Universität Potsdam
Institut für Mathematik

Helmholtz-Zentrum Potsdam
Deutsches GeoForschungsZentrum

# Stochastic Inversion for Core Field Modeling Using Satellite Data

Masterarbeit

zur Erlangung des akademischen Grades
Master of Science (M. Sc.)

von
Jan Möhring

Potsdam, im Januar 2021

**Erstgutachterin**: Prof. Dr. Claudia Stolle
**Zweitgutachter**: apl. Prof. Dr. Gert Zöller
**Betreuer**: Dr. Achim Morschhauser

# Zusammenfassung

Magnetfeldmodellierung mit Kugelflächenfunktionen basiert auf der Inversion nach hunderten bis tausenden von Parametern. Dieses hochdimensionale Problem kann grundsätzlich als ein Optimierungsproblem formuliert werden, bei dem ein globales Minimum einer gewissen Zielfunktion berechnet werden soll. Um dieses Problem zu lösen, gibt es eine Reihe bekannter Ansätze, dazu zählen etwa gradientenbasierte Verfahren oder die Methode der kleinsten Quadrate und deren Varianten. Jede dieser Methoden hat verschiedene Vor- und Nachteile, beispielsweise bezüglich der Anwendbarkeit auf nicht-differenzierbare Funktionen oder der Laufzeit zugehöriger Algorithmen. In dieser Arbeit verfolgen wir das Ziel, einen Algorithmus zu finden, der schneller als die etablierten Verfahren ist und sich auch für nichtlineare Probleme anwenden lässt. Solche nichtlinearen Probleme treten beispielsweise bei der Abschätzung von Euler-Winkeln oder bei der Verwendung der robusteren $L_1$-Norm auf. Dazu untersuchen wir die Anwendbarkeit stochastischer Optimierungsverfahren aus der CMAES-Familie auf die Modellierung des geomagnetischen Feldes des Erdkerns. Es werden sowohl die Grundlagen der Kernfeldmodellierung und deren Parametrisierung anhand einiger Beispiele aus der Literatur besprochen, als auch die theoretischen Hintergründe der stochastischen Verfahren gegeben. Ein CMAES-Algorithmus wurde erfolgreich angewendet, um Daten der Swarm-Satellitenmission zu invertieren und daraus das Magnetfeldmodell EvoMag abzuleiten. EvoMag zeigt gute Übereinstimmung mit etablierten Modellen, sowie mit Observatoriumsdaten aus Niemegk. Wir thematisieren einige beobachtete Schwierigkeiten und präsentieren und diskutieren die Ergebnisse unserer Modellierung.

Diese Arbeit ist in vier Kapitel unterteilt. Im Kapitel 1 werden Eigenschaften des Erdmagnetfeldes und die Modellierung des Kernfeldes mit Kugelflächenfunktionen beschrieben. Außerdem werden verschiedene etablierte Magnetfeldmodelle dargestellt. Kapitel 2 behandelt zunächst Inversionsalgorithmen und stochastische Optimierung. Dann folgt eine Beschreibung von Evolutionsstrategien, einer für uns wichtigen Klasse von stochastischen Optimierungsalgorithmen. Anschließend werden der CMAES-Algorithmus und verwandte Algorithmen erklärt, die in dieser Arbeit angewendet wurden. Schließlich wird der Einfluss der Konditionszahl des Vorwärtsoperators auf die Anzahl der für die Inversion benötigten Funktionsaufrufe untersucht. In Kapitel 3 stellen wir zunächst die Parametrisierung unseres Kernfeldmodells EvoMag dar, und beschreiben dabei insbesondere die verwendeten Daten und einige Details der Implementierung in Python. Dann werden die Ergebnisse unseres Modells präsentiert und mit etablierten Magnetfeldmodellen verglichen. Schließlich werden im vierten Kapitel die Vor- und Nachteile der stochastischen Optimierung für die Kernfeldmodellierung zusammengefasst. Wir benennen nicht gelöste Probleme und skizzieren dazu einige Lösungsideen.

# Summary

Geomagnetic field modeling using spherical harmonics requires the inversion for hundreds to thousands of parameters. This large-scale problem can always be formulated as an optimization problem, where a global minimum of a certain cost function has to be calculated. A variety of approaches is known in order to solve this inverse problem, e.g. derivative-based methods or least-squares methods and their variants. Each of these methods has its own advantages and disadvantages, which affect for example the applicability to non-differentiable functions or the runtime of the corresponding algorithm. In this work, we pursue the goal to find an algorithm which is faster than the established methods and which is applicable to non-linear problems. Such non-linear problems occur for example when estimating Euler angles or when the more robust $L_1$ norm is applied. Therefore, we will investigate the usability of stochastic optimization methods from the CMAES family for modeling the geomagnetic field of Earth's core. On one hand, basics of core field modeling and their parameterization are discussed using some examples from the literature. On the other hand, the theoretical background of the stochastic methods are provided. A specific CMAES algorithm was successfully applied in order to invert data of the Swarm satellite mission and to derive the core field model EvoMag. The EvoMag model agrees well with established models and observatory data from Niemegk. Finally, we present some observed difficulties and discuss the results of our model.

This work is divided into four chapters. In Chapter 1 we describe properties of the Earth's magnetic field and core field modeling using spherical harmonics. Furthermore, we portray some established magnetic field models. Chapter 2 starts with a treatment of inversion algorithms and random optimization, followed by a description of Evolution Strategies, an important class of stochastic optimization algorithms for our purposes. Then, the CMAES algorithm and related algorithms, which were used for this work, will be explained. Finally, the influence of the condition number of the forward operator on the number of function calls that are required for the inversion, will be investigated. In Chapter 3 we present the parameterization of our core field model EvoMag. Thereby, we describe the used data and some details of the implementation in Python. We present the results of our model, and compare it with established magnetic field models. Finally, in the fourth chapter, the advantages and disadvantages of stochastic optimization applied to core field modeling are summarized. We list non-solved problems and sketch some ideas for their solution.

# Contents

# List of Figures

# 1  Introduction

In this chapter we state the assumptions and considerations which will lead us to the problem of geomagnetic field inversion. We also comment on some properties of this inverse problem. For further details we refer to Morschhauser (2016).

## 1.1  The Earth's magnetic field

The Earth's magnetic field is composed of several sources, which can be separated into internal sources and external sources. Internal sources include the core field and the lithospheric field, and external fields arise through electric currents in the ionosphere and magnetosphere (see figure 1.1).

The largest contribution comes from the core field, generated in the Earth's outer core by a self-generating geodynamo. For the core field, $20,000$ nT up to $70,000$ nT can be measured on the Earth's surface. This magnetic field can be approximated within $90\%$ by a dipole with its origin centered at the center of the Earth. An important property of the core field is that the field is time dependent. The time changes of the core field are also referred to as secular variation and vary with timescales ranging from several months to thousands of years.

The magnetic field of the Earth has several distinct features. One of these features is the South Atlantic Anomaly (SAA). The anomaly is an intrinsic characteristic of the core field and arises directly in the core. This area stretches from southern Africa over to South America and is described by a zone of weak and decreasing field strength (see also figure 1.2). It is noteworthy that the radiation comes closer to the Earth over the SAA, imposing a threat to satellites.

Compared to the core field, the lithospheric field, which arises from induced and remanent magnetization of minerals in the Earth's crust, is rather weak. The field strength can reach several thousands of nanoteslas at the Earth's surface and the corresponding time scales vary between thousands and millions of years. For this reason, the lithospheric field is assumed to be constant in field models.

Another weak contribution to the magnetic field comes from induced sources. For example, currents are induced within the oceans. The corresponding field strength usually varies between 2 nT and 2.5 nT at satellite altitude. Moreover, an even larger induced contribution arises from induced currents in the mantle.

Finally, there are contributions from the ionosphere and magnetosphere. The ionosphere extends from the mesosphere up to the exosphere, which includes heights between 60 km and 2000 km. More precisely, the magnetic field in the ionosphere is generated by electric currents which are induced within the E- and F- region in approximately 110 kilometers altitude. These currents occur more frequently on the day
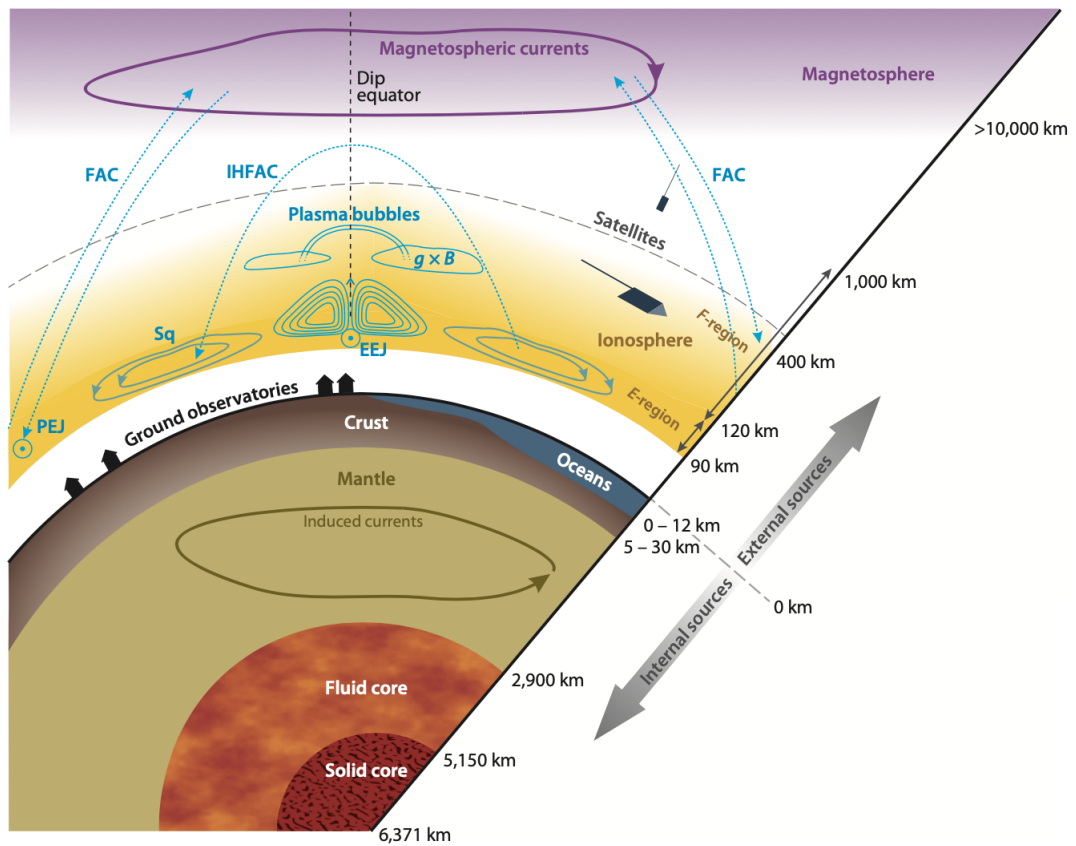
Figure 1.1: Sources contributing to the near-Earth magnetic field. Abbreviations: $B$, ambient magnetic field; EEJ, equatorial electrojet; FAC, field-aligned current; $g$, Earth's gravity vector; IHFAC, interhemispheric field-aligned current; PEJ, polar electrojet; Sq, solar quiet daily magnetic variation. Figure taken from Olsen and Stolle (2012, figure 2).

side because of the increased conductivity through the ionization of charged particles. The magnetosphere is the region where the magnetic field of the Earth dominates. Behind the magnetopause, the solar magnetic field is dominating. The solar wind is a current of charged particles which interacts with the magnetosphere, such that currents are induced. The field strength in this region is in the order of 6 nT, measured at Earth's surface, but can rise to several hundred nT during geomagnetic storms.

