An Ensemble Algorithm Based Component for Geomagnetic Data Assimilation

Zhibin Sun[1, *] and Weijia Kuang[2]

[1] *USDA UV-B Monitoring and Research Program, Natural Resource Ecology Laboratory, Colorado State University, Fort Collins, CO, USA*
[2] *Planetary Geodynamics Laboratory, NASA Goddard Space Flight Center, Greenbelt, MD, USA*

ABSTRACT

Geomagnetic data assimilation is one of the most recent developments in geomagnetic studies. It combines geodynamo model outputs and surface geomagnetic observations to provide more accurate estimates of the core dynamic state and provide accurate geomagnetic secular variation forecasting. To facilitate geomagnetic data assimilation studies, we develop a stand-alone data assimilation component for the geomagnetic community. This component is used to calculate the forecast error covariance matrices and the gain matrix from a given geodynamo solution, which can then be used for sequential geomagnetic data assimilation. This component is very flexible and can be executed independently. It can also be easily integrated with arbitrary dynamo models.

1. INTRODUCTION

The Earth possesses a magnetic field of internal origin (geomagnetic field). It is now widely accepted that this magnetic field is generated and maintained by convective flow in the Earth fluid outer core (geodynamo). Surface geomagnetic observations over the past several centuries, as well as paleomagnetic and acheomagnetic records, have demonstrated that the geomagnetic field varies in broad spatial and temporal scales (Jackson et al. 2000; Sabaka et al. 2004; Korte and Constable 2008; Olsen et al. 2010). Such variations are the manifestation of dynamic processes in the Earth's liquid outer core, and are therefore critical for understanding the interior, the evolution and the interactions among the different Earth components.

In addition to surface geomagnetic observations, various field and dynamic models have been utilized to understand and interpret geomagnetic variability. Among them are various dynamo models in which the convection in the outer core is described by the Navier-Stokes equation, while the equations describing magnetic induction and en-

ergy conservation (when combined these are called the dynamo equations). In the dynamo models the outer core fluid properties are also defined using several non-dimensional parameters. These equations are then solved using specified boundary conditions. For more detail we refer the reader to Kono and Roberts (2002).

Numerical dynamo models have been used successfully for qualitative understanding of geomagnetic secular variation. However, due to limited knowledge of the Earth's outer core, and computational constraints, these models are still far from the "truth". This is the main reason for substantial discrepancies between the observed geomagnetic field and those from dynamo simulations (Kuang et al. 2009). Therefore, a fundamental question arises: how could one obtain the most accurate estimate of the core dynamic state with currently available information?

One answer to this question is geomagnetic data assimilation: using geomagnetic observations to constrain and improve numerical dynamo models. Generally speaking, data assimilation is a methodology in which observational data are assimilated into numerical model outputs for better estimation of the true physical states. Data assimilation has been widely used in atmospheric (e.g., Talagrand 1997) and

---
* *Corresponding author*
  E-mail: sunzhib1@gmail.com

oceanic modeling (e.g., Ghil and Malanotte-Rizzoli 1991, Sun et al. 2013). There are two types of assimilation methods: sequential and variational data assimilation. Detailed descriptions of the algorithms and history of data assimilation can be found in, e.g., Kalnay (2003). Both methods have been used in geomagnetic data assimilation studies (Fournier et al. 2007; Liu et al. 2007; Sun et al. 2007; Kuang et al. 2008, 2010; Fournier et al. 2010), with very promising results. However, extensive studies are needed to understand both the responses of numerical dynamo models to observational constraints and the consistencies of the parameters and approximations used in dynamo simulation with geomagnetic observations. Geomagnetic data assimilation is also computationally very expensive. The required computing resources could be orders of magnitude more than those for numerical dynamo simulations. There are a variety of dynamo models currently used by different research groups, thus calling for algorithm portability among these dynamo models.

One approach to solve these conflicts between research demands and computational constraints is to develop an independent assimilation component. It could then be used in the geomagnetic community. With this component community efforts can be cohesively utilized to address the fundamental questions in geomagnetic data assimilation.

In this paper we present our geomagnetic component based on an ensemble sequential assimilation algorithm. This paper is organized as follows: in section 2, the assimilation algorithm is summarized. In section 3, the basic structure and workflow of the component are described. Numerical examples are given in section 4, followed by concluding remarks in section 5.

2. ALGORITHM

There are currently many dynamo models that differ in many aspects, including boundary conditions, algorithms and geophysical approximations. Regardless, these models can be symbolically written as the following initial-value problem,

$$\frac{\partial \mathbf{x}}{\partial t} = \mathbf{F}(\mathbf{x}, t)$$
$$\mathbf{x}_0 = \mathbf{x}(t_0) \tag{1}$$

where $\mathbf{x}$ is the state vector (an array describing the dynamo state in the outer core), $\mathbf{F}$ is the generalized force governing the time evolution of $\mathbf{x}$, and $\mathbf{x}_0$ is the initial sate at t_0.

