

# Geomagnetism package

François Bertin \*

11/2020

## Abstract

In this note we derive all the equations used in the geomagnetism package and in a companion [Jupyter Notebook](#)

---

\*F. Bertin retired from CEA. Email: [francois.bertin7@wanadoo.fr](mailto:francois.bertin7@wanadoo.fr)

# Contents

<b>1</b>	<b>Introduction</b>	<b>2</b>
<b>2</b>	<b>Geomagnetism calculation</b>	<b>2</b>
<b>3</b>	<b>Spherical harmonics normalisation</b>	<b>3</b>
<b>4</b>	<b>Geotetic to geocentric referentials</b>	<b>4</b>
4.1	Ellipse equation . . . . .	4
4.2	Geotetic to geocentric transformation . . . . .	5
4.3	Computational aspect . . . . .	6
<b>5</b>	<b>Base transformation</b>	<b>7</b>
<b>6</b>	<b>Geomagnetic field computation at North and South Pole</b>	<b>8</b>
<b>7</b>	<b>Spherical harmonic coefficients</b>	<b>10</b>
7.1	Gauss coefficients . . . . .	10
7.2	Gauss coefficients secular variation . . . . .	11
<b>8</b>	<b>The B<sub>componants</sub> function</b>	<b>11</b>
<b>9</b>	<b>Benchmark</b>	<b>13</b>
<b>10</b>	<b>Examples</b>	<b>14</b>
10.1	2D plots: Example 1 . . . . .	14
10.2	2D plots: Example 2 . . . . .	16
	<b>References</b>	<b>18</b>

# 1 Introduction

The package `geomagnetim` intends to serve pedagogical purposes rather than to replace the well established `FORTRAN` or `C` programs developed by the National Oceanic and Atmospheric Administration<sup>1</sup>. In contrast with these programs, which favor compactness and minimize time execution, the geomagnetic package, tentatively, focuses on lisibility. You can download `geomagnetis` using:

**pip install geomagnetism**

Use case examples are given in a companion [Jupyter Notebook](#).

## 2 Geomagnetism calculation

As the terrestrial magnetic field obeys both  $\nabla \mathbf{B} = 0$  and  $\nabla \times \mathbf{B} = 0$ , it can be shown that the magnetic field can be expressed as the gradient of a scalar potential  $V$  which satisfies the Laplace equation:

$$\Delta V = 0. \quad (2.1)$$

For a spherical geometry the geomagnetic potential is given by the following spherical harmonic expansion (SH) [1, 2, 4]:

$$V(r, \theta, \phi, t) = a \sum_{n=1}^N \left(\frac{a}{r}\right)^{n+1} \sum_{m=0}^n [g_n^m(t) \cos(m\phi) + h_n^m(t) \sin(m\phi)] P_{(s),n}^m(\cos(\theta)) \quad (2.2)$$

where  $P_{(s),n}^m(x)$  are the Schmidt quasi-normalized associated Legendre polynomials (for more details see section 3):

$$P_{(s),n}^m(x) = \begin{cases} \frac{1}{2^n n!} \frac{d^n}{dx^n} (x^2 - 1)^n & : m = 0 \\ \sqrt{\frac{2(n-m)!}{(n+m)!}} (1-x^2)^m \frac{1}{2^n n!} \frac{d^{n+m}}{dx^{n+m}} (x^2 - 1)^n & : m > 0 \end{cases} \quad (2.3)$$

where  $g_n^m$  and  $h_n^m$  are the Gauss's coefficients. Note that the sum over  $n$  begins with the the value  $n = 1$  as the index  $n = 0$  would correspond to a monopole. The dipole, quadrupole, octupole,... contribution correspond to  $n = 1, 2, 3, \dots$ . These coefficients vary with time and are tabulated by the [National Oceanic and Atmospheric Administration](#). The coefficient  $a$  is the mean radius of the earth (6371.2 km);  $r$ , the radial distance from the center of the Earth;  $\theta$ , the geocentric colatitude;  $\phi$ , the east longitude measured from Greenwich. We note that the Condon-Shortley phase correction  $(-1)^m$  is omitted in the definition of the associated Legendre polynomial and the polynomes are normalized using Schmidt quasi-normalization [5]. The relation  $\mathbf{B} = -\nabla V$  leads to:

$$\begin{aligned} X_c &\equiv \\ \mathbf{B}_x &= -B_\theta = \frac{1}{r} \frac{\partial V}{\partial \theta} = \\ &\sum_{n=1}^N \left(\frac{a}{r}\right)^{n+2} \sum_{m=0}^n [g_n^m \cos(m\phi) + h_n^m \sin(m\phi)] \frac{dP_{(s),n}^m(\cos \theta)}{d\theta}, \\ Y_c &\equiv \\ \mathbf{B}_y &= B_\phi = \frac{-1}{r \sin \theta} \frac{\partial V}{\partial \phi} = \\ &\sum_{n=1}^N \left(\frac{a}{r}\right)^{n+2} \sum_{m=0}^n m [g_n^m \sin(m\phi) - h_n^m \cos(m\phi)] \frac{P_{(s),n}^m(\cos \theta)}{\sin \theta}, \\ Z_c &\equiv \\ \mathbf{B}_z &= -B_r = \frac{\partial V}{\partial r} = \\ &\sum_{n=1}^N (n+1) \left(\frac{a}{r}\right)^{n+2} \sum_{m=0}^n [g_n^m \cos(m\phi) + h_n^m \sin(m\phi)] P_{(s),n}^m(\cos \theta), \end{aligned} \quad (2.4)$$

where  $\mathbf{B}_x$ ,  $\mathbf{B}_y$ ,  $\mathbf{B}_z$  are the field components respectively in the northward, eastward and downward directions. Theses components are expressed in the geocentric referential as recall by the index  $c$ . The parameter  $N$  stands for the order of the SH decomposition.

---

<sup>1</sup>In addition, a [Python code](#) is available on PyPI. Two inline calculators have been developed by [NOAA](#) and by the [International Geomagnetic Reference Field](#).

### 3 Spherical harmonics normalisation

In the field of geomagnetism the Schmidt quasi-normalized Legendre polynomials  $P_{(s),n}^m$  are widely used<sup>2</sup>. They are proportional to the Legendre polynomial:

$$P_{(s),n}^m = N_n^m P_n^m \quad (3.2)$$

where the associated Legendre polynomials<sup>3</sup> are defined as<sup>4</sup> [6]:

$$P_n^m(x) = \frac{(-1)^m}{2^n n!} \sqrt{(1-x^2)^m} \frac{d^{n+m}}{dx^{n+m}} (x^2-1)^n. \quad (3.3)$$

and where the normalization coefficients  $N_n^m$  are equal to [5]:

$$N_{n,m} = \begin{cases} (-1)^m \sqrt{\frac{(2-\delta_m^0)(n-m)!}{(n+m)!}} & : n - |m| \geq 0 \\ 0 & : n - |m| < 0 \end{cases} \quad (3.4)$$

The Schmidt quasi-normalized polynomials obey, for  $\forall n, \forall N, \forall m, \forall M$ , the identities [5]:

$$\begin{aligned} \frac{1}{4\pi} \int_0^{2\pi} \int_0^\pi C_n^m(\theta, \phi) C_N^M(\theta, \phi) \sin \theta d\theta d\phi &= \frac{1}{2n+1} \delta_n^N \delta_m^M \\ \frac{1}{4\pi} \int_0^{2\pi} \int_0^\pi S_n^m(\theta, \phi) S_N^M(\theta, \phi) \sin \theta d\theta d\phi &= \frac{1}{2n+1} \delta_n^N \delta_m^M \\ \frac{1}{4\pi} \int_0^{2\pi} \int_0^\pi C_n^m(\theta, \phi) S_N^M(\theta, \phi) \sin \theta d\theta d\phi &= 0 \end{aligned} \quad (3.5)$$

where the following notations are used:

$$\begin{aligned} C_n^m(\theta, \phi) &\equiv P_{(s),n}^m \cos \theta \cos m\theta & : m = 0, 1, 2 \dots n \\ S_n^m(\theta, \phi) &\equiv P_{(s),n}^m \cos \theta \sin m\theta & : m = 1, 2 \dots n \end{aligned} \quad (3.6)$$