In this thesis, we will investigate stochastic inversion schemes for the derivation of magnetic field models from satellite data. Such models are important for a variety of reasons. For example, they allow to describe the field in regions where no measurements are available (within certain assumptions). As a result, several scientific studies become possible. This includes for example the study of ionospheric currents and the possibility to understand how the field changes with time. Further, by downward-continuation to the core mantle boundary, information on the material flux and the core dynamo can be obtained. Further usage includes important civic applications such as navigation.

Geomagnetic field model / GFZ
Swarm satellite mission / ESA
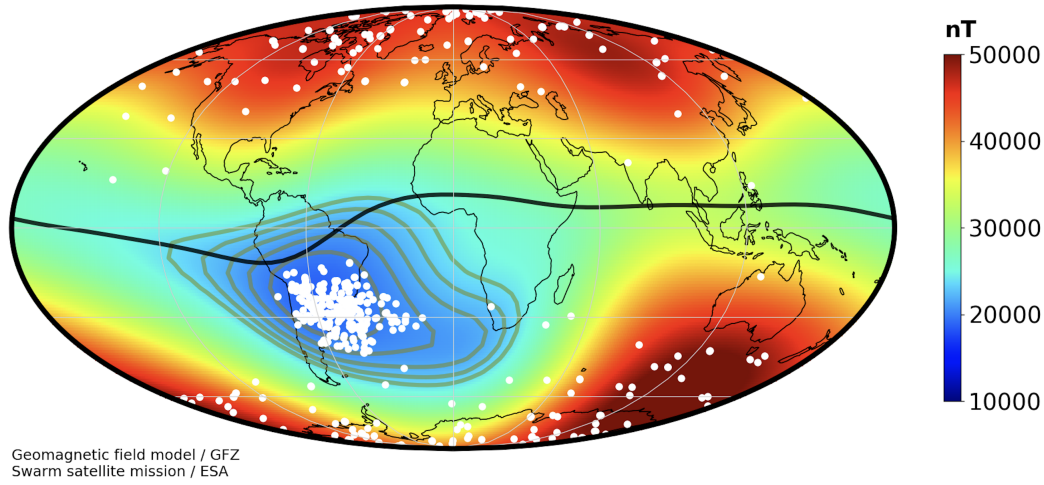
Figure 1.2: Swarm single event upsets and global geomagnetic field strength. Within the region of the South Atlantic Anomaly, a huge number of upsets is recorded. The figure was created by Ingo Michaelis at GFZ Potsdam.

## 1.2 Mathematical description of the core field

In this section, we will describe the common approach to describe the core magnetic field in terms of spherical harmonic functions. Here, we restrict ourselves to regions $\Omega$ without electric currents and temporarily changing electric fields. In this case, it follows from Ampère's law and the non-existence of magnetic monopoles (Maxwell's equations) that

$$
\begin{aligned}
\nabla \cdot B &= 0, \\
\nabla \times B &= 0.
\end{aligned}
\tag{1.1}
$$

Here, $B$ denotes the magnetic induction field. The second equation in (1.1) precisely describes the integrability condition for vector fields. Hence, $B$ is conservative within the region $\Omega$, which implies that there is a scalar potential $V$ such that

$$
B = -\nabla V.
\tag{1.2}
$$

Therefore, $V$ must be a solution of Laplace's equation:

$$
\Delta V = 0.
\tag{1.3}
$$

It is well-known that a separation ansatz in spherical coordinates $V = V(r, \theta, \varphi)$ reveals a representation of $V$ which is given by

$$
\begin{aligned}
V = a \sum_{l=1}^{\infty} \sum_{m=0}^{l} \Bigg[ &\left( g_l^m \cos(m\phi) + h_l^m \sin(m\phi) \right) \left( \frac{a}{r} \right)^{l+1} P_l^m(\cos(\theta)) \\
&+ \left( q_l^m \cos(m\phi) + s_l^m \sin(m\phi) \right) \left( \frac{r}{a} \right)^{l} P_l^m(\cos(\theta)) \Bigg],
\end{aligned}
\tag{1.4}
$$

where $a$ is the reference radius and $P_l^m(\cos(\theta))$ are the Schmidt semi-normalized associated Legendre functions of degree $l$ and order $m$. Usually, we take $a = 6371.2$ km (Earth's mean radius). The real coefficients $\{g_l^m, h_l^m\}$ and $\{q_l^m, s_l^m\}$ are called Gauss coefficients and have the same physical dimensions as $B$. The term for $l = 0$ is omitted from the series since it describes a magnetic monopole.

In equation (1.4), the angular contributions are referred to as real spherical harmonic functions $Y_l^m$. We use the convention that negative orders ($m < 0$) are associated with $\sin(m\phi)$ terms whereas positive orders ($m \geq 0$) are associated with $\cos(m\phi)$ terms. Then the $Y_l^m$ take the form

$$Y_l^m(\theta, \phi) = \begin{cases} \sin(|m|\phi)P_l^{|m|}(\cos(\theta)) & \text{for } m < 0 \\ \cos(m\phi)P_l^m(\cos(\theta)) & \text{for } m \geq 0, \end{cases} \tag{1.5}$$

where $-l \leq m \leq l$. The usage of spherical harmonics is very natural since they are eigenfunctions of the Laplacian when $\Omega$ is the 2-sphere.

The potential equation (1.4) can be separated into an internal part $V_{\text{int}}$ and an external part $V_{\text{ext}}$ such that

$$\begin{aligned} V_{\text{int}} &= a \sum_{l=1}^{\infty} \sum_{m=-l}^{l} \left(\frac{a}{r}\right)^{l+1} g_l^m Y_l^m(\theta, \varphi), \\ V_{\text{ext}} &= a \sum_{l=1}^{\infty} \sum_{m=-l}^{l} \left(\frac{r}{a}\right)^{l} q_l^m Y_l^m(\theta, \varphi), \end{aligned} \tag{1.6}$$

where we set $g_l^m = h_l^m$ and $q_l^m = s_l^m$ for $m < 0$. We remark that the term $(a/r)^{l+1}$ of the internal potential $V_{\text{int}}$ vanishes at infinity ($r \to \infty$). This corresponds to internal fields. The terms in $(r/a)^l$ describe fields of external origin with respect to the surface on which the data were measured, not necessarily with respect to the Earth's mean radius or the reference radius $a$.

For numerical calculations, the series expansions in equations (1.6) are truncated at a certain degree $L$. The choice of $L$ depends, for example, on the quality of the data and the spatial wavelength of the field which has to be modeled (Finlay, Lesur, et al., 2017). We refer to $L$ as the maximum spherical harmonic degree. The number $L$ can also be regarded as a measure for the complexity of a model.

The three vector components $X, Y$ and $Z$ of the field are related to the potential $V$ by

$$X = \frac{1}{r}\frac{\partial V}{\partial \theta}, \qquad Y = -\frac{1}{r\sin(\theta)}\frac{\partial V}{\partial \phi}, \qquad Z = \frac{\partial V}{\partial r}, \tag{1.7}$$

where $X$ is horizontal and north, $Y$ is horizontal and east, and $Z$ is vertically downwards (Whaler and Gubbins, 1981). Since differentiation in equations (1.6) is linear and the coefficients in the representation of $V$ appear linear, a substitution of equations (1.6) (with a truncated series) into equation (1.7) shows that the parameterization by spherical harmonics can be described as a linear equation system

$$Gm = d, \tag{1.8}$$

where $G \in \mathbb{R}^{n \times k}$ describes the spherical harmonics model, $m \in \mathbb{R}^k$ is the model vector which is associated to the Gauss coefficients, and $d \in \mathbb{R}^n$ is the vector containing the data. An important property of $G$ is that the columns of the matrix are linearly independent due to the fact that the spherical harmonics form an orthonormal basis of the Hilbert space of square-integrable functions.

Suppose that the three components of $B$ are measured at $N_{\mathrm{obs}}$ locations, then one possible and convenient sorting scheme for the data vector is given by

$$d = (B_{X_1}, B_{Y_1}, B_{Z_1}, B_{X_2}, B_{Y_2}, B_{Z_2}, \ldots, B_{X_{N_{\mathrm{obs}}}}, B_{Y_{N_{\mathrm{obs}}}}, B_{Z_{N_{\mathrm{obs}}}})^{\intercal} \in \mathbb{R}^{3N_{\mathrm{obs}}}. \qquad (1.9)$$

The data usually consist of a set of satellite or ground observatory measurements, and often more than a million of data points are available. Further, if the series expansions in equations (1.6) are truncated at degree $L$, it can be shown that $k = L(L+2)$. Hence, the model vector is given as $m \in \mathbb{R}^{L(L+2)}$ and the linear equation system (1.8) has a coefficient matrix $G$ of size $3N_{\mathrm{obs}} \times L(L+2)$. For better interpretation and separation of the core field, the Gauss coefficients describing the magnetic potential $V$ have to be estimated from the data with the help of the forward model as described by equation (1.8). This inverse problem is highly over-determined, and is usually solved by a least-squares inversion scheme. In this work, we will investigate the usability of stochastic inversion methods for solving the inverse problem, as detailed in section 2.1.

## 1.2.1 Time dependence

In the parameterization described by equations (1.6) and (1.7), it is assumed that the Gauss coefficients are static. Since the core field is time dependent, this will not result in a satisfactory model, especially as the data necessary to invert for a global field model are usually obtained over several years. Hence, the Gauss coefficients are treated as time dependent and will be expanded as

$$g_l^m(t) = \sum_{i=1}^{N_{\mathrm{b}}} {}^i g_l^m \psi_i(t), \qquad (1.10)$$

where the $\psi_i$ are some sort of basis functions, $N_{\mathrm{b}}$ is the number of basis functions, and ${}^i g_l^m$ are the corresponding expansion coefficients. Independent of the choice of the $\psi_i$, equation (1.10) is linear in the expansion coefficients $\{{}^i g_l^m, {}^i h_l^m\}$ and can be written as a linear equation system as in equation (1.8). As a result, the matrix $G$ has $L(L+2)N_{\mathrm{b}}$ columns, and the number of unknown model parameters has hence increased by the factor $N_{\mathrm{b}}$.

A typical choice (Olsen, Lühr, et al., 2014; Rother, Korte, et al., 2020) for the basis functions are sixth-order B-splines (De Boor, 1978). B-splines are defined over a set of knots $t = (t_i : i \in I)$, which represent a non-decreasing sequence of real numbers, where the index set $I \subseteq \mathbb{N}$ is finite or countably infinite. The $i$-th B-spline of order $k = 1$ is defined by

$$B_{i,1}(x) := \begin{cases} 1 & t_i \leq x < t_{i+1} \\ 0 & \text{otherwise} \end{cases} \qquad (1.11)$$

and the higher order B-splines are defined recursively by

$$B_{i,k+1}(x) := \omega_{i,k}(x)B_{i,k}(x) + [1 - \omega_{i+1,k}(x)]B_{i+1,k}(x), \tag{1.12}$$

with

$$\omega_{i,k}(x) := \begin{cases} \frac{x-t_i}{t_{i+k}-t_i} & t_{i+k} \neq t_i \\ 0 & \text{otherwise.} \end{cases} \tag{1.13}$$

B-splines have small support, i.e. $B_{i,k}(x) = 0$ for $x \notin [t_i, t_{i+k}]$. Furthermore, the $B_{i,k}$ are positive on their support, i.e. $B_{i,k}(x) > 0$ for $t_i < x < t_{i+k}$. Using B-splines as basis functions, equation (1.10) becomes

$$g_l^m(t) = \sum_{i=1}^{N_{\mathrm{b}}} {}^i g_l^m B_{i,k}(t). \tag{1.14}$$

Nowadays, this parameterization is used in a large number of models and was essentially inspired by the example of Bloxham and Jackson (1992). There are several reasons why B-splines are highly popular in magnetic field modeling. One point is that since the $B_{i,k}$ vanish outside a small neighborhood, the matrix $G$ in equation (1.8) becomes sparse. This is a major advantage as memory-efficient algorithms exist for sparse matrix linear operations. Furthermore, a $k$-th order B-spline is infinitely often differentiable, except at the knots. Here, the B-spline is $(k-2)$-times differentiable at a single knot. The differentiability is reduced by one for each multiplicity on multiple knots.

