Annales
Geophysicae

Estimating a planetary magnetic field with time-dependent global MHD simulations using an adjoint approach

Christian Nabert[1], Carsten Othmer[1], and Karl-Heinz Glassmeier[1,2]

[1]Institut für Geophysik und extraterrestrische Physik, Technische Universität Braunschweig, Mendelssohnstr. 3, 38106 Braunschweig, Germany
[2]Max-Planck-Institut für Sonnensystemforschung, Justus-von-Liebig-Weg 3, 37077 Göttingen, Germany

Correspondence to: Christian Nabert (c.nabert@tu-bs.de)

Abstract. The interaction of the solar wind with a planetary magnetic field causes electrical currents that modify the magnetic field distribution around the planet. We present an approach to estimating the planetary magnetic field from in situ spacecraft data using a magnetohydrodynamic (MHD) simulation approach. The method is developed with respect to the upcoming BepiColombo mission to planet Mercury aimed at determining the planet's magnetic field and its interior electrical conductivity distribution. In contrast to the widely used empirical models, global MHD simulations allow the calculation of the strongly time-dependent interaction process of the solar wind with the planet. As a first approach, we use a simple MHD simulation code that includes time-dependent solar wind and magnetic field parameters. The planetary parameters are estimated by minimizing the misfit of spacecraft data and simulation results with a gradient-based optimization. As the calculation of gradients with respect to many parameters is usually very time-consuming, we investigate the application of an adjoint MHD model. This adjoint MHD model is generated by an automatic differentiation tool to compute the gradients efficiently. The computational cost for determining the gradient with an adjoint approach is nearly independent of the number of parameters. Our method is validated by application to THEMIS (Time History of Events and Macroscale Interactions during Substorms) magnetosheath data to estimate Earth's dipole moment.

Keywords. Magnetospheric physics (magnetosheath; planetary magnetospheres; solar wind–magnetosphere interactions)

1 Introduction

Planets with an intrinsically generated magnetic field, such as Earth or Mercury, interact with the solar wind. This causes electrical currents that modify the planetary magnetic field. The properties of the interaction not only depend on the planetary magnetic field but also on the continuously varying solar wind conditions. A spacecraft orbiting a planet in such a highly variable environment measures the modified magnetic field distribution.

In 2025 the BepiColombo mission (Benkhoff et al., 2010) of the ESA and the Japan Aerospace Exploration Agency (JAXA) is expected to reach planet Mercury. In contrast to the previous MESSENGER (Mercury Surface, Space Environment, Geochemistry, and Ranging) mission (Solomon et al., 2001), two spacecraft will simultaneously measure the magnetic field distribution around the planet. The planetary magnetic field at Mercury is about 100 times weaker than the field of Earth. Therefore, the magnetosheath is much closer to the surface of the planet. As a consequence, the magnetic field of the electric currents of the interaction is not negligible, even in the immediate proximity of the planet (e.g., Glassmeier, 2000). Furthermore, electromagnetic induction effects within the planet might be important (e.g., Grosser et al., 2004; Jia et al., 2015). To estimate the planetary magnetic field precisely, the time-dependent interaction needs to be determined. With its two spacecraft, the BepiColombo mission is most suitable for determining the interaction because of simultaneous observations of the magnetic field distribution in the magnetosphere and the solar wind. If both spacecraft are within the interaction region, the solar wind reconstruction method by Nabert et al. (2015) can be used to estimate the time-varying solar wind conditions from the ob-

servations of one spacecraft. Then the data of the other spacecraft provide still observations within the interaction region while the solar wind conditions are known.

So far, the planetary magnetic field of Mercury has been determined using empirical models of the interaction between the solar wind and the planetary magnetic field with spacecraft data from MESSENGER or Mariner 10 (e.g., Korth et al., 2004; Alexeev et al., 2010; Johnson et al., 2012). The electrical current density of the interaction in empirical models is parametrized by pre-described functional relations. Typically, the current system is described as a superposition of localized electrical currents such as the magnetopause current, which is parametrized by its subsolar location and ellipsoidal shape. Taking only a few parameters into account, these prescribed functional relations do not include, for example, effects such as magnetic pile-up, which correspond to a distribution of the magnetopause current within the entire magnetosheath. Furthermore, the parameters are distinguished between only a few discrete solar wind scenarios such as strong and weak solar wind pressure. If more parameters or solar wind scenarios are considered to parametrize the current system more accurately, it is not always possible to determine all parameters with small statistical error due to the finite data coverage. This is especially true if strongly time-dependent nonlinear phenomena occur.

Mercury's magnetic field close to the subsolar magnetopause has a strength of about 60 nT (Johnson et al., 2012). Using an average solar wind velocity of $430 \, \mathrm{km \, s^{-1}}$, this corresponds to a gyroradius of the interaction of about 37.5 km. Compared to global structures of the interaction, such as a subsolar magnetosheath thickness of about 1220 km (Winslow et al., 2013), a magnetohydrodynamic (MHD) approximation seems to be a valid approximation. The inverse gyrofrequency is about 0.5 s, which limits the time resolution for this approximation. In regions dominated by heavy ions, a kinetic approach might be necessary. Here, we restrict our considerations to the MHD approximation. Taking the observations of the two spacecraft of the BepiColombo mission into account, the interaction can be calculated as fully time-dependent with a MHD model. We investigate a procedure to estimate the planetary magnetic field in a strongly modified magnetic environment of the planet using a global MHD simulation. In contrast to empirical models, a MHD simulation requires only parameters of the solar wind conditions, planetary magnetic field, and plasma properties. This approach calculates the interaction self-consistently and does not contain parameters to fit electrical currents. Note that such a model also allows taking a conductivity distribution of the planet into account. Then, the parameters of the planet's interior conductivity can be estimated in addition to the planetary magnetic field parameters in a further step.

As a first approach, we consider a simple MHD simulation code based on the MHD code presented by Ogino (1993) to examine our method. A cost function quantifies the misfit of the spacecraft observations in the magnetosphere to the corresponding MHD simulation results. The cost function needs to be minimized with respect to the planetary magnetic field parameter to estimate these planetary parameters. Different methods can be used to minimize the cost function. Methods such as downhill simplex or Markov chain Monte Carlo algorithms are usually used if derivatives of the cost function cannot be calculated directly. If the gradient can be calculated, gradient-based minimization algorithms can be used, which often offers faster convergence speed. However, these methods are restricted to find a local minimum in parameter space instead of the global minimum. Here, we expect a global minimum, which is not superposed by local minima, so that a gradient-based optimization procedure is considered. The gradient-based methods can provide fast convergence only if the gradient can be determined quickly. However, the calculation of the gradient with respect to several parameters using, for example, finite difference quotients can be very time-consuming. Thus, an adjoint approach is considered, which can theoretically compute gradients nearly independent of the number of parameters (e.g., Jameson, 1988; Giles and Pierce, 2000).

In this paper we investigate the applicability of an adjoint approach to a MHD simulation code using automatic differentiation (Wengert, 1964). Although the adjoint approach can be much faster than using finite differences, it requires larger memory capacities. An adjoint approach using automatic differentiation was successfully applied to a reduced MHD model, the magnetosheath model by Nabert et al. (2013), to estimate the solar wind parameters of the model (Nabert et al., 2015). The reduced MHD model uses series expansions along the bow shock and magnetopause geometry of the MHD quantities. This transfers the stationary partial differential MHD equations into a set of ordinary differential equations. Close to the stagnation streamline, only low-order series expansions are necessary to obtain a valid representation of the interaction. Not only the numerical effort for solving the corresponding ordinary differential equations is significantly lower compared to solving the full MHD system, the required storage capacity is also much lower. Therefore, the automatic differentiation procedure could be applied without regarding memory limitations. Here, an automatic differentiation tool is applied to a full MHD simulation code and thus special emphasis needs to be put on memory consumptions.

Our approach to estimating planetary parameters using data from a multi-spacecraft mission is validated with the THEMIS (Time History of Events and Macroscale Interactions during Substorms) mission (Angelopoulos, 2008) at Earth with its well-known planetary magnetic field. The five spacecraft of the mission (THA, THB, THC, THD, and THE) provide simultaneous observations of the interaction region and the solar wind. However, in contrast to the situation at Mercury, the interaction of the solar wind near the planet's surface is negligible at Earth. Due to the weak magnetic field at Mercury, the interaction region of the solar wind is

much closer to the planet (Winslow et al., 2013). In particular, the subsolar bow shock distance to the center of the planet is on average about 1.89 R_M at Mercury and 13 R_E at Earth ($R_E = 6371$ km). The average distance of the subsolar magnetopause at Mercury is 1.45 R_M and 10 R_E at Earth. As a consequence, only close to the magnetosheath region, are the modifications of Earth's magnetic field comparable to the strong modifications throughout the magnetosphere of Mercury. To validate our procedure with respect to the future measurements of the BepiColombo mission, THEMIS data from the terrestrial magnetosheath is used. However, in its final application for the BepiColombo mission, spacecraft data of the entire interaction region including the magnetosphere will be taken into account to estimate the planetary magnetic field.

2   MHD simulation code

The interaction of the planetary magnetic field with the solar wind is computed by a MHD simulation code. The MHD simulation has to be efficient to perform the time-consuming estimation procedure of the planetary parameters. Furthermore, the simulation code should be simple in its numerical implementation structure to simplify the application of the adjoint approach using automatic differentiation. For these reasons, as a first approach, a simple MHD simulation code is developed, which is based on the simulation code described by Ogino (1993). The MHD simulation code described by Ogino (1993) was already used in studies of magnetospheric convection, for example, depending on the solar wind magnetic field (Ogino et al., 1985) or field-aligned currents (Ogino, 1986). The code is modified and extended for the application to the parameter estimation process as explained in the following paragraphs. Furthermore, some details about the numerical implementation of the simulation code are summarized to understand the application of the adjoint method via automatic differentiation, which is explained in the next section.

2.1   Planetary magnetic field

The magnetic field in the simulation code by Ogino (1993) is restricted to a dipole along the planet's axis of rotation. Here, a more general representation of the magnetic field is required. The planetary magnetic field can be represented by a multipole expansion using a spherical harmonic analysis (Gauss, 1839; Glassmeier and Tsurutani, 2014). Note that this part of the magnetic field does not contain contributions due to the interaction with the solar wind such as induction or magnetopause currents. As a consequence, the planetary magnetic field outside the planet can be represented by a scalar potential V_{pot}:

$$\boldsymbol{B} = -\nabla V_{pot}. \tag{1}$$

Table 1. Dipole coefficients of the IGRF in 2010. On the left side are the Gauss coefficients according to Finlay et al. (2010). On the right side are the corresponding dipole vectors using Eq. (3).

