Data Assimilation and Driver Estimation for Space Weather Models using Ensemble Filters

Alexey V. Morozov

UofM Committee:
Dennis S. Bernstein
Aaron J. Ridley
Pierre T. Kabamba
Ilya V. Kolmanovsky

NCAR Collaborators:
Nancy Collins
Timothy J. Hoar
Jeffrey L. Anderson

March 29, 2013
Overview

1. Summary
2. Problem Statement
3. GITM
4. EAKF
5. Assimilating CHAMP Neutral Density
6. Assimilating GPS Total Electron Content
7. Conclusions and Future Work
1 Summary
   - Main Contributions
   - Publications

2 Problem Statement

3 GITM

4 EAKF

5 Assimilating CHAMP Neutral Density

6 Assimilating GPS Total Electron Content

7 Conclusions and Future Work
Main Contributions

Contributions relating to data assimilation are:

- **modified** the Data Assimilation Research Testbed (DART) to interface it with the Global Ionosphere-Thermosphere Model (GITM),

- **developed** a novel inflation technique for the Ensemble Adjustment Kalman Filter (EAKF) as applied to GITM for purposes of data assimilation and driver estimation, and

- **introduced** an ability to assimilate Total Electron Content (TEC) measurements into the DART-GITM interface.
Main Contributions

Contributions relating to data assimilation are:

- **modified** the Data Assimilation Research Testbed (DART) to interface it with the Global Ionosphere-Thermosphere Model (GITM),

- **developed** a novel inflation technique for the Ensemble Adjustment Kalman Filter (EAKF) as applied to GITM for purposes of data assimilation and driver estimation, and

- **introduced** an ability to assimilate Total Electron Content (TEC) measurements into the DART-GITM interface.

Contributions relating to adaptive control are:

- **described** the Retrospective Cost Adaptive Control (RCAC) stability margins for plants with uncertain nonminimum-phase zeros,

- **introduced** a convex constraint on the controller pole locations to improve RCAC transient and steady-state performance, and

- **modeled** nonlinear system and described the achievable amplitude and frequency control ranges.


1 Summary

2 Problem Statement
   - Big Picture
   - Problem Statement

3 GITM

4 EAKF

5 Assimilating CHAMP Neutral Density

6 Assimilating GPS Total Electron Content

7 Conclusions and Future Work
steering wheel angle \rightarrow \text{Car} \rightarrow \text{lane position}
Where does model inversion come in?

The goal is to make $\hat{y} = y$, that is $\frac{\hat{y}}{y} = 1$ or $e = 0$, that is $\frac{e}{y} = 0$

$\hat{y} = Pd$

$\hat{y} = P(y - C\hat{y})$

$\hat{y} = Py - PC\hat{y}$

$(1 + PC)\hat{y} = Py$

$\frac{\hat{y}}{y} = \frac{P}{1 + PC}$

$\frac{P}{1 + PC} \overset{want}{=} 1$

$P \overset{w}{=} 1 + PC$

$C \overset{w}{=} 1 - P^{-1}$

$e = y - \hat{y}$

$e = y - \frac{P}{1 + PC}y$

$e = 1 + PC - P$

$e = 1 + PC - P$

$\frac{1 + PC - P}{1 + PC} \overset{w}{=} 0$

$1 + PC - P \overset{w}{=} 0$

$C \overset{w}{=} 1 - P^{-1}$
Example of inversion in frequency domain

Consider a linear plant $P = \frac{1}{z+0.5}$, which can be described in time domain as $\hat{y}_k = -0.5\hat{y}_{k-1} + d_{k-1}$. Suppose the goal is for the plant output $\hat{y}$ to follow desired command $y$ (command following). RCAC converges to controller $C = \frac{0.7104}{z-0.5797}$.
How is data assimilation on GITM similar to driving a car?
Problem Statement

We want to drive GITM output (black) to match the satellite data (red).

We want to drive GITM output (black) to match the satellite data (red).
Summary

Problem Statement

3 GITM
   - Inputs and Outputs
   - Equations
   - Implementation

4 EAKF

5 Assimilating CHAMP Neutral Density

6 Assimilating GPS Total Electron Content

7 Conclusions and Future Work
### Inputs and Outputs

<table>
<thead>
<tr>
<th>Inputs</th>
<th>Outputs</th>
</tr>
</thead>
<tbody>
<tr>
<td>Solar flux index</td>
<td>Neutral number densities</td>
</tr>
<tr>
<td>( F_{10.7} ) or ( I_{\infty} )</td>
<td>( N_s )</td>
</tr>
<tr>
<td>Cooling rates</td>
<td>Neutral mass density</td>
</tr>
<tr>
<td>( L_e(X) )</td>
<td>( \rho )</td>
</tr>
<tr>
<td>Heating efficiency</td>
<td>Neutral pressure</td>
</tr>
<tr>
<td>( \varepsilon )</td>
<td>( p )</td>
</tr>
<tr>
<td>Thermal conductivity</td>
<td>Neutral temperature normalized</td>
</tr>
<tr>
<td>( \kappa_c )</td>
<td>( T )</td>
</tr>
<tr>
<td>Earth magnetic field</td>
<td>Neutral velocity</td>
</tr>
<tr>
<td>simplified</td>
<td>( u )</td>
</tr>
<tr>
<td>or APEX</td>
<td>Ion number densities</td>
</tr>
<tr>
<td>Interplanetary magnetic field</td>
<td>( N_j )</td>
</tr>
<tr>
<td>simplified</td>
<td>Ion temperature normalized</td>
</tr>
<tr>
<td>or ACE data</td>
<td>( T_j )</td>
</tr>
<tr>
<td>Hemispheric power index</td>
<td>Ion velocity</td>
</tr>
<tr>
<td>simplified</td>
<td>( v )</td>
</tr>
<tr>
<td>or POES data</td>
<td></td>
</tr>
<tr>
<td>Initialize using</td>
<td></td>
</tr>
<tr>
<td>simplified</td>
<td></td>
</tr>
<tr>
<td>or MSIS/IRI</td>
<td></td>
</tr>
</tbody>
</table>