## 1.2.2 Remarks on ill-posedness

Let us recall that a mathematical problem is called well-posed in the sense of HADA-MARD if the problem has a solution for all admissible data, if this solution is unique, and if the solution depends continuously on the data (Engl, Hanke, and Neubauer, 2000). A problem which is not well-posed is called ill-posed. Nearly all (geophysical) inverse problems are ill-posed. In particular this is also true for the problem of the determination of Gauss coefficients:

- In general we have $n \gg k$, i.e. more data than model parameters. In $\mathbb{R}^k$, a set of $n > k$ vectors is always linearly dependent. Since all of the data are measured at different points in space and time, and since the magnetic field always differs at least slightly at these points, it follows that the linear equation system (1.8) is not uniquely solvable. A reduction to $n = k$ data points in principle could result in an uniquely solvable system. However, in practice, data errors would lead to non-physical results.

- The used parameterization does not exactly describe reality. For example, iono-spheric sources are usually greatly simplified. Only with a perfect parameteri-zation (or model), one could simply measure on $n = k$ positions on the Earth's

surface and would obtain a robust and unique solution. What is even more important to consider is that the geomagnetic potential on the Earth's surface is not uniquely determinable. This is a consequence of the fact that the series expansion of $V_{\text{int}}$ in equations (1.6) is truncated at a certain degree $L$, since the potential is not any longer known from this truncation degree on (Wardinski, 2005).

- In general, the condition number of $G$ is high. Small perturbations in the data lead to significant changes in the Gauss coefficients, such that the solution is not stable a priori. This problem is amplified by downward-continuation, where the term $(a/r)^{l+1}$ increases with decreasing altitude, $r$, and small errors in the Gauss coefficients will hence become more and more dominant.

Caused by the mentioned properties of the problem, regularization techniques are widely applied. We will give some examples in the following field model descriptions.

## 1.3  Existing field models

Two common possible approaches are known for developing geomagnetic field models: "Comprehensive modeling" and "consecutive modeling". The idea of comprehensive modeling is to include all sources and then to invert them together with the internal core field model. A known representative of this method is the CM model series (Sabaka et al., 2018). Using this approach, one obtains a self-consistent model that is aimed to represent the field in its entirety, at least theoretically. On the other hand, the model will contain a lot of parameters, making the model and inversion more complex. Moreover, dependencies between different parameters need to be properly considered and accounted for. In other words, the different sources of the field may be hard to separate due to similar spatial or temporal characteristics. A common workaround is to include a-priori information from physics-based models.

Regarding "consecutive modeling", the influence of external sources is reduced by proper data selection and data "cleaning", e.g. by the subtraction of dedicated models. As an example, we point out the POMME-BOUMME model series (Maus et al., 2006). However, we have to be aware of one danger: It is very difficult to control whether the field of two models overlap, i.e. part of the field may be erroneously subtracted. Consider for example the situation that a core field model shall be developed and therefore, a dedicated crustal field model will be subtracted a priori. In that case, the subtracted part may be already too large as it may contain parts of the core field. This implies the need for a good understanding about the parameterization and other properties of the used models.

Finally, we mention that there are models that are a mix of the above explained strategies. Examples are the CHAOS-model (Finlay, Olsen, et al., 2016) and the GRIMM model (Lesur, Wardinski, et al., 2010).

In the following, we exemplarily give more details on the International Geomagnetic Reference Model (IGRF), the CHAOS core field model, and the GFZ Mag.num core field model.

## 1.3.1 IGRF

The IGRF is not only used by scientists, but also by commercial institutions, making it the most widely used geomagnetic field model. The model is updated every five years by the International Association of Geomagnetism and Aeronomy (IAGA). These regular revisions are necessary in order to take the continuous time development of the geomagnetic field in the outer Earth's core into account. The model is updated by the IAGA working group V-MOD and starts with a call for candidate models several months ahead of the new release date. In order to obtain coefficients of new constituent models, weighted means of the coefficients of selected candidate models are taken (Macmillan and Finlay, 2010). We shortly describe its parameterization, and more details can be found in Thébault et al. (2015).

The model describes the core field without describing external sources. Here, the usual expansion of the scalar potential in spherical harmonics is carried out in geocentric coordinates. The IGRF is composed of the coefficients for a global representation of the Earth's magnetic field from years beginning at 1900, in five-year distances (epochs). In addition, the time derivatives of the coefficients are also included for the most recent epoch.

With the usual notation, the scalar potential satisfies the equation

$$V(r, \theta, \phi, t) = a \sum_{l=1}^{L} \sum_{m=0}^{l} \left(\frac{a}{r}\right)^{l+1} [g_l^m(t)\cos(m\phi) + h_l^m(t)\sin(m\phi)] P_l^m\big(\cos(\theta)\big). \quad (1.15)$$

It is assumed that the Gauss coefficients depend linearly on the time over a five year interval by

$$
\begin{aligned}
g_l^m(t) &= g_l^m(T_0) + \dot{g}_l^m(T_0)(t - T_0), \\
h_l^m(t) &= h_l^m(T_0) + \dot{h}_l^m(T_0)(t - T_0),
\end{aligned}
\qquad (1.16)
$$

where $\dot{g}_l^m$ (respectively $\dot{h}_l^m$) is a five-year average of the first time derivative between two epochs. By convention, the Gauss coefficients are given in nanotesla (nT). The five-year averages are given in units of nT/year. The time $t$ is given in units of year and $T_0$ is an epoch preceding $t$ which fulfills $T_0 \leq t < (T_0 + 5.0)$ and is an exact multiple of five years.

In equation (1.15) the maximal spherical harmonics degree $L = 10$ was chosen for coefficients up to including 1995.0, and $L = 13$ afterwards. As a result, the crustal magnetic field contributions are reduced, which dominate at higher degrees (Langel and Estes, 1982). The secular variation coefficients $\dot{g}_l^m(t)$ and $\dot{h}_l^m(t)$ are given up to degree and order eight to a nominal precision of 0.1-nT/year.

## 1.3.2 CHAOS

The CHAOS series is a family of global geomagnetic field models with the goal to provide a description of the Earth's magnetic field with high spatial and temporal resolution. Satellite data of Ørsted, CHAMP and SAC-C from different time periods

are used. Different data selection criteria are applied, for example, the contributions of ionospheric currents shall be reduced (Mandea and Korte, 2011) by the usage of data only from dark regions (sun 10° below the horizon). Further, data were selected according to the Kp index which measures geomagnetic activity. Here, data from non-polar regions with Kp > 2o, which corresponds to a variation (peak-to-peak) range of $\leq 7$ nT, were excluded, leaving data from mainly undisturbed days.

Although the latest version of the model is CHAOS-7, we will mostly have a look on the parameterization of CHAOS-4 (Olsen, Lühr, et al., 2014), as the parameterization of CHAOS-4 is the basis for later models, and no comprehensive description has been published since. The final model is composed of a combination of coefficients of two sub models, namely CHAOS-4l and CHAOS-4h, that are parameterized according to equation (1.15). Here, the letters "l" and "h" stand for "low" and "high", respectively. Both models differ in the choice of the maximal degree and order of the internal expansion $L = N_{\text{int}}$, where $N_{\text{int}} = 80$ and $N_{\text{int}} = 100$ was chosen for CHAOS-4l and CHAOS-4h, respectively. For the low-degree-model, CHAOS-4l, analogously to equation (1.14), the coefficients $\{g_l^m(t), h_l^m(t)\}$ up to $l = 20$ are assumed to be time dependent. The time dependence is described by sixth-order B-splines, where the knots are separated by sixth months. The coefficients for $l = 21 - 80$ are static. Within CHAOS-4l, Euler angles are additionally estimated. This is done separately for Ørsted and CHAMP and leads to 1107 additional model parameters. On the other hand, for the high degree model the core field coefficients up to $l = 16$ are assumed to depend quadratically on time, whereas the coefficients for $l = 17 - 100$ are static. For the final model the coefficients up to $l = 24$ are taken from the low degree model and the coefficients for the degrees $l = 25 - 100$ are obtained from the high degree model.

Overall, the model contains 25177 parameters. These parameters are estimated by applying a regularized iteratively reweighted least-squares approach with Huber weights in order to minimize the objective function

$$f(m) = e^{\mathsf{T}} C^{-1} e + \lambda_3 m^{\mathsf{T}} \Lambda_3 m + \lambda_2 m^{\mathsf{T}} \Lambda_2 m. \tag{1.17}$$

Here, $m$ is the model vector, $e = d_{\text{obs}} - d_{\text{mod}}$ is the residual vector as a difference between the observation $d_{\text{obs}}$ and the model prediction $d_{\text{mod}}$, and $C$ is the data covariance matrix. Furthermore, $\Lambda_3$ and $\Lambda_2$ are block-diagonal regularization matrices. The matrix $\Lambda_3$ is designed such that it acts as a minimizer of the average of the squared third time derivative, by fulfilling

$$\frac{1}{\Delta t} \int_{t=1997}^{2013.5} \int \left| \frac{\partial^3 B_r}{\partial t^3} \right|^2 \mathrm{d}\Omega_c \, \mathrm{d}t = m^{\mathsf{T}} \Lambda_3 m, \tag{1.18}$$

with $\Delta t = 2013.5 - 1997.0 = 16.5$ years. The usage of this regularization can lead to field oscillations (Olsen, Lühr, et al., 2014). In order to suppress them, the expression $|\partial_t^2 B_r|$ is minimized at the time points $t = 1997$ and $2013.5$, leading to the matrix $\Lambda_2$. The scalars $\lambda_3$ and $\lambda_2$ are regularization parameters which control the influence of the respective terms. By testing a series of values, finally $\lambda_2 = 10(\text{nT yr}^{-2})^{-2}$ and $\lambda_3 = 0.33(\text{nT yr}^{-3})^{-2}$ were chosen.

### 1.3.3  Mag.num

The `Mag.num.IGRF13` core field model (thereafter called `Mag.num`) is the parent for GFZ's IGRF-13 candidate models (Rother, Korte, et al., 2020).

The model is based on vector field data from three Swarm satellites obtained between November 2013 and August 2019. As for CHAOS, the data were selected according to several criteria, such as the usage of nighttime data for low-latitude regions, or the requirement that the $z$-component of the interplanetary magnetic field must be positive for selected data (for more details, see Rother et al. (2013)). Additionally to ordinary satellite data, their along-track and cross-track differences, as well as observatory data were used. The usage of observatory data has the goal to better constrain secular variation.