For computational efficiency we define the matrice  $\underline{\underline{\mathbf{P}}}$  and  $\underline{\underline{\mathbf{P}}}_{(s)}$ :

$$\underline{\underline{\mathbf{P}}} = \begin{bmatrix} P_0^0 & P_1^0 & \dots & P_N^0 \\ 0 & P_1^1 & \dots & P_N^1 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & P_N^M \end{bmatrix}, \quad (3.7)$$

$$\underline{\underline{\mathbf{P}}}_{(s)} = \begin{bmatrix} P_{(s),0}^0 & P_{(s),1}^0 & \dots & P_{(s),N}^0 \\ 0 & P_{(s),1}^1 & \dots & P_{(s),N}^1 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & P_{(s),N}^M \end{bmatrix} \quad (3.8)$$

and we compute  $\underline{\underline{\mathbf{P}}}_{(s)}$  as the matrix component-wise product:

$$\underline{\underline{\mathbf{P}}}_{(s)} = \underline{\underline{\mathbf{P}}} \odot \underline{\underline{\mathbf{N}}}, \quad (3.9)$$

In the `geomagnetism` package:

- (a) The function `Norm_Schmidt(m,n)` computes the normalisation matrix (3.8) using coefficients (3.4).
- (b) The function `Norm_Stacey(m,n)` computes the normalisation matrix using coefficients (3.1).

<sup>2</sup>Note that other authors in geophysics use different normalization factors For example, Stacey [4] uses:

$$N_n^m = \begin{cases} (-1)^m \sqrt{(2-\delta_m^0)(2m+1)} \frac{(n-m)!}{(n+m)!} & : |m| \leq n \\ 0 & : |m| > n \end{cases} \quad (3.1)$$

<sup>3</sup>In the `geomagnetism` package, the associated Legendre polynomials be computed by the scipy function `lpmn(M,N,x)`

<sup>4</sup>To stick with the `scipy` package conventions, these polynomials are defined using the Condon-Shortley phase correction  $(-1)^m$ .

```

# Example of computation of normalization matrix
geo.Norm_Stacey(3,4)
>> array([[ 1.,          1.,          1.,          1.,          1.],
          [ 0., -1.73205081, -1., -0.70710678, -0.54772256],
          [ 0.,          0.,  0.64549722,  0.28867513,  0.16666667],
          [ 0.,          0.,          0., -0.13944334, -0.05270463]])

geo.Norm_Schmidt(3,4)
>> array([[ 1.,          1.,          1.,          1.,          1.],
          [ 0., -1., -0.57735027, -0.40824829, -0.31622777],
          [ 0.,          0.,  0.28867513,  0.12909944,  0.0745356 ],
          [ 0.,          0.,          0., -0.05270463, -0.01992048]])

```

## 4 Geotetic to geocentric referentials

The computation of the geomagnetic field is done in a geocentric coordinate system. So if we provide the geotetic coordinates we have to convert them into geocentric ones. In the following we deduce the transformation relation used in the function `geotetic_to_geocentric`.

### 4.1 Ellipse equation

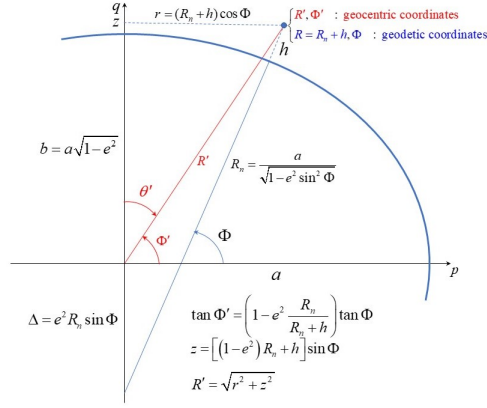


Figure 1: Ellipse notations convention.

Using the notation of the figure 1, the equation of the ellipse reads:

$$\frac{q^2}{b^2} + \frac{p^2}{a^2} = 1. \quad (4.1)$$

The geotetic latitude  $\Phi$  can be expressed through the derivative:

$$\cot \Phi = -\frac{dq}{dp} = \frac{b^2}{a^2} \frac{p}{q} \quad (4.2)$$

From equation (4.2) we can express  $q$  as:

$$q = p \frac{b^2}{a^2} \tan \Phi \quad (4.3)$$

Using equations (4.1) and (4.3) we can express the ellipse coordinates  $p$  and  $q$  as a function of the geotetic latitude  $\Phi$  as:

$$p = \frac{a \cos \Phi}{\sqrt{1 - e^2 \sin^2 \Phi}} \quad (4.4a)$$

$$q = \frac{a(1 - e^2) \sin \Phi}{\sqrt{1 - e^2 \sin^2 \Phi}} \quad (4.4b)$$

where  $e \equiv \sqrt{1 - \frac{b^2}{a^2}}$  is the eccentricity. The prime vertical curvature radius  $R_n$  (see figure 1) can be deduced from  $p$  as:

$$R_n = \frac{p}{\cos \Phi} = \frac{a}{\sqrt{1 - e^2 \sin^2 \Phi}} \quad (4.5a)$$

$$R_n = \frac{a^2}{\sqrt{a^2 - (a^2 - b^2)\sin^2\Phi}} \quad (4.5b)$$

Using the prime vertical curvature we can re-express  $p$  and  $q$  as:

$$p = R_n \cos \Phi \quad (4.6a)$$

$$q = (1 - e^2)R_n \sin \Phi \quad (4.6b)$$

## 4.2 Geotetic to geocentric transformation

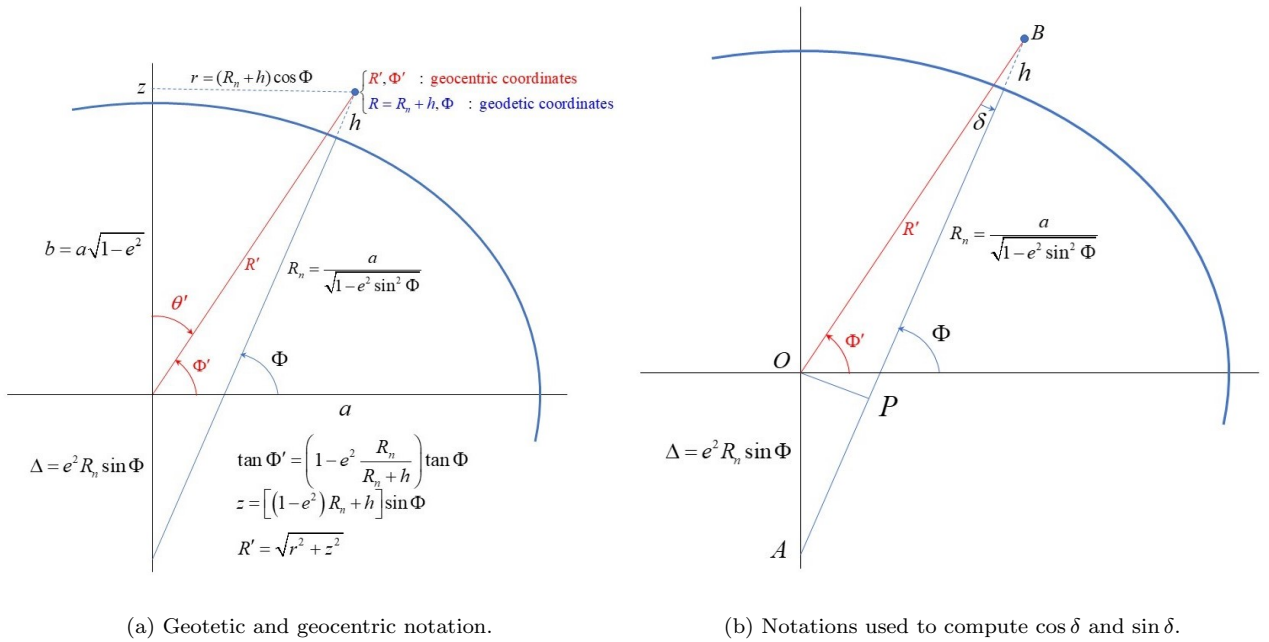


Figure 2: Notation conventions.

