Data Assimilation Models

L. Scherliess

Center for Atmospheric & Space Sciences
Utah State University
Logan, Utah 84322

CEDAR Workshop
Santa Fe
June, 2013
The Daily Weather Forecast is a Product of Data Assimilation
On the One Hand, we have large Quantities of Data

- Different kinds of instruments measuring different quantities (apples and oranges)
- Observations are in different places
- Observations have different cadence and availability
- Observations have different error statistics

Difficult to create coherent Picture
On the Other Hand, we have Mature Theoretical/Numerical Models

» Models contain our ‘knowledge’ of the physics

Uncertain Parameters in Physics-Based Model

- $O^+$ - $O$ Collision Frequency
- Secondary Electron Production
- Downward Heat Flow
- Chemical Reaction Rates
- External Forcing
- Etc.
Objectives

- Optimally combine Data and the Model to create coherent Picture of the Space Environment

- Solution satisfies the physical laws and ‘agrees’ with the data and the model as best as possible (within their error bounds)
Data Assimilation Tasks

- Develop Physical Model
- Develop Assimilation Algorithm
- Data Acquisition Software
- Data Quality Control
- Executive System
- Validation Software
Brief Historical Background

Data Assimilation in the Atmosphere:
- Initial Attemps started in the 1950th (NWP)

Data Assimilation in the Oceans:
- Began with large scales (mean properties) about 30 yrs ago
- Regional efforts (e.g., Gulf stream) [15-20 yrs ago]
- Produce operational upper ocean now- and forecast.
Data Assimilation in Space Sciences

- Assimilative Mapping of Ionospheric Electrodynamics (AMIE, Richmond and Kamide, 1988)
- Initial Testing of Kalman Filter for Ionospheric Electron Density Reconstructions (Howe et al., 1998)
- Data Assimilation Models for the Ionosphere (late 1990): GAIM models, IDA4D
- Data Assimilation Models for the Thermosphere (Minter et al., Fuller-Rowell et al.)
- Data Assimilation for the Radiation Belts
- Initial Attempts for Solar Data Assimilation
What can we learn from Meteorology?

Data Assimilation Techniques have been used in Meteorology for the last 50 years

- Most Accurate Specifications and Forecast Models are Those that Assimilate Measurements into a *Physics-Based* Numerical Model

- Better Predictions are Obtained for the Atmosphere
  - When the Data are Assimilated with a Rigorous Mathematical Approach
Data Assimilation Techniques

- **3-d Var**
  \[ J(\delta x) = 1/2 \delta x^T P^{-1} \delta x + 1/2 [H(\delta x + x^b) - y^o]^T R^{-1} [H(\delta x + x^b) - y^o] \]

- **4-d Var**
  \[ J(\delta x) = 1/2 \delta x_0^T P^{-1} \delta x_0 + 1/2 \sum_{i=0}^{n} [H_i(M_{i,0}(x_0)) - y_i^o]^T R^{-1} [H_i(M_{i,0}(x_0)) - y_i^o] \]

- **Kalman Filter**
  \[ x^f = Mx + \eta \]
  \[ P^f = MPM^T + Q \]
  \[ y^o = Hx + \epsilon \]
  \[ K = P^fH^T (HP^fH^T + R)^{-1} \]
  \[ x^a = x^f + K(y^o - Hx^f) \]
  \[ P^a = (I-KH)P^f \]
The Data Assimilation Cycle

Start with a forecast or an estimate of the state (background)

Data Collection

Quality Control

‘Best-Guess’ Background → Short-Term Forecast → Analysis → Forecast

Background can come from a Physical Model or Empirical Model.
The Data Assimilation Cycle

Collect Data and Quality Control them.

Data can come from a variety of different observation systems.
The Data Assimilation Cycle

Minimize the difference between the analysis and a weighted combination of

- the background and
- the observations.
Minimize the difference between the analysis and a **weighted combination** of
- the background and
- the observations.

The analysis serves as the start for the next model forecast.
The Cost Function

To produce the analysis we want to minimize a “Cost Function” $J$ which consists of:

$$J = J_B + J_O + J_C$$

$J_B$: Weighted fit to the background field

$J_O$: Weighted fit to the observations

$J_C$: Constraint which can be used to impose physical properties (e.g., analysis should satisfy Maxwell’s equations, continuity equation, …)
The Cost Function, cont.

A typical form for the \( J_B \) term is:

\[
J_B = (x_A - x_B)^T B^{-1} (x_A - x_B)
\]

Where:

- \( x_A \): Analysis Variable (e.g., Electron Density, Temperature, …)
- \( x_B \): Background Field, obtained from the Model Forecast

\( B \) : Background Error Covariance Matrix:

- How good is your Background
- What are covariances between different elements

The background error covariances are only poorly known
The Cost Function, cont.