The internal potential is parameterized as in equations (1.6) and the maximum spherical harmonic degree $L$ for the internal potential was assumed to be $L = 20$. Similar to equation (1.14), the Gauss coefficients are time dependent and expanded in a basis of sixth-order B-splines on the interval from December 31, 2012 to January 1st, 2020. The knots are half-year separated and $N_{\mathrm{b}} = 9$ spline functions represent each Gauss coefficient. The higher coefficients of the model were assumed to be static.

The inversion of the data is done by minimizing the function

$$f(m) = (d - Gm)^{\mathsf{T}} W^{\mathsf{T}} W (d - Gm), \tag{1.19}$$

where the diagonal matrix $W$ contains the estimated data weights $w_i$ as described below. In order to reduce the influence of outliers, a modified Huber norm (Morschhauser, Lesur, and Grott, 2014; Lesur, Rother, Wardinski, et al., 2015) was used. Iteratively re-weighted least-squares (Farquharson and Oldenburg, 1998) are used for approximating this norm, and the resulting weight of the data point $d_i$ at iteration $(j + 1)$ is given by

$$w_i^{j+1} = \begin{cases} \frac{1}{\sigma_i} & \text{for } |d_i - (Gm^j)_i| \leq k_i \\ \frac{1}{\sigma_i} \left[ \frac{k_i}{|d_i - (Gm^j)_i|} \right]^{1 - a_i/2} & \text{for } |d_i - (Gm^j)_i| > k_i. \end{cases} \tag{1.20}$$

Here, $m^j$ denotes the estimated Gauss coefficients at iteration $j$, and $a_i$ and $k_i$ are used to adapt the modified Huber norm. The $\sigma_i$ are assumed standard deviations, they differ between different data subsets, can be regarded as representatives of the noise associated with the data, and need to be estimated a priori.

Due to unmodeled signals in the data, the model is regularized in a similar way as CHAOS. In more detail, unrealistically strong spatial and temporal field variations are reduced by damping the third time derivative of the radial field at Earth's core radius $c$, i.e. by minimizing

$$\Phi_1 = \frac{1}{4\pi c^2 (t_2 - t_1)} \int_{t_1}^{t_2} \int_{\Omega(c)} |\partial_t^3 B_r| \, \mathrm{d}\omega \, \mathrm{d}t \tag{1.21}$$

(beginning at the start point $t_1$ and ending at the end point $t_2$ of the model). Moreover,

by minimizing the functions

$$\Phi_2(t) = \frac{1}{4\pi c^2} \int_{\Omega(c)} |\partial_t^2 B_r| \, \mathrm{d}\omega \,, \qquad \Phi_3(t) = \frac{1}{4\pi c^2} \int_{\Omega(c)} |\partial_t B_r| \, \mathrm{d}\omega \,, \tag{1.22}$$

at the times $t = t_1$ and $t = t_2$, oscillations due to the lack of data near the end points can be avoided.

# 2 Optimization with Evolution Strategies

In the following, we give an overview about inversion algorithms and random optimization. Then we continue with a description of Evolution Strategies, these are nature-inspired optimization methods which are based on the idea of evolution. Within an iterative procedure, better candidate solutions are evolved by applying certain principles to individuals, such as random mutation. Then, we present the particular algorithms which we want to use for the inversion of geomagnetic data.

## 2.1 Inversion algorithms

The development of geomagnetic core field models requires the inversion of magnetic field data as described in section 1.1. A first approach is to solve the linear equation system (1.8) in a least-squares sense, i.e. to minimize the corresponding $L_2$ norm of the residuals. Therefore, we seek for a minimum norm solution $m_{L_2}$ which fulfills

$$m_{L_2} = \underset{m \in \mathbb{R}^k}{\arg\min} \|Gm - d\|_2^2. \tag{2.1}$$

Due to the orthogonality of the spherical harmonic functions, $G$ has linearly independent columns. Hence, there exists a unique solution $m_{L_2}$ to equation (2.1) which is given by the normal equations (Aster, Borchers, and Thurber, 2019)

$$m_{L_2} = (G^\intercal G)^{-1} G^\intercal d. \tag{2.2}$$

However, due to numerical instabilities, the calculation of the inverse matrix in equation (2.2) is avoided in practical calculations.

Furthermore, it is desirable to use other norms than the $L_2$ norm as a least-squares fit is in general very sensitive to non-normally distributed errors. By using the $L_2$ norm, errors enter quadratically into the result. The usage of norms as the $L_1$ norm can help to circumvent this problem. There are also more sophisticated norms which can be used, for instance the already mentioned (modified) Huber norm is a mixture of the $L_1$ norm and the $L_2$ norm. However, the $L_2$ norm is widely used since there exists a closed-form solution of the least-squares problem, given by equation (2.2), and since the least-squares solution corresponds to the maximum-likelihood solution, if the errors are assumed to be normally distributed. The normal distribution arises frequently due to the central limit theorem.

As described in subsection 1.2.2, applying regularization techniques is often necessary. Therefore, the problem is often reformulated as an optimization problem. This

is done since it is possible to obtain every geomagnetic field model by minimizing a certain cost function (Gubbins and Herrero-Bervera, 2007). Such a cost function always takes the form "error + regularization". For example, consider the inversion with respect to the $L_2$ norm and an additional regularization term $R(m)$. A corresponding cost function $f : \mathbb{R}^k \longrightarrow \mathbb{R}$ can be written as

$$f(m) = \|Gm - d\|_2^2 + \lambda R(m), \tag{2.3}$$

where $\lambda$ is a regularization parameter which has to be chosen. The regularization term $R(m)$ could, for example, force the solution to approach a given power spectrum at a given altitude. Solving the inverse problem is equivalent to finding a global minimum of the function $f$.

There exists a variety of popular algorithms for numerically calculating a minimum of a given function. Many of these algorithms rely on using derivatives, i.e. they use the gradient of a (sufficiently smooth) function. However, derivative-based methods often only find local minima and strongly depend on an initial guess (e.g. Grayver and Kuvshinov (2016)). This is one reason why we are not interested in using such a method for our optimization problem.

Moreover, since the $L_1$ norm of a vector $x = (x_1, \ldots, x_n)^\intercal$ is given by

$$\|x\|_1 = \sum_{i=1}^{n} |x_i|, \tag{2.4}$$

it is clear that a corresponding objective function $f$ becomes non-differentiable in general, such that derivative-based methods are not suitable to tackle such problems. Although there are approaches to circumvent this problem (Borsic and Adler, 2012), such methods often influence the convergence rate of the algorithm and need more computational resources (Grayver and Kuvshinov, 2016).

We are therefore interested to apply stochastic optimization methods for finding the minimum-norm solution and solving the inverse problem (Sen and Stoffa, 2013). Such methods can be considered as global optimization methods. They are also suitable for non-differentiable and non-convex functions. Some established and well-known algorithms like Particle Swarm Optimization exist, which iteratively calculate a global minimum without using derivatives. Here, we would like to use another stochastic optimization method called Covariance Matrix Adaptation Evolution Strategy (CMAES). Besides the fact that the method can be used for non-smooth functions, it can also be applied to ill-posed problems, which are often encountered in geophysics. CMAES was shown to exceed other established stochastic algorithms such as Genetic Algorithms and to reveal extraordinary robustness on ill-conditioned problems (Auger et al., 2009; Hansen, Ros, et al., 2011). Despite these convincing results, the usage of CMAES in geophysics is still not common. For that reason, our aim is to investigate the usability of the algorithm for magnetic field modeling.

$$x, y \in \mathbb{R}^n \Longrightarrow \begin{array}{l} f(x) < f(y)? \\ f(x) \geq f(y)? \end{array}$$

Figure 2.1: Comparison-based optimization. In the black-box scenario, the optimization algorithm is only allowed to use function value comparisons.

## 2.2 Random optimization

Given a function

$$f : \mathbb{R}^n \longrightarrow \mathbb{R}, \ x \longmapsto f(x), \tag{2.5}$$

a point $x_0 \in \mathbb{R}^n$ is called global minimum of $f$, if

$$f(x_0) \leq f(x) \quad \text{for all } x \in \mathbb{R}^n \tag{2.6}$$

holds. If there is an $\varepsilon > 0$ such that the inequality in equation (2.6) is only true for all $x$ within the $\varepsilon$-environment $U_\varepsilon(x_0) \subseteq \mathbb{R}^n$, then $x_0$ is called a local minimum. The terms global maximum and local maximum are defined accordingly. We remark that the formulation of a minimization problem is equivalent to the formulation of a maximization problem, since $f$ has a (local, global) minimum at $x_0$ if and only if $g = -f$ has a (local, global) maximum at $x_0$.

The function $f$ is also called objective function or fitness function. Our objective is to find a point $x \in \mathbb{R}^n$ such that the corresponding function value $f(x)$ becomes as small as possible. We explicitly note that in general this will not mean that we are searching for a global minimum of $f$, since we often cannot assure the existence of such a minimum. However, there are functions from which we know in advance that they have a global minimum. An example is the least-squares optimization problem, where $f(m) = \|Gm - d\|_2^2$, and $\|Gm - d\|_2$ corresponds to the $L_2$ norm of the residuals (with the notation as in chapter 1). If $G$ has linearly independent columns, then there exists a unique least-squares solution $m_{L_2}$ which corresponds to the global minimum of $f$. The convexity of the least-squares optimization function also guarantees that there is no local minimum.

Here, we will mostly consider black box optimization scenarios. This term refers to the situation that the gradient of the objective function is not available, for example if $f$ is not differentiable. In such cases the function values $f(x)$ can be regarded as the only available information on $f$, and comparison-based algorithms are used to simply compare the function values (see figure 2.1). Due to this situation, the costs of corresponding optimization algorithms are mostly measured by the number of required function calls. Even more, we do not constrain $f$ by any other assumptions, such that $f$ may be non-convex, discontinuous, non-differentiable, or ill-conditioned.

Now, in order to select good candidates $x$ for function evaluation, a widely applied number of algorithms is based on stochastic search methods. This is also referred to as randomized black box search and is well-suited in order to search for minima in a rugged search landscape (see algorithm 1). A basic random optimization algorithm first initializes the distribution parameters and a candidate solution. Then the probability

distribution is used in order to sample a new candidate solution. The two candidate solutions are compared by their function value. The point which corresponds to the lower function value is set to be the candidate solution for the next iteration, and the distribution parameters get updated.

---

**Algorithm 1:** Outline of a basic random optimization algorithm

Initialize distribution parameters $\theta$
Initialize a candidate solution $x \in \mathbb{R}^n$
**repeat**
    Sample a new candidate solution $y \in \mathbb{R}^n$ from the distribution $p_\theta$
    Evaluate $y$ on $f$; if $f(y) < f(x)$, set $x = y$
    Update the distribution parameters $\theta \leftarrow F(\theta, (x, f(x)), (y, f(y)))$
**until** *Termination criterion fulfilled*;

---

It is also possible to select multiple random candidate solutions $x_1, \ldots, x_\lambda$ from the probability distribution. These $\lambda$ values are then evaluated on $f$, and get sorted by the corresponding function values. The number $\lambda$ is usually called population size (note that $\lambda$ referred to the regularization parameter in chapter 1).

## 2.3  Evolution Strategies

CMAES is a specific Evolution Strategy. Evolution Strategies are methods which are inspired by natural evolution processes and they are applied for deriving new candidate solutions of the optimization problem. Charles Darwin was a famous naturalist who defined *Natural Selection* as the *preservation of favorable individual differences and variations, and the destruction of those that are injurious*, see also Coello Coello (2005). In this context, the term *evolution* refers to a process in which individuals try to survive with an adaptation to their environment. The idea of an Evolution Strategy is to numerically simulate such a process. This is done iteratively, where the iterations can be thought of as generations.