Table 21 compiled from \(^1\)

GITM Vertical Equations

\[
\frac{\partial N_s}{\partial t} + \frac{\partial u_{r,s}}{\partial r} + \frac{2u_{r,s}}{r} + u_{r,s} \frac{\partial N_s}{\partial r} = \frac{1}{N_s} \mathcal{J}_s,
\]

(continuity) \hspace{1cm} (1)

\[
\frac{\partial u_{r,s}}{\partial t} + u_{r,s} \frac{\partial u_{r,s}}{\partial r} + \frac{u_{\theta}}{r} \frac{\partial u_{r,s}}{\partial \theta} + \frac{u_{\phi}}{r \cos(\theta)} \frac{\partial u_{r,s}}{\partial \phi} + \frac{k}{M_s} \frac{\partial T}{\partial r} + \frac{k}{M_s} T \frac{\partial N_s}{\partial r} =
\]

\[g + F_s + \frac{u_{\theta}^2 + u_{\phi}^2}{r} + \cos^2(\theta) \Omega^2 r + 2 \cos(\theta) \Omega u_{\phi},\]

(momentum) \hspace{1cm} (2)

\[
\frac{\partial T}{\partial t} + u_r \frac{\partial T}{\partial r} + (\gamma - 1) T \left( \frac{2u_r}{r} + \frac{\partial u_r}{\partial r} \right) = \frac{k}{c_v \rho \bar{m}_n} Q,
\]

(energy) \hspace{1cm} (3)
GITM Vertical Equations and source terms

(continuity) \[ \frac{\partial N_s}{\partial t} + \frac{\partial u_{r,s}}{\partial r} + \frac{2u_{r,s}}{r} + u_{r,s} \frac{\partial N_s}{\partial r} = \frac{1}{N_s} \mathcal{J}_s, \] (4)

(momentum) \[ \frac{\partial u_{r,s}}{\partial t} + u_{r,s} \frac{\partial u_{r,s}}{\partial r} + \frac{u_\theta}{r} \frac{\partial u_{r,s}}{\partial \theta} + \frac{u_\phi}{r \cos(\theta)} \frac{\partial u_{r,s}}{\partial \phi} + \frac{k}{M_s} \frac{\partial T}{\partial r} + \frac{k}{M_s} T \frac{\partial N_s}{\partial r} = g + \mathcal{F}_s + \frac{u_\theta^2 + u_\phi^2}{r} + \cos^2(\theta)\Omega^2 r + 2 \cos(\theta)\Omega u_\phi, \] (5)

(energy) \[ \frac{\partial \mathcal{T}}{\partial t} + u_r \frac{\partial \mathcal{T}}{\partial r} + (\gamma - 1) \mathcal{T} \left( \frac{2u_r}{r} + \frac{\partial u_r}{\partial r} \right) = \frac{k}{c_v \rho \bar{m}_n} 2, \] (6)

\[ \mathcal{J}_s = \frac{\partial}{\partial r} \left[ N_s K_e \left( \frac{\partial N_s}{\partial r} - \frac{\partial N}{\partial r} \right) \right] + C_s, \] (7)

\[ \mathcal{F}_s = \frac{\rho_i}{\rho_s} \nu_{in} (v_r - u_{r,s}) + \frac{kT}{M_s} \sum_{q \neq s} \frac{N_q}{ND_{qs}} (u_{r,q} - u_{r,s}), \] (8)

\[ 2 = Q_{EUV} + Q_{NO} + Q_O + \frac{\partial}{\partial r} \left( (\kappa_c + \kappa_{eddy}) \frac{\partial T}{\partial r} \right) + N_e \frac{\bar{m}_i \bar{m}_n}{\bar{m}_i + \bar{m}_n} \nu_{in} (v - u)^2. \]
In the CHAMP simulations, $5^\circ$ resolution in longitude and latitude is used.

Variable resolution in altitude (from about $2\text{km}$ to about $18\text{km}$) is used to span the range between about $100\text{km}$ and $660\text{km}$.

The atmosphere is broken up into 32 (8 in longitude, 4 in latitude) blocks to allow for parallel computation.

A typical EAKF run (20 ensemble members) requires 640 CPUs (cores) and about 8 wall hours per 24 simulated hours on NASA Pleiades supercomputer.
1. Summary

2. Problem Statement

3. GITM

4. EAKF
   - Big Picture
   - Kalman Filter
   - Extended Kalman Filter (EKF)
   - Unscented Kalman Filter (UKF)
   - Ensemble Kalman Filter (EnKF)
   - Ensemble Adjustment Kalman Filter (EAKF)

5. Assimilating CHAMP Neutral Density

6. Assimilating GPS Total Electron Content
Pros and Cons of These Filters

KF → EKF → UKF → EnKF → EAKF

- The advantage of EKF over KF is that it allows the system to be nonlinear.
- The advantage of UKF over EKF is that it is more accurate in propagating means and covariances through the nonlinear system and does not require linearization.
- The advantage of EnKF over UKF is that it does not require \(2^n\) ensemble members (sigma points, particles), and allows the user to pick \(N\), the number of ensemble members. The newest versions of EnKF allow for localization, which improves computational performance even more than just reduction in number of ensemble members.
- EAKF is similar to EnKF in being an ensemble filter, but utilizes different update equations. Some advantages of EAKF over UKF are addition of localization functionality and decreased computational load.
The advantage of EKF over KF is that it allows the system to be nonlinear.

