}^s$

$\mathbf{M}_t^a \equiv \mathbf{Q}_t \mathbf{G}^T (\mathbf{G} \mathbf{Q}_t \mathbf{G}^T + \mathbf{A} \mathbf{P}_t^a \mathbf{A}^T)^{-1}$

Least-squares solution;

$$\mathbf{E} \mathbf{a} \approx \mathbf{b}$$



$$\hat{\mathbf{a}} = \mathbf{a}_0 + \mathbf{R}_{aa} \mathbf{E}^T (\mathbf{E} \mathbf{R}_{aa} \mathbf{E}^T + \mathbf{R}_{bb})^{-1} (\mathbf{b} - \mathbf{E} \mathbf{a}_0)$$

$$\mathbf{R}_{\hat{\mathbf{a}}\hat{\mathbf{a}}} = \mathbf{R}_{aa} - \mathbf{R}_{aa} \mathbf{E}^T (\mathbf{E} \mathbf{R}_{aa} \mathbf{E}^T + \mathbf{R}_{bb})^{-1} \mathbf{E} \mathbf{R}_{aa}$$

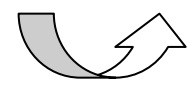
Smoother as Least-Squares

$$\begin{pmatrix} \hat{\mathbf{x}}_t^s \\ \hat{\mathbf{u}}_t^s \end{pmatrix} = \begin{pmatrix} \hat{\mathbf{x}}_t^a \\ \mathbf{u}_t \end{pmatrix} + \begin{pmatrix} \mathbf{P}_t^a \mathbf{A}^T (\mathbf{A} \mathbf{P}_t^a \mathbf{A}^T + \mathbf{G} \mathbf{Q}_t \mathbf{G}^T)^{-1} \\ \mathbf{Q}_t \mathbf{G}^T (\mathbf{G} \mathbf{Q}_t \mathbf{G}^T + \mathbf{A} \mathbf{P}_t^a \mathbf{A}^T)^{-1} \end{pmatrix} \begin{pmatrix} \hat{\mathbf{x}}_{t+1}^s - \mathbf{A} \hat{\mathbf{x}}_t^a - \mathbf{G} \mathbf{u}_t \end{pmatrix}$$

- This is the **Rauch-Tung-Striebel (RTS) fixed-interval smoother** and is an **inversion of the model equations**.

$$\begin{pmatrix} \vdots \\ \mathbf{H} \mathbf{x}_t \\ \mathbf{H} \mathbf{x}_{t+1} \\ \vdots \\ \mathbf{x}_{t+1} - \mathbf{A} \mathbf{x}_t - \mathbf{G} \mathbf{u}_t \\ \mathbf{x}_{t+2} - \mathbf{A} \mathbf{x}_{t+1} - \mathbf{G} \mathbf{u}_{t+1} \\ \vdots \end{pmatrix} \approx \begin{pmatrix} \vdots \\ \mathbf{y}_t \\ \mathbf{y}_{t+1} \\ \vdots \\ \mathbf{0} \\ \mathbf{0} \\ \vdots \end{pmatrix}$$

- This **solves the entire assimilation problem** by sequentially solving smaller calculations.
- Another virtue is its results' **physical consistency**.



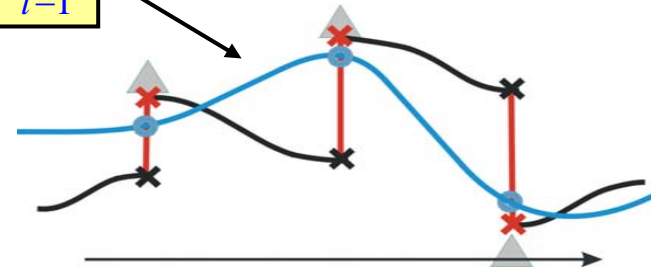
Smoother as Least-Squares

$$\begin{pmatrix} \hat{\mathbf{x}}_t^s \\ \hat{\mathbf{u}}_t^s \end{pmatrix} = \begin{pmatrix} \hat{\mathbf{x}}_t^a \\ \mathbf{u}_t \end{pmatrix} + \begin{pmatrix} \mathbf{P}_t^a \mathbf{A}^T (\mathbf{A} \mathbf{P}_t^a \mathbf{A}^T + \mathbf{G} \mathbf{Q}_t \mathbf{G}^T)^{-1} \\ \mathbf{Q}_t \mathbf{G}^T (\mathbf{G} \mathbf{Q}_t \mathbf{G}^T + \mathbf{A} \mathbf{P}_t^a \mathbf{A}^T)^{-1} \end{pmatrix} (\hat{\mathbf{x}}_{t+1}^s - \mathbf{A} \hat{\mathbf{x}}_t^a - \mathbf{G} \mathbf{u}_t)$$

- By construction, this satisfies the model equation,

$$\hat{\mathbf{x}}_t^s = \mathbf{A} \hat{\mathbf{x}}_{t-1}^s + \mathbf{G} \hat{\mathbf{u}}_{t-1}^s$$

making it amenable to process studies. This should **not** be confused with the “strong constraint” solution; In fact, $\hat{\mathbf{x}}_t^s \neq \mathbf{A} \hat{\mathbf{x}}_{t-1}^s + \mathbf{G} \mathbf{u}_{t-1}$

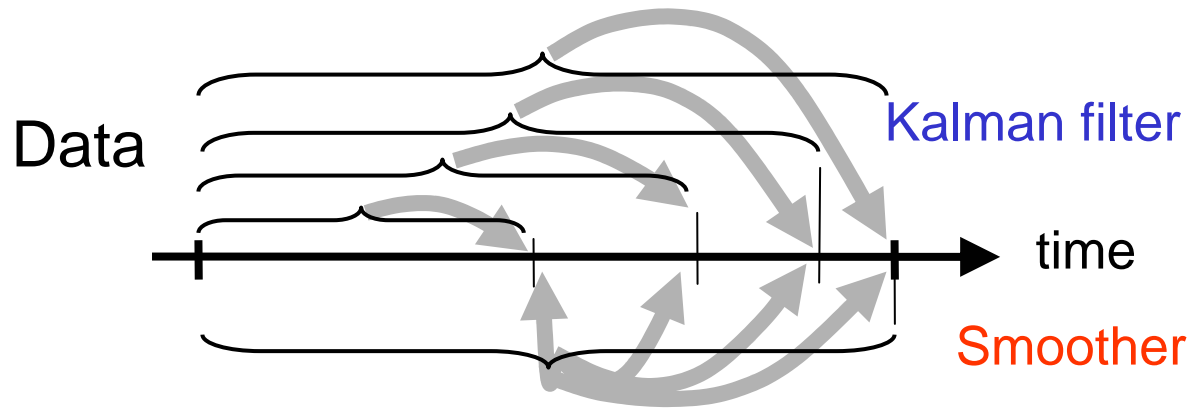


- Control estimates are of the essence.** Smoothed state estimates can be computed from the control estimates.
- The model for model **process noise** \mathbf{u}_t **must be physically based** for the result to be physically sensible.

Smoother as Least-Squares

$$\begin{pmatrix} \hat{\mathbf{x}}_t^s \\ \hat{\mathbf{u}}_t^s \end{pmatrix} = \begin{pmatrix} \hat{\mathbf{x}}_t^a \\ \mathbf{u}_t \end{pmatrix} + \begin{pmatrix} \mathbf{P}_t^a \mathbf{A}^T (\mathbf{A} \mathbf{P}_t^a \mathbf{A}^T + \mathbf{G} \mathbf{Q}_t \mathbf{G}^T)^{-1} \\ \mathbf{Q}_t \mathbf{G}^T (\mathbf{G} \mathbf{Q}_t \mathbf{G}^T + \mathbf{A} \mathbf{P}_t^a \mathbf{A}^T)^{-1} \end{pmatrix} (\hat{\mathbf{x}}_{t+1}^s - \mathbf{A} \hat{\mathbf{x}}_t^a - \mathbf{G} \mathbf{u}_t)$$

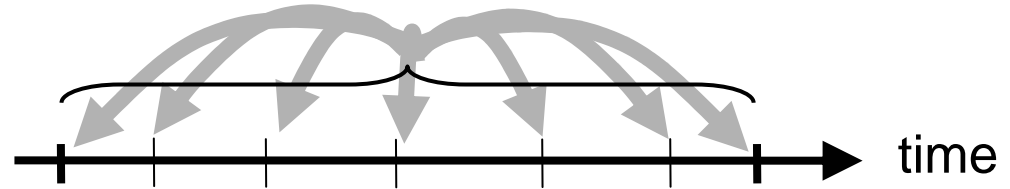
- The RTS smoother recursively carries **information of formally future observations** backwards in time.
- Utilizes **all data within a fixed time-interval** whereas the Kalman filter only utilizes data up to that instant.



- **More accurate than the filter's** estimate.

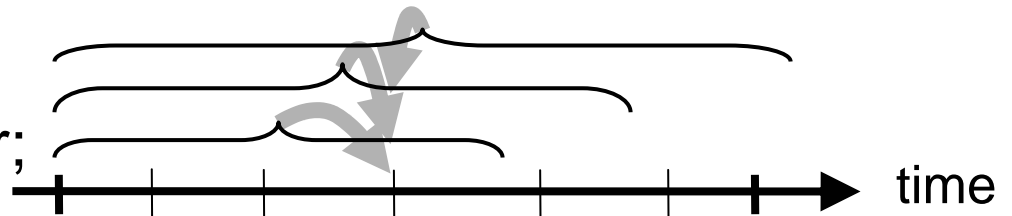
Other Smoothers

- Fixed-Interval Smoother;

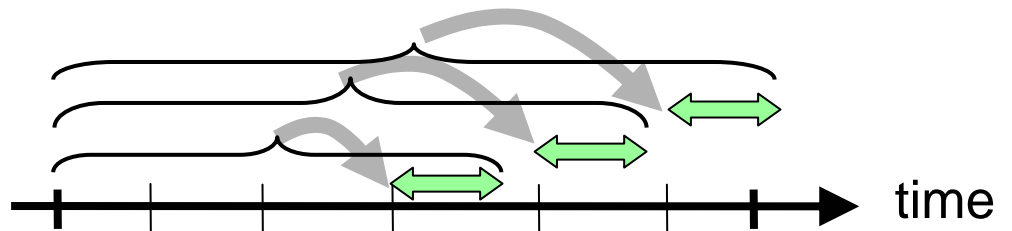


Besides the recursive RTS smoother, the **adjoint method** and **Green's function optimizations** are other commonly used fixed-interval smoothers.

- Fixed-Point Smoother;



- Fixed-Lag Smoother;



What are we really solving? How can we solve it?