The numerical modeling objective is that the model output $\mathbf{x}^f$ (called "forecast" in data assimilation) can be a good approximation of truth $\mathbf{x}^t$. However, it often falls short in achieving this objective for many reasons, such as large model errors, inaccurate initial guesses, turbulent dynamic processes, etc. To reduce the difference between the two, one approach is the sequential data assimilation in which

the model output $\mathbf{x}^f$ and (often partial) observations $\mathbf{y}^o$ are combined to obtain a better approximation $\mathbf{x}^a$ (called analysis) of truth $\mathbf{x}^t$:

$$\mathbf{x}^a = (1 - \mathbf{W})\mathbf{x}^f + \mathbf{W}\mathbf{y}^o \tag{2}$$

The key is to find an appropriate weight $\mathbf{W}$ in Eq. (2) such that $\mathbf{x}^a$ is the optimal estimation of $\mathbf{x}^t$. The analysis $\mathbf{x}^a$ is then used as the new initial state of Eq. (1) to obtain a more accurate forecast in the future. If the domain of $\mathbf{y}^o$ is different from that of $\mathbf{x}^f$, a mapping is then needed in Eq. (2). In general, the analysis Eq. (2) can be derived whenever both observations and forecasts are available.

The mathematical reasoning can be summarized as follows. There are two different sources of information about the truth: one is the model output $\mathbf{x}^f$ (often called *a priori* information) and the other is the observation $\mathbf{y}^o$:

$$\mathbf{z} = \begin{bmatrix} \mathbf{x}^f \\ \mathbf{y}^o \end{bmatrix} \tag{3}$$

Both are different from the truth,

$$\mathbf{x}^f = \mathbf{x}^t + \mathbf{e}^f, \quad \mathbf{y}^o = \mathbf{H}\mathbf{x}^t + \mathbf{e}^o \tag{4}$$

where $\mathbf{H}$ is the observational operator (a mapping between the observation domain and the truth). The errors $\mathbf{e}^f$ and $\mathbf{e}^o$ are often assumed to have Gaussian random distributions with zero mean and variances $\mathbf{P}^f$ and $\mathbf{R}$, respectively. It is often appropriate to assume that both are independent (i.e., uncorrelated). Thus,

$$\mathbf{z} = \begin{bmatrix} \mathbf{x}^t \\ \mathbf{H}\mathbf{x}^t \end{bmatrix} + \begin{bmatrix} \mathbf{e}^f \\ \mathbf{e}^o \end{bmatrix} = \mathbf{z}^t + \mathbf{e}^z \tag{5}$$

where the error $\mathbf{e}^z$ is also a Gaussian distribution with zero mean and the variance,

$$\Sigma = \begin{bmatrix} \mathbf{P}^f & 0 \\ 0 & \mathbf{R} \end{bmatrix} \tag{6}$$

To obtain the optimal estimation of truth $\mathbf{x}^t$ from $\mathbf{z}$, a linear unbiased estimator $\mathbf{x}^a = \mathbf{A}\mathbf{z}$ is proposed with the transform $\mathbf{A}$ chosen to minimize the trace of the error covariance,

$$\min(Tr\{E[(\mathbf{x}^a - \mathbf{x}^t)(\mathbf{x}^a - \mathbf{x}^t)^T]\}) \tag{7}$$

where $E(\cdot)$ denotes the expectation of an ensemble. This leads to

$$\mathbf{A} = (\Gamma^T \Sigma^{-1} \Gamma)^{-1} \Gamma^T \Sigma^{-1}, \quad \Gamma = \begin{bmatrix} \mathbf{I} \\ \mathbf{H} \end{bmatrix} \tag{8}$$

where $\mathbf{I}$ is the identity matrix. Thus,

$$\mathbf{x}^a = (\Gamma^T \Sigma^{-1} \Gamma)^{-1} \Gamma^T \Sigma^{-1} \mathbf{z} \tag{9}$$

Using Eqs. (3), (6), (8), and (9),

$$\mathbf{x}^a = \mathbf{x}^f + \mathbf{K}(\mathbf{y}^o - \mathbf{H}\mathbf{x}^f) \tag{10a}$$

$$\mathbf{K} = [(\mathbf{P}^f)^{-1} + \mathbf{H}^T \mathbf{R}^{-1} \mathbf{H}]^{-1} \mathbf{H}^T \mathbf{R}^{-1} \tag{10b}$$

In data assimilation $\mathbf{K}$ is called the gain matrix, and $\mathbf{y}^o - \mathbf{H}\mathbf{x}^f$ is called the innovation. Equation (10b) can be also interpreted as follows: consider the sum of the distances (weighted by the error variance) from $\mathbf{x}^a$ to the forecast $\mathbf{x}^f$, and from $\mathbf{x}^a$ to the observation $\mathbf{y}^o$

$$J(\mathbf{x}^a) = (\mathbf{x}^a - \mathbf{x}^f)^T (\mathbf{P}^f)^{-1} (\mathbf{x}^a - \mathbf{x}^f) + \\ (\mathbf{y}^o - \mathbf{H}\mathbf{x}^a)^T \mathbf{R}^{-1} (\mathbf{y}^o - \mathbf{H}\mathbf{x}^a) \tag{11}$$

If $\mathbf{K}$ is chosen to be Eq. (10b), then the distance J is the minimum. It can also be shown that Eq. (10b) is equivalent to (Kalnay 2003),

$$\mathbf{K} = \mathbf{P}^f \mathbf{H}^T (\mathbf{H}\mathbf{P}^f \mathbf{H}^T + \mathbf{R})^{-1} \tag{12}$$

Equations (10a) and (12) are used in our system. There are two extreme scenarios in Eq. (12): $\mathbf{K} \approx \mathbf{I}$ if $\|\mathbf{P}^f\| \gg \|\mathbf{R}\|$, i.e., observation replaces the forecast in the analysis; or $\mathbf{K} \approx 0$ if $\|\mathbf{P}^f\| \ll \|\mathbf{R}\|$, i.e., the observation is negligible in the analysis.