The advantage of UKF over EKF is that it is more accurate in propagating means and covariances through the nonlinear system and does not require linearization.

The advantage of EnKF over UKF is that it does not require $2^n$ ensemble members (sigma points, particles), and allows the user to pick $N$, the number of ensemble members. The newest versions of EnKF allow for localization, which improves computational performance even more than just reduction in number of ensemble members.

EAKF is similar to EnKF in being an ensemble filter, but utilizes different update equations. Some advantages of EAKF over UKF are addition of localization functionality and decreased computational load.
Pros and Cons of These Filters

- **KF → EKF**: The advantage of EKF over KF is that it allows the system to be nonlinear.
- **EKF → UKF**: The advantage of UKF over EKF is that it is more accurate in propagating means and covariances through the nonlinear system and does not require linearization.

- **UKF → EnKF**: The advantage of EnKF over UKF is that it does not require $2^n$ ensemble members (sigma points, particles), and allows the user to pick $N$, the number of ensemble members. The newest versions of EnKF allow for localization, which improves computational performance even more than just reduction in number of ensemble members.

- **UKF → EAKF**: EAKF is similar to EnKF in being an ensemble filter, but utilizes different update equations. Some advantages of EAKF over UKF are addition of localization functionality and decreased computational load.
Pros and Cons of These Filters

- **KF → EKF**: The advantage of EKF over KF is that it allows the system to be nonlinear.
- **EKF → UKF**: The advantage of UKF over EKF is that it is more accurate in propagating means and covariances through the nonlinear system and does not require linearization.
- **UKF → EnKF**: The advantage of EnKF over UKF is that it does not require $2n$ ensemble members (sigma points, particles), and allows the user to pick $N$, the number of ensemble members. The newest versions of EnKF allow for localization, which improves computational performance even more than just reduction in number of ensemble members.
Pros and Cons of These Filters

- **KF → EKF** The advantage of EKF over KF is that it allows the system to be nonlinear.
- **EKF → UKF** The advantage of UKF over EKF is that it is more accurate in propagating means and covariances through the nonlinear system and does not require linearization.
- **UKF → EnKF** The advantage of EnKF over UKF is that it does not require $2n$ ensemble members (sigma points, particles), and allows the user to pick $N$, the number of ensemble members. The newest versions of EnKF allow for localization, which improves computational performance even more than just reduction in number of ensemble members.
- **UKF → EAKF** EAKF is similar to EnKF in being an ensemble filter, but utilizes different update equations. Some advantages of EAKF over UKF are addition of localization functionality and decreased computational load.
Consider the linear discrete-time system given by

\[
\begin{align*}
x_k &= F_{k-1} x_{k-1} + G_{k-1} u_{k-1} + w_{k-1}, \\
y_k &= H_k x_k + v_k,
\end{align*}
\]

where \( x_k \in \mathbb{R}^n, y_k \in \mathbb{R}^m, u_k \in \mathbb{R}^p, w_k \) is the process noise, \( v_k \) is the measurement noise, and \( w_k \) and \( v_k \) are uncorrelated (i.e. \( E(w_k v_j^T) = 0 \)).

If \( F_k, G_k, H_k, Q_k, R_k, \) are known, the discrete time Kalman Filter\(^2\) is given by

\[
\begin{align*}
\hat{x}_{k}^− &= F_{k-1} \hat{x}_{k-1}^+ + G_{k-1} u_{k-1}, \\
P_k^− &= F_{k-1} P_{k-1}^+ F_{k-1}^T + Q_{k-1}, \\
K_k &= P_k^− H_k^T (H_k P_k^− H_k^T + R_k)^{-1}, \\
P_k^+ &= (I - K_k H_k) P_k^−, \\
\hat{x}_k^+ &= \hat{x}_k^− + K_k (y_k - H_k \hat{x}_k^−).
\end{align*}
\]

As an aside, we demonstrate that Discrete Algebraic Riccati Equation (DARE) can be derived from the update equations derived so far.

Substituting (16) into (12), we realize that prior state estimate can be updated directly without computation of the posterior estimate as in

\[
\hat{x}_{k+1}^- = F_k(I - K_k H_k)\hat{x}_{k-1}^+ + F_k K_k y_k G_k u_k.
\]