$g_1^0 = -29\,496.5\,\text{nT}$	$m_x = -0.41 \times 10^{15}\,\text{T m}^3$
$g_1^1 = -1585.9\,\text{nT}$	$m_y = 1.28 \times 10^{15}\,\text{T m}^3$
$h_1^1 = 4945.1\,\text{nT}$	$m_z = -7.63 \times 10^{15}\,\text{T m}^3$

Thereby, the scalar potential V_{pot} satisfies a Laplace equation. Using spherical coordinates (r, λ, θ), the solution outside the planet ($r > R_P$), where R_P denotes the planet's radius, is given by

$$V_{pot}(r, \theta, \lambda) = R_P \sum_{l=1}^{\infty} \sum_{m=0}^{l} \left(\frac{R_P}{r}\right)^{l+1} \left(g_l^m \cos(m\lambda)\right.$$
$$\left. + h_l^m \sin(m\lambda)\right) P_l^m(\cos(\theta)),$$

with the Gauss coefficients g_l^m, h_l^m and the Schmidt seminormalized associated Legendre polynomials $P_l^m(\cos(\theta))$, e.g., $P_1^1 = \cos(\theta)$ or $P_2^1 = \sqrt{3}\cos(\theta)\sin(\theta)$ (e.g., Langel, 1987; Clauser, 2016).

The lowest-order coefficients for $l = 1$ are associated with the dipole moment corresponding to the magnetic field vector $\boldsymbol{B}_{dipole}$. The simulation code uses a Cartesian representation of the magnetic field. For the dipole moment, this is

$$\boldsymbol{B}_{dipole} = \frac{3(\boldsymbol{r} \cdot \boldsymbol{m})\boldsymbol{r} - r^2 \boldsymbol{m}}{r^5}. \tag{2}$$

Here, $\boldsymbol{m} = (m_x, m_y, m_z)^T$ is the vector of the dipole moment, which is related to the Gauss coefficients via

$$m_x = R_E^3 g_1^1,$$
$$m_y = R_E^3 h_1^1, \tag{3}$$
$$m_z = R_E^3 g_1^0.$$

The Gauss coefficients for Earth's magnetic field in 2010 were published using the International Geomagnetic Reference Field (IGRF) by Finlay et al. (2010). Thereby, the geographic coordinate system, a body-fixed coordinate system, is used, with its z axis along the axis of rotation. Magnetic field data of spacecraft close to Earth's surface as well as ground stations were used to determine the coefficients. The influence of external currents due to the interaction of the solar wind with the planetary magnetic field was neglected. The magnetic field of Earth outside the planet is dominated by the dipole coefficients, which are summarized in Table 1. Note that a similar estimation procedure at Mercury leads to large errors because of insufficient data coverage in the southern hemisphere of the planet. Furthermore, the solar wind interaction has a strong influence on the magnetic field distribution.

Similar to the dipole, higher-order moments of the planetary magnetic field can be taken into account. Thereby,

the tensor structure of the Cartesian representation becomes more complex for higher orders. For example, the quadrupole can be expressed by a symmetric, traceless matrix **Q**, which is defined by (e.g., Vogt and Glassmeier, 2000; Stadelmann et al., 2010)

$$\mathbf{Q} := \begin{pmatrix} Q_{xx} & Q_{xy} & Q_{xz} \\ Q_{xy} & Q_{yy} & Q_{yz} \\ Q_{xz} & Q_{yz} & -(Q_{xx}+Q_{yy}) \end{pmatrix}. \tag{4}$$

The magnetic field related to the quadrupole can be expressed as

$$\boldsymbol{B}_{\text{quadrupole}} = \frac{2\mathbf{Q}\boldsymbol{r}\,r^2 - 5\,Q_{\text{sp}}\,\boldsymbol{r}}{2\,r^7}, \tag{5}$$

with Q_{sp} defined by

$$\begin{aligned} Q_{\text{sp}} := {} & Q_{xx}\,x^2 + 2\,Q_{xy}\,x\,y + Q_{yy}\,y^2 + 2\,Q_{xz}\,x\,z \\ & + 2\,Q_{yz}\,y\,z - (Q_{xx}+Q_{yy})\,z^2. \end{aligned}$$

The simulation code according to Ogino (1993) uses normalizations for the physical quantities. The normalization constants for the additional magnetic field parameters are

$$m_{\text{norm}} = \begin{cases} 8.07 \times 10^{15}\,\text{T}\,\text{m}^3, & \text{for a dipole component,} \\ 5.14 \times 10^{22}\,\text{T}\,\text{m}^4, & \text{for a quadrupole component.} \end{cases} \tag{6}$$

The simulations take the dipole and quadrupole moments into account. Thus, the resulting planetary magnetic field is

$$\boldsymbol{B}_{\text{Planet}} = \boldsymbol{B}_{\text{dipole}} + \boldsymbol{B}_{\text{quadrupole}}. \tag{7}$$

2.2 MHD equations and boundary conditions

The interaction of the solar wind with the planetary magnetic field is calculated by solving the MHD equations. Thereby, these equations are solved within a box sketched in Fig. 1. The simulation uses a model solar wind planet (MSP) coordinate system, whereby the origin is in the planet's center. The x axis is along the unperturbed solar wind velocity vector, the z axis is parallel to the rotation axis, and the y axis completes a right-handed coordinate system. The length of the simulation box is in x direction x_{L}, in y direction y_{L}, and in z direction z_{L}.

The MHD equations provide solutions for the mass density ρ, the plasma velocity $\boldsymbol{v} := (v_x, v_y, v_z)^{\text{T}}$, the pressure p, and the magnetic field $\boldsymbol{B} := (B_x, B_y, B_z)^{\text{T}}$. The solutions are summarized in the vector

$$\boldsymbol{u} := (\rho, v_x, v_y, v_z, p, B_x, B_y, B_z)^{\text{T}}. \tag{8}$$
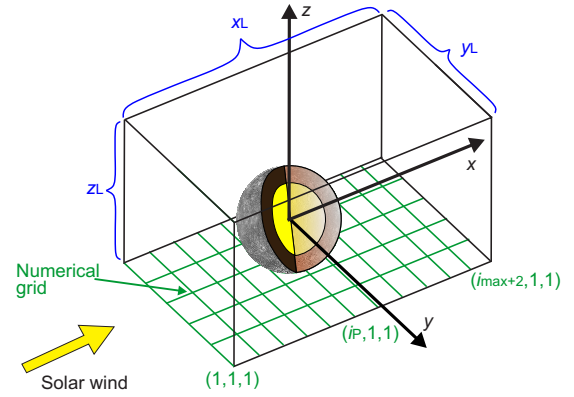

Figure 1. The simulation box contains the planet with its magnetic field. The origin of the coordinate system is in the planet's center and the x axis is along the unperturbed solar wind velocity.

The following representation of the MHD equations is solved by the MHD simulation code:

$$\partial_t \rho = -\nabla \cdot (\rho\,\boldsymbol{v}) + D_\rho\,\nabla^2 \rho, \tag{9}$$

$$\partial_t \boldsymbol{v} = -(\boldsymbol{v}\cdot\nabla)\,\boldsymbol{v} - \frac{1}{\rho}\,(\nabla p - \boldsymbol{j}\times\boldsymbol{B}) + \frac{D_{\text{v}}}{\rho}\,\nabla^2 \boldsymbol{v}, \tag{10}$$

$$\partial_t p = -(\boldsymbol{v}\cdot\nabla)\,p - \gamma\,p\,\nabla\,\boldsymbol{v} + D_p\,\nabla^2 p, \tag{11}$$

$$\partial_t \boldsymbol{B} = \nabla\times(\boldsymbol{v}\times\boldsymbol{B}) - \nabla\times(D_B\,\nabla\times\boldsymbol{B}). \tag{12}$$

Here, D_ρ, D_{v}, D_p, and D_B are diffusion coefficients of the density, the velocity, the pressure, and the magnetic field, respectively. The magnetic diffusion coefficient is related to the electrical resistivity η by $D_B := \eta/\mu_0$, with the vacuum permeability $\mu_0 := 4\pi \times 10^{-7}$. The current density $\boldsymbol{j}$ is calculated with Ampere's law, neglecting the displacement current:

$$\boldsymbol{j} = \frac{1}{\mu_0}\nabla\times\boldsymbol{B}. \tag{13}$$

According to Ogino (1993), the MHD equations are solved using a two-step Lax–Wendroff method (Lax and Wendroff, 1960), which has an accuracy of second order in space and time. This numerical scheme uses finite difference approximations, which require the solution to be described on a discrete grid. The discretization of the MSP coordinates (x, y, z) is related to the indices (i, j, k), with i for x, j for y, and k for z. Thereby, valid values for the indices are $i = 1, \ldots, i_{\max}+2$; $j = 1, \ldots, j_{\max} + 2$; and $k = 1, \ldots, k_{\max} + 2$. The number of spatial grid points N_{grid} is given by

$$N_{\text{grid}} = (i_{\max} + 2) \cdot (j_{\max} + 2) \cdot (k_{\max} + 2). \tag{14}$$

The boundaries of the simulation box along the x direction are located at $(i = 1, j, k)$ and $(i = i_{\max} + 2, j, k)$. Along the y direction, the boundaries are located at $(i, j = 1, k)$ and $(i, j = j_{\max} + 2, k)$ and along the z direction at $(i, j, k = 1)$

and $(i, j, k_{\max}+2)$. The distance between grid points is $\Delta x = x_L/(i_{\max}+1)$ in x direction, $\Delta y = y_L/(j_{\max}+1)$ in y direction, and $\Delta z = z_L/(k_{\max}+1)$ in z direction. Within the grid, the planet is located at (i_P, j_P, k_P), with $i_P = (i_{\max}+1)/2$, $j_P = (j_{\max}+1)/2$, and $k_P = (k_{\max}+1)/2$. The grid points (i, j, k) are related to a position (x, y, z) by

$$x = 0.5 \, \Delta x \, (2i - i_{\max} - 3),$$
$$y = 0.5 \, \Delta y \, (2j - j_{\max} - 3), \qquad (15)$$
$$z = 0.5 \, \Delta z \, (2k - k_{\max} - 3).$$

The time t is discretized by the index l with $l = 0, \ldots, l_{\max}$, whereby $l = 0$ is related to $t = 0$ and $l_{\max}$ to $t = t_E$. This corresponds to a constant time step of $\Delta t = t_E/l_{\max}$. The spatial and time-dependent solution of the MHD equations $\boldsymbol{u}(t, x, y, z)$ defined by Eq. (8) can be represented by $u^n_{l,i,j,k}$, whereby $n = 1, \ldots, N_{\text{var}}$ refers to a component of the vector $\boldsymbol{u}$. Here, the number of the MHD variables is $N_{\text{var}} = 8$.