OUTLINE

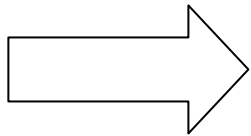
1. What is assimilation really solving?
 - a) Filtering vs Smoothing
 - b) Observability & Controllability
 - c) Data Error and Model Error

2. Examples and Practical Issues; e.g., ECCO
 - a) A hierarchical approach
 - b) Practical derivation and implementation
 - c) Assessment

Observability & Controllability

Kalman Filter is an inversion of observation \mathbf{H} .

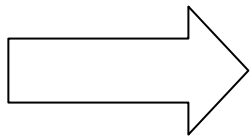
RTS smoother is an inversion of model evolution \mathbf{A} and \mathbf{G} .



Identifies **the problem** the filter & smoother are solving (inverting).

Observability identifies if the model state that can be determined.

Controllability identifies if controls can be determined.



Identifies **the solution** that could be determined.

Observability

If there are no errors in the observations and no errors in the model (except state is unknown), **can we determine the model state uniquely** ?

$$\mathbf{y}_1 = \mathbf{H}_1 \mathbf{x}_1 = \mathbf{H}_1 (\mathbf{A} \mathbf{x}_0 - \mathbf{G} \mathbf{u}_0) \longrightarrow \mathbf{y}_1 + \mathbf{H}_1 \mathbf{G} \mathbf{u}_0 = \mathbf{H}_1 \mathbf{A} \mathbf{x}_0$$

⋮

$$\mathbf{y}_t + \mathbf{H}_t \mathbf{G} \mathbf{u}_{t-1} + \mathbf{H}_t \mathbf{A} \mathbf{G} \mathbf{u}_{t-2} + \cdots + \mathbf{H}_t \mathbf{A}^{t-1} \mathbf{G} \mathbf{u}_0 \equiv \hat{\mathbf{y}}_t = \mathbf{H}_t \mathbf{A}^t \mathbf{x}_0$$

$$\longrightarrow \begin{pmatrix} \hat{\mathbf{y}}_1 \\ \vdots \\ \hat{\mathbf{y}}_t \end{pmatrix} = \mathbf{O} \mathbf{x}_1 \quad \text{where} \quad \mathbf{O} \equiv \begin{pmatrix} \mathbf{H}_1 \mathbf{A} \\ \vdots \\ \mathbf{H}_t \mathbf{A}^t \end{pmatrix}$$

If \mathbf{O} is full rank (column), there is a unique solution for the state, and the system is said to be observable.

Controllability

Can we determine controls that can drive the model to an arbitrary state?

$$\mathbf{x}_1 = \mathbf{A}\mathbf{x}_0 + \mathbf{G}\mathbf{u}_0$$

$$\mathbf{x}_2 = \mathbf{A}\mathbf{x}_1 + \mathbf{G}\mathbf{u}_1 = \mathbf{A}^2\mathbf{x}_0 + \mathbf{A}\mathbf{G}\mathbf{u}_0 + \mathbf{G}\mathbf{u}_1$$

\vdots

$$\mathbf{x}_t = \mathbf{A}\mathbf{x}_{t-1} + \mathbf{G}\mathbf{u}_{t-1} = \mathbf{A}^t\mathbf{x}_0 + \mathbf{A}^{t-1}\mathbf{G}\mathbf{u}_0 + \mathbf{A}^{t-2}\mathbf{G}\mathbf{u}_1 \cdots + \mathbf{G}\mathbf{u}_{t-1}$$

$$\longrightarrow \mathbf{x}_t - \mathbf{A}^t\mathbf{x}_0 = \mathbf{C} \begin{pmatrix} \mathbf{u}_0 \\ \mathbf{u}_1 \\ \vdots \\ \mathbf{u}_{t-1} \end{pmatrix} \quad \text{where} \quad \mathbf{C} \equiv (\mathbf{G} \quad \mathbf{A}\mathbf{G} \quad \cdots \quad \mathbf{A}^{t-1}\mathbf{G})$$

If \mathbf{C} is full rank (row), there is a solution for the control that satisfies this equation, and the system is said to be controllable.

Observability & Controllability

These conditions are **conceptually important**. But the physical state being observable does not mean any meaningful estimate can be made; i.e., Difference between \mathbf{P}^a and \mathbf{P}^f may be indistinguishable.

What are we really solving? How can we solve it?

OUTLINE

1. What is assimilation really solving?
 - a) Filtering vs Smoothing
 - b) Observability & Controllability
 - c) Data Error and Model Error

2. Examples and Practical Issues; e.g., ECCO
 - a) A hierarchical approach
 - b) Practical derivation and implementation
 - c) Assessment

What is assimilation really solving?

“Data Error” and “Model Error”

1. The two are better thought of as **observation constraint error** and **model constraint error**, respectively, as opposed to errors of the data and model, per se.
2. “Data Error” includes **representation errors**, i.e., things that the models do not represent,
3. **How to specify** the two prior errors.

“Data Error” & “Model Error”

$$\begin{pmatrix} \vdots \\ \mathbf{H}\mathbf{x}_t \\ \mathbf{H}\mathbf{x}_{t+1} \\ \vdots \\ \mathbf{x}_{t+1} - \mathbf{A}\mathbf{x}_t - \mathbf{G}\mathbf{u}_t \\ \mathbf{x}_{t+2} - \mathbf{A}\mathbf{x}_{t+1} - \mathbf{G}\mathbf{u}_{t+1} \\ \vdots \end{pmatrix} \approx \begin{pmatrix} \vdots \\ \mathbf{y}_t \\ \mathbf{y}_{t+1} \\ \vdots \\ \mathbf{0} \\ \mathbf{0} \\ \vdots \end{pmatrix}$$

What are error covariance matrices **R** and **Q**?

These two in effect define the inverse solution, and therefore, their understanding and proper specification are fundamental to solving the assimilation problem.

Kalman filter estimate :

“forecast” $\rightarrow \hat{\mathbf{x}}_t^f = \mathbf{A}\hat{\mathbf{x}}_{t-1}^a + \mathbf{G}\mathbf{u}_{t-1}$

“analysis” $\rightarrow \hat{\mathbf{x}}_t^a = \hat{\mathbf{x}}_t^f + \mathbf{P}_t^f \mathbf{H}^T \left(\mathbf{H}\mathbf{P}_t^f \mathbf{H}^T + \mathbf{R}_t \right)^{-1} \left(\mathbf{y}_t - \mathbf{H}\hat{\mathbf{x}}_t^f \right)$

$$\mathbf{P}_t^f = \mathbf{A}\hat{\mathbf{P}}_{t-1}^a \mathbf{A}^T + \mathbf{G}\mathbf{Q}\mathbf{G}^T$$

$$\hat{\mathbf{P}}_t^a = \mathbf{P}_t^f - \mathbf{P}_t^f \mathbf{H}^T \left(\mathbf{H}\mathbf{P}_t^f \mathbf{H}^T + \mathbf{R}_t \right)^{-1} \mathbf{H}\mathbf{P}_t^f$$

“Data Error”

$$\mathbf{E}\mathbf{w}_t - \mathbf{H}\mathbf{\Pi}\mathbf{w}_t : \text{Representation error}$$

Error-free
data

Error-free model
equivalent of data

$\mathbf{\Pi}$: Defines the model state.
 \mathbf{E} : Defines observation.
 \mathbf{H} : Defines model equivalent
of data

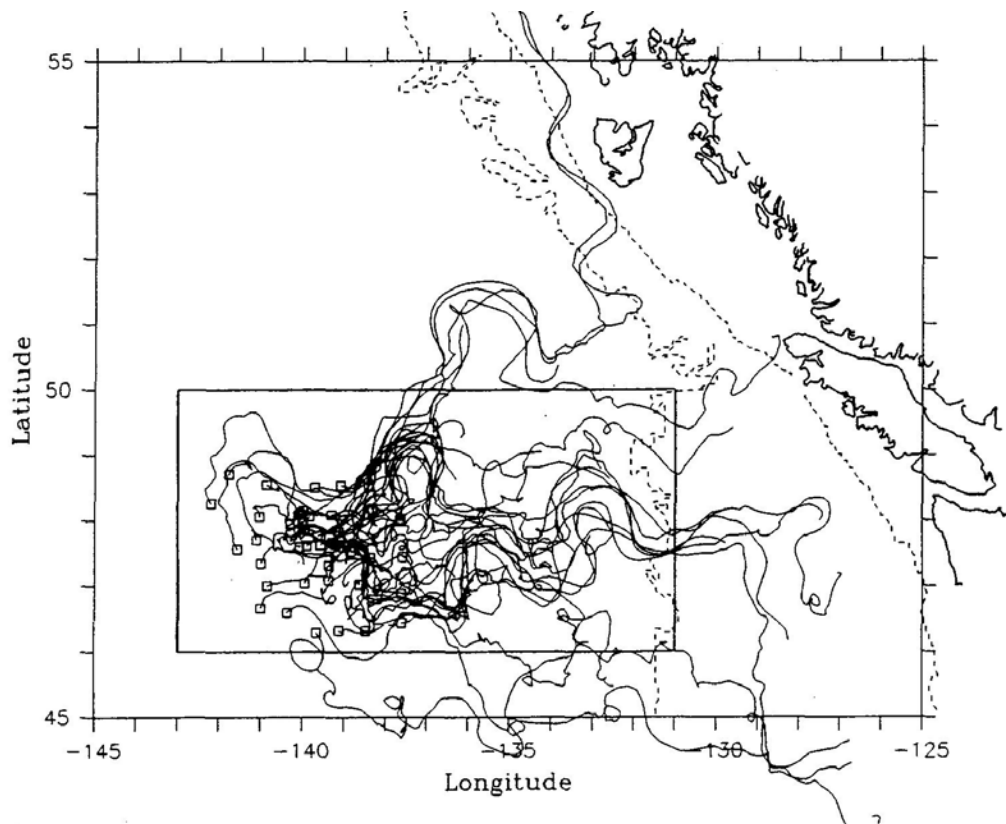
Examples;

- **Meso-scale variability** for a non-eddy resolving model,
- External **tides** for a rigid lid model,
- **Skin temperature** for model with thick surface layer,
- **Micro-structure** for most models,
- **Barotropic** variability for a reduced-gravity model,
- **Baroclinic** variability for a barotropic model,
- Meteorologist do **not** force models **too close to observations** in numerical weather forecasting,
- Individual float & drifter **trajectories** for any finite difference model.