Similarly, by substituting (15) and (14) into (13), it can be shown that prior error covariance matrix can also be updated in one step, as given by

\[
P_{k+1}^- = F_k(P_k^- - \left[ P_k^- H_k^T (H_k P_k^- H_k^T + R_k)^{-1} H_k P_k^- \right] F_k^T + Q_k)
\]

\[
= F_k P_k^- F_k^T - F_k P_k^- H_k^T (H_k P_k^- H_k^T + R_k)^{-1} H_k P_k^- F_k^T + Q_k,
\]

which is the Discrete Algebraic Riccati Equation.
Now consider the nonlinear discrete-time system given by

\[
    x_k = f_{k-1}(x_{k-1}, u_{k-1}, w_{k-1}), \quad w_k \sim N(0, Q_k), \quad (20)
\]

\[
    y_k = h_k(x_k, v_k), \quad v_k \sim N(0, R_k), \quad (21)
\]

where functions \( f_k(\cdot) \) and \( h_k(\cdot) \) are known explicitly and hence can be linearized via

\[
    F_{k-1} = \left. \frac{\partial f_{k-1}}{\partial x} \right|_{\hat{x}_{k-1}}, \quad L_{k-1} = \left. \frac{\partial f_{k-1}}{\partial w} \right|_{\hat{x}_{k-1}}, \quad (22)
\]

\[
    H_k = \left. \frac{\partial h_k}{\partial x} \right|_{\hat{x}_k}, \quad M_k = \left. \frac{\partial h_k}{\partial v} \right|_{\hat{x}_k}. \quad (23)
\]

Next, EKF\(^3\) can be updated via

\[
    \hat{x}_k = f_{k-1} (\hat{x}_{k-1}, u_{k-1}, 0), \quad (24)
\]

\[
    P_k^- = F_{k-1} P_{k-1}^+ F_{k-1}^T + L_{k-1} Q_{k-1} L_{k-1}^T, \quad (25)
\]

\[
    K_k = P_k^- H_k^T (H_k P_k^- H_k^T + M_k R_k M_k^T)^{-1}, \quad (26)
\]

\[
    P_k^+ = (I - K_k H_k) P_k^-, \quad (27)
\]

\[
    \hat{x}_k^+ = \hat{x}_k^- + K_k \left[ y_k - h_k(\hat{x}_k^-, 0) \right]. \quad (28)
\]

Unscented Kalman Filter (UKF)

- Consider the nonlinear discrete-time system with additive noise given by
  \[
  x_k = f_{k-1}(x_{k-1}, u_{k-1}) + w_{k-1}, \\
  y_k = h_k(x_k) + v_k.
  \]  
  (29)  
  (30)

- UKF propagates 2n realizations (sigma points, \( \hat{x}^{(i)} \)) of the state's probability density function, as defined and propagated by
  \[
  \hat{x}^+_{k-1} = \hat{x}^+_{k-1} + (-1)^{\lfloor \frac{i-1}{n} \rfloor} \sqrt{nP^+_{k-1}} T_i,
  \]  
  (31)

  \[
  \hat{x}^{(i)}_k = f_{k-1}(\hat{x}^{(i)}_{k-1}, u_{k-1}), \quad \hat{y}^{(i)}_k = h_k(\hat{x}^{(i)}_k).
  \]  
  (32)

- Accordingly, UKF can be updated via
  \[
  \hat{x}^-_k = \frac{1}{2n} \sum_{i=1}^{2n} \hat{x}^{(i)}_k, \quad \hat{y}^-_k = \frac{1}{2n} \sum_{i=1}^{2n} \hat{y}^{(i)}_k,
  \]  
  (33)

  \[
P^-_k = \sum_{i=1}^{2n} (\hat{x}^{(i)}_k - \hat{x}^-_k)(\hat{x}^{(i)}_k - \hat{x}^-_k)^T / (2n) + Q_{k-1},
  \]  
  (34)

  \[
  K_k = P_{xy} P_y^{-1}, \quad P_{xy} = \sum_{i=1}^{2n} (\hat{x}^{(i)}_k - \hat{x}^-_k)(\hat{y}^{(i)}_k - \hat{y}^-_k)^T / (2n)
  \]  
  (35)

  \[
P^+_k = P^-_k - K_k P_y K_k^T, \quad P_y = \sum_{i=1}^{2n} (\hat{y}^{(i)}_k - \hat{y}^-_k)(\hat{y}^{(i)}_k - \hat{y}^-_k)^T / (2n) + R_k
  \]  
  (36)

  \[
  \hat{x}^+_k = \hat{x}^-_k + K_k(y_k - \hat{y}_k).
  \]  
  (37)
Ensemble Kalman Filter (EnKF)

- EnKF generates $N$ vectors of initial conditions ($\hat{x}_1^+$) and feeds these into the model given by (21). Model started from different initial conditions is referred to as different ensemble members.
- Mean states estimate is calculated via $\mu \hat{x}_k^- = \sum_{i=1}^{N} \frac{\hat{x}_{k,i}^-}{N}$.
- $h_k(\cdot)$ still needs to be known explicitly and its linearization $H_k$ needs to be computed about ensemble mean.
- Accordingly, $N$ EnKF ensemble members can be updated via\textsuperscript{4}

$$\hat{x}_{k,i}^- = f_{k-1}(\hat{x}_{k-1,i}^+, u_{k-1}, 0),$$

$$P_k^- = \sum_{i=1}^{N} (\hat{x}_{k,i}^- - \mu \hat{x}_k^-) (\hat{x}_{k,i}^- - \mu \hat{x}_k^-)^T / (N-1),$$

$$K_k = P_k^- H_k^T (H_k P_k^- H_k^T + M_k R_k M_k^T)^{-1},$$

$$P_k^+ = (I - K_k H_k) P_k^-,$$

$$\hat{x}_{k,i}^+ = \hat{x}_{k,i}^- + K_k \left[ y_k - h_k(\hat{x}_{k,i}^-, 0) \right].$$

First, define joint state-observation vector as

\[ z_k = \begin{bmatrix} x_k \\ y_k \end{bmatrix}. \]  

(43)

Recall that \( x_k \in \mathbb{R}^n, y_k \in \mathbb{R}^m \). We then define \( \mathcal{H} \in \mathbb{R}^{m \times (n+m)} \) such that \( y_k = \mathcal{H}z_k \), i.e. \( \mathcal{H} = [0_{m \times n} I_{m \times m}] \).