In order to underline the idea of Evolution Strategies, we consider a closed natural environment which is located in a huge box. Suppose that within this environment a family of individuals of a certain species lives. Although these individuals are of the same species, it is clear that they can have some different properties, for example they can be distinguished by their height, by their weight, or by the location where they live (see also Scherrmann (2020)). If the search space is $\mathbb{R}^n$, individuals can be thought of as vectors $x \in \mathbb{R}^n$ together with their associated fitness values $f(x)$, where $f$ is the objective function. Now, the individuals with the best properties for the given environment, i.e. fitness value, will survive. These properties should be inherited to later generations in order to improve their ability to survive in the environment. The characteristics of the individuals can be imagined as their coordinates $x_1, \ldots, x_n \in \mathbb{R}$, if $x = (x_1, \ldots, x_n)^\intercal$. In order to generate new individuals from former ones, natural processes such as recombination, mutation and selection are applied (see algorithm

2). Due to the black-box scenario, we will later see that some of these processes are implemented using probabilistic approaches.

---

**Algorithm 2:** General outline of an Evolutionary Algorithm (Algorithm 2.1 from Bäck, Foussette, and Krause (2013))

---

Initialization
**repeat**
 Recombination
 Mutation
 Evaluation
 Selection
**until** *Termination criterion fulfilled*;

---

Evolution Strategies refer to a specific implementation of the recombination, mutation, evaluation, and selection steps. For example, the mutation operator may be parameterized such that it can be adapted during the optimization process (consider for example the phase of approaching the optimum). More such requirements can be found in Rudolph (2012).

We continue with an explanation of recombination, mutation and selection, the three core ingredients of an Evolution Strategy. Thereby we will use the material which is contained in Bäck and Hoffmeister (1994), Bäck, Foussette, and Krause (2013), Hansen, Arnold, and Auger (2015), Hansen (2016), and Rudolph (2012).

## 2.3.1 Recombination

Recombination describes the process of passing characteristics of several parents (also more than two) to their children (offspring). In biology, recombination processes are in general very difficult to understand, if possible at all. For Evolution Strategies, the principles are simplified. Here, we will describe some of the most common recombination operators.

The easiest principle is discrete recombination. Suppose that there are $\rho$ parents, out of which two are randomly selected. For each component of the $x$-vector (such a component is also referred to as variable) one parent is chosen with probability 1/2 and the variable of this parent is copied to the offspring individual. There is also a global version of this type of recombination, where for each single variable one parent is chosen randomly from the parent population.

A slightly more complex possibility is called intermediate recombination. First, two parents are randomly chosen. Then the average value of their two variables is assigned to the offspring individuals' variable. Here, for all new offspring variables, the same parents are chosen, such that only two parents are involved within the recombination process. This method can also be extended when one takes the average value of all $\rho$ parents' attributes.

An even more sophisticated possibility is to take a weighted mean of the $\rho$ parents' variables, and this method is called weighted recombination. The weights may, for

example, depend on the fitness ranking. It is also possible to generalize the weight selection by taking the weights randomly.

All possible recombination procedures have in common that the possible values for the new individuals are located and limited within an area which is spanned by the properties of the parents. In general, the optimum which describes the best suitable family of variables does not have to be located within this area. Therefore, the concept of mutation was introduced.

## 2.3.2 Mutation

In biology, the term mutation describes random changes occurring in the DNA. These changes can have various reasons, such as UV radiation. Mutation can result in individuals with very distinct and different characteristics.

Mutation in Evolution Strategies is described by a random change of parameters that is applied after the recombination process. This random change makes it possible to leave the area which is spanned by the parents' properties. In detail, a random number is added to each variable, such that

$$x^{(g+1)} = x_{\mathrm{rec}}^{(g+1)} + r^{(g)}, \qquad r^{(g)} = (r_1^{(g)}, \dots, r_n^{(g)})^{\intercal}, \tag{2.7}$$

where $x_{\mathrm{rec}}^{(g+1)}$ is the offspring after recombination, $r_i^{(g)}$ is a random mutation of the $i$-th characteristic and the uppercase $(g)$ denotes the generation $g$. Equation (2.7) can be thought of as a perturbation of $x_{\mathrm{rec}}^{(g+1)}$ by $r^{(g)}$.

However, the random changes which are described by equation (2.7) should not be completely arbitrary. Therefore, Rudolph (2012) defines three guiding principles for mutation, namely

- Reachability: Every point of the search space should be attainable with a probability strictly larger than zero after applying the mutation operator finitely many times. This allows to test every possible parameter set to be the optimal set.

- Unbiasedness: The operator should treat all possible solutions as equal, without preferences. This can be achieved by using a maximum entropy distribution.

- Control: The shape of the distribution should be adjustable by parameters of the variation operator. This allows to vary the mutation operator at all times, particularly when approaching the minimum.

In $\mathbb{R}^n$, the multivariate normal distribution fulfills all of the three requirements, and is also used in CMAES. Therefore, we stop a moment and review some mathematical facts which will be needed in order to understand the sampling procedure better.

### The Eigendecomposition of a positive definite matrix

We recall that a symmetric matrix $C \in \mathbb{R}^{n \times n}$ is called positive definite if $x^{\intercal} C x > 0$ for all $x \in \mathbb{R}^n \setminus \{0\}$. It can be proven that this is the case if and only if all eigenvalues

$\lambda_1, \ldots, \lambda_n$ of $C$ are strictly larger than zero. Hence, the eigenvalues can be written as $d_1^2, \ldots, d_n^2$ with $d_i = \sqrt{\lambda_i}$. Furthermore, it is a well-known fact that there exists an orthonormal basis of $\mathbb{R}^n$ which consists of eigenvectors of $C$. This leads to the existence of an eigenvalue decomposition

$$C = UD^2U^\intercal, \tag{2.8}$$

where $U \in \mathbb{R}^{n \times n}$ is an orthogonal matrix ($U^{-1} = U^\intercal$, and the columns of $U$ are the eigenvectors of $C$) and $D^2 = \mathrm{diag}(d_1^2, \ldots, d_n^2)$ is a diagonal matrix which contains the eigenvalues of $C$ as diagonal elements.

The eigendecomposition in equation (2.8) of $C$ leads to two important consequences which will be required in the following considerations. First, we can directly obtain an expression for the inverse matrix of $C$. By means of equation (2.8) and the orthogonality of $U$ it follows that

$$C^{-1} = UD^{-2}U^\intercal, \tag{2.9}$$

with $D^{-2} = \mathrm{diag}(1/d_1^2, \ldots, 1/d_n^2)$. Second, an expression for the square root of $C$ is also obtained automatically. Recall that a symmetric positive definite matrix $B$ is called square root of $C$, if $B^2 = C$ holds. We will write $B = C^{1/2}$. This notation is not ambiguous since the assumptions on $B$ and $C$ imply that there exists exactly one such $B$ which is given by

$$C^{1/2} = UDU^\intercal, \tag{2.10}$$

with $D = \mathrm{diag}(d_1, \ldots, d_n)$. A straightforward consequence is that

$$C^{-1/2} = UD^{-1}U^\intercal. \tag{2.11}$$

### The Multivariate Normal Distribution

Let $X$ be an $n$-dimensional random vector. We say that $X$ is multivariate normally distributed with expectation $m \in \mathbb{R}^n$ and positive definite covariance matrix $C \in \mathbb{R}^{n \times n}$ if $X$ has a probability density function of the form

$$f_X(x) = \frac{1}{(2\pi)^{n/2}|C|^{1/2}} \exp\left(-\frac{1}{2}(x-m)^\intercal C^{-1}(x-m)\right), \tag{2.12}$$

where we denote $|C| = \det C$. In short, we will write $X \sim N(m, C)$. An important property of $N(m, C)$ is that it can be expressed in terms of the standard multivariate normal distribution $N(0, I)$ by

$$N(m, C) \sim m + N(0, C), \tag{2.13}$$

where "$\sim$" denotes equality in distribution. Furthermore, $N(0, C)$ fulfills (Tong, 1990, Theorem 3.3.2)

$$N(0, C) \sim C^{1/2}N(0, I). \tag{2.14}$$

The usage of $N(0, I)$ is desirable by several reasons. First, $N(0, I)$ generates an isotropic distribution with the shape of a sphere. This distribution is invariant under rotations around its mean. More generally, if we consider the contour lines of the probability density function, given by equation (2.12), for an arbitrary positive definite symmetric covariance matrix $C$, then we can identify $C$ with the (hyper-)ellipsoid $\{x \in \mathbb{R}^n : x^\intercal C^{-1} x = 1\}$, which is a surface of equal density of the distribution. Second, $N(0, I)$ is a vector which has independent $(0, 1)$-normally distributed numbers as coordinates. Such a distribution can be easily realized in computer programs.

### Different types of mutation operators

The perturbation in equation (2.7) shall be point symmetric and is sampled from a multivariate normal distribution $N(0, C)$ with zero mean and covariance matrix $C$. Since $a + N(0, C) \sim N(a, C)$ holds for all $a \in \mathbb{R}^n$, it follows that $a$ is a representative of the expected value of the new individual. Hence, we can rewrite equation (2.7) as

$$x \sim m + N(0, C), \tag{2.15}$$

where $x$ is a new search point and $m$ is the mean vector which represents the favorite solution and is drawn from the recombination procedure. Different mutation operators can be constructed depending on the choice of $C$. In order to better understand the sampling, we rewrite $N(0, C)$ (c.f. equation (2.15)) by using equation (2.14) such that

$$x \sim m + C^{1/2} N(0, I). \tag{2.16}$$

Now, plugging equation (2.10) into equation (2.16) implies that

$$x \sim m + U D U^\intercal N(0, I). \tag{2.17}$$

Since $U$ is orthogonal, we have $U^\intercal N(0, I) \sim N(0, I)$. Hence, our final equation for the sampling becomes