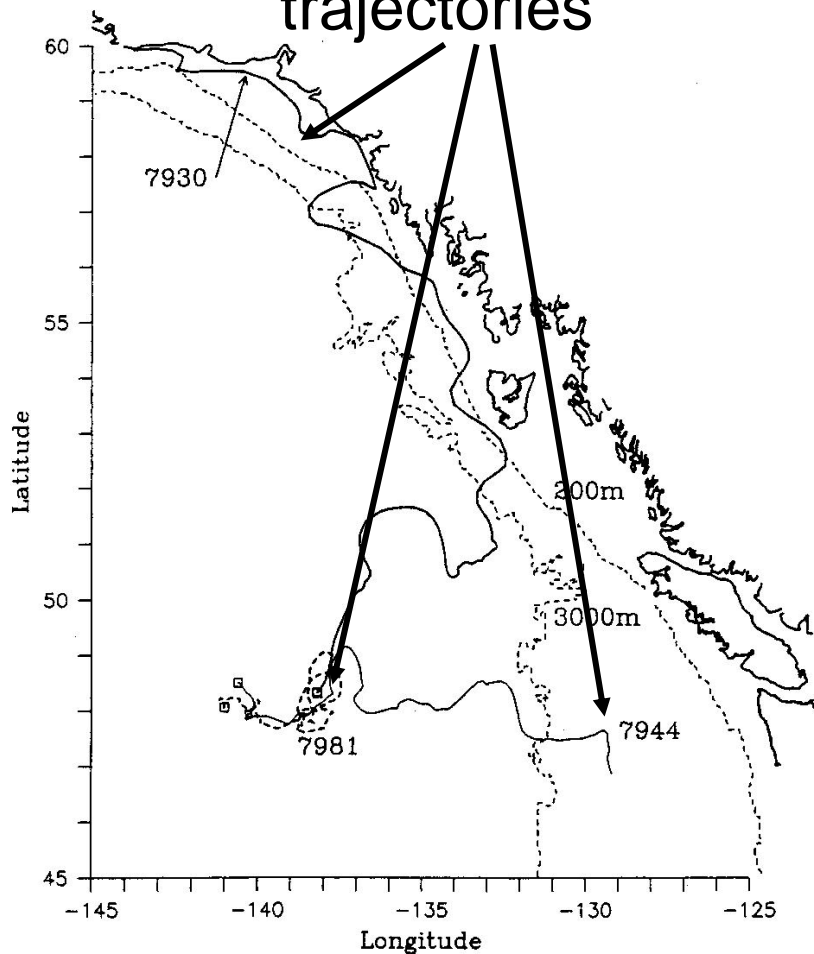
Examples of Drifter Trajectory

(Paduan and Niiler, 1993, JPO)

Typical “spaghetti” diagram



Example of diverging float trajectories



“Model Error”

Q represents the error of the model constraint.

Model evolution: $\mathbf{x}_{t+1} = A(\mathbf{x}_t, \hat{\mathbf{u}}_t^0)$ Forcing, boundary condition, mixing parameters, model error sources.
 Ocean evolution: $\mathbf{w}_{t+1} = L(\mathbf{w}_t, \mathbf{v}_t)$ Real forcing & boundary condition.

Functions describing the evolution

The true model evolution can be written as,

$$\bar{\mathbf{x}}_{t+1} = \mathbf{\Pi} \mathbf{w}_{t+1} = \mathbf{\Pi} L(\mathbf{w}_t, \mathbf{v}_t)$$

$$\bar{\mathbf{x}}_{t+1} = A(\bar{\mathbf{x}}_t, \hat{\mathbf{u}}_t^0) + \underbrace{\left\{ \mathbf{\Pi} L(\mathbf{w}_t, \mathbf{v}_t) - A(\mathbf{\Pi} \mathbf{w}_t, \hat{\mathbf{u}}_t^0) \right\}}$$

Error of the model evolution; $\mathbf{x}_{t+1} = A(\mathbf{x}_t, \hat{\mathbf{u}}_t^0)$

$$\mathbf{\Pi} L(\mathbf{w}_t, \mathbf{v}_t) - A(\mathbf{\Pi} \mathbf{w}_t, \hat{\mathbf{u}}_t^0) : \text{Model Process Noise}$$

True evolution

Model evolution given true model state

“Model Error”

$$\mathbf{\Pi}L(\mathbf{w}_t, \mathbf{v}_t) - A(\mathbf{\Pi}\mathbf{w}_t, \hat{\mathbf{u}}_t^0) : \text{Model Process Noise}$$

Examples;

- 1) Differences in $\hat{\mathbf{u}}_t^0$ and \mathbf{v} (errors in non-state variables); *forcing, boundary condition, model parameters,*
- 2) Errors due to differences in A and L (errors in model algorithm); *finite difference, interaction with scales and processes ignored*

$$\begin{aligned}\bar{\mathbf{x}}_{t+1} &= A(\bar{\mathbf{x}}_t, \hat{\mathbf{u}}_t^0) + \left\{ \mathbf{\Pi}L(\mathbf{w}_t, \mathbf{v}_t) - A(\mathbf{\Pi}\mathbf{w}_t, \hat{\mathbf{u}}_t^0) \right\} \\ &= A(\bar{\mathbf{x}}_t, \bar{\mathbf{u}}_t)\end{aligned}$$

Collectively, sources of model process noise are **modeled as elements of the model control vector.**

Prior Error Specification

Covariance Matching (Fu et al., 1993, *JPO*)

Estimate **R** & **Q** from model-data comparison.

Data **y** and its model simulation equivalent **m** could be written as;

$$\mathbf{y} = \mathbf{s} + \mathbf{r}$$

$$\mathbf{m} = \mathbf{s} + \mathbf{n}$$

where, **s**: true signal ($\mathbf{H}\bar{\mathbf{x}}$), **r**: data error,

m: simulation of **y** ($\mathbf{H}\mathbf{x}_{sim}$),

n: simulation error ($\mathbf{H}\mathbf{x}_{sim} - \mathbf{H}\bar{\mathbf{x}}$)

Then,

$$\begin{aligned}\langle \mathbf{y}\mathbf{y}^T \rangle &= \langle \mathbf{s}\mathbf{s}^T \rangle + \langle \mathbf{s}\mathbf{r}^T \rangle + \langle \mathbf{r}\mathbf{s}^T \rangle + \langle \mathbf{r}\mathbf{r}^T \rangle \\ \langle \mathbf{m}\mathbf{m}^T \rangle &= \langle \mathbf{s}\mathbf{s}^T \rangle + \langle \mathbf{s}\mathbf{n}^T \rangle + \langle \mathbf{n}\mathbf{s}^T \rangle + \langle \mathbf{n}\mathbf{n}^T \rangle \\ \langle \mathbf{m}\mathbf{y}^T \rangle &= \langle \mathbf{s}\mathbf{s}^T \rangle + \langle \mathbf{s}\mathbf{r}^T \rangle + \langle \mathbf{n}\mathbf{s}^T \rangle + \langle \mathbf{n}\mathbf{r}^T \rangle\end{aligned}$$



Prior Error Specification

If, $\langle \mathbf{sn}^T \rangle \approx 0$ etc,

$$\mathbf{R} = \langle \mathbf{rr}^T \rangle = \langle \mathbf{yy}^T \rangle - \langle \mathbf{my}^T \rangle \quad \leftarrow \mathbf{R} \text{ estimate.}$$

$$\mathbf{HP}_{sim} \mathbf{H}^T = \langle \mathbf{nn}^T \rangle = \langle \mathbf{mm}^T \rangle - \langle \mathbf{my}^T \rangle \quad \leftarrow \text{Implicit } \mathbf{Q} \text{ estimate.}$$

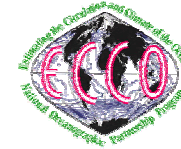
Adjust process noise \mathbf{Q} to match the solution of the “Lyapunov Equation” to that of the $\langle \mathbf{nn}^T \rangle$ estimate.

$$\text{Solve } \mathbf{P} \text{ as } t \rightarrow \infty \quad \mathbf{P}_t = \mathbf{A}\mathbf{P}_{t-1}\mathbf{A}^T + \mathbf{G}\mathbf{Q}\mathbf{G}^T$$

$$\text{Then choose } \mathbf{Q} \text{ such that } \mathbf{HPH}^T \approx \langle \mathbf{nn}^T \rangle$$

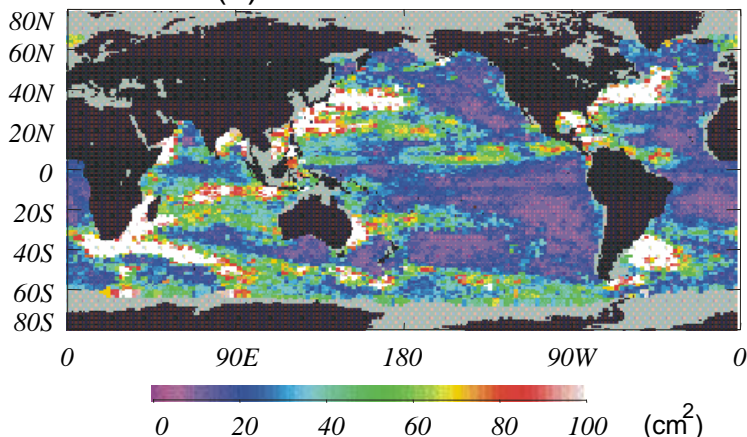
Prior Error Specification

Error variance estimate (Fukumori et al., 1999, *JGR*)



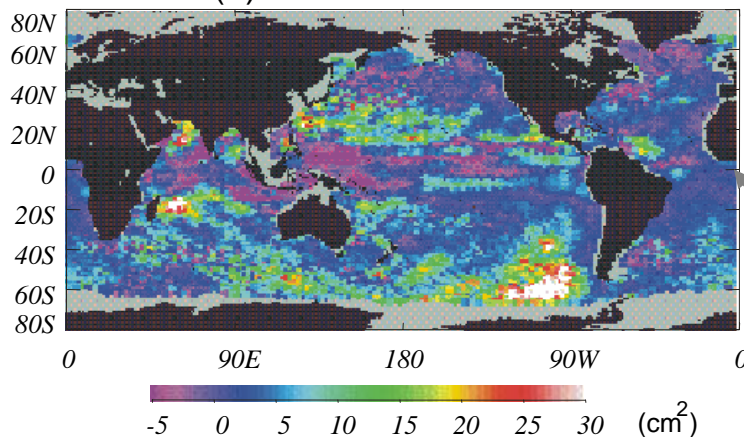
$$\mathbf{R} = \langle \mathbf{r}\mathbf{r}^T \rangle = \langle \mathbf{y}\mathbf{y}^T \rangle - \langle \mathbf{m}\mathbf{y}^T \rangle$$