Accordingly, \( N \) EAKF ensembles can be updated via

\[
\hat{z}_{k,i}^- = [f_{k-1}(\hat{x}_{k-1,i}^+, u_{k-1}, 0); h_k(\hat{x}_{k-1,i}^+, 0)],
\]

(44)

\[
P_k^- = \sum_{i=1}^N (\hat{z}_{k,i}^- - \mu_{\hat{z}_k^-})(\hat{z}_{k,i}^- - \mu_{\hat{z}_k^-})^T / (N - 1),
\]

(45)

\[
\mathcal{A}_k = (\mathcal{F}_k^T)^{-1} \mathcal{G}_k^T U_k^T (U_k^T)^{-1} B_k^T (\mathcal{G}_k^T)^{-1} \mathcal{F}_k^T,
\]

(46)

\[
P_k^+ = \left[ (P_k^-)^{-1} + \mathcal{H}^T R_k^{-1} \mathcal{H} \right]^{-1}, \quad \mu_{\hat{z}_k^+} = P_k^+ \left[ (P_k^-)^{-1} \mu_{\hat{z}_k^-} + \mathcal{H}^T R_k^{-1} y_k \right],
\]

(47)

\[
\hat{z}_{k,i}^+ = \mathcal{A}_k^T (\hat{z}_{k,i}^- - \mu_{\hat{z}_k^-}) + \mu_{\hat{z}_k^+}.
\]

(48)

---


Filter Divergence

- Consider linear system

\[
x_k = 0.5x_{k-1} + u_{k-1}, \quad u_k = 1.0 + \sin(0.5k),
\]
\[
y_k = x_k + v_k, \quad v_k \sim N(0, 0.2).
\]

- If the driver varies with time but we assume it is constant, ensemble might collapse.
Ensemble Inflation

Ensemble inflation given by

\[ \hat{x}_k = \sqrt{\lambda}(\hat{x}_k - \mu[\hat{x}_k]) + \mu[\hat{x}_k], \]

with \( \lambda = 2.0 \) alleviates filter divergence.
Inflating driver by

\[
\hat{u}_k = \frac{\sqrt{\sigma_i^2}}{\sqrt{\sigma^2[k]}}(\hat{u}_k - \mu[\hat{u}_k]) + \mu[\hat{u}_k],
\]

with \(\sigma_i^2 = 0.12\) removes the lag in the driver estimate.
1 Summary

2 Problem Statement

3 GITM

4 EAKF

5 Assimilating CHAMP Neutral Density
   - Simulated CHAMP data
   - Real CHAMP data

6 Assimilating GPS Total Electron Content

7 Conclusions and Future Work
CHAMP and GRACE orbits
Sim: Does actual density track the desired density?

- **Measurement source:** \( \rho \) from GITM truth simulation \((F_{10.7} = 148)\) is interpolated to the CHAMP location.
- **Ensemble members:** 20 GITM instances with \( F_{10.7} = N(130, 25) \)
Same plot, but orbit averages
Sim: Does error at the measurement location go to zero?

Define RMS percentage error \( \text{RMSPE} \triangleq \frac{\sqrt{(\rho - \hat{\rho})^2}}{\sqrt{\rho^2}} \)
Sim: Does error at the validation location go to zero?

RMSPE (2nd day) along GRACE path = 4%

GITM without EAKF
EAKF posterior

Absolute percentage error along GRACE path

(b) Hours since 00UT 01/12/2002

AV Morozov, UM Data Assimilation and Driver Estimation
Sim: Does $F_{10.7}$ estimate converge to the true value?
Real: Does actual density track the desired density?

- **Measurement source:** real CHAMP $\rho$ with real CHAMP uncertainty
- **Ensemble members:** 20 GITM instances with $F_{10.7} = N(130, 25)$
Real: Does error at the measurement location go to zero?

RMSPE (2nd day) along CHAMP path ≈ 7%
Real: Does error at the validation location go to zero?

\[ \text{RMSPE (2nd day) along GRACE path} = 52\% \]

 GITM without EAKF
 EAKF posterior

(d) Hours since 00UT 01/12/2002
Absolute percentage error along GRACE path

RMSPE (2nd day) along GRACE path = 52%
Real: Does $F_{10.7}$ estimate converge to the NOAA value?
Summary

Problem Statement

GITM

EAKF

Assimilating CHAMP Neutral Density

Assimilating GPS Total Electron Content
  - Simulated TEC data
  - Real TEC data

Conclusions and Future Work
Real TEC data
Simulated TEC data

![Map of Simulated TEC data with color scale representing TEC in [TECU].]
Sim: True and Estimated TEC

2002-12-01 00:30:00
Sim: TEC error
Sim: average error without driver estimation

- **Average VTEC error**
- **Average VTEC spread (± SD)**

![Graph showing VTEC error and spread over hours since 00UT 12/1/2002]

- **F10.7**
  - **measured**
  - **estimated**
  - **estimated ± SD**

![Graph showing F10.7 values over hours since 00UT 12/1/2002]
Real: True and Estimated TEC
Real: TEC error
Real: average error and driver

Graph 1: Average VTEC error and spread 

Graph 2: Estimated $F_{10.7}$
Real: average error without driver estimation
1 Summary

2 Problem Statement

3 GITM

4 EAKF

5 Assimilating CHAMP Neutral Density

6 Assimilating GPS Total Electron Content

7 Conclusions and Future Work
Conclusions

1. **DART-GITM interface**: This dissertation developed the framework for data assimilation of satellite data into a space weather model. In particular, it introduced driver estimation and demonstrated that drivers need to be inflated differently from the model states.