$$x \sim m + U D N(0, I) \tag{2.18}$$

for the individuals with the distribution

$$N(m, C) \sim m + U D N(0, I). \tag{2.19}$$

The term $U D N(0, I)$ offers a good possibility for a geometrical explanation of the realization of the perturbation distribution. First of all, $N(0, I)$ initializes a spherical distribution as in figure 2.2, left. Within figure 2.2, the blue points represent the individuals, and the green lines depict 1-$\sigma$ and 2-$\sigma$ lines of the corresponding distributions. By applying the diagonal matrix $D$, the spherical distribution gets scaled as shown in figure 2.2, middle. Finally, the matrix $U$ acts as a rotation (figure 2.2, right).
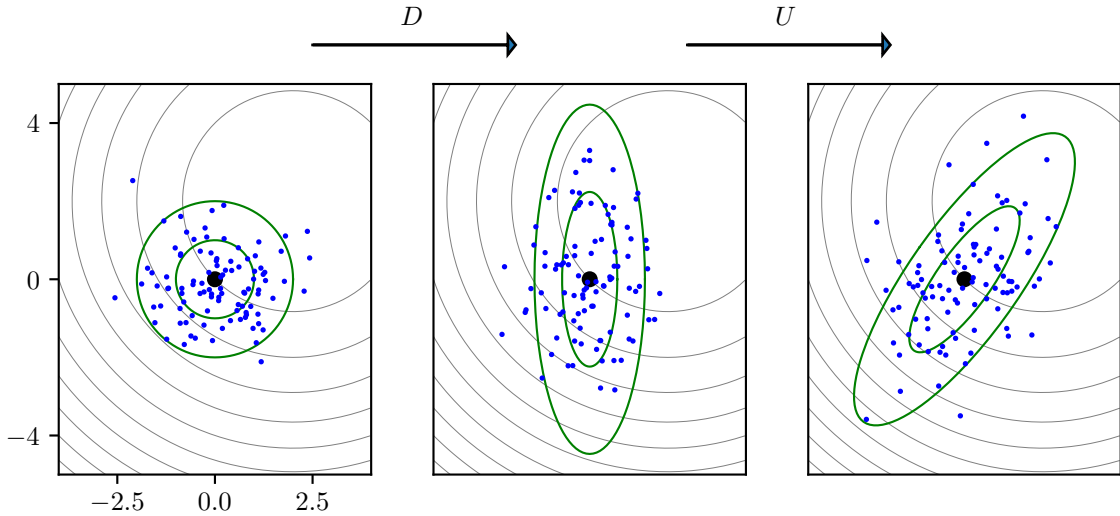
Figure 2.2: Two-dimensional multivariate normal distributions $N(0, C) \sim C^{1/2} N(0, I)$. From left to right, the covariance matrix $C$ of the distribution is $I$, the diagonal matrix $\mathrm{diag}(1/2, 5)$ and the matrix $\begin{pmatrix} 2 & \sqrt{18}/2 \\ \sqrt{18}/2 & 7/2 \end{pmatrix}$ with the same eigenvalues as the diagonal matrix. In each figure the mean of the distribution is shown as a black dot at 0 and the green curves show the 1-$\sigma$ and 2-$\sigma$ lines of equal density. The blue dots represent a sample of 100 points, where it is possible that some of these points lie outside the shown area. The thin grey lines depict contour lines of a possible objective function. Here, the objective function was chosen to be $f(x, y) = (x-2)^2 + (y-2)^2$. The figure is inspired by figure 1 in Hansen, Arnold, and Auger (2015).

## 2.3.3 Selection

Finally, we address the question of how individuals of one parent generation are chosen for the recombination process. This process is called selection and it works completely deterministic. Two common approaches will be explained, the $(\mu, \lambda)$-selection ("comma"-selection) and the $(\mu + \lambda)$-selection ("plus"-selection).

We assume that the parent population contains $\mu$ individuals and that we select $\lambda$ individuals as parents for the next generation. Furthermore, we consider in general the case $\mu \leq \lambda$. The both methods differ in whether the individuals' age is taken into account. Firstly, suppose that the individuals die out after one iteration step. If so, the $\mu$ most suitable individuals out of $\lambda$ offspring are taken. Hence, only the youngest individuals are considered. Secondly, suppose that also previous parents ("grand-parents") are considered. Here, the new parents are chosen from $(\mu + \lambda)$ individuals. In this case, the parents can be regarded as the all-time best individuals.

The performance of both approaches in general varies depending on the problem. Often, $(\mu, \lambda)$-selection is recommended to use since the $(\mu + \lambda)$-selection is not suitable for the application to non-stationary environments (see Bäck and Hoffmeister (1994)).

We remark that sometimes the notation as $(\mu/\rho, \lambda)$ can be encountered. This in-

dicates that $\rho \leq \mu$ parent individuals (out of $\mu$) are used in the recombination step. Additionally, there are sometimes subscripts to $\rho$ used for denoting the type of recombination, for example $\rho_I$ or $\rho_W$ for intermediate or weighted recombination, respectively.

Finally, we remark that $\lambda$ should be chosen accordingly to the optimization situation. Smaller values lead in general to a faster convergence, whereas larger values help to avoid local optima. A default value which is often used is $\lambda = 4 + \lfloor 3 \ln(n) \rfloor$, where $n$ is the problem dimension. A typical choice for $\mu$ is $\mu = \lfloor \lambda/2 \rfloor$.

### 2.3.4 Advantages and disadvantages of Evolution Strategies

We continue with mentioning some upsides and downsides of Evolution Strategies (see also Fogel (1997)).

One very remarkable point is that such algorithms can be applied to black-box optimization scenarios in which no additional knowledge on the topology of the problem is available. Speaking in terms of functions, for example no information about derivatives is available. Therefore, Evolution Strategies can be applied to a large class of (difficult) problems. This also includes the case where it is unknown whether the problem has a solution. Another positive point is that due to the simplicity of the above steps mutation, recombination and selection (see also algorithm 2), the optimization process can be parallelized up to the selection process. Even more, the parallelization will scale almost linearly with the number of available cores.

On a downside, it is possible that there are many function evaluations needed for solving a problem. This is often the case if the number of variables increases. Furthermore, the running time of such an algorithm suffers significantly if the cost of function evaluation is high. If we consider for example a function whose evaluation depends in some sense on a dot product between a large matrix $G$ and a vector $m$, then the matrix multiplication can take some time. Therefore, the function calls have to be implemented in a way such that the evaluation is very efficient. We discuss this issue in more detail in subsection 3.1.3.

## 2.4 The CMAES algorithm

The CMAES algorithm is a particular Evolution Strategy which can be regarded as a very popular and very efficient one. This is achieved by two special features: Covariance Matrix Adaptation (CMA) and Cumulative Step-Size Adaptation (CSA). In this section, we give an overview about the most important ingredients. However, we will not give very detailed mathematical derivations and motivations for some formulas. For details, we refer to Hansen (2016).

In view of our considerations in subsection 2.3.2, new search points $x_i \in \mathbb{R}^n$ are sampled by usage of a normal distribution such that

$$x_i \sim m + \sigma N_i(0, C) \qquad \text{for } i = 1, \ldots, \lambda, \tag{2.20}$$

where the roles of the expectation value $m$ and the covariance matrix $C$ are analog to equation (2.15) and $\sigma$ is a global step size which controls the step length.

In order to explain the principles of the algorithm, we will discuss how the parameters $m, C$ and $\sigma$ are updated.

## 2.4.1 The selection and recombination procedure

In what follows and for better readability we will occasionally omit the subscripts which indicate the generation. In principle, CMAES uses a slightly modified version of the $(\mu, \lambda)$-selection algorithm (see also subsection 2.3.3). Let $x_1, \ldots, x_\lambda$ be a sample obtained from equation (2.20), such that there are $\lambda$ offspring points. Then a number $\mu > 1$ is chosen such that $\mu \leq \lambda$, and $\mu$ is the number of selected individuals (see also figure 2.3). Furthermore, let $x_{i:\lambda}$ denote the $i$-th best individual out of $x_1, \ldots, x_\lambda$ such that $f(x_{1:\lambda}) \leq f(x_{2:\lambda}) \leq \cdots \leq f(x_{\lambda:\lambda})$. Then the new mean $m$ is calculated by means of a weighted average with

$$m = \sum_{i=1}^{\mu} w_i x_{i:\lambda}, \tag{2.21}$$

where the $w_{i=1,\ldots,\mu} > 0$ are weights such that

$$\sum_{i=1}^{\mu} w_i = 1 \qquad \text{and} \qquad w_1 \geq w_2 \geq \cdots \geq w_\mu > 0. \tag{2.22}$$

We give several comments on this:

- Equation (2.21) is an application of weighted intermediate recombination. Furthermore, a $(\mu, \lambda)$-selection is used and the usage of different weights $w_i$ can also be regarded as part of the selection. In CMAES-related algorithms, $\mu \approx \lambda/2$ is a frequently applied choice.

- The multiplication of the decreasing weights with the sequence of $i$-th best ranked search points favors better solutions and downweights worse solutions.

- A possible choice for the $w_i$ is $w_i = w_i' / \sum w_i'$, where

$$w_i' = \ln\left(\frac{\lambda + 1}{2}\right) - \ln(i), \qquad i = 1, \ldots, \mu. \tag{2.23}$$

In view of equation (2.21), it is desirable to rewrite the equation as an update of $m$. This is usually done by considering the assignment

$$m \leftarrow m + c_{\mathrm{m}} \sum_{i=1}^{\mu} w_i (x_{i:\lambda} - m). \tag{2.24}$$

In equation (2.24), the number $c_{\mathrm{m}}$ is a learning rate which is usually set to 1, and was also set to 1 in our calculations. The choice of $c_{\mathrm{m}}$ can for example depend on the noisyness of the objective function. A usage of $c_{\mathrm{m}} < 1$ results in a decrease of the distance between the old mean vector and the new one. For further details concerning the choice of $c_{\mathrm{m}}$ we refer to Miyazawa and Akimoto (2017).
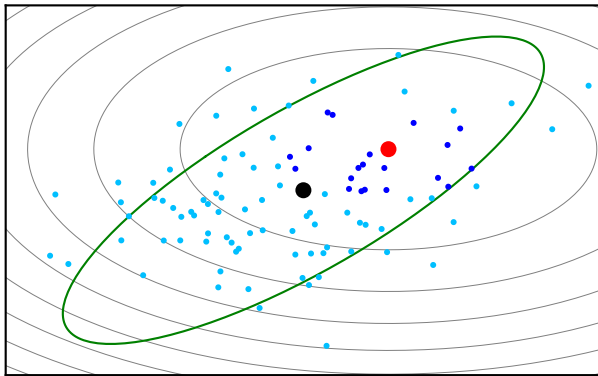
Figure 2.3: For recombination, $\mu \leq \lambda$ parents are chosen from the $\lambda$ offspring points. The generation contains $\lambda = 100$ individuals. Here, the $\mu = 20$ parents are depicted by dark blue points, which are closest to the optimum (red point), and remaining individuals are shown in light blue.

## 2.4.2 The update of the covariance matrix

We continue with a description of the update of the covariance matrix. The covariance matrix adaptation aims to approximate the contour lines of the fitness function $f$. The derivation of the update is a long and winding road which contains a lot of technical considerations which are not particularly relevant within the scope of this work. Therefore, we restrict ourselves to a short overview of its main ingredients. Details can for example be found in Hansen (2016).

We start with evolution paths, which describe former consecutive steps of particular search strategy parameters. The idea behind them is to realize an accumulation of information over several generations, allowing to increase the likelihood of formerly successful steps. In CMAES, two evolution paths are used, namely $p_c$ for the update of the covariance matrix and $p_\sigma$ for the update of the global step size (for $p_\sigma$ see subsection 2.4.3). Starting with $p_c = 0$ for the first generation, the path $p_c$ is defined recursively by

$$p'_c = (1 - c_c)p_c + \sqrt{c_c(2 - c_c)\mu_{\text{eff}}} \, \frac{m' - m}{\sigma}. \tag{2.25}$$

We comment on the meaning of variables in equation (2.25).

- $(1 - c_c)$ is a decay factor which regulates the influence of the former path.

- $\sqrt{c_c(2 - c_c)\mu_{\text{eff}}}$ is a normalization factor and the number $\mu_{\text{eff}} = 1/\sum_{i=1}^{\mu} w_i^2$ is called "variance effective selection mass" (Hansen, 2016).

- $(m' - m)/\sigma$ can be considered as new input, since the concept of the evolution path relies on a sum of consecutive steps of the mean $m$. Primes indicate the variables of the current generation.

The evolution paths $p_c$ are used for the so-called rank-one-update of the covariance matrix, such that

$$C' = (1 - c_c)C + c_c p_c p_c^\intercal. \tag{2.26}$$

It can be shown that this update can reduce the number of function evaluations to adapt to a straight ridge from $O(n^2)$ to $O(n)$ (Hansen, Müller, and Koumoutsakos, 2003).