The observational error covariance $\mathbf{R}$ is determined with the measurements and the relevant analysis (e.g., geomagnetic field modeling). It is therefore an "input" to the assimilation component. For the details we refer the reader to, e.g., Olsen (2002) and Korte and Constable (2008). Evaluation of the forecast error covariance $\mathbf{P}^f$ is the centerpiece of the algorithm. There are various means used to calculate $\mathbf{P}^f$, e.g., a statistical study of the innovations (Hollingsworth and Lönnberg 1986) or the lagged-forecast difference method (Parrish and Derber 1992). In our system an ensemble-covariance estimation of $\mathbf{P}^f$ is employed that could also serve as a component for the more complex ensemble Kalman filter algorithm (Evensen 1994). This ensemble method can be carried out for any numerical dynamo model without knowing its details and can also be parallelized.

In this estimation an ensemble of N_{ens} initial states $\{\mathbf{x}_0^i \mid i = 1, 2, ..., N_{ens}\}$ is first generated by adding arbitrary perturbations to the given model state vector $\mathbf{x}_0$,

$$\mathbf{x}_0^i = \mathbf{x}_0 + \varepsilon^i \tag{13}$$

where the perturbations ε^i are generated randomly via the Monte Carlo technique and their magnitudes are only a small fraction to that of $\mathbf{x}_0$. These perturbed initial states are then integrated in time via Eq. (1) for a specified time interval. The final states $\{\mathbf{x}^i\}$ are used to evaluate $\mathbf{P}^f$ as follows:

$$\mathbf{P}^f = \frac{1}{N_{ens} - 1} \sum_{i=1}^{N_{ens}} (\mathbf{x}^i - \bar{\mathbf{x}})(\mathbf{x}^i - \bar{\mathbf{x}})^T \\ \bar{\mathbf{x}} = \frac{1}{N_{ens}} \sum_{i=1}^{N_{ens}} \mathbf{x}^i \tag{14}$$

In geodynamo modeling the state vector is comprised of spherical harmonic coefficients of the velocity, magnetic field and temperature perturbation. These harmonic coefficients also vary in the radius r and time t. Therefore $\mathbf{P}^f$ is a function of the degree l and order m of the expansions. It also varies in radius r. From our previous experiences, the covariance between different degrees and orders are much weaker than those of the same degree and order and are therefore ignored in the evaluation.

Since the ensemble size N_{ens} is finite and substantially smaller than that of the model output, spurious behaviors exist in the covariance of two distant points (e.g., in radius r). They could normally be eliminated by applying filters, such as the following proposed by Gaspari and Cohn (1999):

$$C_0\left(z, \frac{1}{2}, c\right) = \begin{cases} 0, \ 2c \leq z \\ -\frac{1}{4}\zeta^5 + \frac{1}{2}\zeta^4 + \frac{5}{8}\zeta^3 - \frac{5}{3}\zeta^2 + 1, \\ 0 \leq z \leq c \\ \frac{1}{12}\zeta^5 - \frac{1}{2}\zeta^4 + \frac{5}{8}\zeta^3 + \frac{5}{3}\zeta^2 - 5\zeta + 4 - \frac{2}{3}\zeta^{-1}, \\ c \leq z \leq 2c \end{cases} \tag{15}$$

$$\zeta \equiv \frac{z}{c} \equiv \frac{|r - r_0|}{c}$$

where r is the given model grid location in radius, r_0 is the observational location, and c is the correlation length parameter (e.g., 10% of the outer core thickness). In Eq. (15), the radius is scaled by that of the the core-mantle boundary (CMB) (i.e., $r = 1$).

Different dynamo models have different definitions of the state vector $\mathbf{x}$. To make this component applicable to arbitrary dynamo models it is necessary to introduce a procedure in which the state vectors $\tilde{\mathbf{x}}$ of an arbitrary dynamo model can be transformed back and forth with that $\mathbf{x}$ defined in this component:

$$\tilde{\mathbf{x}} = \mathbf{T}\mathbf{x}, \ \mathbf{x} = \mathbf{T}^{-1}\tilde{\mathbf{x}} \\ \tilde{\mathbf{y}}^o = \mathbf{T}_y \mathbf{y}^o, \ \mathbf{y}^o = \mathbf{T}_y^{-1}\tilde{\mathbf{y}}^o \tag{16}$$

where $\mathbf{T}$ is the transform for the dynamo model state vectors

and $\mathbf{T}_y$ is the transform for the observational subspace. Using the definitions Eqs. (14) and (16), we have

$$\widetilde{\mathbf{P}}^f = \mathbf{T}\mathbf{P}^f\mathbf{T}^T \tag{17}$$

and

$$\widetilde{\mathbf{K}} = \mathbf{T}\mathbf{K}\mathbf{T}_y^{-1}, \quad \widetilde{\mathbf{H}} = \mathbf{T}_y\mathbf{H}\mathbf{T}^{-1} \tag{18}$$

Thus, the analysis

$$\widetilde{\mathbf{x}}^a = \widetilde{\mathbf{x}}^f + \widetilde{\mathbf{K}}(\widetilde{\mathbf{y}}^o - \widetilde{\mathbf{H}}\widetilde{\mathbf{x}}^f) \tag{19}$$

will be used for assimilation with the given dynamo model.

The above formulations are for the state vectors $\mathbf{x}$ in the real space. If $\mathbf{x}$ is complex, e.g., complex spherical harmonic coefficients, one can simply replace the transpose matrices in Eqs. (10b), (12), and (14) using their complex conjugate transpose.