2. **Differential inflation**: The driver estimate was inflated differently from the rest of the variables, that is its spread was set to be constant to allow for continuous updating.

3. **CHAMP density assimilation**: The data assimilation technique was first demonstrated by using sparse thermospheric measurements.

4. **GPS TEC assimilation**: The interface was then modified to handle more global (ionospheric) measurements coming from the GPS satellites. It was found that the driver estimation in this more global case was not needed.
1. **Localization**: $F_{10.7}$ estimate currently affects both the day-side and the night-side. It would be interesting to see the effect of localizing $F_{10.7}$.

2. **Slant TEC**: Only vertical total electron content measurements were considered so far. Implementing slant TEC would allow us to solve a broader class of problems.

3. **Heating efficiency**: Heating efficiency is a stronger driver for TEC, so estimating it using TEC data should be easier than estimating $F_{10.7}$.

4. **Geomagnetic storms**: This study only considered geomagnetic quiet times, so performing data assimilation during geomagnetic storms is subject of future work.
Questions?
Consider the multi-input, multi-output discrete-time system

\[
x(k + 1) = Ax(k) + Bu(k) + D_1 w(k), \\
y(k) = Cx(k) + Du(k) + D_2 w(k), \\
z(k) = E_1 x(k) + E_2 u(k) + E_0 w(k).
\]

(53, 54, 55)

Our goal is to develop an adaptive controller that generates a control signal \( u \) that minimizes the performance \( z \) in the presence of the exogenous signal \( w \). For this presentation we consider SISO plants with no direct feedthrough and \( z(k) = y(k) \) in command following context (i.e. \( E_1 = C, E_2 = D = 0, E_0 = D_2 \neq 0, D_1 = 0 \)). We define Markov parameters as \( H_i \triangleq CA^{i-1}B \) for \( i > 0 \).

\[
z(k) = \sum_{i=1}^{n} -\alpha_i z(k - i) + \sum_{i=d}^{n} \beta_i u(k - i) + \sum_{i=0}^{n} \gamma_i w(k - i),
\]

(56)

\[
u(k) \triangleq \sum_{i=1}^{n_c} M_i(k) u(k - i) + \sum_{i=1}^{n_c} N_i(k) y(k - i), \quad u(k) = \theta(k) \phi(k), \quad \hat{u}(k) = \hat{\theta}(k) \phi(k),
\]

(57)

\[
Z(k) \triangleq \begin{bmatrix} z(k - 1) \\ \vdots \\ z(k - p_c) \end{bmatrix}, \quad U(k) \triangleq \begin{bmatrix} u(k - 1) \\ \vdots \\ u(k - p_c) \end{bmatrix}, \quad \hat{U}(k) \triangleq \begin{bmatrix} \hat{u}(k - 1) \\ \vdots \\ \hat{u}(k - p_c) \end{bmatrix},
\]

(58)

\[
\bar{B}_{zu} \triangleq [ 0_d \quad H_d \times \text{poly}(NMPz) ], \text{ or } \bar{B}_{zu} = [ 0_d \quad H_d \quad H_{d+1} ],
\]

(59)

\[
\hat{Z}(\hat{\theta}(k), k) \triangleq Z(k) - \bar{B}_{zu} \begin{bmatrix} U(k) \\ \hat{U}(k) \end{bmatrix},
\]

(60)

where green highlighting represents known variables, yellow - measured variables, and red - variables to be found.
We now consider the cost function

\[ J(\hat{\theta}, k) \triangleq \hat{Z}^T(\hat{\theta}, k)\hat{Z}(\hat{\theta}, k) + \zeta(k)\text{tr}\left[ (\hat{\theta} - \theta)^T(\hat{\theta} - \theta) \right], \quad (61) \]

where the positive scalar \( \zeta(k) \) is the learning rate. Substituting (60) into (61), the cost function can be written as the quadratic form

\[ J(\hat{\theta}, k) = \left( \text{vec} \hat{\theta} \right)^T A(k) \text{vec} \hat{\theta} + B(k)^T \text{vec} \hat{\theta} + C(k), \quad (62) \]

where

\[ D(k) \triangleq \sum_{i=1}^{p_c} \phi^T(k-i) \otimes (\bar{B}_{zu}L_i), \]

\[ f(k) \triangleq Z(k) - \bar{B}_{zu}U(k), \]

\[ A(k) \triangleq D^T(k)D(k) + \zeta(k)I_{nc \tilde{l}_u(l_u+l_y)}, \]

\[ B(k) \triangleq 2D^T(k)f(k) - 2\zeta(k)\text{vec} \theta(k), \]

\[ C(k) \triangleq f(k)^Tf(k) + \zeta(k)\text{tr}\left[ \theta^T(k)\theta(k) \right]. \quad (63) \]

Since \( A(k) \) is positive definite, \( J(\hat{\theta}, k) \) has the strict global minimizer

\[ \hat{\theta}(k) = -\frac{1}{2} \text{vec}^{-1}(A(k))^{-1}B(k). \quad (64) \]

The controller gain update law is \( \theta(k+1) = \hat{\theta}(k) \).
Consider the discrete-time system

\[ G(z) = \frac{z - 1.4}{(z - 0.5)(z - 0.6)(z - 0.7)}. \] (65)

With NMP zero location known exactly RCAC achieves transient performance of about 1.6. Transient and steady state performances are defined as

\[ z_{tr} = \max_k |z(k)|, \] (66)

\[ z_{ss} = \max_{k=900:1000} |z(k)|. \] (67)
When location of the nonminimum phase zero is uncertain, transient performance can become unbounded.