In this section we derive the relation between the geotetic colatitude  $\Phi$  and the geocentric colatitude  $\Phi'$ . Using the conventions of the figure 2a, we have:

$$\frac{\tan \Phi'}{\tan \Phi} = \frac{z}{z + \Delta} = \frac{(1 - e^2)R_n + h}{(1 - e^2)R_n + h + e^2 R_n} = 1 - e^2 \frac{R_n}{R_n + h} \quad (4.7)$$

The flattening  $f$  is defined as follow:

$$f = \frac{a-b}{a} \quad (4.8)$$

Usually, the geodetic reference ellipsoid is specified by its reciprocal flattening  $f^{-1}$ . The reciprocal flattening is related to the eccentricity  $e$  by:

$$\begin{aligned} e &= \sqrt{1 - \frac{b^2}{a^2}} = \sqrt{f(2-f)} \\ 1 - e^2 &= \frac{b^2}{a^2} \end{aligned} \quad (4.9)$$

We can express  $z$  and  $r$  as:

$$\begin{cases} z = (R_n + h) \sin \Phi - \Delta = [(1 - e^2)R_n + h] \sin \Phi \\ r = (R_n + h) \cos \Phi \end{cases} \quad (4.10)$$

where  $h$  is the height above the reference ellipsoid. Thanks to equation (4.10) we can obtain the geocentric radius as:

$$R' = \sqrt{z^2 + r^2} \quad (4.11)$$

Using the equations (4.5b) and (4.10) we express the geocentric as a function both the semi major and the semi minor axis. We obtain [3]:

$$R'^2 = \frac{h^2 + 2h\sqrt{a^2 - (a^2 - b^2)\sin^2\Phi} + [a^4 - (a^4 - b^4)\sin^2\Phi]}{a^2 - (a^2 - b^2)\sin^2\Phi} \quad (4.12)$$

Combining the equations (4.11) and (4.10) we can express the geocentric colatitude as:

$$\cos\theta = \frac{z}{\sqrt{r^2 + z^2}} = \frac{\sin\Phi}{\sqrt{\left[\frac{R_n+h}{(1-e^2)R_n+h}\right]^2 \cos^2\Phi + \sin^2\Phi}} \quad (4.13)$$

The relation (4.13) can be rewritted using the semi major and minor axis. Using equations (4.5b) and (4.13) we obtain [3]:

$$\cos\theta = \frac{\sin\Phi}{\sqrt{c \cos^2\Phi + \sin^2\Phi}} \quad \text{where} \quad c = \left[ \frac{a^2 + h\sqrt{a^2 - (a^2 - b^2)\sin^2\Phi}}{b^2 + h\sqrt{a^2 - (a^2 - b^2)\sin^2\Phi}} \right]^2 \quad (4.14)$$

In the triangle AOB of the figure 2b the length of AB leads to the equality:

$$\Delta \sin\Phi + R' \cos\delta = R_n + h, \quad (4.15)$$

and, after rearranging:

$$\cos\delta = \frac{1}{R'} \left[ h + R_n(1 - e^2(\sin\Phi)^2) \right]. \quad (4.16)$$

Using the relation (4.5) equation (4.16) reads:

$$\cos\delta = \frac{1}{R'} \left[ h + \frac{a^2}{R_n} \right]. \quad (4.17)$$

The length of the common side OP of the two rectangles triangle AOP and BOP of the figure 2b leads to the equality:

$$R' \sin\delta = \Delta \cos\Phi.$$

So:

$$\sin\delta = \frac{R_n}{R'} e^2 \cos\Phi \sin\Phi \quad (4.18)$$

### 4.3 Computational aspect

The function `geodetic_to_geocentric(ellipsoid, co_latitude, height)` computes the geocentric colatitude and radius using respectively the equations (4.7) and (4.11). The angle  $\delta = \theta' - \theta$  between the geocentric and the geotetic colatitude is also computed. The figure 3 shows the variation of  $\delta$  versus the geotetic colatitude  $\theta$  and the height  $h$ .

```
# Exemple of computation of r_geocentric, co_latitude_geocentric, delta
import geomagnetism as geo
ellipsoid = geo.geomagnetism.WGS84 # tuple (mean earth radius in meter, inverse flattening)
r_geocentric, co_latitude_geocentric, delta = geo.geodetic_to_geocentric(ellipsoid , 170,
100_000)
>> (6457402.34844737, 2.965925285681976, -0.0011344427083841424)
ellipsoid = geo.geomagnetism.GRS80
r_geocentric, co_latitude_geocentric, delta = geo.geodetic_to_geocentric(ellipsoid , 170,
100_000)
>> (6457402.348345751, 2.9659252856763882, -0.001134442713972117)
```

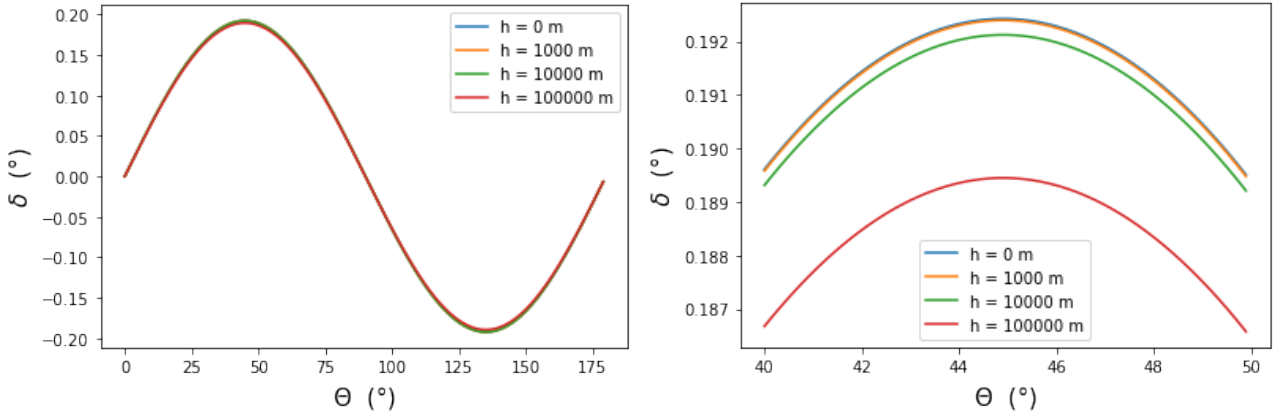
```

# Exemple of computation of r_geocentric, cos(co_latitude_geocentric), sin(
# cos(delta), sin(delta)
import geomagnetism as geo
ellipsoid = geo.geomagnetism.WGS84 # tuple (earth major axis in meter, earth minor axis in
                                     meter)

ellipsoid = geo.geomagnetism.WGS84_
r, ct, st, cd, sd = geo.geodetic_to_geocentric_IGRF13(ellipsoid, 170, 100_000)
>> (6457402.34844737 -0.9846101254413312 0.17476527366272146 0.9999993565199398 -0.
    0011344424650535443)

ellipsoid = geo.geomagnetism.GRS80_
r, ct, st, cd, sd = geo.geodetic_to_geocentric_IGRF13(ellipsoid, 170, 100_000)
>> (6457402.348345758 -0.9846101254403548 0.17476527366822295 0.9999993565199334 -0.
    0011344424706410008)

```



(a) Variation of  $\delta$  for  $\theta$  varying from  $0^\circ$  to  $180^\circ$ .

(b) Zoom of the variation of  $\delta$  near a maximum.

Figure 3: Variation of  $\delta$  versus the geotetic colatitude  $\theta$  and the height  $h$  above the Earth.

Alternatively, the function `geodetic_to_geocentric_IGRF13(ellipsoid, co_latitude, height)` is a translation of the [FORTRAN routine](#) where the authors compute the geocentric radius as well as  $\cos \delta$  and  $\sin \delta$  using respectively the equations (4.12), (4.17), (4.18).

## 5 Base transformation

As geomagnetic field components are computed in a geocentric referential (see section 2), we have to express  $B_x, B_y, B_z$  in a geotetic referential.