3. COMPONENT STRUCTURE AND WORKFLOW

In this component the state vector includes the velocity field, the magnetic field and the density perturbation:

$$\mathbf{x} = [P_v, T_v, P_B, T_B, \Theta]^T \tag{20}$$

where P_v and T_v are the poloidal and toroidal velocity scalars; P_B and T_B are the poloidal and toroidal magnetic field scalars; and Θ is the density perturbation. Each scalar in Eq. (20) is described by a spherical harmonic expansion. For example,

$$\begin{bmatrix} P_B \\ T_B \end{bmatrix} = \sum_{0 \le m \le l}^{L} \begin{bmatrix} b_l^m(r) \\ j_l^m(r) \end{bmatrix} Y_l^m(\theta, \phi) + C.C. \tag{21}$$

where $C.C.$ stands for the complex conjugate (and L is the truncation order of the expansion). Surface geomagnetic observations (with maximal degree L_{obs}) are downward continued to the radial position r_0. Thus

$$\mathbf{y}^0 = [P_B^o] = \sum_{0 \le m \le l}^{l \le L_{obs}} b_l^m(r_0) Y_l^m(\theta, \varphi) + C.C. \tag{22}$$

For a given degree l and order m, the corresponding observation operator is of a simple form

$$\mathbf{H}_l^m = [0, 0, ..., 0, 0, 1, 0, 0, ..., 0, 0]^T \tag{23}$$

where the only non-zero element corresponding to the ob-

servational location r_0. Therefore,

$$\begin{aligned} \mathbf{P}^f\mathbf{H}^H &= \mathbf{P}^f\mathbf{H}^T = [\cdots P_{lm}^f(r, r_0) \cdots]^T \\ \mathbf{H}\mathbf{P}^f\mathbf{H}^H &= \mathbf{H}\mathbf{P}^f\mathbf{H}^T = \{P_{lm}^{bb}(r_0, r_0)\} \end{aligned} \tag{24}$$

For example, using Eq. (14), we have

$$\begin{aligned} P_{lm}^{bb}(r, r_0) &= \frac{1}{N_{ens} - 1} \sum_{i=1}^{N_{ens}} [b_l^{mi}(r) - \overline{b}_l^m(r)][b_l^{mi}(r) - \overline{b}_l^m(r)]^H \\ P_{lm}^{jb}(r, r_0) &= \frac{1}{N_{ens} - 1} \sum_{i=1}^{N_{ens}} [j_l^{mi}(r) - \overline{j}_l^m(r)][b_l^{mi}(r) - \overline{b}_l^m(r)]^H \end{aligned} \tag{25}$$

for the self-covariance between $b_l^m(r)$ and $b_l^m(r_0)$, and the cross-covariance between $j_l^m(r)$ and $b_l^m(r_0)$, respectively. The superscript H in Eqs. (24) and (25) stands for the complex conjugate transpose.

The component structure is described in Fig. 1: it is comprised of three independent sections, each with its own functions: the gain matrix calculation (the right column in Fig. 1), the transformation between the state vectors (the center column in Fig. 1), and the dynamo simulation (the left column in Fig. 1).

The dynamo section is normally imported from a chosen dynamo model. This part does not need any reengineering effort. The transformation section needs input from users to define the transformation matrices ($\mathbf{T}$ and $\mathbf{T}^{-1}$) for the chosen dynamo model. The gain matrix section includes all "intrinsic" procedures and functions. Users do not need to know the details for their applications. By introducing such a structure this component is simple to implement and easy to learn.

The workflow of this component can be summarized as follows: first, select an initial state vector (a solution of a given dynamo model). Then, transform the state vector into the standard representation for the component. Next, add randomly generated perturbations to the transformed state vector. The modified state vector will then be inversely transformed and sent back to the dynamo model for time integration. The final output from the dynamo simulation will be sent back to the component for calculating the forecast error covariance $\mathbf{P}^f$. This process will be repeated until N_{ens} final states are successfully obtained. The covariance $\mathbf{P}^f$ and the gain matrix $\mathbf{K}$ are calculated and exported for geomagnetic data assimilation.

4. APPLICATION EXAMPLE

This component can be easily integrated with arbitrary dynamo models. To demonstrate this we consider the MoSST core dynamics model series developed by Kuang and Bloxham (1999), Kuang and Chao (2003), Jiang and Kuang (2008). The state vector $\widetilde{\mathbf{x}}$ defined in MoSST is slightly different from that used in this component: the radial variation

of the velocity scalars P_v and T_v are expressed by Chebyshev polynomial expansions in MoSST, while they are defined on the radial grid points in the component. Therefore, the transform matrix **T** is simply the Chebyshev transform.

In this application the entire MoSST dynamo model (including its source codes and executable) is in one directory (hereafter called the "dynamo directory"), while the component is in a separate directory (call the "component directory"). The component directory includes the three subdirectories for the dynamo, interface and gain matrix sections, respectively. The first step is to copy the necessary source codes for defining the dynamo state vector and parameters from the dynamo directory into the component directory. Users then modify the interface section accordingly, including the dynamo directory information. They also need to define the parameters for the component, e.g., the ensemble size N_{ens}, the correlation length c defined in Eq. (15), and the perturbation amplitude. The component execution script includes the MoSST execution for dynamo simulation. The simulation process ends when an ensemble of N_{ens} successful final states is obtained. It should be pointed out that dynamo simulation results from a given initial state are stored in a designated subdirectory (thus N_{ens} subdirectories will be created in the process). These outputs are used for calculating the gain matrix **K** (part of the component output) and will be used for future analysis if needed.