(a) Data Constraint Error



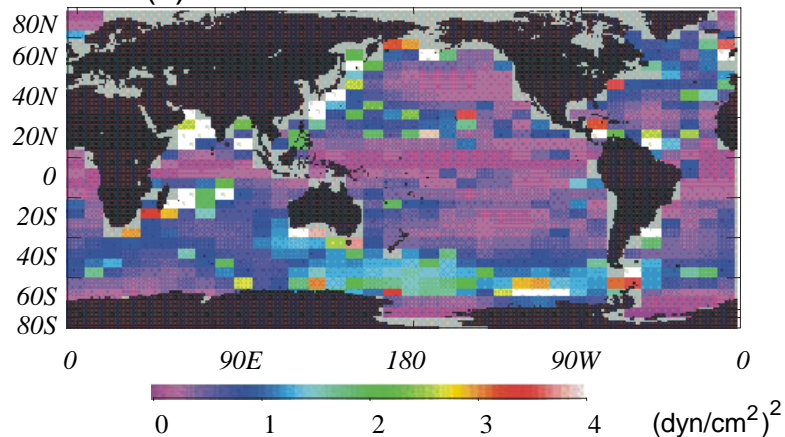
$$\mathbf{HP}_{sim} \mathbf{H}^T = \langle \mathbf{n}\mathbf{n}^T \rangle = \langle \mathbf{m}\mathbf{m}^T \rangle - \langle \mathbf{m}\mathbf{y}^T \rangle$$

(b) Model Simulation Error



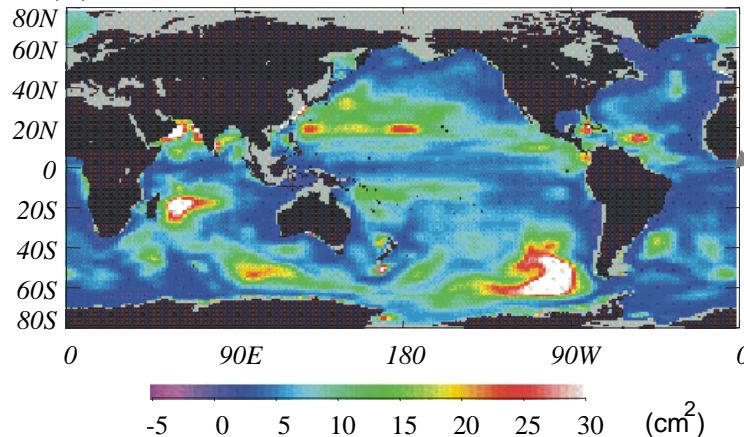
\mathbf{Q}

(c) Calibrated Model Process Noise



$\mathbf{HP}_{sim} \mathbf{H}^T$

(d) Calibrated Model Simulation Error Estimate



1st order match

What are we really solving? How can we solve it?

OUTLINE

1. What is assimilation really solving?
 - a) Filtering vs Smoothing
 - b) Observability & Controllability
 - c) Data Error and Model Error

2. Examples and Practical Issues; e.g., ECCO
 - a) A hierarchical approach
 - b) Practical derivation and implementation
 - c) Assessment



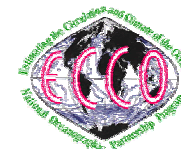
“Estimating the Circulation and Climate of the Ocean”

<http://www.ecco-group.org>

GOAL: Advance data assimilation into an operational tool to help understand ocean circulation.

- 3 groups: JPL, Scripps Institution of Oceanography, Massachusetts Institute of Technology.
- Employ as much observations as possible.
- State-of-the-art ocean general circulation model.
- Advanced assimilation (**Kalman filter/smoothen, adjoint, Green's function**) characterized by its consistency.

ECCO Near Real-Time Analysis

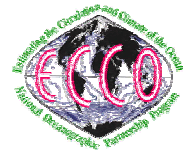


- Nowcasts every **10-days**.
- Highlights @ <http://ecco.jpl.nasa.gov/external>
- **SSH**, Temperature profiles & time-mean sea level.
- Controls: **winds**, heat & fresh water fluxes and mixing parameters.

- Green's function, **Kalman filter & RTS Smoother**, Adjoint method)

- LAS server from **1993 to present**: <http://www.ecco-group.org/las>

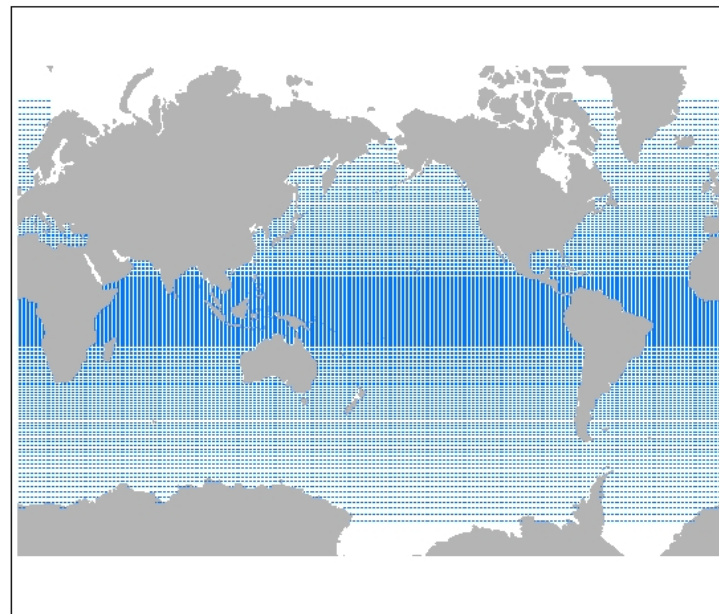
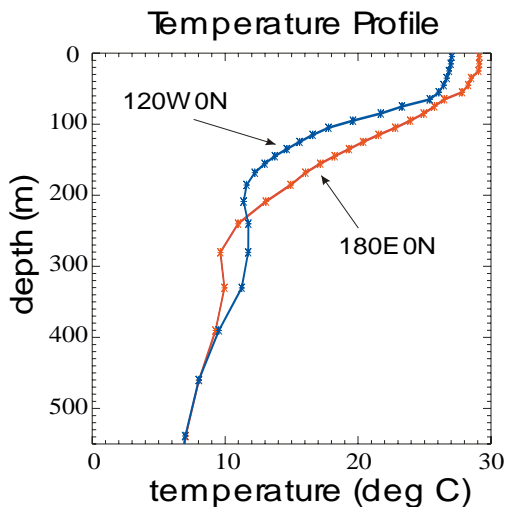
ECCO Routine Assimilation



MIT General Circulation Model (MITgcm)

(Marshall et al., 1997, *JGR*)

- Nonlinear primitive equation model
- Adaptable to massively parallel supercomputers
- Advanced physics (e.g., KPP & GM mixing)
- Global, high resolution
 - ✓ $80^{\circ}\text{S} \sim 80^{\circ}\text{N}$
 - ✓ $0.3^{\circ} \times 10\text{m}$ in tropics



Approximate Kalman filter & Smoother

Kalman filter estimate :

$$\hat{\mathbf{x}}_t^f = \mathbf{A}\hat{\mathbf{x}}_{t-1}^a + \mathbf{G}\mathbf{u}_{t-1}$$

$$\hat{\mathbf{x}}_t^a = \hat{\mathbf{x}}_t^f + \mathbf{P}_t^f \mathbf{H}^T \left(\mathbf{H}\mathbf{P}_t^f \mathbf{H}^T + \mathbf{R}_t \right)^{-1} \left(\mathbf{y}_t - \mathbf{H}\hat{\mathbf{x}}_t^f \right)$$

$$\mathbf{P}_t^f = \mathbf{A}\hat{\mathbf{P}}_{t-1}^a \mathbf{A}^T + \mathbf{G}\mathbf{Q}\mathbf{G}^T$$

$$\hat{\mathbf{P}}_t^a = \mathbf{P}_t^f - \mathbf{P}_t^f \mathbf{H}^T \left(\mathbf{H}\mathbf{P}_t^f \mathbf{H}^T + \mathbf{R}_t \right)^{-1} \mathbf{H}\mathbf{P}_t^f$$

The explicit evaluation of \mathbf{P} (“Riccati Equation”) is impractical because of their large dimension and time-evolution.

ECCO employs **three approximations** of \mathbf{P} ;

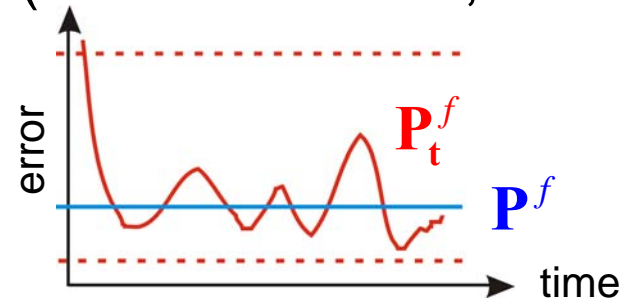
1. Time-asymptotic
2. Reduced-state
3. Partitioned



ECCO Approximate KFS

1) **Time-asymptotic** approximation (Fukumori et al., *JPO*, 1993);

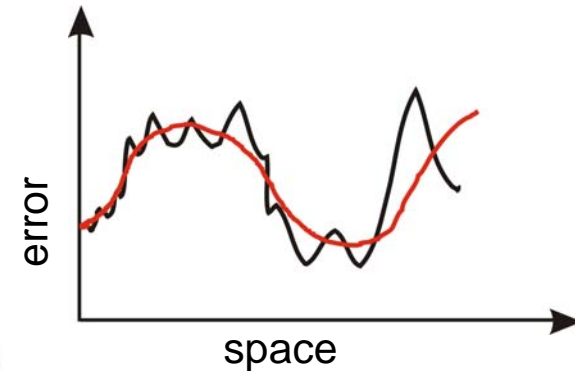
$$\mathbf{P}_t^f \approx \mathbf{P}^f$$



2) **State reduction** (Fukumori and Rizzoli, *JGR*, 1995).

$$\delta \mathbf{x} \approx \mathbf{B} \delta \mathbf{x}', \quad \dim(\delta \mathbf{x}) \square \dim(\delta \mathbf{x}')$$

$$\mathbf{P} \equiv \langle \delta \mathbf{x} \delta \mathbf{x}^T \rangle \approx \mathbf{B} \langle \delta \mathbf{x}' \delta \mathbf{x}'^T \rangle \mathbf{B}^T \approx \mathbf{B} \mathbf{P}' \mathbf{B}^T$$

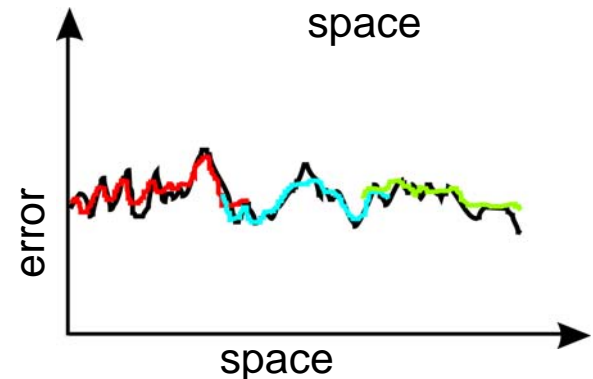


3) **Partitioning** (Fukumori, *MWR*, 2002).

$$\delta \mathbf{x} \approx \mathbf{B}_1 \delta \mathbf{x}'_1 + \dots + \mathbf{B}_L \delta \mathbf{x}'_L \approx \sum_i^L \mathbf{B}_i \delta \mathbf{x}'_i$$

$$\dim(\delta \mathbf{x}) \square \dim(\delta \mathbf{x}'_i)$$

$$\mathbf{P} \approx \sum_i^L \mathbf{B}_i \mathbf{P}'_i \mathbf{B}_i^T, \quad \mathbf{P}'_i \equiv \langle \delta \mathbf{x}'_i \delta \mathbf{x}'_i^T \rangle$$



What are we really solving? How can we solve it?