Passing from geocentric to geotetic referential the magnetic field undergoes the following transformation:

$$\begin{pmatrix} B_x \\ B_y \\ B_z \end{pmatrix}_{\text{geotetic}} = \begin{bmatrix} \cos \delta & 0 & -\sin \delta \\ 0 & 1 & 0 \\ \sin \delta & 0 & \cos \delta \end{bmatrix} \begin{pmatrix} B_x \\ B_y \\ B_z \end{pmatrix}_{\text{geocentric}} \quad (5.1)$$

with  $\delta = \theta' - \theta = \Phi - \Phi'$  (Figure 4a) where  $\cos \delta$  and  $\sin \delta$  can be obtained thanks to the equations (4.17) and (4.18). Using Peddie notation [3] we have:

$$\begin{pmatrix} X \\ Y \\ Z \end{pmatrix} = \begin{bmatrix} \cos \delta & 0 & -\sin \delta \\ 0 & 1 & 0 \\ \sin \delta & 0 & \cos \delta \end{bmatrix} \begin{pmatrix} X_c \\ Y_c \\ Z_c \end{pmatrix} \quad (5.2)$$

Using the notations of the Figure 4b, the geomagnetic horizontal intensity  $H$ , total intensity  $F$ , declination  $D$  and inclination  $I$  can be obtained from:

$$\begin{cases} H = \sqrt{X^2 + Y^2} \\ F = \sqrt{H^2 + Z^2} \\ D = \text{atan2}(Y, X) \\ I = \text{atan2}(Z, H) \end{cases} \quad (5.3)$$



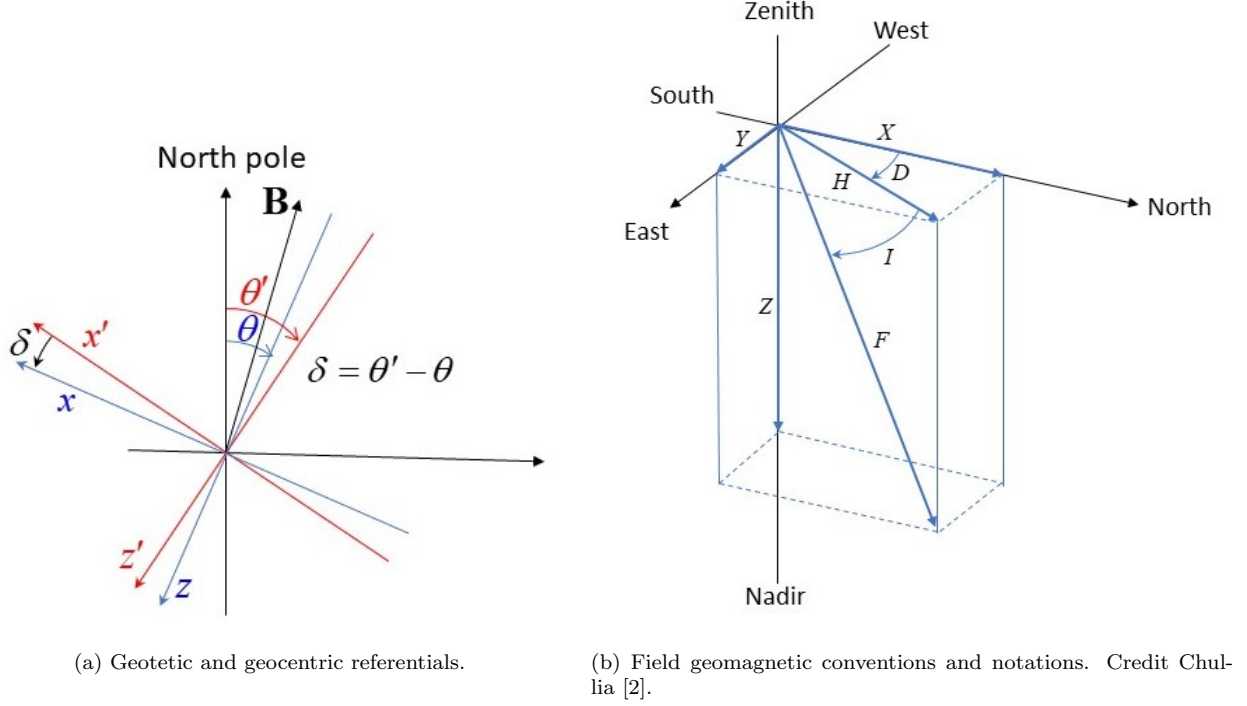


Figure 4: Notation conventions.

## 6 Geomagnetic field computation at North and South Pole

When the colatitude  $\theta$  tends towards 0 (North Pole) or towards  $\pi$ , the equations (2.3) are numerically instable. To deal with that problem we have to evaluate  $\frac{dP_{(s),n}^m(\cos \theta)}{d\theta}$ ,  $\frac{P_{(s),n}^m(\cos \theta)}{\sin \theta}$ ,  $P_{(s),n}^m(\cos \theta)$  for  $\theta = 0$  and for  $\theta = \pi$ .

**Identity 1.**  $P_{(s),n}^m(1) = \delta_m^0$ .

**Identity 2.**  $P_{(s),n}^m(-1) = (-1)^n \delta_m^0$ .

*Proof.* The associate Legendre polynomials can be defined as:

$$P_n^m(x) = (-1)^m (1-x^2)^{\frac{m}{2}} \frac{d^m}{dx^m} P_n(x) \quad (6.1)$$

Using the change of variable  $x = \cos \theta$  equation (6.1) reads:

$$P_n^m(\cos \theta) = (-1)^m (\sin \theta)^m \frac{d^m}{d(\cos \theta)^m} P_n(\cos \theta) \quad (6.2)$$

From the equation (6.2) we deduce that for  $\theta = 0$  or for  $\theta = \pi$ ,  $P_n^m(\cos \theta)$  is not null iff  $m = 0$ . As  $P_n(1) = 1$  [8, p. 752] we deduce the Identity 1. As  $P_n(-1) = (-1)^n$  [8, p. 752] we deduce the Identity 2.  $\square$

**Identity 3.**  $\lim_{\theta \rightarrow 0} \frac{P_{(s),n}^m(\cos \theta)}{\sin \theta} = \delta_1^m \sqrt{\frac{n(n+1)}{2}}$ .

**Identity 4.**  $\lim_{\theta \rightarrow \pi} \frac{P_{(s),n}^m(\cos \theta)}{\sin \theta} = (-1)^n \delta_1^m \sqrt{\frac{n(n+1)}{2}}$ .

*Proof.* From equation (6.2) we deduce that  $\frac{P_n^m(\cos \theta)}{\sin \theta}$  is not null iff  $m = 1$ . This condition leads to:

$$\lim_{\theta \rightarrow 0} \frac{P_n^1(\cos \theta)}{\sin \theta} = \frac{d}{dx} P_n(x) \Big|_{x=1} \quad (6.3)$$

As the Legendre polynomials  $P_n(x)$  satisfy the differential equation [8, p. 751]:

$$(1-x^2) \frac{d^2}{dx^2} P_n(x) - 2x \frac{d}{dx} P_n(x) + n(n+1) P_n(x) = 0, \quad (6.4)$$

and also satisfy the two identities  $P_n(1) = 1$  and  $P_n(-1) = (-1)^n$  [8, p. 752], we have:

$$\frac{d}{dx} P_n(1) = \frac{n(n+1)}{2} \quad (6.5a)$$

$$\frac{d}{dx} P_n(-1) = (-1)^{n-1} \frac{n(n+1)}{2} \quad (6.5b)$$

Taking into account respectively equation (6.5a) and equation (6.5b) in conjunction with the Schmidt normalisation coefficients (3.4) we obtain the Identity 3 and the Identity 4.  $\square$

**Identity 5.**  $\left. \frac{dP_{(s),n}^m(\cos \theta)}{d\theta} \right|_{\theta=0} = \delta_1^m \sqrt{\frac{n(n+1)}{2}}.$

**Identity 6.**  $\left. \frac{dP_{(s),n}^m(\cos \theta)}{d\theta} \right|_{\theta=\pi} = (-1)^n \delta_1^m \sqrt{\frac{n(n+1)}{2}}.$

*Proof.* The derivative versus  $\theta$  of the equation (6.2) leads to the expression:

$$\frac{d}{d\theta} P_n^m(\cos \theta) = (-1)^m (\sin \theta)^{m-1} \left[ m \cos \theta \frac{d^m}{d(\cos \theta)^m} P_n(\cos \theta) - (\sin \theta)^2 \frac{d^{m+1}}{d(\cos \theta)^{m+1}} P_n(\cos \theta) \right] \quad (6.6)$$

showing that  $\frac{d}{d\theta} P_n^m(\cos \theta)$  is not null iff  $m = 1$ . Taking into account respectively equation (6.5a) equation (6.5b) in conjunction with the Schmidt normalisation coefficients (3.4) we obtain the Identity 5 and the Identity 6.  $\square$