Boundary conditions are required to solve the MHD equations. The inflow boundary conditions at $(i = 1, j, k)$ are determined by the solar wind conditions. The solar wind velocity vector is restricted to the x axis, perpendicular to the planet's rotation axis. In contrast to the more simple inflow boundary conditions of Ogino (1993), we use time-varying solar wind conditions:

$$\boldsymbol{u}_{\text{SW}}(t) := (\rho_{\text{SW}}(t), v_{\text{SW}}(t), 0, 0, \, p_{\text{SW}}(t), B_{x,\text{SW}}(t),$$
$$B_{y,\text{SW}}(t), B_{z,\text{SW}}(t))^{\text{T}}.$$

Instead of using the mass density, the ion density $N_{\text{SW}} = \rho_{\text{SW}}/m_P$ can be used as well, with the proton mass $m_P = 1.672621898 \times 10^{-27}$ kg. In general, the physical properties at grid points at $(i = 1, j, k)$ can be replaced by the solar wind conditions in every time step. The solar wind vector discretized in time is $\boldsymbol{u}_{\text{SW},l} := \boldsymbol{u}_{\text{SW}}(l \, \Delta t)$. All other outer boundaries are outflow boundaries according to Ogino (1993).

In addition to the boundary conditions, our simulation requires initial conditions. Therefore, at time step $l = 0$, the physical quantities have to be determined in the entire simulation domain. The velocity $\boldsymbol{v}$ is assumed to be zero, so that $u^2_{1,i,j,k} = u^3_{1,i,j,k} = u^4_{1,i,j,k} = 0$. The density ρ and pressure p are initialized by their solar wind values, $\rho_{\text{SW}}(0)$ and $p_{\text{SW}}(0)$, respectively. Thus, $u^1_{1,i,j,k} = \rho_{\text{SW}}(0)$ and $u^5_{1,i,j,k} = p_{\text{SW}}(0)$ are used. The initial values of the magnetic field are determined by the planetary magnetic field. Taking only the dipole and the quadrupole moments into account, the planetary magnetic field can be calculated by Eq. (7) with Eqs. (2) and (5). The initial conditions determine a stationary solution at a certain time step $l = l_{\text{st}}$ with $0 < l_{\text{st}} < l_{\max}$. The solar wind conditions for $l < l_{\text{st}}$ are set to the values at l_{st}. For $l > l_{\text{st}}$, time-dependent solar wind conditions from spacecraft observations are applied in the simulation and the results are compared to spacecraft observations.

The extended magnetic field geometry, especially the arbitrarily aligned dipole moment, can cause a complex motion
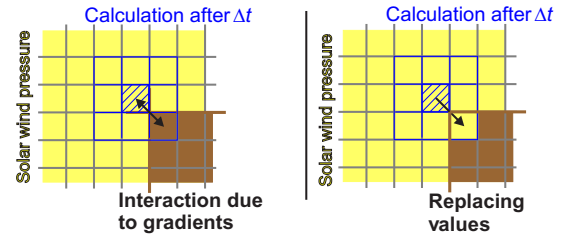

Figure 2. Density and pressure gradients across the planetary surface are related to forces if the MHD equations are solved within the entire simulation box. To minimize this interaction, the density and the pressure at the planetary surface are set to the values outside the planet.

of the plasma in the magnetosphere due to co-rotation of the plasma or magnetic reconnection, for example. Therefore, different from the simulation described by Ogino (1993), the simulation requires appropriate inner boundary conditions to allow a stable simulation for long time intervals. The planetary surface is approximated by a spherical surface with the distance R_P to the planet's center. The distance of a grid point (i, j, k) to the center is defined by

$$r_{i,j,k} := \sqrt{((i - i_P) \Delta x)^2 + ((j - j_P) \Delta y)^2 + ((k - k_P) \Delta z)^2}. \quad (16)$$

The velocity of the plasma inside the planet, i.e., $r_{i,j,k} < R_P$, is

$$u^n_{l,i,j,k} = 0, \quad \text{for } r_{i,j,k} < R_P, \ \ n = 2, 3, 4. \quad (17)$$

It is also possible to set only the normal component of the velocity to zero.

Density and pressure gradients between the planet's interior and the plasma outside must not cause forces on the plasma. However, the MHD equations are solved within the entire simulation domain. This can lead to an interaction as sketched in Fig. 2. The values at grid points inside the planet, which have at least one neighboring grid point outside, are replaced in every time step by average values of the surrounding neighboring grid points outside the planet, i.e., the non-boundary neighbors. Neighboring grid points of a grid point (i_P, j_P, k_P) are $\{(i_P \pm 1, j_P \pm 1, k_P \pm 1)\}$. This procedure suppresses the interaction of density and pressure gradients across the planet.

In contrast to the density and gas pressure, the magnetic field can interact with the planet due to electromagnetic induction, which is additionally implemented. Time-dependent variations in the magnetic field inside the planet are calculated by Eq. (12). We assume the velocity inside the planet to be zero, not considering the detailed time-dependent dynamo action. This is justified due to the very different timescales of magnetospheric and dynamo action. Concerning a possible coupling between magnetosphere and dynamo, see Glassmeier et al. (2007) and Heyner et al. (2011). Thus, the in-

duction equation simplifies to the diffusion equation

$$\partial_t \boldsymbol{B} = -\nabla \times \left(\frac{\eta}{\mu_0} \nabla \times \boldsymbol{B} \right). \tag{18}$$

The resistivity distribution in the simulation box is modeled by

$$\eta = \begin{cases} \eta_{\text{Core}}, & r < R_{\text{Core}} \\ \eta_{\text{Mantle}}, & R_{\text{Core}} \leq r \leq R_{\text{P}} \\ \eta_{\text{A}}, & \text{otherwise} \end{cases} \tag{19}$$

Thereby, the planet's interior consists of two regions with different resistivity, a core with $\eta_{\text{Core}} = 1/\sigma_{\text{Core}}$ and a mantle with $\eta_{\text{Mantle}} = 1/\sigma_{\text{Mantle}}$. The resistivity outside the planet is assumed to be constant with η_{A}. As a consequence, the interaction due to diffusion is allowed depending on the electrical resistivity of the planet. This is of particular importance if Mercury is considered; however, it is of minor importance for Earth.

2.3　Using spacecraft data

The simulation uses solar wind parameters as boundary conditions. With respect to the two spacecraft in mission Bepi-Colombo, simultaneous observations of the solar wind as well as the magnetic field close to the planet will be available in the future. Thereby, the solar wind conditions can be determined either directly by in situ measurements or by using the reconstruction method by Nabert et al. (2015) from data within the interaction region. This allows a precise determination with a high time resolution of the solar wind conditions. The THEMIS mission provides data from similar spacecraft constellations at Earth. The solar wind conditions observed by a spacecraft need to be transferred to the inflow boundary of the simulation box. Therefore, the solar wind data are shifted by $\Delta t_{\text{SC/in}}$, which is given by

$$\Delta t_{\text{SC/in}} = \frac{\boldsymbol{n}_{\text{P}} \cdot \Delta \boldsymbol{r}_{\text{SC/in}}}{\boldsymbol{n}_{\text{P}} \cdot \boldsymbol{v}_{\text{SW}}}, \tag{20}$$

where $\boldsymbol{n}_{\text{P}}$ is the solar wind's phase plane normal vector, $\Delta \boldsymbol{r}_{\text{SC/in}} := \boldsymbol{r}_{\text{SC}} - \boldsymbol{r}_{\text{in}}$ is the distance vector between the spacecraft's position $\boldsymbol{r}_{\text{SC}}$, with the center of the inflow boundary $\boldsymbol{r}_{\text{in}}$.

For a comparison between simulation results and spacecraft data, the data need to be transferred into MSP coordinates. Therefore, the THEMIS data are first transferred into geographic (GEO) coordinates. Vectors in these coordinates can be transferred into MSP coordinates by rotation matrices $R_y(\theta_{\text{K}})$ and $R_z(\lambda_{\text{K}})$. These matrices are defined by

$$\boldsymbol{R}_y(\theta_{\text{K}}) = \begin{pmatrix} \cos(\theta_{\text{K}}) & 0 & \sin(\theta_{\text{K}}) \\ 0 & 1 & 0 \\ -\sin(\theta_{\text{K}}) & 0 & \cos(\theta_{\text{K}}) \end{pmatrix},$$

$$\boldsymbol{R}_z(\lambda_{\text{K}}) = \begin{pmatrix} \cos(\lambda_{\text{K}}) & -\sin(\lambda_{\text{K}}) & 0 \\ \sin(\lambda_{\text{K}}) & \cos(\lambda_{\text{K}}) & 0 \\ 0 & 0 & 1 \end{pmatrix}. \tag{21}$$

The rotation angles θ_{K} and λ_{K} are determined by the solar wind velocity vector:

$$\lambda_{\text{K}} = \frac{v_{y,\text{SW,GEO}}}{|v_{y,\text{SW,GEO}}|} \arccos \left(\frac{v_{x,\text{SW,GEO}}}{\sqrt{v_{x,\text{SW,GEO}}^2 + v_{y,\text{SW,GEO}}^2}} \right),$$

$$\theta_{\text{K}} = \frac{\widetilde{v}_{z,\text{SW,GEO}}}{|\widetilde{v}_{z,\text{SW,GEO}}|} \arccos \left(\frac{\widetilde{v}_{x,\text{SW,GEO}}}{\sqrt{\widetilde{v}_{x,\text{SW,GEO}}^2 + \widetilde{v}_{z,\text{SW,GEO}}^2}} \right). \tag{22}$$

Here, $\boldsymbol{v}_{\text{SW,GEO}} = (v_{x,\text{SW,GEO}}, v_{y,\text{SW,GEO}}, v_{z,\text{SW,GEO}})^{\text{T}}$ is the solar wind velocity vector using GEO coordinates and $\widetilde{\boldsymbol{v}}_{\text{SW,GEO}}$ is defined by

$$\widetilde{\boldsymbol{v}}_{\text{SW,GEO}} := \left(\widetilde{v}_{x,\text{SW,GEO}}, \widetilde{v}_{y,\text{SW,GEO}}, \widetilde{v}_{z,\text{SW,GEO}} \right)^{\text{T}}$$
$$:= R_z(\lambda_{\text{K}}) \boldsymbol{v}_{\text{SW,GEO}}.$$

Then, a vector in GEO coordinates $\boldsymbol{g}_{\text{GEO}}$ can be transferred into a vector in MSP coordinates $\boldsymbol{g}_{\text{MSP}}$ by

$$\boldsymbol{g}_{\text{MSP}} = R_y(\theta_{\text{K}}) R_z(\lambda_{\text{K}}) \boldsymbol{g}_{\text{GEO}}. \tag{23}$$

Applying the coordinate transformation, the solar wind velocity becomes parallel to the x axis.