The magnitude of the random perturbations is small: it is set as 1% relative to the spectral coefficients of the original dynamo solutions. In Fig. 2 is the coefficient $b_2^1(r)$ of the initial state defined in Eq. (21). Figure 3 shows an example of random perturbation. The original random perturbation is applied to the 2nd order radial derivative of $b_2^1(r)$ (Fig. 3b): perturbations are added at every four grid point; those in between are set via linear interpolation. The final perturbation ε [see Eq. (13)] is shown in Fig. 3a.

In our application, the forecast error covariance **P**^{f} is simply of the form defined in Eqs. (24) and (25). The filter Eq. (15) used in our example here is defined with $c = 0.1D$ (D is the thickness of the outer core). As an example, we show in Fig. 4a the filtered self-covariance $P_{l1}^{bb}(r, r_0)$, in Fig. 4b the non-filtered $P_{l1}^{bb}(r, r_0)$, and in Fig. 5 the cross-covariance $P_{l1}^{jb}(r, r_0)$. The corresponding gain matrix elements with **R** = 0 are shown in Figs. 6 and 7, respectively.

5. DISCUSSION

This paper described our geomagnetic data assimilation component with an ensemble-based algorithm Eq. (14). This component is stand-alone. It can be implemented independent of dynamo models and the final result can be transformed easily for the designated dynamo model via Eqs. (17), (18), and (19). Given an initial dynamo state and a dynamo model (either serial or parallel), this data assimilation component can generate the gain matrix **K** automatically without any interactions from users. Its implementation is very simple: only the transform section is customer-designed. The gain matrix calculation component is very user-friendly: its details are not needed for applications. The dynamo section is imported directly from the selected dynamo model. However, since the definitions of the dynamo state vectors **x** and $\widetilde{\mathbf{x}}$ are often very simple, transform matrix **T** construction is not very complicated. The component is easy to maintain. Only the transform section needs to be updated if any change is made to the dynamo model.

This component is efficient for parallelization: its scalability can be achieved either through the original dynamo model (since the dynamo executable will be called directly by the component), or, if the original dynamo model is not scalable, by evenly distributing individual dynamo runs.