OUTLINE

1. What is assimilation really solving?
 - a) Filtering vs Smoothing
 - b) Observability & Controllability
 - c) Data Error and Model Error

2. Examples and Practical Issues; e.g., ECCO
 - a) A hierarchical approach
 - b) Practical derivation and implementation
 - c) Assessment

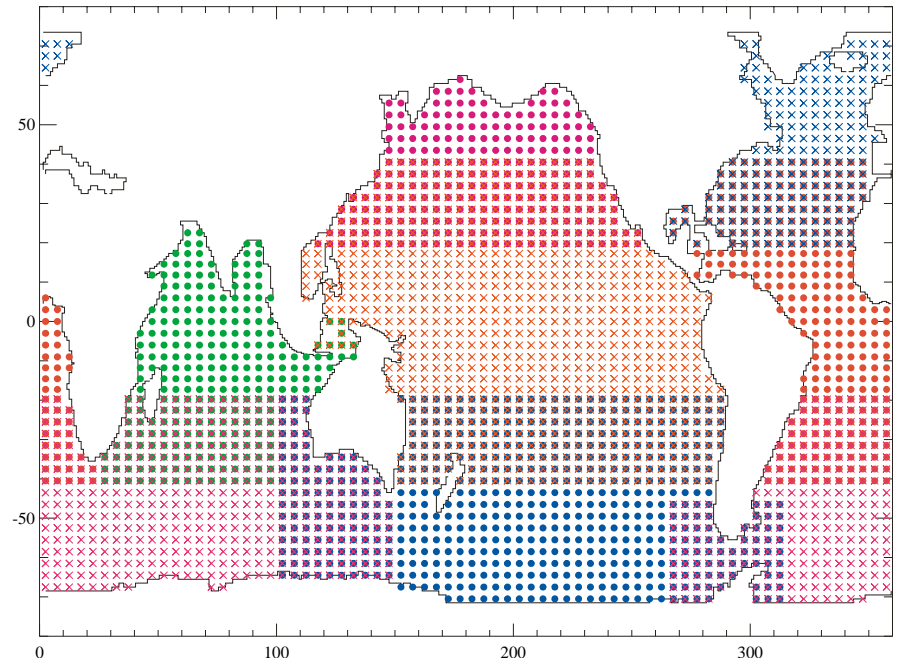
ECCO Approximate KFS

1. Errors due to wind error (process noise) are estimated.

2. Partition global domain into 8 different regions (basins)

- Global barotropic cell
- 7 Regional baroclinic cells

- I. Tropical Indian ($>40^{\circ}\text{S}$)
- II. Tropical Pacific ($40^{\circ}\text{S} \sim 40^{\circ}\text{N}$)
- III. Tropical Atlantic ($40^{\circ}\text{S} \sim 40^{\circ}\text{N}$)
- IV. South Pacific ($<20^{\circ}\text{S}$)
- V. South Atlantic-Indian ($<20^{\circ}\text{S}$)
- VI. North Pacific ($>20^{\circ}\text{N}$)
- VII. North Atlantic ($>20^{\circ}\text{N}$)



ECCO Approximate KFS

3. State and control reduction: Horizontal velocity and vertical displacement in terms of vertical dynamic modes on a coarse horizontal grid. The reduced control is wind error on the same coarse grid.

Vertical projection (dynamic modes) Horizontal interpolation (objective mapping) Reduced state (amplitude of dynamic mode on coarse grid)

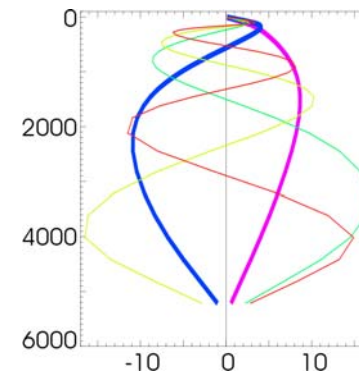
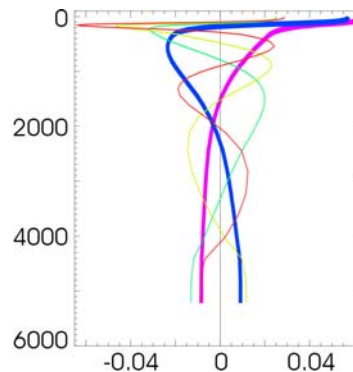
$$\delta \mathbf{u} = \mathbf{D}_{vel} \mathbf{O} \mathbf{a}_u, \quad \delta \mathbf{v} = \mathbf{D}_{vel} \mathbf{O} \mathbf{a}_v, \quad \delta \boldsymbol{\eta} = \mathbf{D}_{disp} \mathbf{O} \mathbf{a}_{disp}, \quad \delta \boldsymbol{\tau} = \mathbf{O} \mathbf{a}_\tau$$

$$\delta \mathbf{T} = \frac{\partial \mathbf{T}}{\partial z} \delta \boldsymbol{\eta}, \quad \delta \mathbf{S} = \frac{\partial \mathbf{S}}{\partial z} \delta \boldsymbol{\eta}, \quad \delta \mathbf{h} = F(\delta \boldsymbol{\eta}, \mathbf{T}, \mathbf{S})$$

3a. Vertical projection: \mathbf{D}_{vel} , \mathbf{D}_{disp}

$$\frac{d}{dz} \left(\frac{1}{N^2} \frac{dp}{dz} \right) + \frac{1}{c^2} p = 0$$

$$\frac{d^2 \eta}{dz^2} + \left(\frac{N}{c} \right)^2 \eta = 0$$



ECCO Approximate KFS

3b. Horizontal Interpolation: ○

“Objective Mapping” (Bretherton et al., 1976, *DSR*)

Given measurements φ , the underlying field θ is estimated by

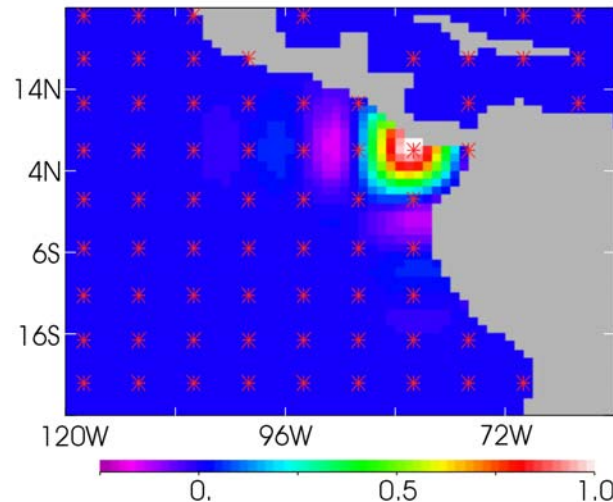
$$\theta = \mathbf{C}\mathbf{A}^{-1}\varphi \quad \text{where} \quad \mathbf{C} = \langle \theta \varphi^T \rangle \quad \mathbf{A} = \langle \varphi \varphi^T \rangle$$

This is just another
least-squares

$$\hat{\mathbf{a}} = \mathbf{a}_0 + \mathbf{R}_{aa} \mathbf{E}^T (\mathbf{E} \mathbf{R}_{aa} \mathbf{E}^T + \mathbf{R}_{bb})^{-1} (\mathbf{b} - \mathbf{E} \mathbf{a}_0)$$

Horizontal interpolation, ○, is defined as objective mapping from a coarse grid system to the model grid system.

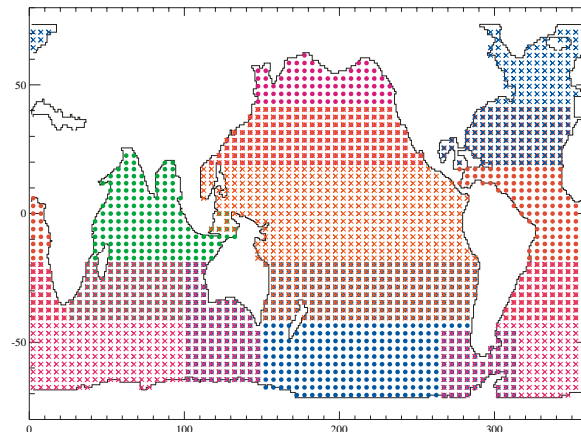
Example of mapping of a perturbation at one coarse grid point;



ECCO Approximate KFS

- **Global barotropic cell:** global $6^{\circ} \times 6^{\circ}$ grid, barotropic UVH (966 \times 3 = 2898 variables)
- **Regional baroclinic cells:** regional $5^{\circ} \times 3^{\circ}$ grid, baroclinic UVD (5 gravest modes)
 - I. Tropical Indian ($>40^{\circ}\text{S}$; 308 \times 3 \times 5 = 4620)
 - II. Tropical Pacific ($40^{\circ}\text{S} \sim 40^{\circ}\text{N}$; 787 \times 3 \times 5 = **11805**)
 - III. Tropical Atlantic ($40^{\circ}\text{S} \sim 40^{\circ}\text{N}$; 350 \times 3 \times 5 = 5250)
 - IV. South Pacific ($<20^{\circ}\text{S}$; 633 \times 3 \times 5 = 9495)
 - V. South Atlantic-Indian ($<20^{\circ}\text{S}$; 664 \times 3 \times 5 = 9960)
 - VI. North Pacific ($>20^{\circ}\text{N}$; 271 \times 3 \times 5 = 4065)
 - VII. North Atlantic ($>20^{\circ}\text{N}$; 198 \times 3 \times 5 = 2970)

In comparison, the model's state dimension is **8 million** (2 million grid points).



ECCO Approximate KFS

4. Compute P by formulating and explicitly deriving partitioned reduced-state models.

4a. Formulate partitioned reduced-state models.

The reduced-state must form a closed system.