Using the identities 1 to 6 we can derive the expression of the magnetic at the North and the South Pole as follow:

- (a) Putting the identities 1, 3, 5 in equation (2.4) we obtain the following expressions of the magnetic field at the North Pole:

$$\left\{ \begin{array}{l} X_c(0) = \sum_{n=1}^N \left(\frac{a}{r}\right)^{n+2} \sqrt{\frac{n(n+1)}{2}} g_n^1 \cos(\phi) + \sum_{n=1}^N \left(\frac{a}{r}\right)^{n+2} \sqrt{\frac{n(n+1)}{2}} h_n^1 \sin(\phi) \\ Y_c(0) = \sum_{n=1}^N \left(\frac{a}{r}\right)^{n+2} \sqrt{\frac{n(n+1)}{2}} g_n^1 \sin(\phi) - \sum_{n=1}^N \left(\frac{a}{r}\right)^{n+2} \sqrt{\frac{n(n+1)}{2}} h_n^1 \cos(\phi) \\ Z_c(0) = \sum_{n=1}^N (n+1) \left(\frac{a}{r}\right)^{n+2} g_n^0 \end{array} \right. \quad (6.7)$$

- (b) Putting the identities 3, 4, 6 in equation (2.4) we obtain the following expressions of the magnetic field at the South Pole:

$$\left\{ \begin{array}{l} X_c(\pi) = \sum_{n=1}^N \left(-\frac{a}{r}\right)^{n+2} \sqrt{\frac{n(n+1)}{2}} g_n^1 \cos(\phi) + \sum_{n=1}^N \left(-\frac{a}{r}\right)^{n+2} \sqrt{\frac{n(n+1)}{2}} h_n^1 \sin(\phi) \\ Y_c(\pi) = \sum_{n=1}^N \left(-\frac{a}{r}\right)^{n+2} \sqrt{\frac{n(n+1)}{2}} g_n^1 \sin(\phi) - \sum_{n=1}^N \left(-\frac{a}{r}\right)^{n+2} \sqrt{\frac{n(n+1)}{2}} h_n^1 \cos(\phi) \\ Z_c(\pi) = - \sum_{n=1}^N (n+1) \left(-\frac{a}{r}\right)^{n+2} g_n^0 \end{array} \right. \quad (6.8)$$

In the **geomagnetism** package the constant EPS monitors the choice of the equation used to compute the electromagnetic field according to Algorithm 1. The default value of EPS is set to  $\text{EPS} = 10^{-5}$  rad. At  $\phi = \text{EPS}$  the electromagnetic field components undergo a discontinuity as shown in figure 5. The value of the constant EPS can be reaffected using:

`geo.geomagnetism.EPS=NewEps.`

Nevertheless as shown by Figure 5b for  $\text{EPS} \lesssim 8 \times 10^{-6}$  rad numerical computations of equations (2.4) are noisy.

---

**Algorithm 1** geomagnetic field computation

---

```
1: EPS  $\leftarrow 10^{-5}$ 
2: if  $\phi - \text{EPS} \in [\pi, \pi - \text{EPS}]$  then
3:   use equations (6.8).
4: else if  $\phi \in [0, \text{EPS}]$  then
5:   use equations (6.7).
6: else
7:   use equations (2.4).
8: end if
```

---

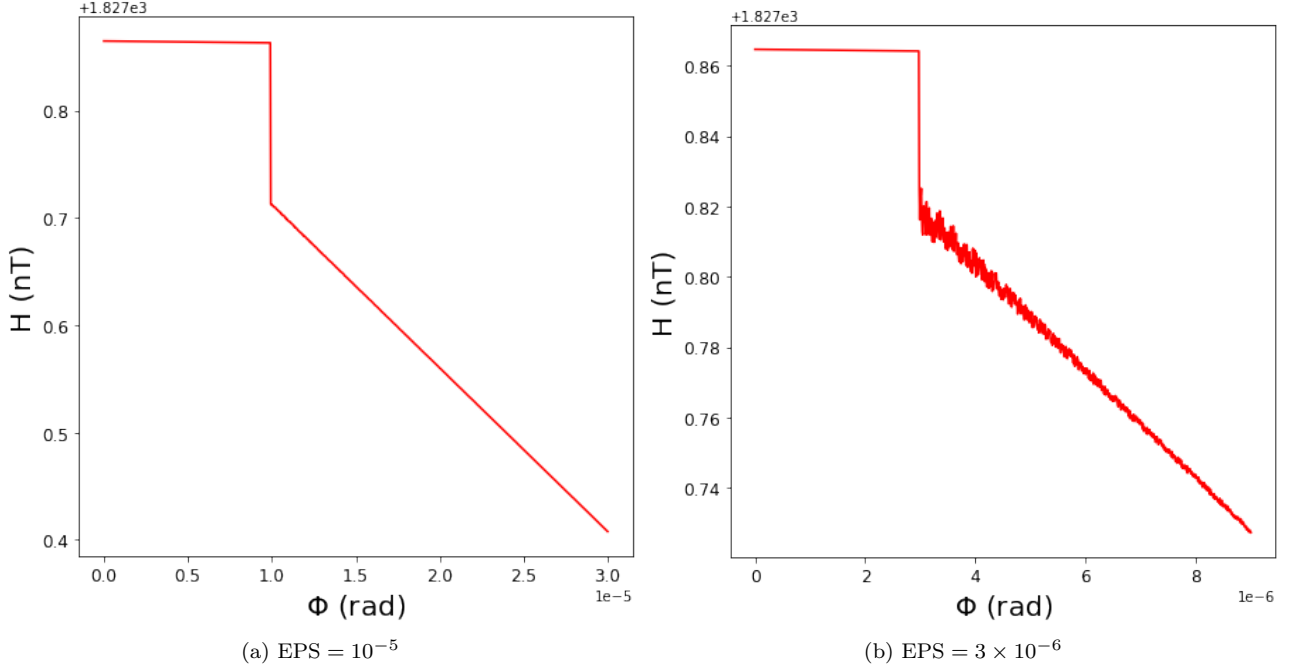


Figure 5: Computational discontinuity of the component  $H$  at  $\phi = \text{EPS}$ .

## 7 Spherical harmonic coefficients

### 7.1 Gauss coefficients

To compute the geomagnetic fields, through the equation (2.4), we must provide: the values of the Gauss coefficients  $h_n^m$  and  $g_n^m$ , the order  $N$  of the SH development. As the Gauss coefficients vary with time we must also provide the values  $\dot{h}_n^m$  and  $\dot{g}_n^m$ . All these values are stored in the following variables:

`dic_dic_h` contains  $h_n^m$  coefficients stored in a dict of dict as `{year: {(m,n):h,...},...}`  
`dic_dic_g` contains  $g_n^m$  coefficients stored in a dict of dict as `{year: {(m,n):g,...},...}`  
`dic_dic_SV_h` contains  $\dot{h}_n^m$  coefficients stored in a dict of dict as `{year: {(m,n):SV_h,...},...}`  
`dic_dic_SV_g` contains  $\dot{g}_n^m$  coefficients stored in a dict of dict as `{year: {(m,n):SV_g,...},...}`  
`dic_N` dict containing the order  $N$  of the SH decomposition as `dic_N[year]=N`  
`Years` contains the list of tabulated years

The function `read_IGRF13.COF` reads the `IGRF13.COF` coefficients.

```
file = 'IGRF13.COF' # downloaded from https://www.ngdc.noaa.gov/IAGA/vmod/coeffs/igrf13coeffs.txt
dic_dic_h, dic_dic_g, dic_dic_SV_h, dic_dic_SV_g, dic_N, Years = geo.read_IGRF13_COF(file)
```