For the validation of the code, the known planetary dipole moment components of Earth according to Table 1 with the normalization constant of Eq. (6) can be used:

$$m_x = -0.051 \, m_{\text{norm}},$$
$$m_y = 0.158 \, m_{\text{norm}}, \tag{24}$$
$$m_z = -0.945 \, m_{\text{norm}}.$$

To include the rotation of the planetary magnetic field due to the planet's rotation, the magnetic moments of the magnetic field are modified according to Eq. (23). The angles of the transformation will continuously vary along the spacecraft's trajectory. The rotation of the planetary magnetic field is performed every 200 time steps by subtracting the planetary field contribution from the total magnetic field at the time step considered and adding the planetary magnetic field corresponding to the new angles θ_{K} and λ_{K}.

2.4　Validation of the simulation code

To validate the modified simulation code, we compare a simulation using the known dipole moment of Earth according to Table 1 with THEMIS magnetosheath data from 24 August 2008 measured by THC (Angelopoulos, 2008). Solar wind conditions are observed by THB during the magnetosheath transition. The size of the simulation box is $x_{\text{L}} = 50.0 \, R_{\text{E}}$, $y_{\text{L}} = 60.0 \, R_{\text{E}}$, and $z_{\text{L}} = 60.0 \, R_{\text{E}}$, with the planet in the center. The simulation uses a grid with $i_{\text{max}} = 200$, $j_{\text{max}} = 150$, and $k_{\text{max}} = 150$. Furthermore, the values of the diffusion coefficients were chosen according to Ogino (1993) for a stable simulation at Earth. The data and corresponding
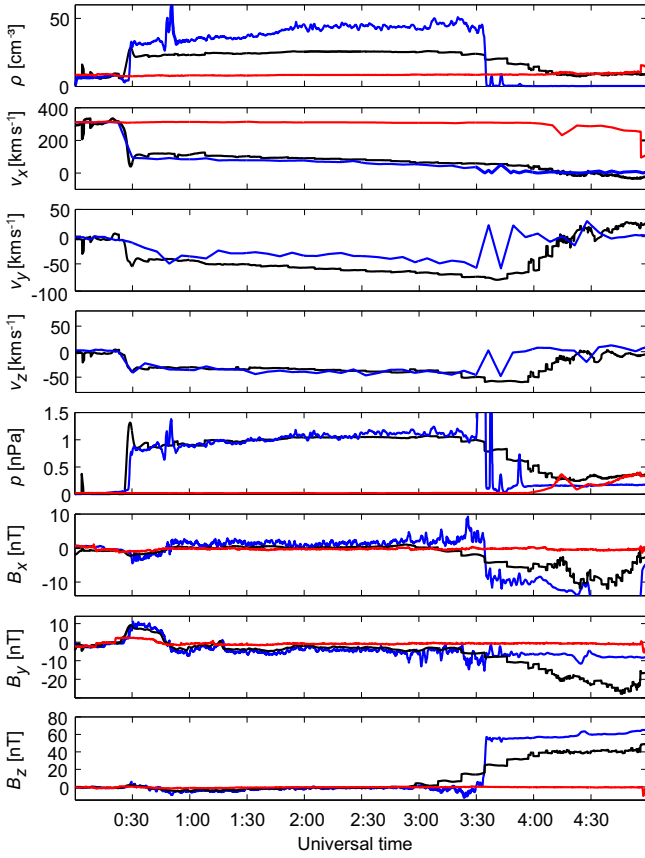
Figure 3. THC magnetosheath data (blue) observed on 24 August 2008 and the corresponding MHD simulation results (black) using the solar wind observations of THB (red).

simulation results on the spacecraft's trajectory are presented in Fig. 3.

The bow shock is observed at about 00:30 UT and the magnetopause at about 03:30 UT in accordance with the simulation results. Most physical quantities show a good agreement between actual observations and simulation results. Only the ion density in the magnetosheath is observed to be higher than in the simulation. Furthermore, the magnetic field in the magnetosphere is about 15 nT weaker than measured by the spacecraft. The magnetopause thickness is observed to be smaller than in the simulation, which is related to the diffusion coefficients required for a stable simulation. These differences between simulation result and data are mainly caused by numerical errors. This can impact an estimation of the planetary magnetic field. The lower magnetospheric magnetic field will tend to overestimate the planetary magnetic field strength. However, this overestimation is limited due to the magnetopause location. A much stronger dipole moment will increase the magnetopause distance and the magnetic field will increase in the magnetosheath, which is not in accord with the observations. In general, the MHD simulation results agree well with the observations made. In a

future step, the simulation code might be improved to reduce differences between simulation results and observations. An adaptive mesh refinement should be introduced to enhance the accuracy close to the magnetopause and reduce numerical errors.

3 Data assimilation

3.1 Cost function and its minimization

In the previous section, spacecraft data were qualitatively compared to the results of the MHD simulation. To quantify the deviations, a cost function is introduced. Therefore, the method of least squares is used. The sum of squared residuals, FQ, of M_{data}-measured values y_m at points x_m with a model f depending on the parameters s is

$$FQ := \sum_{m=1}^{M_{data}} (y_m - f(x_m, s))^2. \tag{25}$$

The parameters s of the MHD model are related to a vector space $\mathcal{P}$. Here, for simplification, we consider only the planetary magnetic field parameters of the dipole and quadrupole. Thus, the parameters are

$$s = (m, Q)^T = (m_x, m_y, m_z, Q_{xx}, Q_{xy}, Q_{xz}, Q_{yy}, Q_{yz})^T, \tag{26}$$

with $Q := (Q_{xx}, Q_{xy}, Q_{xz}, Q_{yy}, Q_{yz})^T$. The vector space corresponding to these parameters is named $\mathcal{P}_{D,Q}$. The parameters of the model s are estimated by minimizing the sum of squared residuals FQ. Transferred to the magnetic field observations $B_{data} := (B_{x,data}, B_{y,data}, B_{z,data})^T$ and MHD simulation results $B_{simu} := (B_{x,simu}, B_{y,simu}, B_{z,simu})^T$ with the spacecraft's position in the orbit $r_{SC,m}$, the cost function K is

$$K(s) = \sum_{m=1}^{M_{data}} \left(\left(B_{x,data}(r_{SC,m}) - B_{x,simu}(r_{SC,m}, s) \right)^2 \right.$$
$$+ \left(B_{y,data}(r_{SC,m}) - B_{y,simu}(r_{SC,m}, s) \right)^2 \tag{27}$$
$$\left. + \left(B_{z,data}(r_{SC,m}) - B_{z,simu}(r_{SC,m}, s) \right)^2 \right).$$

A gradient-based optimization can be used to minimize the cost function with respect to the parameters of the model s. Starting from a point s_0 in parameter space, new points $s_k = (m_k, Q_k)^T$ are determined with every kth gradient calculation. This optimization problem is without constraints and can be solved using a quasi-Newton method. We use the Broyden–Fletcher–Goldfarb–Shanno (BFGS) algorithm (Press et al., 1992) to minimize the cost function.

The algorithm requires the gradient of the cost function K with respect to the parameters of the model at points s_k in parameter space. There are different possibilities to compute these gradients. For example, the gradient can be approximated by difference quotients:

$$\frac{\partial K(s)}{\partial s} \Big|_{s=s_k} \approx \sum_{l=1}^{N_{\mathcal{P}}} \frac{K(s_k + \Delta s_l e_l) - K(s_k)}{\Delta s_l} e_l, \tag{28}$$

where e_l denotes the lth unit vector and Δs_l is the corresponding step size in parameter space $\mathcal{P}$. The sum of Eq. (28) includes all dimensions in parameter space ($N_{\mathcal{P}} := \dim(\mathcal{P})$). Note that the gradient $\partial_s K(s)$ is used as a column vector. The step sizes Δs_l need to be adequately small to approximate the gradient sufficiently well.

3.2 Automatic differentiation and adjoint method

Each calculation of the cost function for a certain set of parameters requires a full global MHD simulation of the data along the spacecraft's trajectory. Thus, ($N_{\mathcal{P}} + 1$) simulations have to be performed in the calculation of the gradient according to Eq. (28). In general, the calculation is extremely time-consuming because of the nonlinearity of the MHD equations.

Another possibility to calculate the gradient $\partial_s K(s)$ is the differentiation using analytical expressions, as explained in detail in the following. The solution of the MHD simulation depending on space and time coordinates $u(t, x)$ can be represented by a vector $u_{t,x}$ on a numerical grid. This vector contains the solution at all time steps and discrete positions in space for all physical quantities in its components. Thus, the number of components of the vector $u_{t,x}$ is

$$N_v = N_{var} N_{grid} l_{max}, \tag{29}$$

with the constants as defined in the previous chapter. The simulation code calculates the time- and spatially dependent solution of the MHD quantities $u(t, x)$ iteratively. The iteration is implemented by a time loop in the simulation code, whereby $u(t = l \Delta t, x)$ is computed from the results of the previous time step $u(t = (l-1) \Delta t, x)$ and the boundary conditions. Equivalently, the time iteration of l can be considered as an iteration that steadily improves the approximation of the final solution vector $u_{t,x}$, sketched in Fig. 4. Thereby, after the lth iteration step, the vector $q^l_{t,x}$ contains the valid solution for all time steps that satisfy $t \leq l \Delta t$. The final solution $q^{l_{max}}_{t,x} = u_{t,x}$ is obtained after l_{max} iteration steps. At the 0th iteration step, the simulation needs to be initialized. The simulation code calculates the solution in the lth iteration step from the previous approximation by a function F. Thus, the lth iteration step can be expressed by

$$q^l_{t,x}(s) = F(q^{l-1}_{t,x}(s)). \tag{30}$$

The cost function $K(q^{l_{max}}_{t,x}(s))$ depends implicitly on the parameters s. With respect to the nested dependences of the solution in Eq. (30), the gradient of the cost function can be expressed by the chain rule:

$$\frac{\partial K(q^{l_{max}}_{t,x}(s))}{\partial s} = \frac{\partial K(q^{l_{max}}_{t,x})}{\partial q^{l_{max}}_{t,x}} \cdot \frac{\partial q^{l_{max}}_{t,x}}{\partial q^{l_{max}-1}_{t,x}} \cdots$$
$$\cdot \frac{\partial q^1_{t,x}}{\partial q^0_{t,x}} \cdot \frac{\partial q^0_{t,x}(s)}{\partial s}. \tag{31}$$
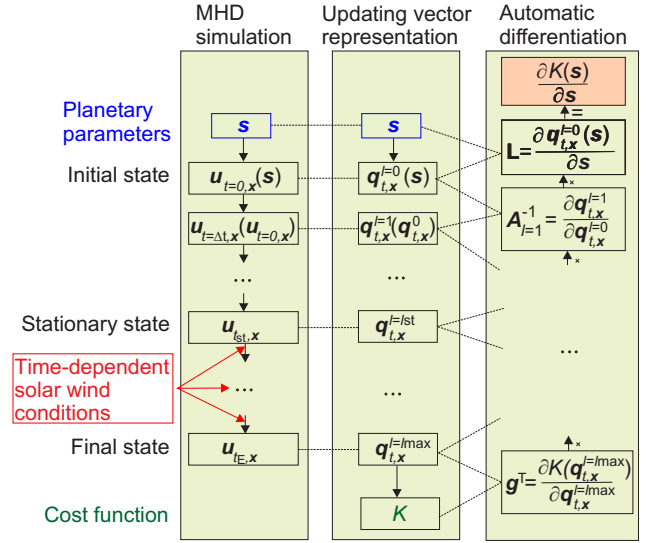

Figure 4. The left column sketches the time iteration of the MHD simulation code starting from the initial state at $t = 0$ determined by the planetary magnetic field parameters s. At a certain time step, a stationary solution $u_{st,x}$ is obtained and time iteration continues using time-dependent solar wind conditions until the simulation ends at $u_{t_E,x}$. In the middle column, the corresponding interpretation of updating the complete time- and spatially dependent solution vector $q_{t,x}$ is presented. The cost function K is calculated from the final vector. On the right side, the automatic differentiation gradient calculation is presented, starting from the bottom and multiplying each factor according to Eq. (31).