Simple modification to the component can be achieved
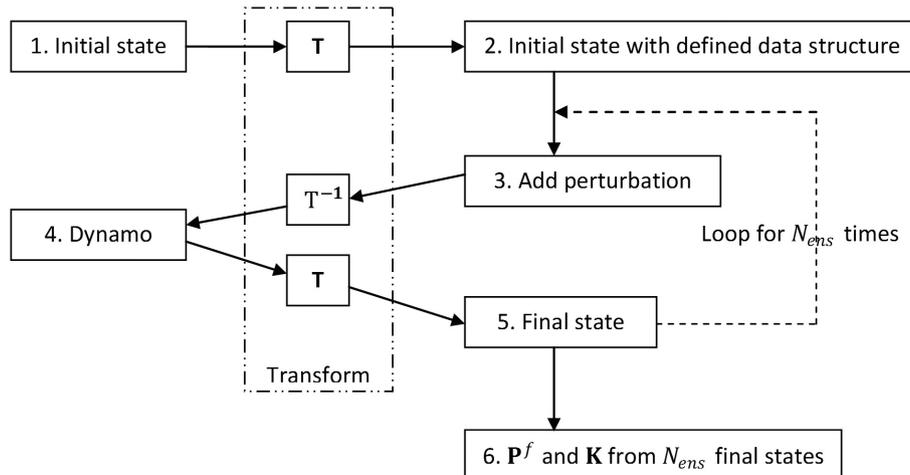
Fig. 1. Sketch of the component structure: in the left column are the dynamo modules and the initial state imported by users; in the right column are the main modules of the component for computing **K**; in the center column are the interface modules to compute the transform **T**.
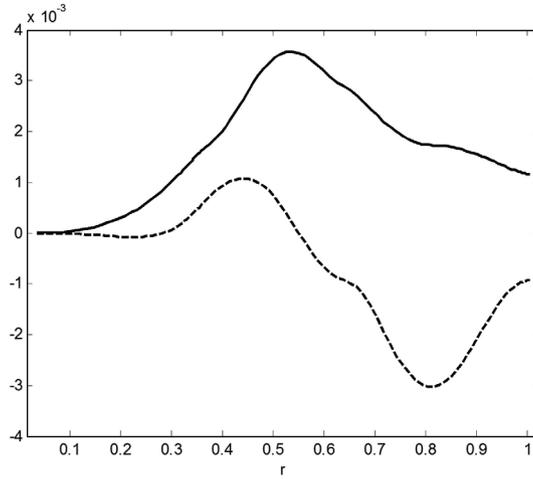
Fig. 2. The radial distribution of the spectral coefficient $b_2^1(r)$ of the initial state: the solid curve is the real part and the dashed curve is the imaginary part. The horizontal axis is the radius r.
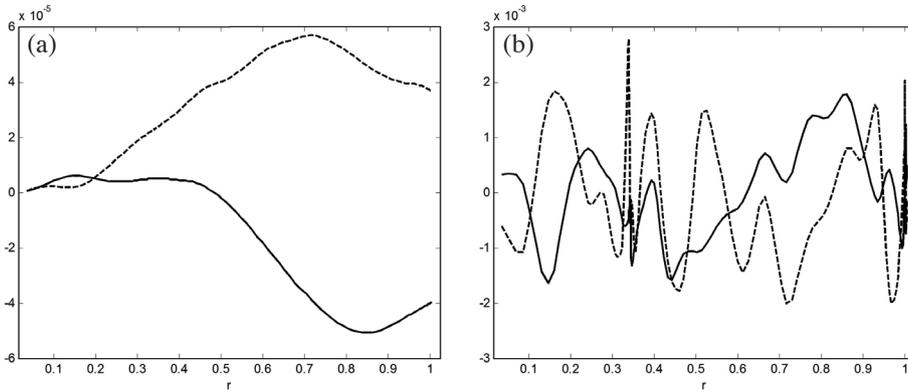
Fig. 3. Similar to Fig. 2, but for the random perturbations added to $b_2^1(r)$. In (a) is the final perturbation to $b_2^1(r)$; in (b) is the perturbation to its 2nd order radial derivative.
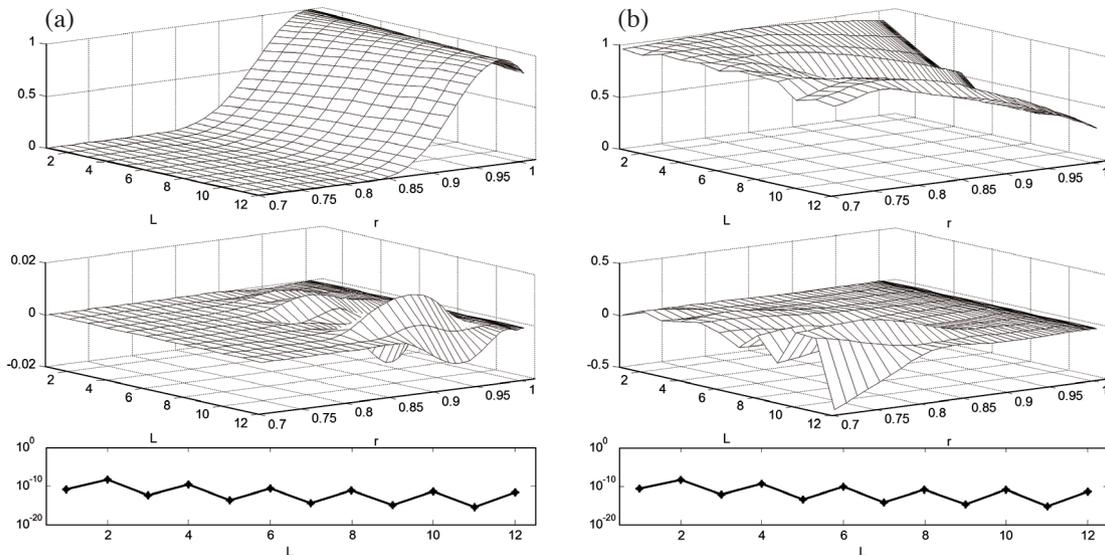
Fig. 4. (a) The radial distribution of the error covariance $P_{l1}^{bb}(r, r_0)$ for $1 \le l \le 12$ (r_0 is at the top of D''-layer): the lower panel is the maximum magnitude $P_{l1}^{mx} \equiv \max \left\{ \left| P_{l1}^{bb}(r, r_0) \right| \right\}$; the upper panel is the real part scaled by P_{l1}^{mx}, $\mathrm{Re}\left[P_{l1}^{bb}(r, r_0) \right]/P_{l1}^{mx}$; and the central panel is the imaginary part scaled by P_{l1}^{mx}, $\mathrm{Im}\left[P_{l1}^{bb}(r, r_0) \right]/P_{l1}^{mx}$. (b) Similar to Fig. 4a, but for the non-filtered error covariance $P_{l1}^{bb}(r, r_0)$.
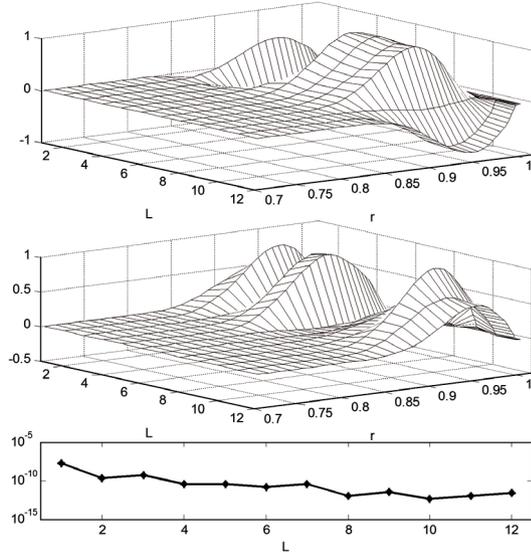
Fig. 5. Similar to Fig. 4, but for the error covariance $P_{l1}^{jb}(r, r_0)$.
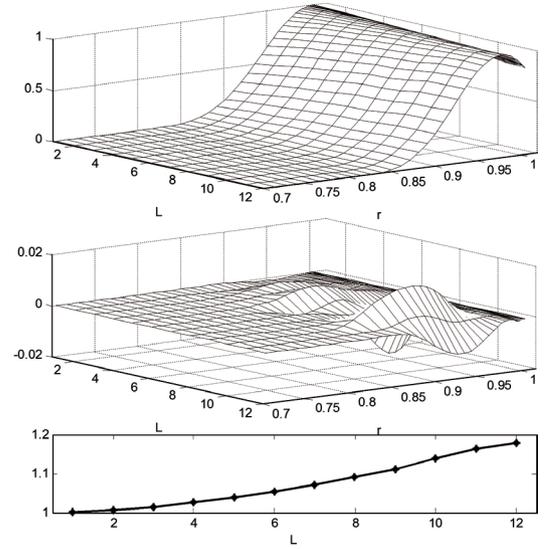
Fig. 6. Similar to Fig. 4, but for the gain matrix **K** corresponding to $P_{l1}^{bb}(r, r_0)$.
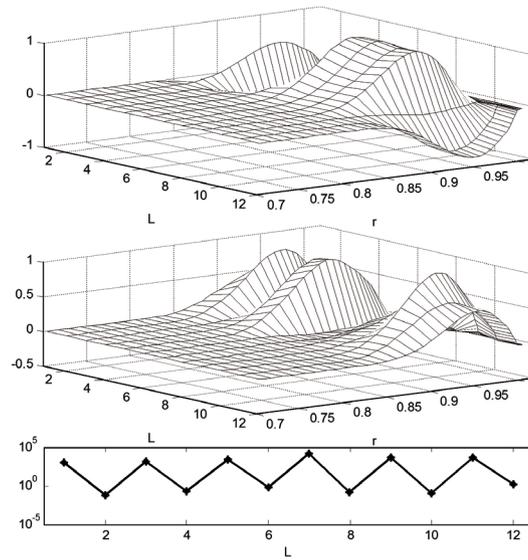
Fig. 7. Similar to Fig. 4, but for the gain matrix **K** corresponding to $P_{l1}^{jb}(r, r_0)$.