The function `read_WMM` reads the `WMM.2020.COF` or `theWMM.2015.COF` coefficients.

```
file = 'WMM_2020.COF' # downloaded from https://www.ngdc.noaa.gov/geomag/WMM/wmm_ddownload.shtml
dic_dic_h, dic_dic_g, dic_dic_SV_h, dic_dic_SV_g, dic_N, Years = geo.read_WMM(file)
```

```
file = 'WMM_2015.COF' # downloaded from https://www.ngdc.noaa.gov/geomag/WMM/wmm_download.shtml
dic_dic_h, dic_dic_g, dic_dic_SV_h, dic_dic_SV_g, dic_N, Years = geo.read_WMM(file)
```

## 7.2 Gauss coefficients secular variation

The Gauss coefficients  $g_n^m$  and  $h_n^m$  vary with time [7]. Their secular variation  $\dot{g}_n^m$  and  $\dot{h}_n^m$  expressed in nT/year are tabulated every year and are considered to be constant during that current year. So, at time  $t$ , the Gauss coefficients can be expressed as:

$$\begin{cases} g_n^m(t) = g_n^m(t_0) + \dot{g}_n^m(t_0)(t - t_0) \\ h_n^m(t) = h_n^m(t_0) + \dot{h}_n^m(t_0)(t - t_0) \end{cases} \quad (7.1)$$

where  $t_0$  stands for the time on the 1 January at 00:00 PM of the current year. Using  $\dot{g}_n^m$  and  $\dot{h}_n^m$  it is straightforward to compute the secular variation of the geomagnetic field components. We have [2]:

$$\begin{aligned} \dot{X}_c &\equiv \\ \dot{\mathbf{B}}_x &= -\dot{B}_\theta = \frac{1}{r} \frac{\partial \dot{V}}{\partial \theta} = \\ &\sum_{n=1}^N \left(\frac{a}{r}\right)^{n+2} \sum_{m=0}^n \left[ \dot{g}_n^m \cos(m\phi) + \dot{h}_n^m \sin(m\phi) \right] \frac{dP_{(s),n}^m(\cos \theta)}{d\theta}, \\ \dot{Y}_c &\equiv \\ \dot{\mathbf{B}}_y &= \dot{B}_\phi = \frac{-1}{r \sin \theta} \frac{\partial \dot{V}}{\partial \phi} = \\ &\sum_{n=1}^N \left(\frac{a}{r}\right)^{n+2} \sum_{m=0}^n m \left[ \dot{g}_n^m \sin(m\phi) - \dot{h}_n^m \cos(m\phi) \right] \frac{P_{(s),n}^m(\cos \theta)}{\sin \theta}, \\ \dot{Z}_c &\equiv \\ \dot{\mathbf{B}}_z &= -\dot{B}_r = \frac{\partial \dot{V}}{\partial r} = \\ &\sum_{n=1}^N (n+1) \left(\frac{a}{r}\right)^{n+2} \sum_{m=0}^n \left[ \dot{g}_n^m \cos(m\phi) + \dot{h}_n^m \sin(m\phi) \right] P_{(s),n}^m(\cos \theta), \end{aligned} \quad (7.2)$$

The time derivative of the equation (5.2) can be expressed as:

$$\begin{pmatrix} \dot{X} \\ \dot{Y} \\ \dot{Z} \end{pmatrix} = \begin{bmatrix} \cos \delta & 0 & -\sin \delta \\ 0 & 1 & 0 \\ \sin \delta & 0 & \cos \delta \end{bmatrix} \begin{pmatrix} \dot{X}_c \\ \dot{Y}_c \\ \dot{Z}_c \end{pmatrix} \quad (7.3)$$

Finally, the time derivative of equation (5.3) reads <sup>5</sup>:

$$\begin{cases} \dot{H} = \frac{X\dot{X} + Y\dot{Y}}{H} \\ \dot{F} = \frac{X\dot{X} + Y\dot{Y} + Z\dot{Z}}{F} \\ \dot{I} = \frac{H\dot{Z} - Z\dot{H}}{F^2} \\ \dot{D} = \frac{X\dot{Y} - Y\dot{X}}{H^2} \end{cases} \quad (7.4)$$

## 8 The B\_components function

The function `B_components` is defined as:

```
def B_components(
    phi_,
    theta_,
    altitude,
```

<sup>5</sup>We use  $\frac{\partial}{\partial x} \text{atan2}(y, x) = -\frac{y}{x^2 + y^2}$  and  $\frac{\partial}{\partial y} \text{atan2}(y, x) = \frac{x}{x^2 + y^2}$

```

Date,
referential="geotetic",
file_gauss_coeff="IGRF13.COF",
ELLIPSOID=WGS84,
SV=False,
):

```

The function `B_components` takes the arguments:

`phi_`: Longitude  $\Phi$  in deg  $\Phi \in [0^\circ, 360^\circ[$  or West East longitude  $\Phi \in [-180^\circ, 180^\circ[$ ;

`theta_`: Colatitude  $\theta$  in deg  $\theta \in [0^\circ, 180^\circ]$ ;

`altitude`: Elevation in metres;

`Date`: time used to compute the magnetic field. `Date` is a dictionary

```

Date["mode"]
    if Date["mode"]=="ymd" time is expressed as yyyy/mm/dd and is contained in Date["year"]
    if Date["mode"]=="dec" we have:
        Date["year"] = year
        Date["month"] = month month = 1,2,...,12
        Date["day"] = day of the month
        Date["hour"] = hour of day
        Date["minute"] = minute of the hour
        Date["second"] = second of the minute

```

`referential`:

the colatitude is expressed in a geotetic referential if `referential = geotetic`;

the colatitude is expressed in a geotetic referential if `referential = geotetic`.

`file_gauss_coeff`: name of the file containing the Gauss coefficients  $h$  and  $g$ .

`file = "IGRF13.COF"` covers the years from 1900 to 2015;

`file = "WMM_2015.COF"` covers the year 2015;

`file = "WMM_2020.COF"` (default value) covers the year 2020 with extrapolation allowed up to year 2025.

`ELLIPSOID`: tuple  $(a, f^{-1})$  where  $a$  is the Earth semi major axis in metres, and  $f^{-1}$  the Earth reciprocal flattening. Conventional values are:

`WGS80 = (6378137, 298.257222100882711)`

`WGS84 = (6378137, 298.257223563)` (default value)

`SV`: if `SV==True` the computation of the field secular variation is done.

The function `B_components` returns the dictionary `result`:

`result["D"]= $D$` : Declination in deg;

`result["F"]= $F$` : Total field intensity in nT;

`result["H"]= $H$` : horizontal field intensity in nT;

`result["I"]= $I$` : Inclination in deg;

`result["X"]= $X$` : North component in nT in the geocentric coordinate;

`result["Y"]= $Y$` : East component in nT in both geocentric and geotetic coordinate;

`result["Z"]= $Z$` : Down component in nT in the geocentric coordinate;

`result["Fd"]= $\dot{F}$` : Total field intensity secular variation in nT/year if `SV==True` and `None` otherwise;

`result["Hd"]= $\dot{H}$` : Horizontal field intensity secular variation in nT/year if `SV==True` and `None` otherwise;

`result["Xd"]= $\dot{X}$` : North component field intensity secular variation in nT/year if `SV==True` and `None` otherwise;

`result["Yd"]= $\dot{Y}$` : East component field intensity secular variation in nT/year if `SV==True` and `None` otherwise;

`result["Zd"]= $\dot{Z}$` : Down component field intensity secular variation in nT/year if `SV==True` and `None` otherwise;

`result["Id"]= $\dot{I}$` : Inclination secular variation in deg/year if `SV==True` and `None` otherwise;

`result["Dd"]= $\dot{D}$` : Declination secular variation in deg/year if `SV==True` and `None` otherwise.

For example:

```
height= 100_000.0
colatitude= 170.0
longitude= 240.0
Date={"mode": "dec", "year": 2022.5}

result = geo.B_components(longitude,colatitude,height,Date,referential="geodetic",
                           file="WMM_2020.COF", SV=True)

>> {'X': 5814.9658886214675,
>> 'Y': 14802.966383932766,
>> 'Z': -49755.31199391833,
>> 'F': 52235.358844960836,
>> 'H': 15904.139148337288,
>> 'I': -72.27367389486136,
>> 'D': 68.55389056498416,
>> 'Xd': 28.038196182663352,
>> 'Yd': 1.3970624624335226,
>> 'Zd': 85.63095330312873,
>> 'Hd': 11.551824423488469,
>> 'Fd': -78.04814717528828,
>> 'Id': 0.040667257177207254,
>> 'Dd': -0.09217565861159664}
```

## 9 Benchmark

Using the latitude, longitude, time and height above the ellipsoid given in the Table 1, the Table 2 shows the geomagnetic field element obtained by both the geomagnetism package and those extracted from the Table 3b High-precision numerical example from [2, p. 14-15].