This expression contains the stationary solution at $l = l_{st}$ because $0 < l_{st} < l_{max}$. Although the cost function is evaluated only after the stationary state has been obtained, i.e. $l > l_{st}$, the cost function also depends implicitly on prior time steps because the stationary solution emerges from the initial state. The magnetic field components of the initial state vector $u^0_{t,x}$ depend on the parameters s because the initial magnetic field distribution in the simulation is created by the dipole and quadrupole parameters. Equation (31) can be written as

$$\frac{\partial K(s)}{\partial s} = g^T \cdot A^{-1}_{l_{max}} \cdot A^{-1}_{l_{max}-1} \cdots A^{-1}_1 \cdot L \tag{32}$$

using the following abbreviations:

$$A^{-1}_l := \frac{\partial q^l_{t,x}}{\partial q^{l-1}_{t,x}}, \qquad l = 1, 2, \ldots, l_{max},$$

$$g^T := \frac{\partial K(q^{l_{max}}_{t,x})}{\partial q^{l_{max}}_{t,x}}, \tag{33}$$

$$L := \frac{\partial q^0_{t,x}(s)}{\partial s}.$$

The function F in Eq. (30) is determined by the Lax–Wendroff scheme of the differential equations, which is related to the MHD equations and the boundary conditions.

Therefore, the derivatives of the matrices in Eq. (31) can be determined by analytical expressions. The time iteration of the simulation code starts at $l = 0$ and ends at l_{max}. After the lth iteration step, the corresponding matrix $\mathbf{A}_l^{-1}$ can be calculated. Starting from a unit matrix, the matrix containing the derivatives is multiplied after every time step to the left side. Finally, after l_{max} iterations, the gradient $\partial K(s)/\partial s$ is obtained.

This procedure is called forward differentiation because the gradient is calculated parallel to the execution of the time loop in the simulation code. The advantage over the computation of the gradient using difference quotients according to Eq. (28) is that no errors due to finite step sizes occur. Forward differentiation can be applied by hand to the simulation code, or alternatively, by an automatic differentiation (AD) tool (Wengert, 1964). Therefore, the cost function K and its dependent parameters s need to be declared in the code. The AD tool identifies all implicit dependences. The required analytical expressions for the derivations are taken from a library of the AD tool and inserted at the correct positions in the code. Note that the library contains elementary analytical derivations of all important expressions such as $\partial_x \sin(x) = \cos(x)$. According to Eq. (31), the inserted expressions are related to each other such that the required gradient is computed. Several different AD tools were developed during the last decades. Here, the Transformation of Algorithms in Fortran (TAF) tool (Giering and Kaminski, 2003) from the company FastOpt was used (http://www.FastOpt.com).

An AD tool is able to differentiate a numerical code automatically, i.e., the tool can be applied without considering details of the implementation. However, for complex numerical codes, such as MHD simulation codes, problems might occur. For example, codes using parallel computing function calls by the message-passing interface (MPI), as they are also used for our simulation code, usually need further treatment. The analytical forward differentiation with an AD tool is also called automatic forward differentiation.

The computational costs for the calculation of the gradient with difference quotients or using automatic forward differentiation do not differ much. However, the latter procedure leads to a more efficient approach, the adjoint method. The adjoint method is extensively used for optimization problems in fluid dynamics, e.g., drag minimization by variations in surface geometry (e.g., Jameson, 1988; Othmer, 2008, 2014; Meader and Martins, 2012) or in seismology (e.g., Fichtner et al., 2006).

The adjoint method can be introduced with systems of linear equations, as described briefly in the following (e.g., Giles and Pierce, 2000; McNamara et al., 2004; Nabert et al., 2015). The symbols used for variables, vectors, and matrices refer to the previous considerations and will be marked by an asterisk as an index for distinction. We consider the following system of equations

$$\mathbf{A}_* \cdot \mathbf{X}_* = \mathbf{L}_*, \tag{34}$$

with the matrices of the coefficients $\mathbf{A}_*$, the solution $\mathbf{X}_*$, and the inhomogeneity $\mathbf{L}_*$. All elements of the matrices are real numbers. The scalar product of a vector $\boldsymbol{g}_*$ with the matrix $\mathbf{X}_*$ should be calculated using the following equation:

$$\boldsymbol{g}_*^T \cdot \mathbf{X}_* = ?. \tag{35}$$

This scalar product can be computed by solving Eq. (34) first, and then calculating the product of vector $\boldsymbol{g}_*$ with the solution $\mathbf{X}_*$. This approach is called forward calculation.

Another possibility is to use the adjoint method. To deduce the method, the product of a vector $\boldsymbol{y}^T$ with both sides of the system of linear Eq. (34) is considered:

$$\boldsymbol{y}_*^T \cdot \mathbf{A}_* \cdot \mathbf{X}_* = \boldsymbol{y}_*^T \cdot \mathbf{L}_*. \tag{36}$$

The vector $\boldsymbol{y}_*$ is defined by

$$\boldsymbol{y}_*^T \cdot \mathbf{A}_* = \boldsymbol{g}_*^T. \tag{37}$$

This equation is transposed, which leads to the adjoint system of equations

$$\mathbf{A}_*^T \boldsymbol{y}_* = \boldsymbol{g}_*. \tag{38}$$

Using Eqs. (36) and (37), the scalar product (Eq. 35) can be written as

$$\boldsymbol{g}_*^T \cdot \mathbf{X}_* = \boldsymbol{y}_*^T \cdot \mathbf{A}_* \cdot \mathbf{X}_* = \boldsymbol{y}_*^T \cdot \mathbf{L}_*. \tag{39}$$

If the adjoint system of Eq. (38) is solved, $\boldsymbol{y}_*^T \cdot \mathbf{L}_*$ can be computed, which is nothing other than the scalar product (Eq. 35) as seen in Eq. (39).

The computational costs are mainly determined by the number of multiplications and differ for both possibilities of calculating the scalar product (Eq. 35). Only in case of a column vector inhomogeneity $\mathbf{L}_*$, is the number of multiplications equal. If the matrix $\mathbf{L}_*$ consists of $N_{*,\mathcal{P}}$ column vectors, $N_{*,\mathcal{P}}$ systems of linear equations with a vector inhomogeneity need to be solved in the forward calculation. The adjoint method is independent of $N_{*,\mathcal{P}}$ and only a single system of linear equations needs to be solved. Therefore, the latter approach requires $N_{*,\mathcal{P}}$ times fewer multiplications.

The adjoint approach can be applied to the calculation of the gradient in Eq. (32). If the product of all matrices $\mathbf{A}^{-1} := \mathbf{A}_{l_{max}}^{-1} \cdot \ldots \cdot \mathbf{A}_1^{-1}$ in Eq. (32) is substituted, this leads to

$$\frac{\partial K(s)}{\partial s} = \boldsymbol{g}^T \cdot \mathbf{A}^{-1} \cdot \mathbf{L}. \tag{40}$$

The second product on the right side of Eq. (40) can be substituted by

$$\mathbf{X} := \mathbf{A}^{-1} \cdot \mathbf{L}. \tag{41}$$

This equation can be related to the system of linear Eq. (34) by identifying the quantities with an asterisk as an index.

During the analytical forward differentiation, the gradient is computed successively using chain rule from the right to the left. This corresponds to a procedure, where, at first, the system of linear Eq. (34) is solved with respect to $\mathbf{X}_*$ and then, the scalar product (Eq. 35) is calculated. If the matrix products of Eq. (40) are computed from left to right, at first, the product

$$\boldsymbol{y}^{\mathrm{T}} := \boldsymbol{g}^{\mathrm{T}} \cdot \mathbf{A}^{-1} \qquad (42)$$

is determined. This corresponds to solving the adjoint Eq. (38) with respect to $\boldsymbol{y}_*$. The scalar product (Eq. 35) is determined by $\boldsymbol{y}_*^{\mathrm{T}} \cdot \mathbf{L}_*$, which is related to the multiplication of $\boldsymbol{y}^{\mathrm{T}} \cdot \mathbf{L}$ to determine the cost function. Thus, the adjoint method for the gradient calculation of the cost function can be identified with the execution of the multiplications from the left to the right in Eq. (40).

The dimensions of the vectors and matrices involved are $\dim(\boldsymbol{g}) = N_{\mathrm{v}} \times 1$, $\dim(\mathbf{L}) = N_{\mathrm{v}} \times N_{\mathcal{P}}$, and $\dim(\mathbf{A}) = N_{\mathrm{v}} \times N_{\mathrm{v}}$. The calculation of the gradient by computing the matrices from the right to the left in Eq. (40) requires N_{rl} multiplications of components, whereby

$$N_{\mathrm{rl}} = N_{\mathrm{v}}^2 N_{\mathcal{P}} (N_{\mathrm{v}} + 1). \qquad (43)$$

If the gradient is calculated from the left to the right in Eq. (40), N_{lr} multiplications of components are performed:

$$N_{\mathrm{lr}} = N_{\mathrm{v}}^2 (N_{\mathrm{v}} + N_{\mathcal{P}}). \qquad (44)$$

The limit $N_{\mathcal{P}} = 1$ leads to $N_{\mathrm{rl}} = N_{\mathrm{lr}}$. Usually, one can assume $N_{\mathcal{P}} \ll N_{\mathrm{v}}$ because the number of grid points exceeds the dimensions of parameter space, which is eight for the dipole and quadrupole parameters. Then, Eq. (44) simplifies to

$$N_{\mathrm{lr}} = N_{\mathrm{rl}} N_{\mathcal{P}}. \qquad (45)$$

Thus, the multiplication of the matrices in Eq. (40) from the left to the right, the adjoint approach, is more efficient for many parameters and requires about $N_{\mathcal{P}}$ times fewer multiplications. The evaluation procedure for the simulation code is depicted in Fig. 4.