There is another core ingredient to the final update. Recall that the new search points $x_i$ are sampled by $x_i = m + \sigma y_i$ with $y_i \sim N_i(0, C)$. The outer product of the vectors $y_i$ can be taken with a weighted mean in order to construct the matrix

$$C_\mu = \sum_{i=1}^{\mu} w_i y_{i:\lambda} y_{i:\lambda}^\mathsf{T}. \tag{2.27}$$

Let $n$ be the dimension of the search space. The matrix $C_\mu$ in equation (2.27) has rank $\min\{\mu, n\}$ almost surely (if $\mu$ weights are non-zero; it is theoretically possible to get a matrix with lower rank by a sample of the distribution) and is used for the rank-$\mu$-update

$$C' = (1 - c_\text{c})C + c_\text{c} C_\mu. \tag{2.28}$$

This update can be regarded as an extension for the update rule for large population sizes and can reduce the number of necessary generations approximately from $O(n^2)$ to $O(n)$ (Hansen, Müller, and Koumoutsakos, 2003).

The final update of the covariance matrix combines the rank-one-update and the rank-$\mu$-update and is given by

$$C' = (1 - c_l - c_\mu)C + c_l p_c p_c^\mathsf{T} + c_\mu \sum_{i=1}^{\mu} w_i y_{i:\lambda} y_{i:\lambda}^\mathsf{T}, \tag{2.29}$$

with decay factors $c_l$ and $c_\mu$. The update feeds the new covariance matrix in every iteration with the data of the most successful preceding steps such that the likelihood of similar steps increases.

## 2.4.3 The update of the step-size

The considerations in the preceding subsection showed that the covariance matrix adaptation regulates the search direction for selected steps. However, it is still desirable to have a control mechanism of the global step size. For example, the update in equation (2.29) cannot realize a dependency of the step size $\sigma$ on $\mu$, as is required for some situations (Beyer, 2001).

There are several possibilities for step-size adaptation which can be used in Evolution Strategies. A classical example is the 1/5-th success rule (Rechenberg, 1973), where the decision on increasing or decreasing the step-size is made by checking whether more than 20% of the new solutions are successful.

CMAES uses a technique called cumulative step length adaptation (CSA). This technique relies on evolution paths like in subsection 2.4.2. In order to understand the approach we consider figure 2.4, which is inspired by figure 1.1 in Hansen (1998). Figure 2.4 depicts three evolution paths which are composed by five intermediate steps with
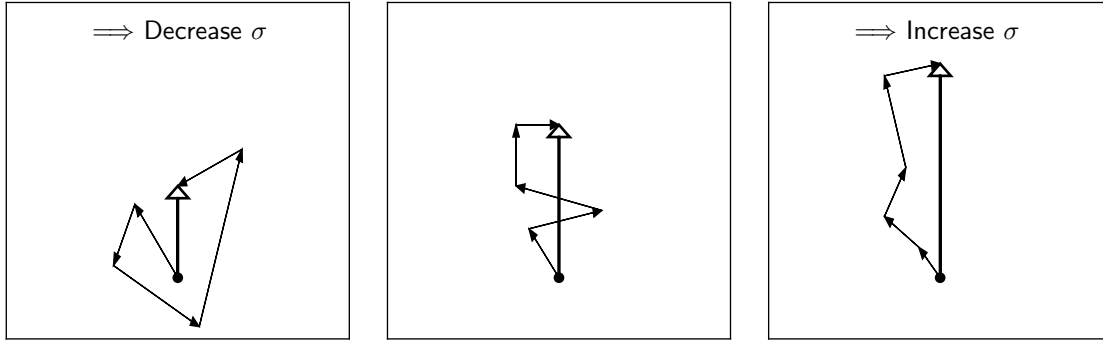
Figure 2.4: Evolution paths. The three evolution paths differ in their total length. Left: The step size should be decreased since the intermediate steps cancel out each other. As a result, the evolution path is short. Right: The step size should be increased since it is possible to cover the distance by fewer but longer steps into the same direction. The evolution path is long. Middle: The desired situation. Here, the intermediate steps can be considered as perpendicular in expectation. The length of the evolution path is in some sense "optimal".

approximately identical step lengths. The left path corresponds to a situation where the step size should be decreased, the path in the middle shows the optimal situation and the path on the right depicts a situation where the step size should be increased. In an optimal situation, the length of an evolution path should be equal to its expected length under random selection. However, it is usually biased by the selection process. Hence, we will adapt the length of an evolution path by comparing with its expected length $E\|N(0, I)\|$ (where $E$ denotes the expectation value), as explained below. If the path is longer than expected, the step size should be increased, and if the path is shorter than expected, the step size should be decreased.

The used evolution path $p_\sigma$ is constructed similar to the one in equation (2.25) and is given by

$$p'_\sigma = (1 - c_\sigma)p_\sigma + \sqrt{c_\sigma(2 - c_\sigma)\mu_{\text{eff}}} \; C^{-1/2} \; \frac{m' - m}{\sigma}, \tag{2.30}$$

with notation analogously as in equation (2.25) and $C^{-1/2}$ given by equation (2.11). It is known that the expected length of $p_\sigma$ in general depends on its direction. This is avoided by using $C^{-1/2}$, making the expected length independent of the direction of the evolution path. Furthermore, given $p_\sigma^{(0)} \sim N(0, I)$, one can assume that $p_\sigma^{(g+1)} \sim N(0, I)$ holds for all generations $g$ (Hansen, 2016). In order to update the step size $\sigma$, the number $\|p_\sigma^{(g+1)}\|$ is compared with its expected length $E\|N(0, I)\|$ by

$$\sigma^{(g+1)} = \sigma^{(g)} \exp\left(\frac{c_\sigma}{d_\sigma}\left(\frac{\|p_\sigma^{(g+1)}\|}{E\|N(0, I)\|} - 1\right)\right), \tag{2.31}$$

with scaling parameters $c_\sigma < 1$ and $d_\sigma \approx 1$. The constant $d_\sigma$ is a damping parameter, similar to the learning rate in equation (2.24). Equation (2.31) shows that if $\|p_\sigma^{(g+1)}\| > E\|N(0, I)\|$, the step size will be increased, and decreased otherwise.

The number $E\|N(0, I)\|$ can be expressed by the Gamma function $\Gamma$. In numerical calculations, equation (2.31) is evaluated by means of the approximation

$$E\|N(0, I)\| = \sqrt{2}\Gamma\left(\frac{n+1}{2}\right) \Big/ \Gamma\left(\frac{n}{2}\right) \approx \sqrt{n}\left(1 - \frac{1}{4n} + \frac{1}{21n^2}\right), \qquad (2.32)$$

with $n$ equal to the search space dimension.

Figure 2.5 depicts a summary of the CMAES algorithm, where the update of the covariance matrix and the sampling of the search space are illustrated.

## 2.5 Remarks on large-scale optimization

Typically, geophysical inversion is done with respect to a large number of parameters. Consider for example our model vector $m$ from chapter 1. There, $m$ was introduced as an element of $\mathbb{R}^k$, with

$$k = L(L + 2)N_\text{b}. \qquad (2.33)$$

A convenient choice for $L$ and $N_\text{b}$ is e.g. given by $(L, N_\text{b}) = (13, 18)$. As a result, the Evolution Strategy reaches an optimum within a search space of dimension 3510.

### 2.5.1 Problems with CMAES

Recall from equations (2.16) and (2.30) that the CMAES algorithm heavily relies on the factorization

$$C = AA^\intercal \qquad (2.34)$$

of the covariance matrix $C$ with a suitable matrix $A$ in order to sample the multivariate normal distribution. The evaluation of $C$ is a core ingredient in order to calculate the evolution path $p_\sigma$ which is given by equation (2.30) and it is needed for the sampling of new solutions according to equation (2.16). The matrices $A$ and $A^\intercal$ in equation (2.34) are also referred to as Cholesky factors. For the computation of the Cholesky factors, an eigendecomposition of $C$ is needed which a priori has to be calculated every time when a new sample shall be created. Even more, it is possible to show that the eigendecomposition of $C$ is in general $O(n^3)$ in complexity, where $n$ is the dimension of the problem. A method to mitigate this problem was already proposed when the original CMAES was published and is based on the suggestion to perform the eigendecomposition only every $n/10$ generations. This results in a complexity of $O(n^2)$ per function evaluation (Hansen and Ostermeier, 2001). However, even $O(n^2)$ is too high for our problem size and excludes the standard CMAES due to the resulting time costs for the optimization process. This implies our interest in a large-scale variant of CMAES which is suitable for high-dimensional problems.

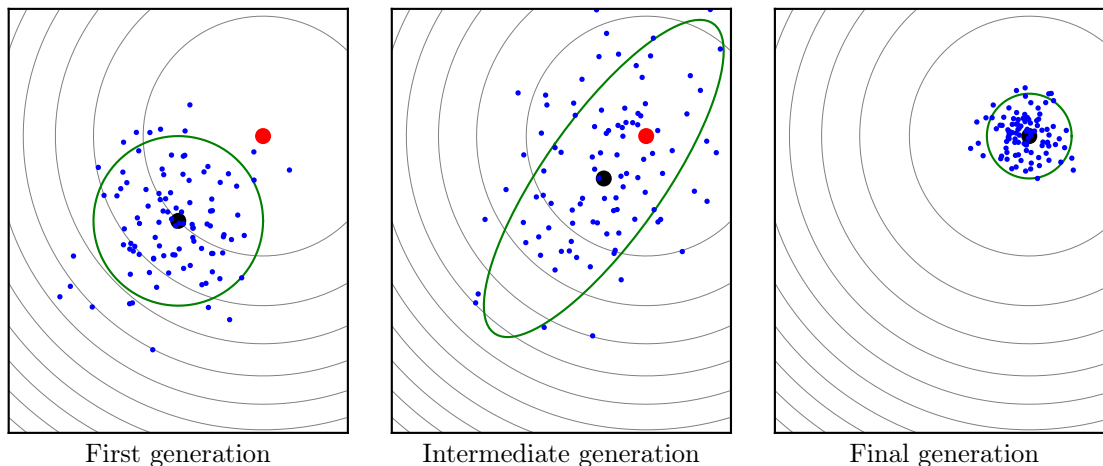First generation          Intermediate generation          Final generation

Figure 2.5: Summary of the CMAES-algorithm. During the optimization process, the parameters get adapted such that new distributions are stretched in directions where better candidate solutions are located. In the final generation, the whole population is located in a very small environment of the optimum, such that the individual with the lowest function value corresponds to the desired approximate solution.

The problem of calculating and storing huge matrices and their factorizations in large-scale optimization is not new. A general approach in order to circumvent such difficulties is to calculate matrix factorizations not directly, but to reconstruct matrices from the last $m$ iterations. This ends up in an iteratively working procedure.

## 2.5.2 LMCMA

The Limited Memory CMAES (LMCMA) is an Evolution Strategy which is very similar to the original CMAES and is highly applicable to high-dimensional problems. Compared to the original CMAES, there are some smaller changes in the algorithm which concern for example the update of the step size. Here, we will focus on the main changes that reduce time and space complexity of the algorithm. For details we refer to Loshchilov (2014).

The key idea is essentially inspired from the Cholesky-CMAES (Suttorp, Hansen, and Igel, 2009). Instead of calculating the factors of $C$ in equation (2.34) directly, they are iteratively computed and updated. More precisely, there exist constants $a$ and $b^{(g)}$ such that

$$A^{(g+1)} = aA^{(g)} + b^{(g)}p_{\mathrm{c}}^{(g)}v^{(g)\mathsf{T}}, \tag{2.35}$$

with $v^{(g)} = A^{(g)}z^{(g)}$ and $z^{(g)}$ sampled from the normal distribution.