A typical form for the cost function for the observations is:

\[ J_O = [y - H(x_A)]^T R^{-1} [y - H(x_A)] \]

Where:

- \( y \) : Represents all Observations
- \( H \) : Forward Operator which maps the Grid Point Values to Observations (can be linear or nonlinear)
- \( R \) : Observation Error Covariance Matrix:
  - **How good is your data?**
  - (also includes the representativeness of the data)
The Cost Function, cont.

The **Physical Properties/Model** were used to:

- Obtain the best possible background field
- To constrain the Analysis

**In 3-D Var:**

- Cost function and the constraints are not explicitly time dependent
- A temporal model is not necessarily required
- Snapshots
Fundamental Concepts of 4D-Var

4D-Var introduces the temporal dimension to data assimilation

Find a close fit to the data that is consistent with the dynamical model over an extended period of time.

⇒ Find the closest trajectory

\[
J(\delta x) = \frac{1}{2} \delta x_0^T P^{-1} \delta x_0 + \frac{1}{2} \sum_{i=0}^{n} \left[ H_i(M_{i,o}(x_0)) - y_i^o \right]^T R^{-1} \left[ H_i(M_{i,o}(x_0)) - y_i^o \right]
\]

The Model

The Data
Fundamental Concepts of 4D-Var

4D-Var introduces the temporal dimension to data assimilation

Find a close fit to the data that is consistent with the dynamical model over an extended period of time.

⇒ Find the closest trajectory

\[
J(\delta x) = 1/2 \delta x_0^T P^{-1} \delta x_0 + 1/2 \sum_{i=0}^{n} \left[ H_i(M_{i,o}(x_0)) - y_i^o \right]^T R^{-1} \left[ H_i(M_{i,o}(x_0)) - y_i^o \right]
\]

Model Error Covariance     Data Error Covariance
Another Data Assimilation Technique
The Kalman Filter

- M - State Transition Matrix
- P - Model Error Covariance
- y - Data Vector
- R - Observation Error Covariance
- X - Model State Vector
- η - Transition Model Error
- Q - Transition Model Error Covariance
- H - Measurement Matrix
- ε - Observation Error
- K - Kalman Gain

\[ \begin{align*}
    x_f^f &= Mx + \eta \\
    P_f^f &= MPM^T + Q \\
    y^o &= Hx + \epsilon \\
    K &= P_f^fH^T(HP_f^fH^T + R)^{-1} \\
    x^a &= x_f^f + K(y^o - Hx_f^f) \\
    P^a &= (I - KH)P_f^f
\end{align*} \]
Another Data Assimilation Technique

The Kalman Filter

- **M** - State Transition Matrix
- **P** - Model Error Covariance
- **y** - Data Vector
- **R** - Observation Error Covariance
- **X** - Model State Vector
- **η** - Transition Model Error
- **Q** - Transition Model Error Covariance
- **H** - Measurement Matrix
- **ε** - Observation Error
- **K** - Kalman Gain

\[
x_f = Mx + \eta \\
P_f = MPMT + Q \\
y_o = Hx + \epsilon \\
K = P_f^T(HP_fH^T + R)^{-1} \\
x_a = x_f + K(y_o - Hx_f) \\
P_a = (I - KH)P_f
\]
Another Data Assimilation Technique
The Kalman Filter

The Dynamical Model entered the Filter:

- Evolution of the State Vector (make a Forecast)
- Evolution of the Error Covariance Matrix

\[
\begin{align*}
x_f^t &= Mx + \eta \\
P_f^t &= MMP^T + Q \\
y^o &= Hx + \epsilon \\
K &= P_f^tH^T(HP_f^tH^T + R)^{-1} \\
x^a &= x_f^t + K(y^o - Hx_f^t) \\
P^a &= (I - KH)P_f^t
\end{align*}
\]
Another Data Assimilation Technique
The Kalman Filter

The Dynamical Model entered the Filter:

- Evolution of the State Vector (make a Forecast)
- Evolution of the Error Covariance Matrix

→ Error Covariance Matrix becomes time-dependent and evolves with the same physical model as the state!

This is computationally the most expensive step in the Kalman filter

\[
\begin{align*}
    x^f &= Mx + \eta \\
    P^f &= MPM^T + Q \\
    y^o &= Hx + \epsilon \\
    K &= P^fH^T(HP^fH^T + R)^{-1} \\
    x^a &= x^f + K(y^o - Hx^f) \\
    P^a &= (I - KH)P^f
\end{align*}
\]
**Example:** Tracking of a Rocket with a Kalman Filter

A rocket is flying through space launched from an initial location with an initial velocity.