A state and control perturbation $\delta \mathbf{x}$, $\delta \mathbf{u}$ can be written as

$\delta \mathbf{x} = \mathbf{B}\mathbf{x}' + \mathbf{N}\mathbf{n}$: \mathbf{B} , \mathbf{N} : range & null space of reduced-state

$\delta \mathbf{u} = \hat{\mathbf{B}}\mathbf{u}' + \hat{\mathbf{N}}\mathbf{m}$: The control equivalent of state reduction.

Given the model $\mathbf{x}_{t+1} = \mathfrak{F}(\mathbf{x}_t, \mathbf{u}_t)$ the perturbation

satisfies $\delta \mathbf{x}_{t+1} = \mathfrak{F}(\tilde{\mathbf{x}} + \delta \mathbf{x}_t, \tilde{\mathbf{u}} + \delta \mathbf{u}_t) - \mathfrak{F}(\tilde{\mathbf{x}}, \tilde{\mathbf{u}})$

or $\mathbf{B}\mathbf{x}'_{t+1} + \mathbf{N}\mathbf{n}_{t+1} = \mathfrak{F}(\tilde{\mathbf{x}} + \mathbf{B}\mathbf{x}'_t + \mathbf{N}\mathbf{n}_t, \tilde{\mathbf{u}} + \hat{\mathbf{B}}\mathbf{u}'_t + \hat{\mathbf{N}}\mathbf{m}_t) - \mathfrak{F}(\tilde{\mathbf{x}}, \tilde{\mathbf{u}})$



ECCO Approximate KFS

$$\mathbf{B}\mathbf{x}'_{t+1} + \mathbf{N}\mathbf{n}_{t+1} = \mathfrak{F}\left(\tilde{\mathbf{x}} + \mathbf{B}\mathbf{x}'_t + \mathbf{N}\mathbf{n}_t, \tilde{\mathbf{u}} + \hat{\mathbf{B}}\mathbf{u}'_t + \hat{\mathbf{N}}\mathbf{m}_t\right) - \mathfrak{F}(\tilde{\mathbf{x}}, \tilde{\mathbf{u}})$$

Multiplying by the pseudo inverse of \mathbf{B} , \mathbf{B}^* ($\mathbf{B}^*\mathbf{B} = \mathbf{I}$, $\mathbf{B}^*\mathbf{N} = 0$)

$$\mathbf{x}'_{t+1} = \mathbf{B}^* \left(\mathfrak{F}\left(\tilde{\mathbf{x}} + \mathbf{B}\mathbf{x}'_t + \mathbf{N}\mathbf{n}_t, \tilde{\mathbf{u}} + \hat{\mathbf{B}}\mathbf{u}'_t + \hat{\mathbf{N}}\mathbf{m}_t\right) - \mathfrak{F}(\tilde{\mathbf{x}}, \tilde{\mathbf{u}}) \right)$$

For this to be a model for the reduced-state we cannot have dependency on the null space, $\mathbf{N}\mathbf{n}_t, \hat{\mathbf{N}}\mathbf{m}_t$;

$$\mathbf{x}'_{t+1} = \mathbf{B}^* \left(\mathfrak{F}\left(\tilde{\mathbf{x}} + \mathbf{B}\mathbf{x}'_t, \tilde{\mathbf{u}} + \hat{\mathbf{B}}\mathbf{u}'_t\right) - \mathfrak{F}(\tilde{\mathbf{x}}, \tilde{\mathbf{u}}) \right)$$

This would be achieved if null (range) space perturbations remain in the null (range) space, because $\mathbf{B}^*\mathbf{N} = 0$; i.e., state reduction must form a closed-system for it to be effective.



ECCO Approximate KFS

4b. Derive partitioned reduced-state model in explicit matrix form to facilitate error evaluation.

$$\mathbf{x}'_{t+1} = \mathbf{B}^* \left(\mathfrak{J}(\tilde{\mathbf{x}} + \mathbf{B}\mathbf{x}'_t, \tilde{\mathbf{u}} + \hat{\mathbf{B}}\mathbf{u}'_t) - \mathfrak{J}(\tilde{\mathbf{x}}, \tilde{\mathbf{u}}) \right)$$



$$\mathbf{x}'_{t+1} = \mathbf{A}'\mathbf{x}'_t + \mathbf{G}'\mathbf{u}'_t$$

\mathbf{A}' and \mathbf{G}' can be derived as **coarse grain Green's functions**;

$$(\mathbf{A}')_i = \mathbf{A}'\mathbf{e}_i + \mathbf{G}'\mathbf{0} = \mathbf{B}^* \left(\mathfrak{J}(\tilde{\mathbf{x}} + \mathbf{B}\mathbf{e}_i, \tilde{\mathbf{u}}) - \mathfrak{J}(\tilde{\mathbf{x}}, \tilde{\mathbf{u}}) \right)$$

$$(\mathbf{G}')_i = \mathbf{A}'\mathbf{0} + \mathbf{G}'\mathbf{e}_i = \mathbf{B}^* \left(\mathfrak{J}(\tilde{\mathbf{x}}, \tilde{\mathbf{u}} + \hat{\mathbf{B}}\mathbf{e}_i) - \mathfrak{J}(\tilde{\mathbf{x}}, \tilde{\mathbf{u}}) \right)$$

Similarly, \mathbf{H}' can be derived by,

$$(\mathbf{H}')_i = \mathbf{H}'\mathbf{e}_i = H(\tilde{\mathbf{x}} + \mathbf{B}\mathbf{e}_i) - H(\tilde{\mathbf{x}})$$

where H is a function defining model equivalent of data.



ECCO Approximate KFS

$$(\mathbf{A}')_i = \mathbf{A}'\mathbf{e}_i + \mathbf{G}'\mathbf{0} = \mathbf{B}^* \left(\mathfrak{I}(\tilde{\mathbf{x}} + \mathbf{B}\mathbf{e}_i, \tilde{\mathbf{u}}) - \mathfrak{I}(\tilde{\mathbf{x}}, \tilde{\mathbf{u}}) \right)$$

Implementing \mathbf{B}^* requires some considerations;

1. \mathbf{O} is sparse but \mathbf{O}^* is not. Thus, we do

$$\mathbf{O}^* = (\mathbf{O}^T \mathbf{O})^{-1} \mathbf{O}^T$$

$(\mathbf{O}^T \mathbf{O})^{-1}$ is small and is, therefore, explicitly computed and stored. \mathbf{O}^T is implemented as the transpose (adjoint) of \mathbf{O} .

2. η is evaluated by **time-integrating vertical velocity** instead of inverting

$$\delta \mathbf{T} = \frac{\partial \mathbf{T}}{\partial \mathbf{z}} \delta \boldsymbol{\eta}, \quad \delta \mathbf{S} = \frac{\partial \mathbf{S}}{\partial \mathbf{z}} \delta \boldsymbol{\eta}$$

which is singular where stratification is weak.



ECCO Approximate KFS

4c. Derive representative time-asymptotic limit of P.

- Stationary solution to the Riccati Equation requires a **stationary system**, i.e., time-invariant \mathbf{A}' , \mathbf{G}' , \mathbf{H}' , \mathbf{Q} , \mathbf{R} .
 - Approximate with a **3-day assimilation cycle**;

$$\mathbf{x}'_{t+1} = \mathbf{A}'\mathbf{x}'_t + \mathbf{G}'\mathbf{u}'_t$$

If $\mathfrak{I}(\tilde{\mathbf{x}}, \tilde{\mathbf{u}})$ is a 1-day integration. Then $\mathbf{A}' = \mathbf{C}'^3$ where

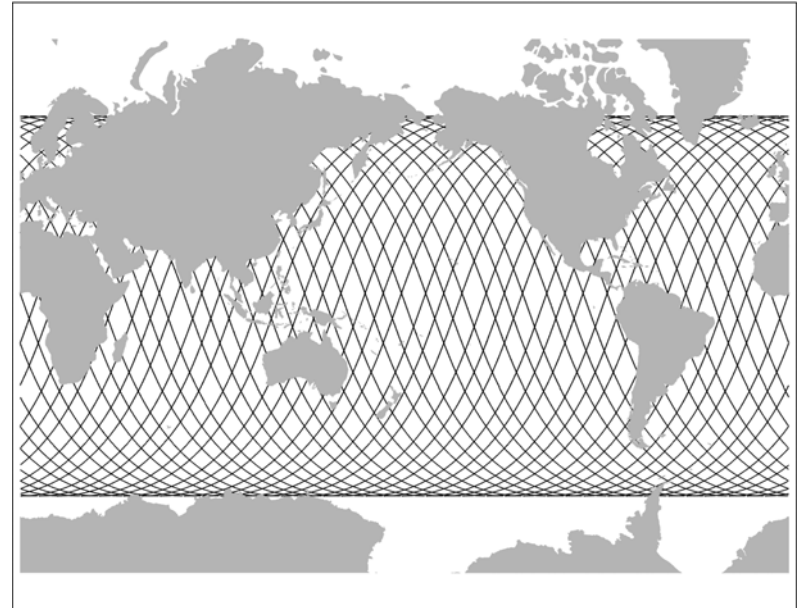
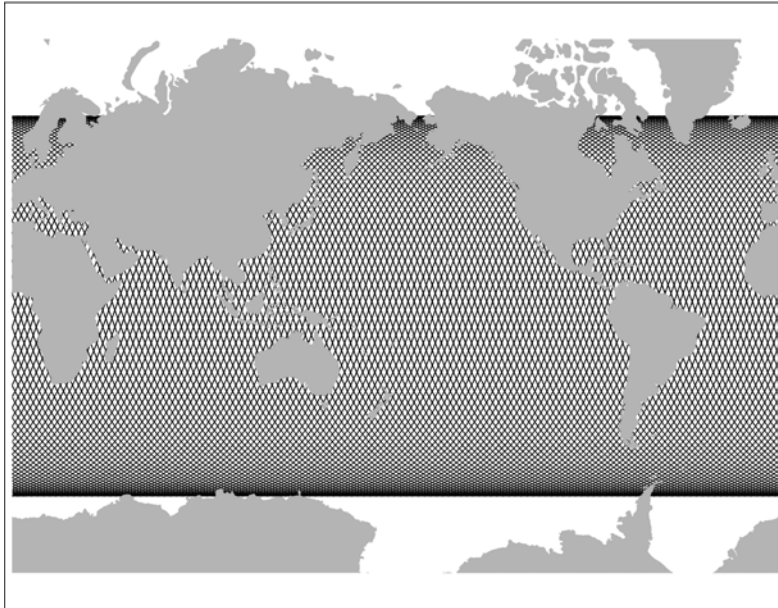
$$(\mathbf{C}')_i = \mathbf{B}^* \left(\mathfrak{I}(\tilde{\mathbf{x}} + \mathbf{B}\mathbf{e}_i, \tilde{\mathbf{u}}) - \mathfrak{I}(\tilde{\mathbf{x}}, \tilde{\mathbf{u}}) \right)$$

- Approximate all observing operation \mathbf{H}' and \mathbf{R} with **representative observations**; e.g.,

Assume a full 10-day cycle of altimeter observations is available every 3-days but with 3.3 times the error variance.