However, the numerical implementation of the adjoint method is more difficult than the analytical forward differentiation. As described, the calculation of the gradient with the analytical forward differentiation is parallel to the execution of the time loop in the simulation code. In contrast, the solution at the last time iteration $\boldsymbol{q}_{t,\boldsymbol{x}}^{l_{\max}}$ has to be known to calculate $\boldsymbol{g}^{\mathrm{T}} \cdot \mathbf{A}_{l_{\max}}^{-1}$. Thus, at first, the simulation needs to be performed once, whereby all calculation results that are required for the matrix multiplications are stored temporarily. Then, the gradient can be computed according to the adjoint approach.

There are AD tools that can derive codes not only according to forward differentiation but also according to the adjoint method. However, the available memory on a computer is often too small to store all the required results in the central memory. The memory consumption M_{Memory} can be estimated by multiplying the number of grid points of the simulation box N_{grid} according to Eq. (14) with the number of time steps $l_{\max}$, the number of MHD variables N_{var}, and the size of a MHD variable M_{var}:

$$M_{\mathrm{Memory}} \approx N_{\mathrm{grid}} \, l_{\max} \, N_{\mathrm{var}} \, M_{\mathrm{var}}. \qquad (46)$$

The number of variables of the MHD simulation is $N_{\mathrm{var}} = 8$ and the size of such a variable is $M_{\mathrm{var}} = 4$ bytes if a float variable is assumed. This gives a memory consumption of about 1600 GB for a simulation grid $i_{\max} = j_{\max} = k_{\max} = 100$ and $l_{\max} = 5 \times 10^5$. The central memory is often much smaller, so that a certain portion of the variables needs to be stored on the hard disk. However, the seek time of the central memory is much smaller, and thus, the runtime of the algorithm becomes longer by storing data on the hard disk.

To minimize the access to the hard disk, checkpointing can be used. Thereby, the main iteration loop of the algorithm is split at certain checkpoints into smaller loops. Then, the smaller loop iterates over $N_{\mathrm{loop,check}}$ iterations instead of the complete time loop with $l_{\max}$ iterations. This reduces the memory requirements for such a loop to

$$M_{\mathrm{Memory,check}} = \frac{N_{\mathrm{loop,check}}}{l_{\max}} M_{\mathrm{Memory}}. \qquad (47)$$

The variables during a calculation of such a smaller loop can be stored within the central memory. After the execution of the smaller loop, the results are stored to the hard disk to combine all results of the smaller loop. However, using smaller loops, the adjoint method can only be applied within these smaller loops. Thus, checkpointing reduces the seek time of the memory, but the adjoint approach is restricted to a smaller part of the algorithm. In total, this reduces the runtime of the algorithm, but the performance is below the theoretical possible performance of the adjoint approach with unlimited central memory space.

Note that instead of using only observations of a single spacecraft, simultaneous measurements from multiple spacecraft at different locations can be calculated in Eq. (27) as well. This can be done without additional computational costs and memory capacity because the solution of the MHD simulation is calculated in the entire simulation domain and stored anyway.

3.3 Adjoint MHD simulation code

The AD tool applied for an automatic backward differentiation transfers a numerical code for the calculation of a cost function into an adjoint code, which can compute the gradient according to the adjoint approach. This was done for the MHD simulation code presented in the previous chapter by the TAF tool of the company FastOpt. Thereby, the parameter space of the dipole and quadrupole parameters $\mathcal{P}_{\mathrm{D,Q}}$ was
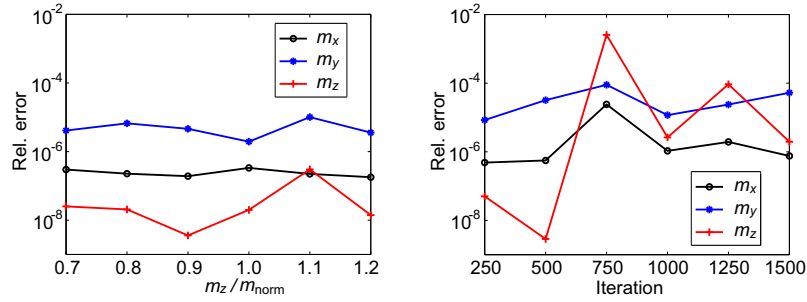
Figure 5. The relative errors for gradients determined by difference quotients and the adjoint method for the dipole components. Thereby, on the left side, different points in parameter space are considered. On the right side, the dependence of the error on a different number of time iteration steps is shown.
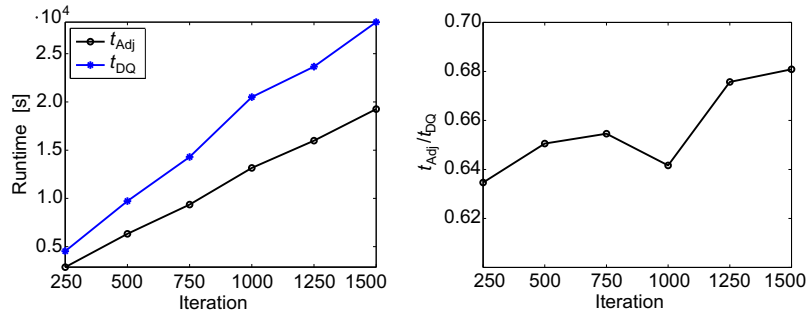

Figure 6. The runtime of calculating gradients using the adjoint method t_{Adj} and using difference quotients t_{DQ}. On the left side, the dependence on the number of time iteration steps is presented. On the right side, the ratio of the runtimes is shown.

considered. Thus, the adjoint code computes the gradient of the cost function (27) with respect to the parameters (26).

To validate the adjoint MHD simulation code, the gradients produced by the adjoint code are compared to those calculated by difference quotients according to Eq. (28). Therefore, at first, the interaction of the solar wind with the planetary magnetic field is neglected and the planetary magnetic field represented by its dipole and quadrupole moments is only taken into account. Gradients at certain points $s_0 = (0, 0, m_z, 0, 0, 0, 0, 0)^T$ in parameter space are considered. Thereby, the m_z component varies between 0.7 and $1.2 m_{norm}$ with a step size of $0.1 m_{norm}$. The spacecraft data B_{data} on a trajectory r_{SC}, required to calculate the cost function, are generated synthetically along the x axis between 20.2 and $9 R_E$ with a step size of $0.42 R_E$. The gradient of the cost function is calculated using difference quotients $\partial_{s_0} K_{DQ}$ and the adjoint method $\partial_{s_0} K_{Adj}$ for different s_0. The relative error of the ith component of the gradient is defined by

$$\text{rel. error} := \frac{(\partial_{s_0} K_{DQ} - \partial_{s_0} K_{Adj}) \cdot e_i}{\max(\partial_{s_0} K_{DQ} \cdot e_i, \partial_{s_0} K_{Adj} \cdot e_i)}. \quad (48)$$

Here, The result of the maximum function $\max(a, b)$ is the larger value of a and b and e_i defines the i unit vector. The relative error of the dipole moment for different s_0 is depicted in Fig. 5. The error is smaller than 10^{-4}, i.e., both gradients agree for different values of m_z.

Now, the interaction of the planet with the solar wind is taken into account. Thereby, the gradients calculated by the adjoint method can be compared to gradients computed by difference quotients for a different number of time iterations. The corresponding relative errors of the dipole components of the gradient are shown in Fig. 5 on the right side. It is seen that the gradients agree very well.

To determine the runtime of the adjoint code, the gradient calculations are performed on a test computer for different numbers of time iteration steps. The test computer uses 64 GB of central memory and has an Intel Xeon E5 processor with 12 cores and 24 threads at 2.5 GHz. The results are shown in Fig. 6 on the left side. The runtime of calculating the gradients increases linearly with the number of iteration steps performed, as expected. The plot on the right side presents the ratio of the runtime of the adjoint code t_{Adj} and the runtime using difference quotients t_{DQ}. On the test computer, the adjoint method calculates the gradient about 33 % faster than using the difference quotients. According to the previous argumentation, eight parameters require nine MHD simulation calls to determine the gradient with difference quotients of Eq. (28). The adjoint method needs to run the simulation once to store all necessary results and another simulation run to calculate the gradient. In Fig. 4, the first simulation run is shown on the left side from top to bottom, storing all results in vector representation presented in
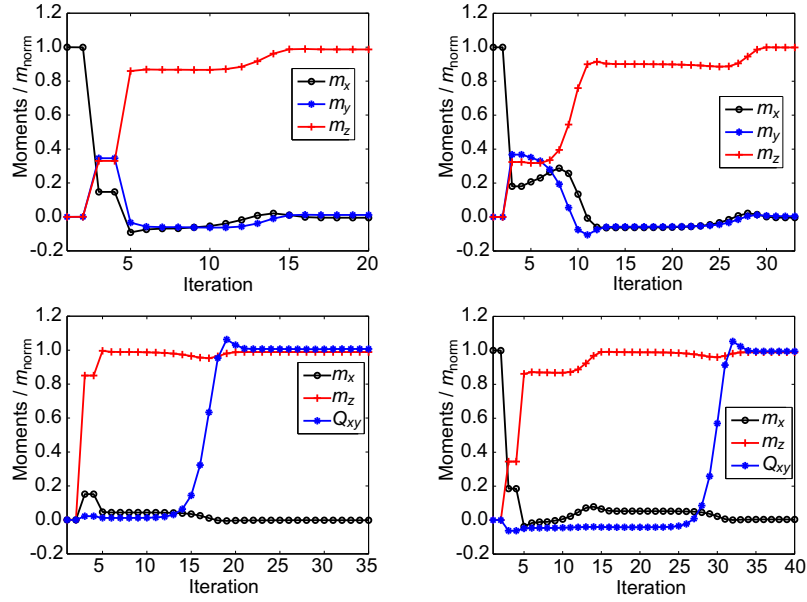
Figure 7. Estimation of planetary moments neglecting the interaction with the solar wind and using synthetic data. The reconstructions shown in the upper panels take only a dipole moment into account. These two results consider a different spacecraft trajectory, with a trajectory closer to the planet corresponding to the left figure. All reconstructions presented use the same dipole moment for calculating the synthetic spacecraft data. However, in the lower panels, an additional quadrupole component was introduced. The reconstructions in the lower panels use a different choice of the initial parameters for the estimation.

the middle. Using these results, the automatic differentiation procedure calculates the gradient as sketched on the right from bottom to top, which corresponds to another simulation run. Thus, in theory, the adjoint method can be up to 78 % faster than the difference quotient calculation. However, our adjoint code uses checkpointing because the central memory is too small, which increases the runtime.

Consequently, the test computer configuration is not optimal to achieve the best performance. The performance can be improved by using a computer cluster with distributed memory space. Then, each core can access its own memory space and checkpointing can be avoided. This can increase the performance. Furthermore, it should be noted that without additional computational costs and memory requirements, more parameters can be introduced in the estimation process of the adjoint approach, such as octupole planetary magnetic field parameters.