In this plot, the true nonminimum phase zero (x-axis) and zero estimate (y-axis) are varied from 1.1 to 8.1. We conclude that RCAC is more robust to overestimating the location of the nonminimum phase zero than to underestimating it. Additionally, the cases with NMP zeros further out on the real axis result in greater stability margins than those with NMP zeros closer to 1.
We now generate 50 stable second order plants with random poles and NMP zero at 2 and test RCAC with NMP zero estimates varying from 1.4 to 10.

For these 50 random plants, the stability of the closed-loop system is less sensitive to overestimating the location of the NMP zero than it is to underestimating the location of the NMP zero. However, none of the 50 plants have an upward margin in excess of 10, which corresponds to 5 times the true value of the NMP zero.
We can extend these findings to plants with higher order and bigger relative degree.

We find that robustness to the estimate of the NMP zero increases in rows from left to right (with decreasing order), and in columns from bottom to top (with increasing relative degree).
Convex Constraint on Pole Locations

The denominator coefficients of the controller (60) are given by
\[
\text{den}(\theta(k)) \triangleq [1 - M_1 - M_2 \cdots - M_{nc}]. \tag{68}
\]

We modify the problem of minimizing (62) by imposing a maximum singular value constraint on the companion-form matrix

\[
K \triangleq \begin{bmatrix}
M_1 & M_2 & \cdots & M_{nc-1} & M_{nc} \\
1 & 0 & \cdots & 0 & 0 \\
0 & 1 & \cdots & 0 & 0 \\
\vdots & \vdots & \ddots & \vdots & \vdots \\
0 & 0 & \cdots & 1 & 0
\end{bmatrix}, \tag{69}
\]

\[
\sigma_{\text{max}}(K) \leq \gamma, \gamma > 1. \tag{70}
\]
Convex Constraint on Pole Locations continued

Convex constraint improves transient performance

RCAC

CC-RCAC
Evolution of the controller poles is shown as a function of time in terms of color. CC-RCAC poles settle twice as fast.
Consider planar multilink arm

\[
\begin{bmatrix}
\frac{m_1 l_1^2}{3} + m_2 l_2^2 & \frac{m_2 l_1 l_2}{2} \cos(\theta_1 - \theta_2) \\
\frac{m_2 l_1 l_2}{2} \cos(\theta_1 - \theta_2) & \frac{m_2 l_2^2}{3}
\end{bmatrix}
\begin{bmatrix}
\ddot{\theta}_1 \\
\ddot{\theta}_2
\end{bmatrix}
+ \begin{bmatrix}
c_1 + c_2 & -c_2 \\
-c_2 & c_2
\end{bmatrix}
\begin{bmatrix}
\dot{\theta}_1 \\
\dot{\theta}_2
\end{bmatrix}
+ \begin{bmatrix}
k_1 + k_2 & -k_2 \\
-k_2 & k_2
\end{bmatrix}
\begin{bmatrix}
\theta_1 \\
\theta_2
\end{bmatrix}
= \begin{bmatrix}
u(t) \\
0
\end{bmatrix}
\]

(71)
Nonlinear Multilink Arm continued

Nonlinear system can be controlled for small command magnitudes and frequencies.

RCAC performance at empirically found maximum command amplitude.

Region plot for RC-MRAC, taken from [1]

Vertical Energy Equation

\[
\frac{\partial \mathcal{T}}{\partial t} + u_r \frac{\partial \mathcal{T}}{\partial r} + (\gamma - 1) \mathcal{T} \left( \frac{2ur}{r} + \frac{\partial u_r}{\partial r} \right) = \frac{k}{c_v \rho \bar{m}_n} \mathcal{Q}, \tag{72}
\]

\[
\mathcal{Q} = Q_{EUV} + Q_{NO} + Q_O + \frac{\partial}{\partial r} \left( (\kappa_c + \kappa_{\text{eddy}}) \frac{\partial T}{\partial r} \right) + N_e \bar{m}_i \bar{m}_n \nu_{in} (v - u)^2,
\]

\[
Q_{EUV} = \sum_s \sum_\lambda \left[ N_s(z) \, I_\infty(\lambda) \, e^{-\sec(\chi)} \sum_s N_s(z) \sigma_s^a(\lambda) H_s(z) \right], \tag{73}
\]

\[
I_\infty(\lambda) = f(\lambda) \left[ 1 + a(\lambda) \left( \frac{F_{10.7}}{2} + \left\langle F_{10.7} \right\rangle_{81d} + 80 \right) \right], \tag{74}
\]

where equation (73) is a combination of equation (9.17)\(^7\) and notes from AOSS 495. Equation (74) is an empirical model.\(^8\)

---


First, some notes on matrices that were not defined above.