Time	2022.5	yr
Height-above-Ellipsoid	100	km
Latitude	-80	deg
Longitude	240	deg

Table 1: parameters values used for the benchmark.

notation	geomagnetism	WMM test value	relative error
D	69.125	69.13	-0.006
Dd	-0.094	-0.09	4.342
F	54912.078	54912.1	0.0
Fd	-83.356	-83.4	-0.052
H	16884.992	16885.0	0.0
Hd	12.551	12.6	-0.388
I	-72.091	-72.09	0.002
Id	0.041	0.04	4.462
X	6016.523	6016.5	0.0
Xd	30.379	30.4	-0.068
Y	15776.705	15776.7	0.0
Yd	1.847	1.8	2.578
Z	-52251.635	-52251.6	0.0
Zd	91.656	91.7	-0.047

Table 2: comparison of the geomagnetic field element obtained by both the geomagnetism package and those extract from the Table 3b High-precision numerical example from Chullia. [2, p. 14-15]

## 10 Examples

### 10.1 2D plots: Example 1

Using the following code generates the xarrays of:  $X, Y, Z, F, H, \dot{X}, \dot{Y}, \dot{Z}, \dot{F}, \dot{H}$  on the grid defined by the two arrays `colatitudes` and `longitudes`.

```
import numpy as np
import geomagnetism as geo
'''
compute the xarrays: dintensities containing X,Y,Z,F,H;
                    dangles containing I, D;
                    dintensities_sv containing the the time derivative of X,Y,Z,F,H;
                    dangles_sv containing the the time derivative of I, D;
for the colatitudes and longitudes specified in the arrays colatitudes, longitudes
'''

colatitudes = np.linspace(0,180,181)
longitudes = np.linspace(-180,179,360)

dintensities, dangles, dintensities_sv, dangles_sv = geo.grid_geomagnetic(colatitudes,
                                                                           longitudes)
```

Using the following code generates the the plots of Figure 6 and of the Figure 7.

```
'''
Examples of plots of geomagnetic components. Note that, to plot the geomagnetic field, we use
three different map projection: \texttt{spstere} for the South Pole; \texttt{npstere} for the
North Pole; \texttt{mill} for the whole Earth.
Figure~\ref{fig:example2} shows an evolving
dent in Earth's magnetic field over South
America.
'''
component = 'F' # sould be 'X','Y','Z','F','H','I','D','Xd','Yd','Zd','Fd','Hd','Id','Dd'
geo.plot_geomagetism(dintensities,
                    dangles,
                    dintensities_sv,
                    dangles_sv,
                    component,
                    {'proj':'spstere',
                    'boundinglat':-55,
                    'lon_0':270})

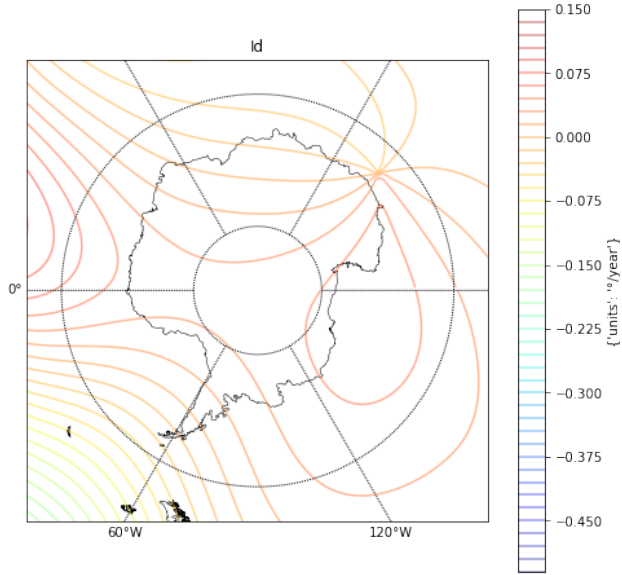
geo.plot_geomagetism(dintensities,
                    dangles,
                    dintensities_sv,
                    dangles_sv,
                    component,
                    {'proj':'npstere',
                    'boundinglat':70,
```

```

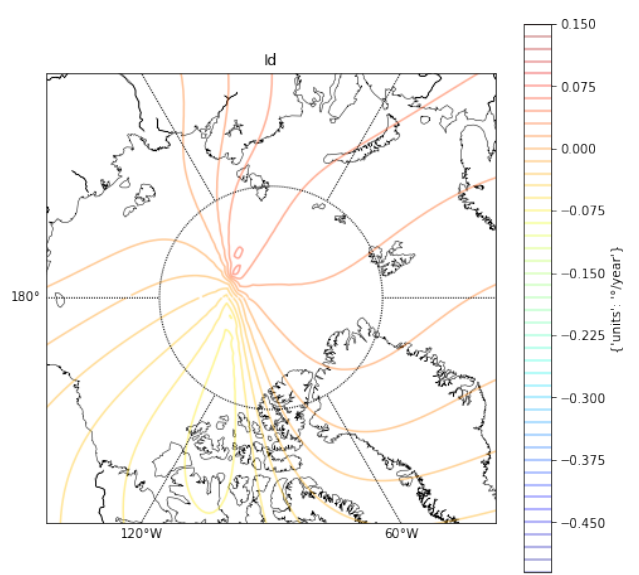
        'lon_0':270} )

geo.plot_geomagnetism(dintensities,
                      dangles,
                      dintensities_sv,
                      dangles_sv,
                      component,
                      {'proj':'mill',
                      'llcrnrlat':-90,
                      'urcrnrlat': 90,
                      'llcrnrlon':0,
                      'urcrnrlon':360} )

```

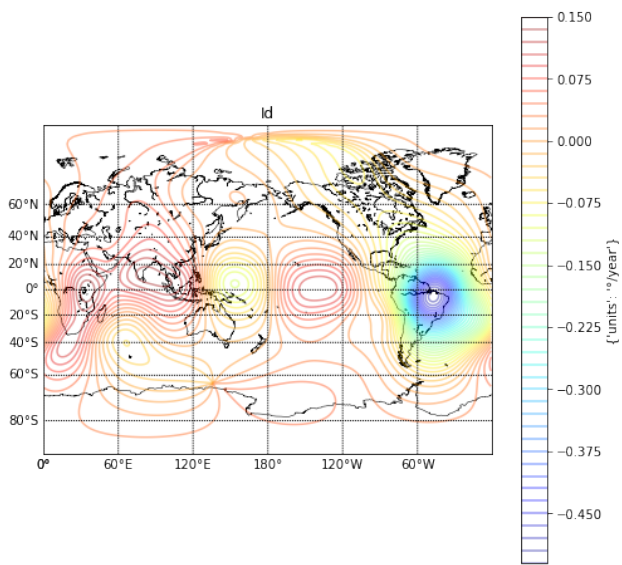


(a)  $\dot{I}$  at South Pole

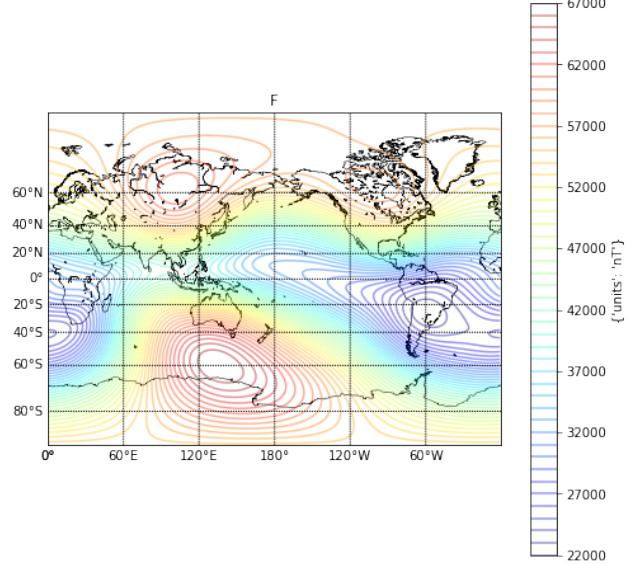


(b)  $\dot{I}$  at North Pole

Figure 6: Examples of the secular variation of the geomagnetic field inclination at the North and South Pole.