ECCO Approximate KFS

e.g., TOPEX/POSEIDON and Jason have a 3-day subcycle.



ECCO Approximate KFS

5. Implementation

- a) Actual assimilation is conducted **every 6-hours** using all observations taken within ± 3 -hours of the assimilation instant.
- b) Filtering is performed using an **alternate formulation of the Kalman gain**;

K-filter increment $\Delta \hat{\mathbf{x}}_t^a = \mathbf{P}_t^f \mathbf{H}^T \left(\mathbf{H} \mathbf{P}_t^f \mathbf{H}^T + \mathbf{R}_t \right)^{-1} \left(\mathbf{y}_t - H \left(\hat{\mathbf{x}}_t^f \right) \right)$

filter gain

$$\mathbf{P}_t^f \mathbf{H}^T \left(\mathbf{H} \mathbf{P}_t^f \mathbf{H}^T + \mathbf{R}_t \right)^{-1} = \mathbf{P}_t^a \mathbf{H}^T \mathbf{R}_t^{-1}$$

This form minimizes computations involving \mathbf{P} and simplifies the matrix inversion that is needed.



ECCO Approximate KFS

K-filter increment $\Delta \hat{\mathbf{x}}_t^a = \mathbf{P}_t^a \mathbf{H}_t^T \mathbf{R}_t^{-1} \left(\mathbf{y}_t - H \left(\hat{\mathbf{x}}_t^f \right) \right)$

c) The K-filter increment is **sum of partitioned increments**;

$$\Delta \hat{\mathbf{x}}_t^a = \sum_i \mathbf{B}_i \Delta \hat{\mathbf{x}}_{t,i}^{\prime a}$$

where $\Delta \hat{\mathbf{x}}_{t,i}^{\prime a} = \mathbf{P}_i^{\prime a} \mathbf{H}_{t,i}^{\prime T} \mathbf{R}_t^{-1} \left(\mathbf{y}_t - H \left(\hat{\mathbf{x}}_t^f \right) \right)$

Or, $\Delta \hat{\mathbf{x}}_{t,i}^{\prime a} = \mathbf{P}_i^{\prime a} \mathbf{B}^T \mathbf{H}_{t,i}^{\prime T} \mathbf{R}_t^{-1} \left(\mathbf{y}_t - H \left(\hat{\mathbf{x}}_t^f \right) \right)$

using the adjoint of H in place of \mathbf{H}^T .

These calculations are performed from **right to left**;

There is no explicit computation of product $\mathbf{P}_i^{\prime a} \mathbf{H}_{t,i}^{\prime T} \mathbf{R}_t^{-1}$ because it is computationally inefficient.



ECCO Approximate KFS

- d) **Anomalies are assimilated** for sea level and temperature profiles because of
- Unknown marine geoid,
 - Temporally uncorrelated process noise assumption.

$$\Delta \hat{\mathbf{x}}_t'^a = \mathbf{P}_i'^a \mathbf{H}'_{t,i}{}^T \mathbf{R}_t^{-1} \left((\mathbf{y}_t - \bar{\bar{\mathbf{y}}}) - (H(\hat{\mathbf{x}}_t^f) - \bar{\bar{\mathbf{m}}}) \right)$$

where $\bar{\bar{\mathbf{y}}}$ and $\bar{\bar{\mathbf{m}}}$ are time-means of data and model (simulation) equivalent of data mean, respectively.

This choice of reference amounts to assimilating temporal anomalies of the data without altering the model time-mean.

- e) **Integrate original model** after state and diagnostic variables are updated; partitioned reduced-state model is never used to integrate state.



ECCO Approximate KFS

f) **Smoothing** is performed

i. Derive **smoothed wind correction** by,

$$\begin{aligned}\Delta \hat{\mathbf{u}}_i^s &= \mathbf{Q}_i \mathbf{G}^T \left(\mathbf{G} \mathbf{Q}_i \mathbf{G}^T + \mathbf{A} \mathbf{P}_i^a \mathbf{A}^T \right)^{-1} \left(\hat{\mathbf{x}}_{i+1}^s - \mathbf{A} \hat{\mathbf{x}}_i^a - \mathbf{G} \mathbf{u}_i \right) \\ &= \mathbf{Q} \mathbf{G}^T \mathbf{P}^{f^{-1}} \left(\Delta \hat{\mathbf{x}}_{i+1}^s + \Delta \hat{\mathbf{x}}_{i+1}^a \right)\end{aligned}$$

In partitioned
reduced-state;

$$\begin{aligned}\Delta \hat{\mathbf{u}}_t^s &= \sum \hat{\mathbf{B}}_i \Delta \hat{\mathbf{u}}_{t,i}'^s \\ \text{where } \Delta \hat{\mathbf{u}}_{t,i}'^s &= \mathbf{Q}'_i \mathbf{G}'_i{}^T \mathbf{P}'_i{}^{f^{-1}} \left(\Delta \hat{\mathbf{x}}_{t+1,i}'^s + \Delta \hat{\mathbf{x}}_{t+1,i}'^a \right)\end{aligned}$$

ii. Re-run model **simulation with smoother corrected wind** because,

$$\hat{\mathbf{x}}_{t+1}^a + \Delta \hat{\mathbf{x}}_{t+1}^s \neq \mathfrak{F} \left(\hat{\mathbf{x}}_t^a + \Delta \hat{\mathbf{x}}_t^s, \mathbf{u}_t + \Delta \hat{\mathbf{u}}_t^s \right)$$

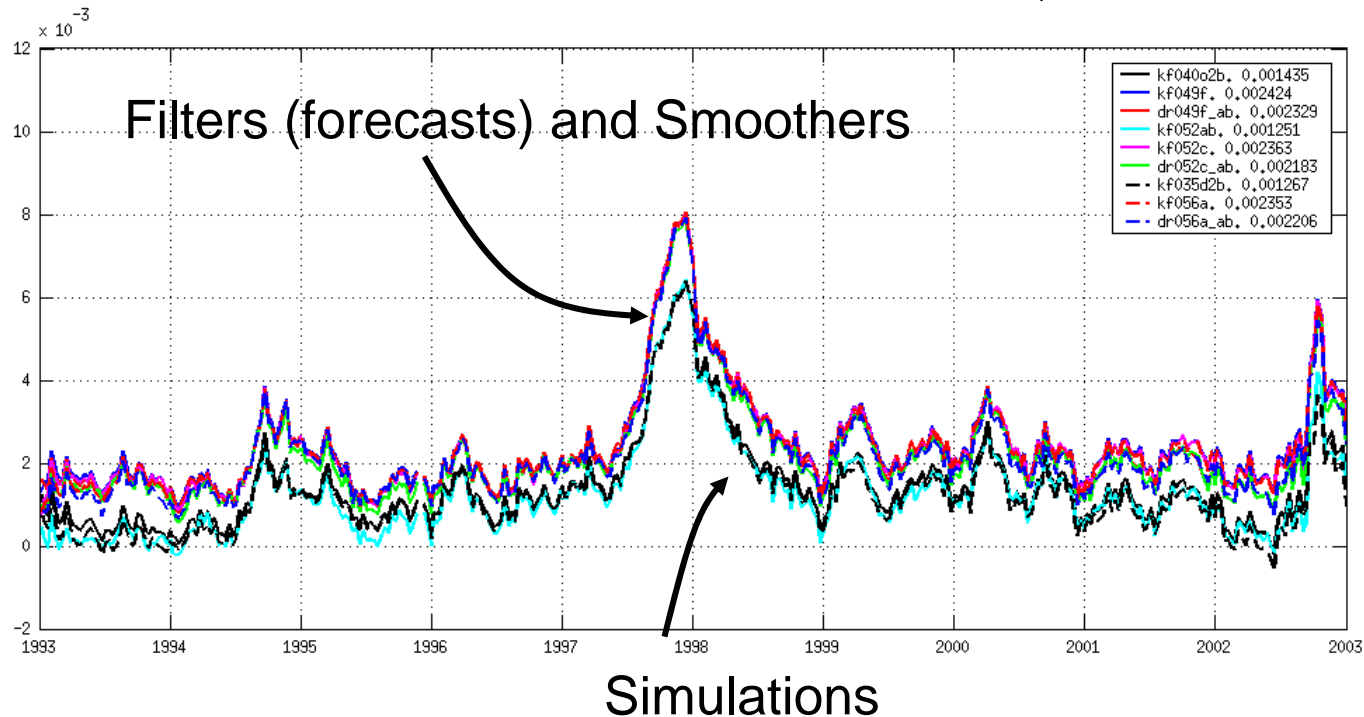
due to approximations, but

$$\hat{\mathbf{x}}_{t+1}^s = \mathfrak{F} \left(\hat{\mathbf{x}}_t^s, \mathbf{u}_t + \Delta \hat{\mathbf{u}}_t^s \right)$$

Assimilation should make models closer to data, both assimilated and independent.

Data not explained by model

Data variance explained by models; $\langle \mathbf{y}\mathbf{y}^T \rangle - \langle (\mathbf{y} - \mathbf{H}\mathbf{x})(\mathbf{y} - \mathbf{H}\mathbf{x})^T \rangle$

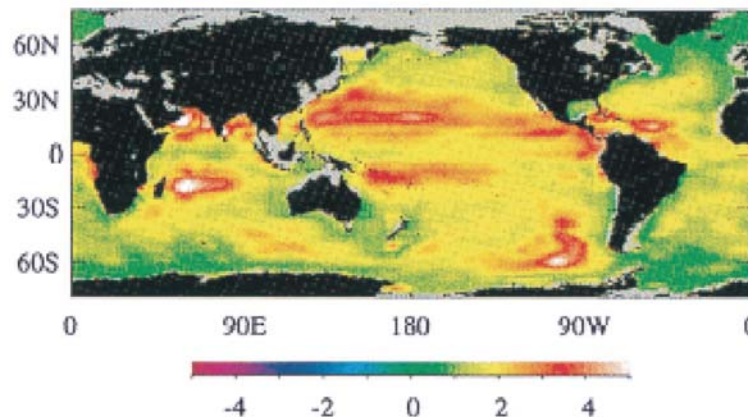
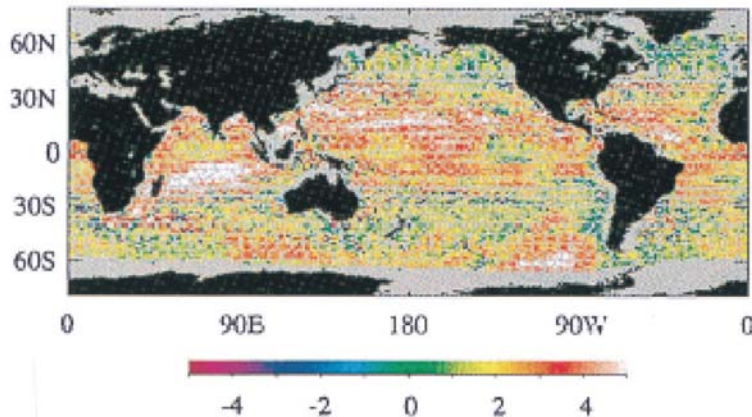