4 Estimation of planetary magnetic field parameters

4.1 Using synthetic data

At first, the results of data assimilation using synthetically produced data are considered, neglecting the interaction of the planetary magnetic field. The simulation box has a length of 60.2 R_E in every dimension with the planet in its center. The number of grid points is $i_{max} = j_{max} = k_{max} = 300$. Synthetic spacecraft data $\boldsymbol{B}_{data}$ are calculated from the magnetic

field distributions of certain dipole and quadrupole parameters $\boldsymbol{s}_{Planet}$ along a trajectory $\boldsymbol{r}_{SC}(x)$. The initialization $\boldsymbol{s}_0$ for the estimation procedure of the planetary parameters differ from these moments. Starting from this initialization, the cost function is minimized.

The first trajectory considered here is $\boldsymbol{r}_{SC}(x) = (x, 10.1\ R_E - x, 0)^T$, which is diagonal within the xy plane. The spacecraft's magnetic field data $\boldsymbol{B}_{data}$ are generated by a dipole along the z axis, i.e., $\boldsymbol{s}_{Planet} = (0, 0, 1, 0, 0, 0, 0, 0)^T m_{norm}$. In parameter space, the starting point of the estimation procedure is $\boldsymbol{s}_0 = (1, 0, 0, 0, 0, 0, 0, 0)^T m_{norm}$, which is nothing other than a dipole along the x axis. The BFGS algorithm iteratively computes new gradients in which direction the cost function (27) is minimized. The corresponding dipole parameters during the minimization, depending on the iteration step of calculating new gradients, are presented in Fig. 7 in the top left panel. The vector of the dipole moment $\boldsymbol{s}_{Planet}$ is reconstructed very well after 15 iteration steps.

Next, a different trajectory is considered to produce the synthetic data: $\boldsymbol{r}_{SC} = (x, 30.1\ R_E - x, 0)^T$. This orbit is farther away from the planet than the previous trajectory. The results of the estimation process are depicted in Fig. 7 in the top right panel. Again, the dipole vector was reconstructed very well, however, about twice as many iterations were required. This is related to the variations in the magnetic field strength, with a smaller percentage ratio of the variations on the trajectory farther out.
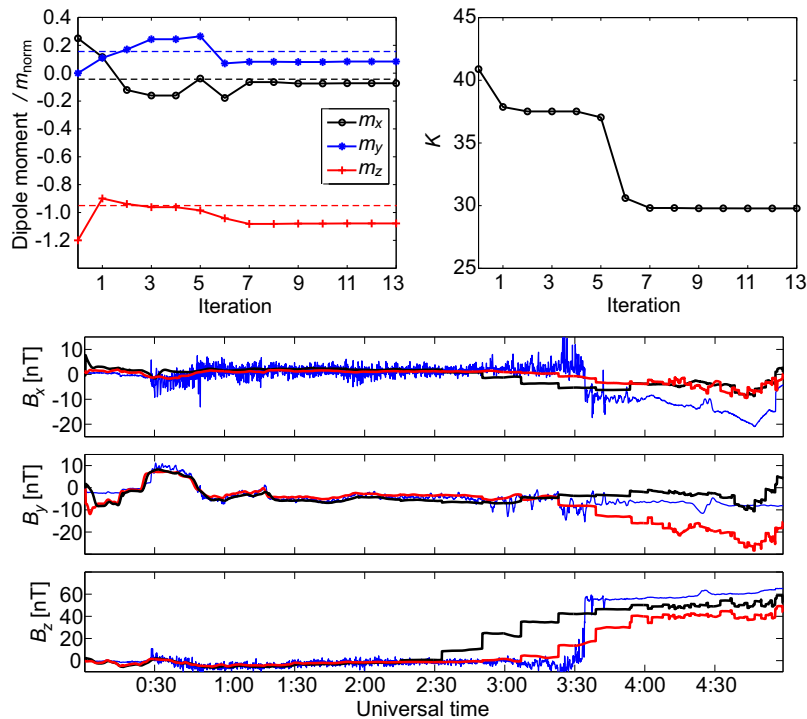
Figure 8. Estimation of Earth's dipole moment from magnetosheath data of THC on 24 August 2008 using the THB solar wind observations, which are presented in Fig. 3. The variations in the dipole moment during the iteration process (top left) and the corresponding values of the cost function (top right) are shown. The values of the dipole moment of Earth are sketched as dashed lines. The plot on the bottom shows the iterative assimilation of the MHD simulation to the THC data (blue) before the first iteration (black) and after the 13th iteration (red).

The simultaneous estimation of dipole and quadrupole parameters is considered as well by using magnetic field data $\boldsymbol{B}_{\text{data}}$ generated by $\boldsymbol{s}_{\text{Planet}} = (0, 0, 1, 1, 0, 0, 0, 0)^{\text{T}} m_{\text{norm}}$. Thereby, the trajectory $\boldsymbol{r}_{\text{SC}}(x) = (x, 10.1 \, R_{\text{E}} - x, 0)^{\text{T}}$ is used. The reconstruction of the planetary magnetic field, starting from $\boldsymbol{s}_0 = (0, 0, 0, 0, 0, 0, 0, 0)^{\text{T}} m_{\text{norm}}$, is shown in Fig. 7 in the bottom left panel. Additionally, the estimation process from a different point in parameter space $\boldsymbol{s}_0 = (1, 0, 0, 0, 0, 0, 0, 0)^{\text{T}} m_{\text{norm}}$ is realized. The results are presented in Fig. 7 in the bottom right panel. In both situations, the moments $\boldsymbol{s}_{\text{Planet}}$ were correctly determined. Thereby, the estimation starting in parameter space farther away from the solution required 15 more iteration steps. Altogether, it is seen that the dipole as well as the quadrupole parameters can be reconstructed from synthetic data, whereby larger magnetic field variations along the trajectory or a starting point $\boldsymbol{s}_0$ closer to the minimum speed up the estimation process.

4.2 Using THEMIS data

After proving the general functionality of the algorithm, it is applied using THEMIS spacecraft data at Earth. Thereby, data from the magnetosheath, a region strongly influenced by the interaction process of the solar wind, is considered. Different to the situation at Earth, spacecraft magnetic field observations in Mercury's magnetosphere are strongly mod-

ified due to the magnetosphere's small size. Here, we restrict our approach to Earth's magnetosheath data to consider a strongly disturbed magnetic environment comparable to the situation at Mercury. However, in final application, magnetospheric data will be used as well to reduce errors. Due to the weak components of the quadrupole at Earth, their influence is negligible close to the magnetopause. The largest quadrupole moment corresponds to a magnetic field of < 0.1 nT at $10 \, R_{\text{E}}$. This is very small compared to the contribution of the dipole's z component of about 30 nT. Thus, only the dipole moment is considered in the estimation at Earth. The estimation process starts from $\boldsymbol{s}_0 = (0.25, 0, -1.2, 0, 0, 0, 0, 0)^{\text{T}} m_{\text{norm}}$ in parameter space. Subsequently, the cost function is minimized iteratively, whereby every new calculation of a gradient denotes a new iteration step.

The magnetosheath transition observed by THC on 24 August 2008, presented in Fig. 3, is used as a first estimation of the dipole parameters. Thereby, the solar wind measurements of the THB spacecraft determine the inflow boundary conditions of the MHD simulations. The results of the estimation process are depicted in Fig. 8. Thereby, the values of the dipole moment as well as the cost function are shown. The dipole components vary mainly during the first iterations. The value of the cost function is strongly reduced from iteration steps 0 to 1

and 6 to 7. After the eighth iteration step, the cost function and the components of the dipole moment do not change much. Finally, the solution vector after 13 iteration steps is $s_{13} = (-0.072, 0.084, -1.078, 0, 0, 0, 0, 0)^T m_{norm}$. Thereby, all components are closer to the value of the dipole moment of Earth according to Eq. (24) than the initial values. The relative errors for the dipole components are $\Delta m_x = (s_{13,1} - m_{x,E})/m_{x,E} = -0.44$, $\Delta m_y = (s_{13,2} - m_{y,E})/m_{y,E} = -0.47$, and $\Delta m_z = (s_{13,3} - m_{z,E})/m_{z,E} = 0.14$. The relative error of the z component is the smallest because Earth's dipole is mainly along the z direction. Considering the magnitude, the relative error is 0.13. The panels showing B_x, B_y, and B_z in Fig. 8 show the magnetic field distribution of the MHD simulation, which corresponds to s_0 and s_{13}.

5   Conclusions

We introduced an approach to estimating planetary parameters using global MHD simulations of the interaction of the solar wind with a planet. A simple MHD simulation code was introduced and prepared for an automatic differentiation tool to obtain an adjoint MHD simulation code. The differences of spacecraft data and corresponding simulation results are quantified by a cost function, which is minimized by a gradient-based optimization. The adjoint code computes the gradient with lower computational effort compared to a difference quotient calculation.

Our approach is designed to estimate planetary magnetic fields, especially if the field strength is weak so that the interaction strongly modifies the magnetic field of the planet's environment. We used THEMIS data of Earth's magnetosheath to simulate such an environment to test our approach. The results of the estimation process can be affected by statistical and systematic errors. Therefore, statistical errors will not contribute to the mean values of the estimated planetary magnetic field if a sufficiently large number of magnetosheath transitions are considered. For example, the solar wind density can be measured incorrectly due to a spacecraft potential (McFadden et al., 2008). However, the density is usually equally overestimated and underestimated. Considering a single magnetosheath transition, the estimated dipole magnitude of Earth differs about 13 % from the expected value. Based on this approach, further transitions can be considered to minimize the errors. Note that including magnetospheric data at Mercury will further reduce the statistical error. The runtime of the parameter estimation using the test computer is about 1 week using the 5 h magnetosheath data. This fast calculation procedure allows taking a lot more data into account, especially if supercomputers are used.

We used a simple MHD simulation code to investigate the automatic differentiation procedure. As a next step, the simulation code needs to be improved, e.g., by an adaptive mesh refinement, to reduce numerical errors. Also, kinetic or hybrid simulation codes can be considered and treated with an automatic differentiation tool. The limiting factor for applying the automatic differentiation is not the complexity of the code but the memory consumption. Using our test computer, the adjoint approach was about 33 % faster than a finite difference approach. Although the adjoint MHD code does not calculate the gradient very much faster than using difference quotients, it has the advantage that further parameters such as higher-order magnetic field moments or parameters of the planet's conductivity can be included with nearly no additional computational costs. Nonetheless, the performance of the adjoint code is, related to memory limitations of our test computer, much lower than expected from theory. Thus, as a further step, the test computer configuration needs to be modified to increase performance. It is beneficial for the adjoint approach that each core has access to its own memory, which is different from our test computer. Thus, instead of using traditional supercomputers with fewer more powerful computers, a computer cluster using many commoditized computers with their own memory should be considered. These computer configurations recently became very popular in big data analysis using Google's well-known MapReduce technique (Dean and Ghemawat, 2004). A similar configuration might be more suitable for the adjoint code and increase its performance. The ability of our approach to perform on clusters with many cores depends on the parallelization of the MHD simulation code. Although this can be limited to a certain number of cores, another possibility to parallelize the estimation process is to split the data into subsets and perform the calculation of these subsets in parallel. Each data set will provide an individual estimator of the planetary parameters that can be applied in an ensemble averaging technique to reduce errors.