(a) Secular variation  $\dot{I}$  of geomagnetic inclination.



(b) Geomagnetic intensity  $F$ .

Figure 7: Examples of geomagnetic field computation for the whole Earth.



## 10.2 2D plots: Example 2

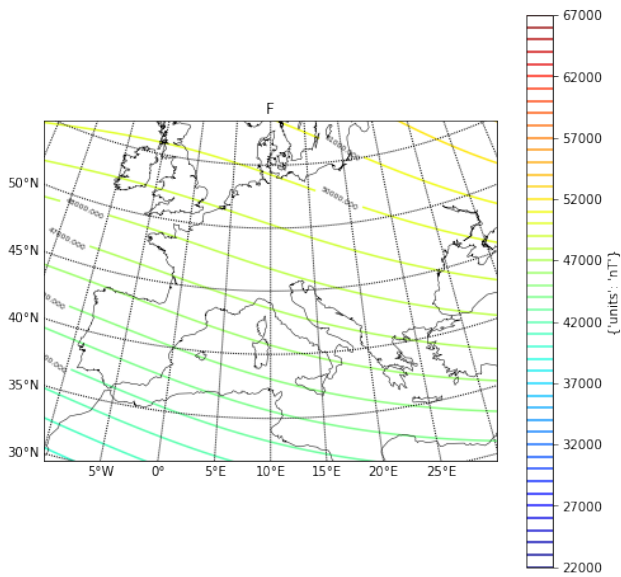
Using the following code generates the the plots of Figure 8 and Figure 9.

```
geo.plot_geomagetism(dintensities,
                    dangles,
                    dintensities_sv,
                    dangles_sv,
                    'F',
                    {'proj':'lcc',
                    'lat_1':45.,
                    'lat_2':55,
                    'lat_0':45,
                    'lon_0':10.})

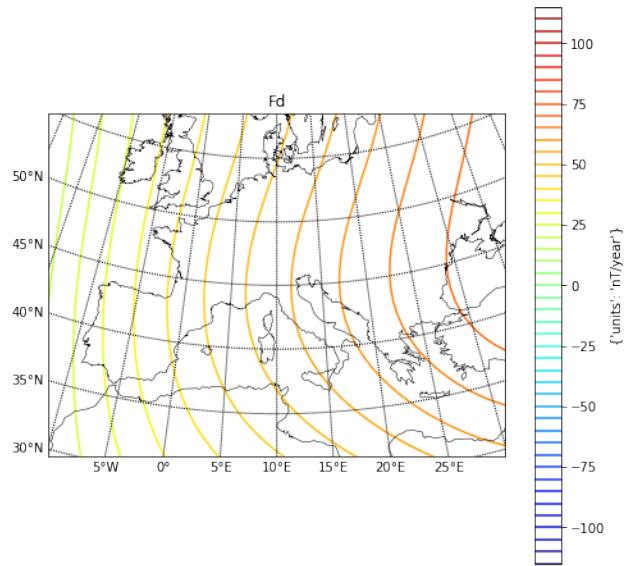
geo.plot_geomagetism(dintensities,
                    dangles,
                    dintensities_sv,
                    dangles_sv,
                    'Fd',
                    {'proj':'lcc',
                    'lat_1':45.,
                    'lat_2':55,
                    'lat_0':45,
                    'lon_0':10.})

geo.plot_geomagetism(dintensities,
                    dangles,
                    dintensities_sv,
                    dangles_sv,
                    'I',
                    {'proj':'lcc',
                    'lat_1':45.,
                    'lat_2':55,
                    'lat_0':45,
                    'lon_0':10.})

geo.plot_geomagetism(dintensities,
                    dangles,
                    dintensities_sv,
                    dangles_sv,
                    'Id',
                    {'proj':'lcc',
                    'lat_1':45.,
                    'lat_2':55,
                    'lat_0':45,
                    'lon_0':10.})
```

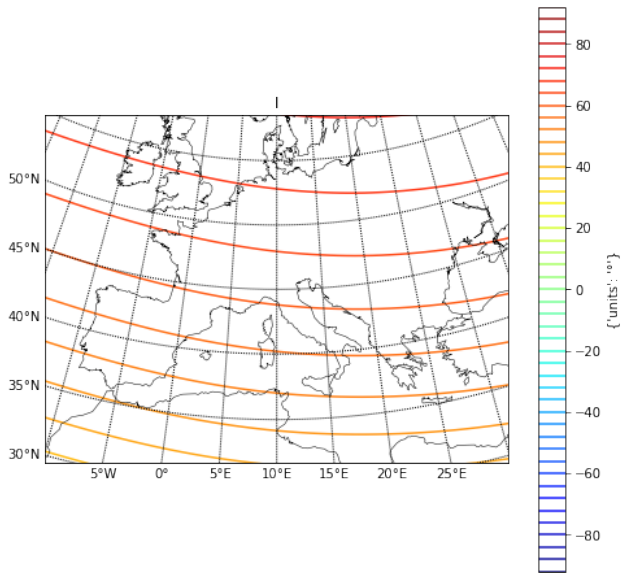


(a) Intensity  $F$  of the geomagnetic field.

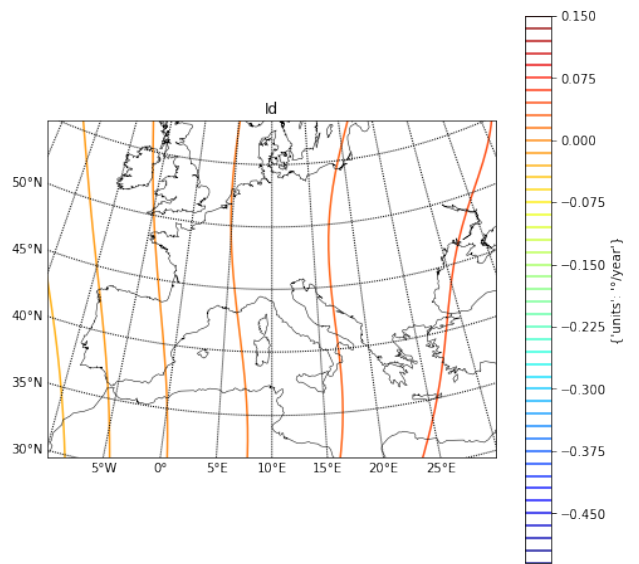


(b) Secular variation  $\dot{F}$  of the geomagnetic field intensity.

Figure 8: Examples of field magnitude and field magnitude secular variation in South Western Europe.



(a) Inclination  $I$  of the geomagnetic field.



(b) Secular variation  $\dot{I}$  of the geomagnetic field inclination.

Figure 9: Examples of field inclination and inclination secular variation in South Western Europe.

## References

- [1] Campbell, W. H. (2007). *Introduction to Geomagnetic Fields* Cambridge University Press.
- [2] Chulliat, A., W. Brown, P. Alken, C. Beggan, M. Nair, G. Cox, A. Woods, S. Macmillan, B. Meyer and M. Panizza (2020). *The US/UK World Magnetic Model for 2020-2025: Technical Report, National Centers for Environmental Information* NOAA, [doi:10.25923/ytk1-yx35](https://doi.org/10.25923/ytk1-yx35).
- [3] Peddie, N. W. (1982). *International Geomagnetic Reference Field: the third generation*. J. Geomag. Geoelectr 34: 309-326.
- [4] Stacey, F. D. and P. M. Davis (2008). *Physics of the Earth* Cambridge University Press
- [5] Winch, D. E., et al. (2005). *Geomagnetism and Schmidt quasi-normalization*. Geophys. J. Int. 160: 487-454.
- [6] Zhang, S. and Jin, J. (1996). *Computation of Special Functions*. Wiley-Interscience.
- [7] Glaßmeier, K.-H., et al. (2008) *Geomagnetic Field Variations (Advances in Geophysical and Environmental Mechanics and Mathematics)*. Springer.
- [8] Arfken, G., Weber H. (2005) *Mathematical Methods for Physicists*. Sixth Edition Elsevier.