Model-data residuals should be comparable to expectation

simulation - forecast residual

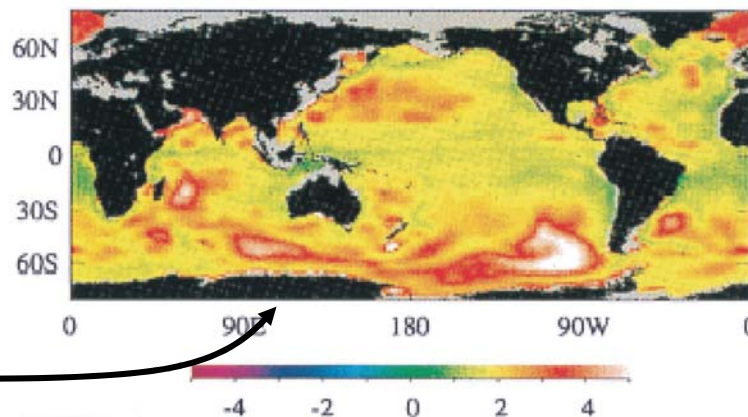
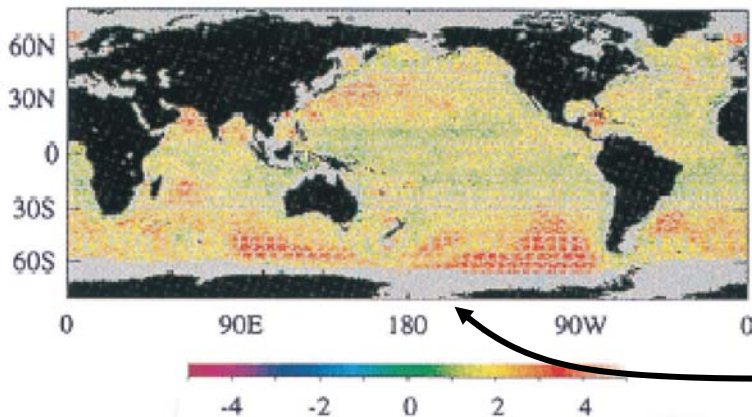
1st order match

$$\mathbf{HP}_{sim} \mathbf{H}^T - \mathbf{HP}^f \mathbf{H}^T$$



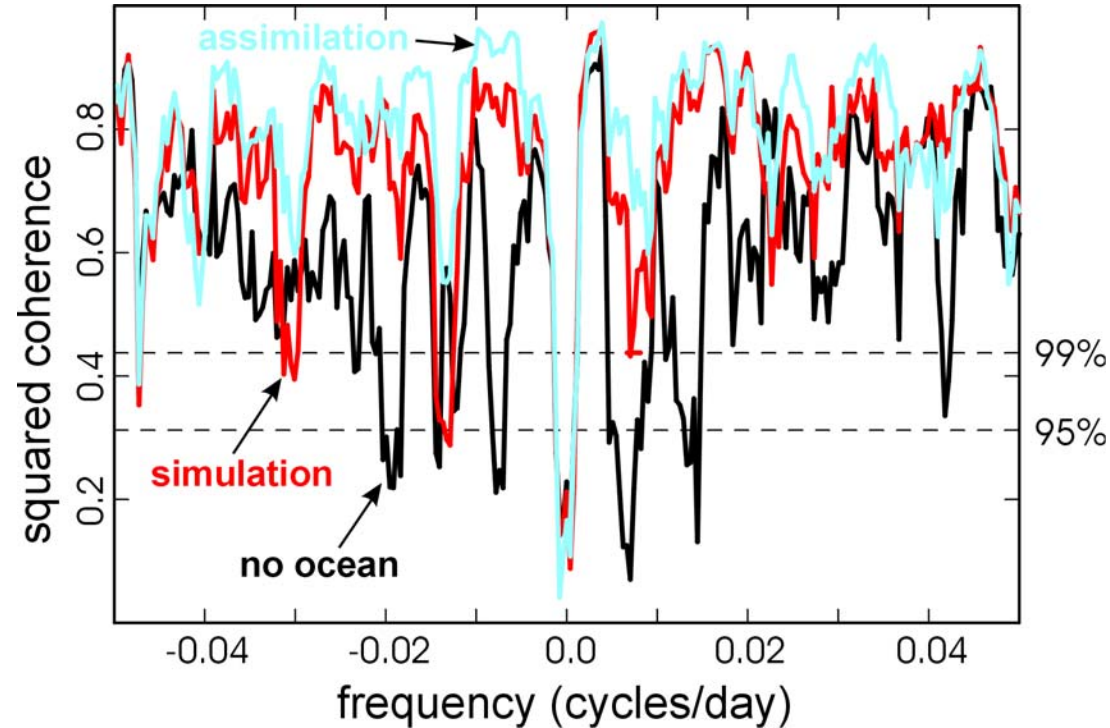
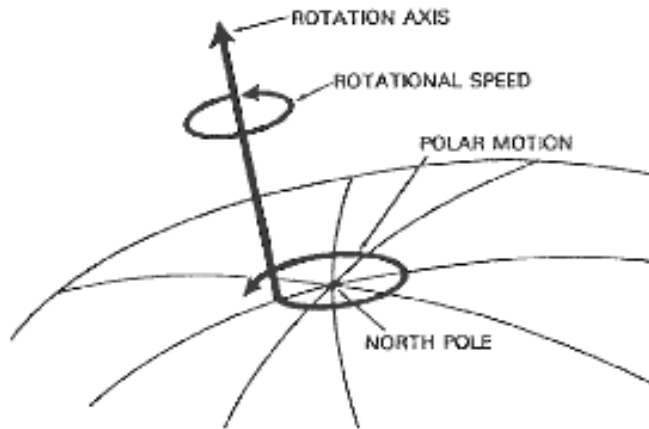
forecast - analysis residual

$$\mathbf{HP}^f \mathbf{H}^T - \mathbf{HP}^a \mathbf{H}^T$$



Comparison to **independent data**: ECCO assimilation explains observed **polar motion** better than ocean simulation does.

Coherence of observed & modeled excitation



(Gross et al., 2003, JGR)

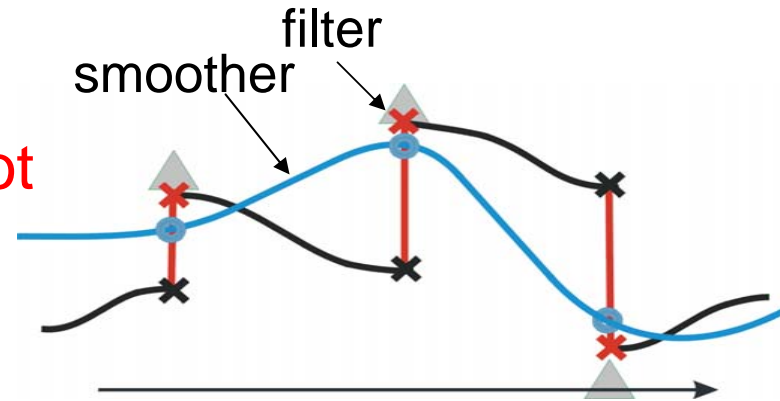
Summary

1. Data assimilation is an **inverse problem** of observations & model with **state & model controls** as unknowns;
2. Data constraint error includes model **representation error**,
3. Kalman filters and RTS smoothers are recursive **least-squares inversions**,
4. Kalman **filters invert observations**; RTS **smoothers invert model**,

$$\begin{pmatrix} \vdots \\ \mathbf{H}\mathbf{x}_t \\ \mathbf{H}\mathbf{x}_{t+1} \\ \vdots \\ \mathbf{x}_{t+1} - \mathbf{A}\mathbf{x}_t - \mathbf{G}\mathbf{u}_t \\ \mathbf{x}_{t+2} - \mathbf{A}\mathbf{x}_{t+1} - \mathbf{G}\mathbf{u}_{t+1} \\ \vdots \end{pmatrix} \approx \begin{pmatrix} \vdots \\ \mathbf{y}_t \\ \mathbf{y}_{t+1} \\ \vdots \\ \mathbf{0} \\ \mathbf{0} \\ \vdots \end{pmatrix}$$

Summary

5. Smoother's **control estimates** are fundamental in solving the assimilation problem,
6. Temporal evolution of filters is **not physically consistent**, but that of smoothers is,
7. Various **approximations & simplifications are possible** that make solving the assimilation problem tractable; ECCO KFS.



Future Challenge

Modeling different model error sources by identifying effective corresponding reduced-state approximations \mathbf{B} is a central issue in data assimilation.

Controls need explicit modeling for estimation;

$$\bar{\mathbf{x}}_{t+1} = A(\bar{\mathbf{x}}_t, \hat{\mathbf{u}}_t^0) + \left\{ \mathbf{\Pi}L(\mathbf{w}_t, \mathbf{v}_t) - A(\mathbf{\Pi}\mathbf{w}_t, \hat{\mathbf{u}}_t^0) \right\}$$

$= A\bar{\mathbf{x}}_t + \mathbf{G}\bar{\mathbf{u}}_t$

\uparrow
 True evolution

\nwarrow
 Model evolution given
 true model state

What are the most **effective basis functions** \mathbf{B} that would represent effects of **various model error sources** ?

Effective = Small dimension but closed system.

$$\mathbf{x}'_{t+1} = \mathbf{B}^* \left(\mathfrak{F}(\tilde{\mathbf{x}} + \mathbf{B}\mathbf{x}'_t + \mathbf{N}\mathbf{n}_t, \tilde{\mathbf{u}} + \hat{\mathbf{B}}\mathbf{u}'_t + \hat{\mathbf{N}}\mathbf{m}_t) - \mathfrak{F}(\tilde{\mathbf{x}}, \tilde{\mathbf{u}}) \right)$$

$$\approx \mathbf{B}^* \left(\mathfrak{F}(\tilde{\mathbf{x}} + \mathbf{B}\mathbf{x}'_t, \tilde{\mathbf{u}} + \hat{\mathbf{B}}\mathbf{u}'_t) - \mathfrak{F}(\tilde{\mathbf{x}}, \tilde{\mathbf{u}}) \right)$$