by selecting different values for the ensemble size N_{ens} and the filtering length scale c. However, more complicated modifications to the algorithm will need substantial reprogramming effort. For example, if the observation data $\mathbf{y}^o$ are different, e.g., the core flow inverted from the observed geomagnetic secular variation is included as part of the observations, the observation operator **H** in Eq. (12) will be different. Thus, the gain matrix of Eq. (12) will be more complicated. Another example is that this component is not very effective when the time variation of the gain matrix **K** becomes important. In this case it will be optimal to integrate it directly to the dynamo model. This is beyond the scope of this paper. This component can be improved fur-

ther as more users implement it for their geomagnetic data assimilation studies.

REFERENCES

Evensen, G., 1994: Sequential data assimilation with a

nonlinear quasi-geostrophic model using Monte Carlo methods to forecast error statistics. *J. Geophys. Res.*, **99**, 10143-10162, doi: 10.1029/94JC00572. [Link]

Fournier, A., C. Eymin, and T. Alboussière, 2007: A case for variational geomagnetic data assimilation: Insights from a one-dimensional, nonlinear, and sparsely observed MHD system. *Nonlinear Proc. Geoph.*, **14**, 163-180, doi: 10.5194/npg-14-163-2007. [Link]

Fournier, A., G. Hulot, D. Jault, W. Kuang, A. Tangborn, N. Gillet, E. Canet, J. Aubert, and F. Lhuillier, 2010: An introduction to data assimilation and predictability in geomagnetism. *Space Sci. Rev.*, **155**, 247-291, doi: 10.1007/s11214-010-9669-4. [Link]

Gaspari, G. and S. E. Cohn, 1999: Construction of correlation functions in two and three dimensions. *Q. J. R. Meteorol. Soc.*, **125**, 723-757, doi: 10.1002/qj.49712555417. [Link]

Ghil, M. and P. Malanotte-Rizzoli, 1991: Data assimilation in meteorology and oceanography. *Adv. Geophys.*, **33**, 141-266, doi: 10.1016/S0065-2687(08)60442-2. [Link]

Hollingsworth, A. and P. Lönnberg, 1986: The statistical structure of short-range forecast errors as determined from radiosonde data. Part I: The wind field. *Tellus A*, **38**, 111-136, doi: 10.1111/j.1600-0870.1986.tb00460.x. [Link]

Jackson, A., A. R. T. Jonkers, and M. R. Walker, 2000: Four centuries of geomagnetic secular variation from historical records. *Phil. Trans. Roy. Soc. Lond.*, **358**, 957-990, doi: 10.1098/rsta.2000.0569. [Link]

Jiang, W. and W. Kuang, 2008: An MPI-based MoSST core dynamics model. *Phys. Earth Planet. Inter.*, **170**, 46-51, doi: 10.1016/j.pepi.2008.07.020. [Link]

Kalnay, E., 2003: Atmospheric Modelling, Data Assimilation and Predictability, Cambridge University Press, New York, 341 pp.

Kono, M. and P. H. Roberts, 2002: Recent geodynamo simulations and observations of the geomagnetic field. *Rev. Geophys.*, **40**, 4-1-4-53, doi: 10.1029/2000RG000102. [Link]

Korte, M. and C. G. Constable, 2008: Spatial and temporal resolution of millennial scale geomagnetic field models. *Adv. Space Res.*, **41**, 57-69, doi: 10.1016/j.asr.2007.03.094. [Link]

Kuang, W. and J. Bloxham, 1999: Numerical modeling of magnetohydrodynamic convection in a rapidly rotating spherical shell: Weak and strong field dynamo action. *J. Comput. Phys.*, **153**, 51-81, doi: 10.1006/jcph.1999.6274. [Link]

Kuang, W. and B. F. Chao, 2003: Geodynamo modeling and core-mantle interactions. In: Dehant, V., K. C. Creager, S. I. Karato, and S. Zatman (Eds.), Earth's Core: Dynamics, Structure, Rotation, Geodynamics Series, Vol. 31, American Geophysical Union, 193-212, doi: 10.1029/GD031. [Link]

Kuang, W., A. Tangborn, W. Jiang, D. Liu, Z. Sun, J. Bloxham, and Z. Wei, 2008: MoSST-DAS: The first generation geomagnetic data assimilation framework. *Commun. Comput. Phys.*, **3**, 85-108.

