Enabling CIG geomagnetic data assimilation: Algorithm, Framework and Implementation

W. Kuang
Planetary Geodynamics Lab, NASA GSFC

Weiyuan Jiang, Zhibin Sun, Andy Tangborn

• Data assimilation algorithm
• Framework structure
• Current Progress and Near future development
Summary and Decisions

(1) A workshop will be proposed for summer 2012, with focus on how to develop a massively parallel geodynamo code. International participation, to include numerical methods experts and programmers as well as geodynamo modellers. [P. Olson]

(2) Leeds' pseudospectral geodynamo code will be donated to CIG, as agreed 2 years ago, when a suitable programmer can be assigned to the task to work with David Gubbins

(3) A Data Assilimilation code will be donated to CIG in modular form for use with users' codes [Weijia Kuang]

(4) CIG's Graphics capability will be explored for application to the output from existing geodynamo codes [Bruce Buffett and Hiroo Matsui]
**Assimilation Algorithm**

**Dynamo model**

\[
\frac{\partial \mathbf{x}}{\partial t} = \mathbf{M}(\mathbf{x}, \alpha)
\]

\[
\mathbf{x}(t_0) = \mathbf{x}_0
\]

**x**: state vector

**α**: system parameters

**Ensemble-based sequential data assimilation**

\[
\frac{\partial \mathbf{x}^f}{\partial t} = \mathbf{M}(\mathbf{x}^f, \alpha)
\]

\[
\mathbf{x}^f(t_a) = \mathbf{x}^a
\]

\[
\mathbf{x}^a = \mathbf{x}^f + \mathbf{K}(\mathbf{y} - \mathbf{Hx}^f)
\]

\[
\mathbf{K} = \mathbf{P}^f \mathbf{H}^T \left[ \mathbf{H} \mathbf{P}^f \mathbf{H}^T + \mathbf{R} \right]^{-1}
\]

\[
\mathbf{P}^f = \left\langle (\mathbf{x}^f - \mathbf{x}^t)(\mathbf{x}^f - \mathbf{x}^t)^T \right\rangle 
\]

\[
\approx \left\langle (\delta \mathbf{x}^f)(\delta \mathbf{x}^f)^T \right\rangle
\]
1. Making dynamo models as independent and externally-operated executable

2. Making DAS framework stand-alone and flexible (accommodating different dynamo models)

3. Leaving scalability for community specification
Assimilation Framework

Work flow

Assimilation Framework
- Perturbation
- Covariance and Gain Matrix

Dynamo model
- Return run
- $x^f$
- $x^a$
**Assimilation Framework**

**Framework Structure**

- **DAS Algorithm Block**
  1. Perturbation generation
  2. Calculate $P^f$ and $K$

- **Dynamo Model Environment Block**
  - Recycle dynamo model components

- **Interface Block**
  1. Data I/O
  2. Definition conversion

- **Independent algorithm development**
  - “black-box”

**CIG Geodynamo workshop, Boulder, Oct.08-10,2012**
Assimilation Framework

CIG Geodynamo workshop, Boulder, Oct.08-10,2012
modif dynamo parameters data file
(change two parameters: 'path_in' and 'path_out')

sys_command = trim('sed ''s///''/path_in/c path_in''
''/''//trim(new_path_in)/''/''/'' ''//
trim(EnK_dynamo_script_dir)/''/''/''//
trim(EnK_dynamo_param_filename)/''/backup > '/'
trim(EnK_dynamo_script_dir)/''/''/''//
trim(EnK_dynamo_param_filename)/''/tmp')

write(0,0) trim(sys_command)
call SYSTEM(trim(sys_command))

sys_command = trim('sed ''s///''/path_out/c path_out''
''/''//trim(new_path_out)/''/''/'' ''//
trim(EnK_dynamo_script_dir)/''/''/''//
trim(EnK_dynamo_param_filename)/''/tmp > '/'
trim(EnK_dynamo_script_dir)/''/''/''//
trim(EnK_dynamo_param_filename)

write(0,0) trim(sys_command)
call SYSTEM(trim(sys_command))

Run MPI dynamo job with alias=EnK_MPI_alias_name

write(str_tmp,0) k_ens+1

sys_command = trim('sed ''s///''/mpiexec/mpiexec -a ''/''//
trim(EnK_MPI_alias_name)/''/case/''/''/''
trim(adjust(str_tmp))/''/''/'' ''//
trim(EnK_dynamo_script_dir)/''/''/''//
trim(EnK_dynamo_script_filename)/''/''/''/''
trim(EnK_dynamo_script_filename)

write(0,0) trim(sys_command)
Assimilation Framework

Flexibility

The state vectors can be defined differently in DAS Framework and in dynamo models

\[ x_{DA} : \text{state vector defined in DAS framework} \]
\[ x_{DM} : \text{state vector defined in dynamo model} \]
\[ (T, T') : \text{transform/inverse transform} \]

\[ x_{DA} = T \cdot x_{DM} \]
\[ x_{DM} = T' \cdot x_{DA} \]
Flexibility

The state vectors can be defined differently in DAS Framework and in dynamo models

\[
x_{DA}^{a} = x_{DA}^{f} + K_{DA} \cdot \left( y_{DA}^{o} - H_{DA} \cdot x_{DA}^{f} \right)
\]

\[
x_{DM}^{a} = x_{DM}^{f} + K_{DM} \cdot \left( y_{DM}^{o} - H_{DM} \cdot x_{DM}^{f} \right)
\]

\[
x_{DM}^{a} = T \cdot x_{DA}^{a} \quad \text{(if export analysis)}
\]

\[
\begin{cases}
  K_{DM} = T \cdot K_{DA} \cdot T^{I} \\
  H_{DM} = T \cdot H_{DA} \cdot T^{I}
\end{cases} \quad \text{(if export Gain matrix)}
\]
Current Progress and near future development

Progress

1. Initial covariance with random perturbations
2. Gain matrix with self-correlation (without observation error)
3. Tested with mMoSST model
Original dynamo solutions

Random perturbations
Gain matrix for axisymmetric poloidal field

Real $K(\text{cob}), m=0$
Gain matrix for order $m = 1$ poloidal field
Current Progress and near future development

Progress

1. Initial covariance with random perturbations
2. Gain matrix with self-correlation (without observation error)
3. Tested with mMoSST model

Future development

4. Modeled observation error
5. Modeled time-varying forecast covariance
6. Transforms for CIG repository dynamo models
7. Hybrid forecast/analysis covariance