References

Alexeev, I. I., Belenkaya, E. S., Slavin, J. A., Korth, H., Anderson, B. J., Baker, D. N., Boardsen, S. A., Johnson, C. L., Purucker, M. E., Sarantos, M., and Solomon, S. C.: Mercury's magnetospheric magnetic field after the first two MESSENGER flybys, Icarus, 209, 23–39, doi:10.1016/j.icarus.2010.01.024, 2010.

Angelopoulos, V.: The THEMIS Mission, Space Sci. Rev., 141, 5–34, doi:10.1007/s11214-008-9336-1, 2008.

Benkhoff, J., van Casteren, J., Hayakawa, H., Fujimoto, M., Laakso, H., Novara, M., Ferri, P., Middleton, H. R., and Ziethe, R.: BepiColombo – Comprehensive exploration of Mercury: Mission overview and science goals, Planet. Space Sci., 58, 2–20, doi:10.1016/j.pss.2009.09.020, 2010.

Clauser, C.: Einführung in die Geophysik, Springer Spektrum, Berlin Heidelberg, 2016.

Dean, J. and Ghemawat, S.: MapReduce: Simplified data processing on large clusters, in: Proceedings of Operating Systems Design and Implementation, 137–150, San Francisco, 2004.

Fichtner, A., Bunge, H.-P., and Igel, H.: The adjoint method in seismology: I – Theory, Phys. Earth Planet. In., 157, 86–104, doi:10.1016/j.pepi.2006.03.016, 2006.

Finlay, C. C., Maus, S., Beggan, C. D., Bondar, T. N., Chambodut, A., Chernova, T. A., Chulliat, A., Golovkov, V. P., Hamilton, B., Hamoudi, M., Holme, R., Hulot, G., Kuang, W., Langlais, B., Lesur, V., Lowes, F. J., Lühr, H., MacMillan, S., Mandea, M., McLean, S., Manoj, C., Menvielle, M., Michaelis, I., Olsen, N., Rauberg, J., Rother, M., Sabaka, T. J., Tangborn, A., Tøffner-Clausen, L., Thébault, E., Thomson, A. W. P., Wardinski, I., Wei, Z., and Zvereva, T. I.: International Geomagnetic Reference Field: the eleventh generation, Geophys. J. Int., 183, 1216–1230, doi:10.1111/j.1365-246X.2010.04804.x, 2010.

Gauss, C. F.: Allgemeine Theorie des Erdmagnetismus, in: Resultate aus den Beobachtungen des Magnetischen Vereins im Jahre 1838, edited by: Gauss, C. F. and Weber, W., 1–59, Göttinger Magnetischer Verein, Leipzig, 1839.

Giering, R. and Kaminski, T.: Applying TAF to generate efficient derivative code of Fortran 77–95 programs, Proceedings in Applied Mathematics and Mechanics, 2, 54–57, 2003.

Giles, M. B. and Pierce, N. A.: An Introduction to the Adjoint Approach to Design, Flow Turbul. Combust., 65, 393–415, doi:10.1007/s11214-008-9365-9, 2000.

Glassmeier, K.-H.: Currents in Mercury's Magnetosphere, in: Magnetospheric Current Systems, Geophysical Monograph 118, 371–380, American Geophysical Union, Washington DC, 2000.

Glassmeier, K.-H. and Tsurutani, B. T.: Carl Friedrich Gauss – General Theory of Terrestrial Magnetism – a revised translation of the German text, History of Geo- and Space Sciences, 5, 11–62, doi:10.5194/hgss-5-11-2014, 2014.

Glassmeier, K.-H., Grosser, J., Auster, U., Constantinescu, D., Narita, Y., and Stellmach, S.: Electromagnetic Induction Effects and Dynamo Action in the Hermean System, Space Sci. Rev., 132, 511–527, doi:10.1007/s11214-007-9244-9, 2007.

Grosser, J., Glassmeier, K.-H., and Stadelmann, A.: Induced magnetic field effects at planet Mercury, Planet. Space Sci., 52, 1251–1260, doi:10.1016/j.pss.2004.08.005, 2004.

Heyner, D., Wicht, J., Gómez-Pérez, N., Schmitt, D., Auster, H.-U., and Glassmeier, K.-H.: Evidence from Numerical Experiments for a Feedback Dynamo Generating Mercury's Magnetic Field, Science, 334, 1690–1693, doi:10.1126/science.1207290, 2011.

Jameson, A.: Aerodynamic Design via Control Theory, J. Sci. Comput., 3, 233–260, doi:10.1007/BF01061285, 1988.

Jia, X., Slavin, J. A., Gombosi, T. I., Daldorff, L. K. S., Toth, G., and Holst, B.: Global MHD simulations of Mercury's magnetosphere with coupled planetary interior: Induction effect of the planetary conducting core on the global interaction, J. Geophys. Res.-Space, 120, 4763–4775, doi:10.1002/2015JA021143, 2015.

Johnson, C. L., Purucker, M. E., Korth, H., Anderson, B. J., Winslow, R. M., Al Asad, M. M. H., Slavin, J. A., Alexeev, I. I., Phillips, R. J., Zuber, M. T., and Solomon, S. C.: MESSENGER observations of Mercury's magnetic field structure, J. Geophys. Res.-Planet., 117, E00L14, doi:10.1029/2012JE004217, 2012.

Korth, H., J. Anderson, B., Acuña, M. H., Slavin, J. A., Tsyganenko, N. A., Solomon, S. C., and McNutt, R. L.: Determination of the properties of Mercury's magnetic field by the MESSENGER mission, Planet. Space Sci., 52, 733–746, doi:10.1016/j.pss.2003.12.008, 2004.

Langel, R. A.: The main field, in: Geomagnetism, edited by: Jacobs, J. A., 249–512, Academic Press, London, 1987.

Lax, P. and Wendroff, B.: Systems of conservation laws, Commun. Pur. Appl. Math., 13, 217–237, doi:10.1002/cpa.3160130205, 1960.

McFadden, J. P., Carlson, C. W., Larson, D., Ludlam, M., Abiad, R., Elliott, B., Turin, P., Marckwordt, M., and Angelopoulos, V.: The THEMIS ESA Plasma Instrument and In-flight Calibration, Space Sci. Rev., 141, 277–302, doi:10.1007/s11214-008-9440-2, 2008.

McNamara, A., Treuille, A., Popović, Z., and Stam, J.: Fluid Control Using the Adjoint Method, ACM T. Graphic., 23, 449–456, doi:10.1145/1015706.1015744, 2004.

Meader, C. A. and Martins, J. R. R. A.: Derivatives for Time-Spectral Computational Fluid Dynamics Using an Automatic Differentiation Adjoint, The American Institute of Aeronautics and Astronautics Journal, 50, 2809–2819, doi:10.2514/1.J051658, 2012.

Nabert, C., Glassmeier, K.-H., and Plaschke, F.: A new method for solving the MHD equations in the magnetosheath, Ann. Geophys., 31, 419–437, doi:10.5194/angeo-31-419-2013, 2013.

Nabert, C., Othmer, C., and Glassmeier, K.-H.: Solar wind reconstruction from magnetosheath data using an adjoint approach, Ann. Geophys., 33, 1513–1524, doi:10.5194/angeo-33-1513-2015, 2015.

Ogino, T.: A three-dimensional MHD simulation of the interaction of the solar wind with the earth's magnetosphere – The generation of field-aligned currents, J. Geophys. Res., 91, 6791–6806, doi:10.1029/JA091iA06p06791, 1986.

Ogino, T.: Two-Dimensional MHD Code, in: Computer Space Plasma Physics: Simulation Techniques and Software, edited by: Matsumoto, H. and Omura, Y., 161–207, Terra Scientific Publishing Company, Tokyo, 1993.

Ogino, T., Walker, R. J., Ashour-Abdalla, M., and Dawson, J. M.: An MHD simulation of By-dependent magnetospheric convection and field-aligned currents during northward IMF, J. Geophys. Res., 90, 10835–10842, doi:10.1029/JA090iA11p10835, 1985.

Othmer, C.: A continuous adjoint formulation for the computation of topological and surface sensitivities of ducted flows, International Journal for Numerical Methods in Fluids, 58, 861–877, doi:10.1002/fld.1770, 2008.

Othmer, C.: Adjoint methods for car aerodynamics, Journal of Mathematics in Industry, 4, 6, doi:10.1186/2190-5983-4-6, 2014.

Press, W. H., Teukolsky, S. A., Vetterling, W. T., and Flannery, B. P.: Numerical recipes in FORTRAN, Cambridge University Press, Cambridge, 1992.

Solomon, S. C., McNutt, R. L., Gold, R. E., Acuña, M. H., Baker, D. N., Boynton, W. V., Chapman, C. R., Cheng, A. F., Gloeckler, G., Head, III, J. W., Krimigis, S. M., McClintock, W. E., Murchie, S. L., Peale, S. J., Phillips, R. J., Robinson, M. S., Slavin, J. A., Smith, D. E., Strom, R. G., Trombka, J. I., and Zuber, M. T.: The MESSENGER mission to Mercury: scientific objectives and implementation, Planet. Space Sci., 49, 1445–1465, doi:10.1016/S0032-0633(01)00085-X, 2001.

Stadelmann, A., Vogt, J., Glassmeier, K.-H., Kallenrode, M.-B., and Voigt, G.-H.: Cosmic ray and solar energetic particle flux in paleomagnetospheres, Earth Planets Space, 62, 333–345, doi:10.5047/eps.2009.10.002, 2010.

Vogt, J. and Glassmeier, K. H.: On the location of trapped particle populations in quadrupole magnetospheres, J. Geophys. Res., 105, 13063–13072, doi:10.1029/2000JA900006, 2000.

Wengert, R. E.: A simple automatic derivative evaluation program, Commun. ACM, 7, 463–464, 1964.

Winslow, R. M., Anderson, B. J., Johnson, C. L., Slavin, J. A., Korth, H., Purucker, M. E., Baker, D. N., and Solomon, S. C.: Mercury's magnetopause and bow shock from MESSENGER Magnetometer observations, J. Geophys. Res.-Space, 118, 2213–2227, doi:10.1002/jgra.50237, 2013.