\[
m \frac{d^2 x}{dt^2} = ma \quad \Rightarrow \quad \frac{dx}{dt} = v \quad \Rightarrow \quad x_{i+1} \approx x_i + v_i \cdot dt
\]

\[
\frac{dv}{dt} = a \quad \Rightarrow \quad v_{i+1} \approx v_i + a_i \cdot dt
\]

\[
\frac{da}{dt} = \Rightarrow \quad a_{i+1} \approx a_i
\]

In Kalman filter we have: \( x_{i+1} = M x_i \)

\[
\begin{pmatrix}
  x_{i+1} \\
  v_{i+1} \\
  a_{i+1}
\end{pmatrix} =
\begin{pmatrix}
  1 & dt & 0 \\
  0 & 1 & dt \\
  0 & 0 & 1
\end{pmatrix}
\begin{pmatrix}
  x_i \\
  v_i \\
  a_i
\end{pmatrix}
\]
Example: Tracking of a Rocket with a Kalman Filter

Propagate Error Covariance Matrix: \[ P_{i+1} = M P_i M^T \]

\[
P_0 = \begin{pmatrix}
\sigma_x^2 & 0 & 0 \\
0 & \sigma_v^2 & 0 \\
0 & 0 & \sigma_a^2
\end{pmatrix}
\]

At the next time step:

\[
P_1 = \begin{pmatrix}
1 & dt & 0 \\
0 & 1 & dt \\
0 & 0 & 1
\end{pmatrix} \begin{pmatrix}
\sigma_x^2 & 0 & 0 \\
0 & \sigma_v^2 & 0 \\
0 & 0 & \sigma_a^2
\end{pmatrix} \begin{pmatrix}
1 & 0 & 0 \\
dt & 1 & 0 \\
0 & dt & 1
\end{pmatrix}
\]
Example: Tracking of a Rocket with a Kalman Filter

Propagate Error Covariance Matrix: \( P_{i+1} = M P_i M^T \)

\[
    P_0 = \begin{pmatrix}
        \sigma_x^2 & 0 & 0 \\
        0 & \sigma_v^2 & 0 \\
        0 & 0 & \sigma_a^2
    \end{pmatrix}
\]

At the next time step:

\[
    P_1 = \begin{pmatrix}
        \sigma_x^2 + \sigma_v^2 dt^2 & \sigma_v^2 dt & 0 \\
        \sigma_v^2 dt & \sigma_v^2 + \sigma_a^2 dt^2 & \sigma_a^2 dt \\
        0 & \sigma_v^2 dt & \sigma_a^2
    \end{pmatrix}
\]
The Rocket

Position

Velocity

Acceleration
Kalman Filter has specified the external Forcing

Forcing is specified based on the Dynamics provide by the physical Model
Next, consider the more complicated situation:

- Much more complicated Differential Equations
- Global Reconstruction
- Many observations
- Different kinds of instruments measuring different quantities
- Observations are in different places

⇒ This is the Situation in the Ionosphere
Another Data Assimilation Technique

The Kalman Filter

Model

\[
\begin{align*}
    x^f &= Mx + \eta \\
    P^f &= MPM^T + Q \\
    y^o &= Hx + \epsilon \\
    K &= P^fH^T(HP^fH^T + R)^{-1} \\
    x^a &= x^f + K(y^o - Hx^f) \\
    P^a &= (I - KH)P^f
\end{align*}
\]

This is computationally the most expensive step in the Kalman filter
Ways to get around the Problem

- Approximate Kalman Filters
  - Band-Limited Kalman Filter
  - Reduced State Kalman Filter
  - Gauss-Markov Kalman Filter
  - Ensemble Kalman Filter

Do not evolve Error Covariance Matrix with Model

INSTEAD

Obtain Error Covariance Matrix from an ENSEMBLE of Model runs
Full Physics Kalman Filter Model

• Ensemble Kalman Filter
  ➢ 30 Global Simulations are Launched at Each Assimilation Time Step

• Physics-based Ionosphere-Plasmasphere Model

• Model Physics is embedded in Kalman filter

• Same 5 Data Sources as Gauss-Markov Model

• Provides both specifications for the ionospheric plasma densities and drivers.
Determination of Ionospheric Drivers Using The Full Physics-Based GAIM Model

- Ionospheric Sensitivities to Drivers are embedded in the Covariances and are automatically and at each Time Step calculated.

- Drivers include:
  - Electric Fields
  - Neutral Wind
  - Composition
  - …
Example of Full Physics-Based Kalman Filter Model

- Several Days in March/April of 2004
- Geomagnetically Quiet Period
- Data Assimilated
  - Slant TEC from 162 GPS Ground Receivers
- Use Ionosonde Data for Validation
Comparison with Ionosonde Data

Ionosonde Data were NOT assimilated!
Summary

NCEP Operational Forecast Skill
36 and 72 Hour Forecasts @ 500 MB over North America
[100 \times (1-S/70) Method]

- 36 Hour Forecast
- 72 Hour Forecast

NCEP Central Operations January 2007