1. $\mathcal{F}_k$ comes from SVD of $\mathcal{P}_k = \mathcal{F}_k D_k \mathcal{F}_k^T$.
2. $G_k$ is a square root of $D_k$, as in $G_k = D_k^{1/2}$.
3. $U_k$ comes from SVD of $G_k^T \mathcal{F}_k^T \mathcal{H}^T R^{-1} \mathcal{H} \mathcal{F}_k G_k = U_k J_k U_k^T$.
4. $B_k$ is a square root of $I + J_k$, as in $B_k = (I_{n+m} + J_k)^{-1/2}$.

Second, here are some notes

- The EAKF procedure presented so far is not exactly what is implemented in DART.
- The procedure presented here is the closest EAKF representation to other filters, but does not incorporate localization and is not optimized for parallel implementation.
- The version that is implemented in DART is described in 9 and 10.

---

Results

- Simulated measurement above Ann Arbor
- Simulated measurement at Subsolar Point
- Simulated measurement at CHAMP location
- Real measurement from CHAMP
- Real measurement from CHAMP with advanced IMF and HPI models
Simulated measurement above Ann Arbor

- **Measurement source:** GITM truth simulation with an input of $F_{10.7}$ fixed at about 150. $\rho$ at (82.5°W, 42.5°N, 394km) is recorded with associated uncertainty ($R$) of $2.6 \times 10^{-12} kg/m^3$.

- **Ensemble members:** 20 GITM instances prespun for 2 days prior to Dec 01 with $\hat{F}_{10.7}$ values coming from normal distribution $\sim N(130, 25)$. $\hat{F}_{10.7}$ is inflated using $P_i = 49$. 

![Diagram](attachment:diagram.png)
EAKF estimates of $\rho$ are brought within the uncertainty bounds of GITM truth simulation.
The effect of measurement assimilation can be restricted to a region to avoid updating uncorrelated states.\textsuperscript{11}

Correlation function with horizontal cutoff of $30^\circ$ is shown to the right and below, and vertical cutoff of $100\text{km}$ is shown bottom right.

\textsuperscript{11}Gaspari, G., and S. E. Cohn. “Construction of correlation functions in two and
Note, $\rho$ at subsolar point does not vary too much since $F_{10.7}$ is relatively constant. The localized AA measurement is tripled in intensity.
ρ at CHAMP location

The localized AA measurement is tripled in intensity.
We define RMS percentage error $\text{RMSPE} = \sqrt{\frac{(\rho - \hat{\rho})^2}{\rho^2}}$.
Simulated measurement above Ann Arbor Summary

RMSPE (2nd day) along CHAMP path = 24%

GITM without EAKF
EAKF posterior

RMSPE (2nd day) along GRACE path = 41%

GITM without EAKF
EAKF posterior
So what we learned is that it is hard to get a good estimate of $F_{10.7}$ based on measurement that is fixed in longitude.
Simulated measurement at Subsolar Point

**Measurement source:** $\rho$ at *subsolary point* is recorded with associated uncertainty ($R$) of $2.6 \times 10^{-12} \text{kg/m}^3$. 

---

**Figure:**

- **Left:** Graph showing mass density $\times 10^{-12}$ (kg m$^{-3}$) over time.
  - Blue line: EAKF ensemble mean.
  - Black dashed line: GITM truth simulation.
  - Red solid line: EAKF posterior.
  - Pink shaded area: EAKF ensemble mean $\pm$ SD.
  - Dot line: SP measurement localized.

- **Right:** Graph showing absolute percentage error along CHAMP path.
  - Blue line: GITM without EAKF.
  - Blue dashed line: EAKF posterior.
  - RMSPE (2nd day) along CHAMP path = 3%.

- **Bottom Left:** Graph showing mass density $\times 10^{-12}$ (kg m$^{-3}$) over time.
  - Blue line: EAKF ensemble mean.
  - Black dashed line: GITM without EAKF.
  - Purple line: GITM truth simulation.
  - Blue dashed line: GITM truth simulation $\pm$ SD.

- **Bottom Right:** Graph showing absolute percentage error along GRACE path.
  - Blue line: GITM without EAKF.
  - Blue dashed line: EAKF posterior.
  - RMSPE (2nd day) along GRACE path = 4%.
Conclusion here is that $\rho$ at subsolar point is more closely related to $F_{10.7}$ than $\rho$ at any point fixed in longitude (for example, Ann Arbor).
Simulated measurement at CHAMP location

- **Measurement source:** $\rho$ at *CHAMP location* is recorded with associated uncertainty ($R$) of $2.6 \times 10^{-12} \text{kg/m}^3$. 

![Graphs showing mass density along CHAMP and GRACE paths with RMSPE (2nd day) along CHAMP path = 2% and along GRACE path = 4%]
We conclude that this case is better conditioned than AA or SP ones.
**Measurement source:** \( \rho \) from *real CHAMP* has associated uncertainty \( (R_k) \) that varies, but has a mean of \( 2.6 \times 10^{-12} \text{kg/m}^3 \).
We conclude that EAKF $F_{10.7}$ estimate did not converge to the commonly accepted value in order to compensate for model bias.
Real measurement from CHAMP with IMF and HPI

- **Measurement source**: \( \rho \) from *real CHAMP* has associated uncertainty \( (R_k) \) that varies, but has a mean of \( 2.6 \times 10^{-12} \text{kg/m}^3 \).
We conclude that EAKF $F_{10.7}$ estimate did not converge to the commonly accepted value in order to compensate for model bias, even more advanced IMF and HPI models are used.