Kuang, W., A. Tangborn, Z. Wei, and T. Sabaka, 2009: Constraining a numerical geodynamo model with 100 years of surface observations. *Geophys. J. Int.*, **179**, 1458-1468, doi: 10.1111/j.1365-246X.2009.04376.x. [Link]

Kuang, W., Z. Wei, R. Holme, and A. Tangborn, 2010: Prediction of geomagnetic field with data assimilation: A candidate secular variation model for IGRF-11. *Earth Planets Space*, **62**, 775-785, doi: 10.5047/eps.2010.07.008. [Link]

Liu, D., A. Tangborn, and W. Kuang, 2007: Observing system simulation experiments in geomagnetic data assimilation. *J. Geophys. Res.*, **112**, doi: 10.1029/2006JB004691. [Link]

Olsen, N., 2002: A model of the geomagnetic field and its secular variation for epoch 2000 estimated from Ørsted data. *Geophys. J. Int.*, **149**, 454-462, doi: 10.1046/j.1365-246X.2002.01657.x. [Link]

Olsen, N., M. Mandea, T. J. Sabaka, and L. Tøffner-Clausen, 2010: The CHAOS-3 geomagnetic field model and candidates for the 11th generation IGRF. *Earth Planets Space*, **62**, 719-727, doi: 10.5047/eps.2010.07.003. [Link]

Parrish, D. F. and J. C. Derber, 1992: The National Meteorological Center's spectral statistical-interpolation analysis system. *Mon. Weather Rev.*, **120**, 1747-1763, doi: 10.1175/1520-0493(1992)120<1747:TNMCSS>2.0.CO;2. [Link]

Sabaka, T. J., N. Olsen, and M. E. Purucker, 2004: Extending comprehensive models of the Earth's magnetic field with Ørsted and CHAMP data. *Geophys. J. Int.*, **159**, 521-547, doi: 10.1111/j.1365-246X.2004.02421.x. [Link]

Sun, Z., A. Tangborn, and W. Kuang, 2007: Data assimilation in a sparsely observed one-dimensional modeled MHD system. *Nonlinear Proc. Geoph.*, **14**, 181-192, doi: 10.5194/npg-14-181-2007. [Link]

Sun, Z., L. Y. Oey, and Y. H. Zhou, 2013: Skill-assessments of statistical and Ensemble Kalman Filter data assimilative analyses using surface and deep observations in the Gulf of Mexico. *Front. Earth Sci.*, **7**, 271-281, doi: 10.1007/s11707-013-0377-8. [Link]

Talagrand, O., 1997: Assimilation of observations, an introduction. *J. Meteorol. Soc. Jpn.*, **75**, 191-209.

APPENDIX: WORKING MANUAL OF THE COMPONENT

1. Component Usage

All codes for this component are stored in five different subdirectories based on their functionalities. The names of these subdirectories are self-explanatory. For example,

the stand-alone Fortran modules are stored in the subdirectory named "modules_EnK".

The component application can be summarized as follows: (1) copy all files to a separate subdirectory (we strongly recommend users for this procedure for better maintenance); (2) compile the code; (3) modify the parameter files according to the dynamo package of the users; (4) execute the component and check the output. The last step may be repeated if the desired ensemble members cannot be reached in a single execution.
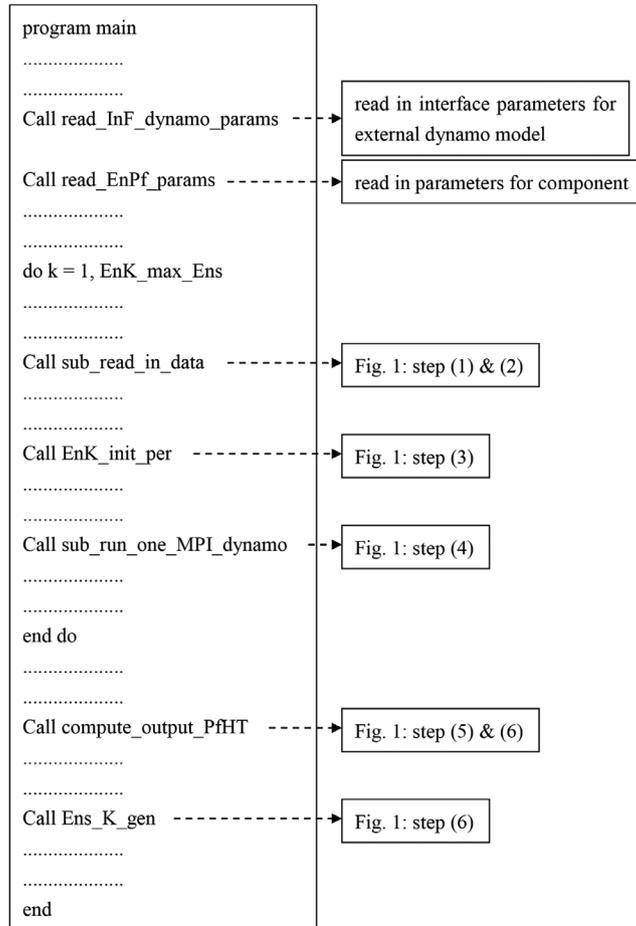
Fig. A1. Program structure.