We consider now first the generation $g = 0$. Then, $A^0 = I$ implies that the first updated Cholesky factor is given by

$$A^1 = aI + b^0 p_{\mathrm{c}}^0 v^{0\mathsf{T}} \tag{2.36}$$

and it holds that

$$A^1 z = (aI + b^0 p_{\mathrm{c}}^0 v^{0\mathsf{T}})z = az + b^0 p_{\mathrm{c}}^0 (v^{0\mathsf{T}} z). \tag{2.37}$$

For $g = 1$ it follows similarly that

$$A^2 = a(aI + b^0 p_{\mathrm{c}}^0 v^{0\mathsf{T}}) + b^1 p_{\mathrm{c}}^1 v^{1\mathsf{T}}. \tag{2.38}$$

Continuing this procedure implies the existence of a simple iterative procedure which can be applied to sample candidate solutions in $\mathbb{R}^n$ using the Cholesky factors $A^{(g)}$. Also, similar considerations hold for the second Cholesky factor of $C$. As an advantage, only $m$ pairs of vectors $p_{\mathrm{c}}^{(g)}$ and $v^{(g)}$ have to be stored to reconstruct $A^{(g)}$. This procedure just scales as $O(mn)$.

Finally, we have to address the question of how the number $m$ is chosen. The choice of $m$ can be thought of as a representation of how much information will be reconstructed from the covariance matrix within the iterative procedure. While there can be seen reasonable differences in performance between different choices of $m$ (Loshchilov, 2015), there is a default value of the algorithm which equals the default value of $\lambda$ in CMAES, i.e.

$$m = 4 + \lfloor 3\ln(n) \rfloor . \tag{2.39}$$

Then, for our problem with $n = 3510$, we have $m = 28$, such that we obtain an algorithm with $O(mn) = O(98280)$ complexity instead of $O(n^3) = O(12320100)$.

## 2.5.3 LMMAES

The Limited Memory Matrix Adaptation Evolution Strategy (LMMAES; Loshchilov, Glasmachers, and Beyer (2017)) is the current state of the art of Evolution Strategies for large-scale optimization and is heavily influenced by CMAES and LMCMA. First, it was shown by Beyer and Sendhoff (2017) that the construction of the covariance matrix $C$ in the CMAES algorithm can completely be avoided. Instead of using $C$, only one transformation matrix $M$ is needed which is a representation of $C^{1/2}$. Hence, all CMAES-related equations (e.g. equation (2.30)), which need $C^{1/2}$ and related terms, can be simplified and reduced to terms which depend on $M$ only, although all the explained principles remain the same.

Following Beyer and Sendhoff (2017), we give the idea of the proof of the main update equation. By the definition of $M$, it holds that

$$M^{(g)}(M^{(g)})^{\mathsf{T}} = C^{(g)} \tag{2.40}$$

for all generations $g$. If equation (2.40) is plugged into the update equation (2.29) for $C$, some term manipulations show that $M$ obeys the equation

$$M^{(g+1)}(M^{(g+1)})^{\mathsf{T}} = M^{(g)}\left[ I + c_1\left( p_\sigma^{(g+1)}(p_\sigma^{(g+1)})^{\mathsf{T}} - I \right) + c_\mu\left( \sum_{i=1}^{\mu} w_i y_{i:\lambda}^{(g)} (y_{i:\lambda}^{(g)})^{\mathsf{T}} - I \right) \right]. \tag{2.41}$$

As shown below, this result allows to obtain an approximation formula for $M$ and therefore to avoid complex covariance matrix updates. Equation (2.41) is of the form

$$AA^\intercal = M(I + B)M^\intercal \tag{2.42}$$

with a symmetric matrix $B$ and $A = M^{(g+1)}$. The matrix $A$ will later be used for an approximation of $M$. In order to calculate $A$, we take a power series ansatz

$$A = M \sum_{k=0}^{\infty} \gamma_k B^k \tag{2.43}$$

with a real coefficient sequence $(\gamma_k)$. Inserting equation (2.43) into equation (2.41) shows that

$$A = M \left( I + \frac{1}{2}B - \frac{1}{8}B^2 + \dots \right) \tag{2.44}$$

satisfies equation (2.42) up to the second power in $B$. Without loss of generality, $\|B\|$ can assumed to be small, such that the power series expansion in equation (2.44) can be truncated after the linear term, yielding the approximation

$$M^{(g+1)} = M^{(g)} \left[ I + \frac{c_1}{2} \left( (p_\sigma^{(g+1)}(p_\sigma^{(g+1)})^\intercal - I \right) + \frac{c_\mu}{2} \left( \sum_{i=1}^{\mu} w_i y_{i:\lambda}^{(g)}(y_{i:\lambda}^{(g)})^\intercal - I \right) \right]. \tag{2.45}$$

The simplified version of CMAES without the covariance matrix is referred to as Matrix Adaptation Evolution Strategy. The "Limited Memory"- aspect (Loshchilov, Glasmachers, and Beyer, 2017) for large-scale optimization comes in, for example, since it can be proven that the sampling procedure

$$d_i^{(g)} = M^{(g)} N(0, I) \tag{2.46}$$

does not require the matrix $M$ to be stored. Instead, the storage of the direction vectors $p_\sigma$, which are used in order to construct $M$, is sufficient.

### 2.5.4 The impact of the condition number

In practice, we observed that the convergence speed heavily decreased when the optimum was approached. Thereby, even after several hundred thousand of function calls, the fitness value decreased only very slowly. This problem is not only related to large-scale optimization, but in particular arises within geophysical inverse problems. In this subsection, we investigate the reason.

As explained in chapter 1, our problem is ill-conditioned and therefore, the design matrix $G$ has a high condition number. In order to investigate the influence of the condition number on the convergence of LMMAES, we construct a test problem with smaller matrices where we can directly control the condition number. Then we evaluate the number of needed function calls to obtain a predefined threshold of the fitness value.

Recall that the condition number $\kappa_A$ of a regular matrix $A \in \mathbb{R}^{n \times n}$ is given by

$$\kappa_A = \frac{\sigma_{\max}(A)}{\sigma_{\min}(A)}, \tag{2.47}$$

where $\sigma_{\max}(A)$ and $\sigma_{\min}(A)$ are the largest and smallest singular values of $A$, respectively. We let $a > 0$ and set $A$ to be a diagonal matrix such that

$$A = \operatorname{diag}(\sqrt{a}, \sqrt{a}, \ldots, \sqrt{a}, 1/\sqrt{a}, 1/\sqrt{a}, \ldots, 1/\sqrt{a}) \in \mathbb{R}^{n \times n}, \tag{2.48}$$

where the first $n/2 = 3510/2 = 1755$ diagonal entries are set to $\sqrt{a}$ and the other half is set to $1/\sqrt{a}$. Since $A$ is diagonal, this implies

$$\kappa_A = \sqrt{a}/(1/\sqrt{a}) = a. \tag{2.49}$$

We evaluate how many function calls are needed in order to reduce the function value of the objective function $f(x) := \|Ax\|_2^2$ below the threshold of $10^3$. Please note that an absolute threshold can be used since the LMMAES algorithm adapts the step size such that the objective function is scaled accordingly. This situation can be regarded as a test problem for an $L_2$ minimization of the residual norm of the linear equation system $Ax = 0$, where the right hand side can be set to $0$ without loss of generality. Furthermore, in each run we start the optimization process at the point $(1, 1, \ldots, 1, 1)^\intercal$ and remark that each fitness function $f$ has a global minimum at $0$. The results are shown in figure 2.6, where the number of needed function calls is depicted, depending on the condition number of $A$.
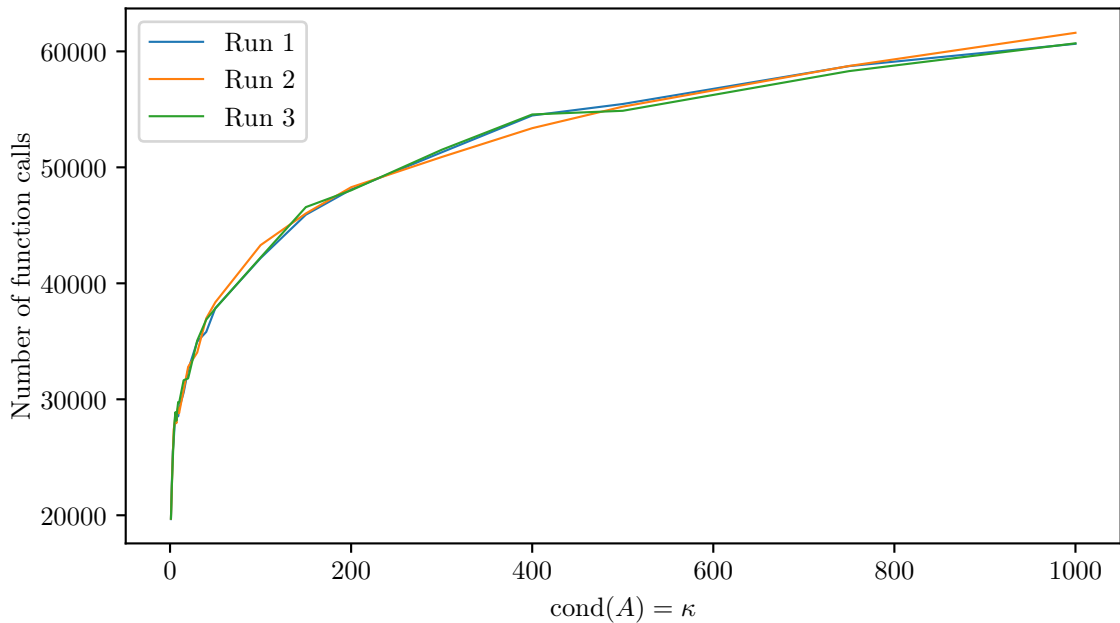


Figure 2.6: Number of needed function calls in order to reduce the residual norm $\|Ax\|_2^2$ below the threshold of $10^3$, depending on the condition number of $A$. Each of the three runs is composed of ten sub-runs, shown is the respective average of these sub-runs. All sub-runs started at the point $(1, 1, \ldots, 1, 1)^\intercal$.

Since the condition number $\kappa_G$ of our forward operator $G$ tends to infinity (see also the remarks in subsection 1.2.2), the results depicted by figure 2.6 explain the large number of needed function calls in order to achieve the optimum. The high condition number implies that small changes in the function argument (which are needed in a small neighborhood of the optimum) often lead to larger changes in the function values, and hence the algorithm needs time in order to adapt the covariance matrix such that it converts the objective function into the sphere function, which is easy to handle. In order to get a faster convergence towards the optimum, a suitable stabilization technique is to regularize $G$ with respect to its singular values. However, such an additional regularization term has to be calculated a priori and is in general very difficult to determine. The reason is that adding such a term corresponds to an additional a-priori assumption, which can lead to an unrealistic model. Moreover, the regularization term can even change the results to physically meaningless ones, if it is not carefully chosen. Hence, instead of regularizing the matrix, it is more suitable to develop an effective implementation of the function calls.

Another regularization technique regarding $G$ is to consider a truncated singular value decomposition. Here, instead of calculating the exact decomposition $G = U\Sigma V^{\mathsf{T}}$, only a truncated decomposition $\tilde{G} = U_t\Sigma_t V_t^{\mathsf{T}}$ is used, where $t$ is the number of column vectors of $U$ and row vectors of $V_t^{\mathsf{T}}$ which correspond to the $t$ largest singular values of $G$. The smaller singular values and the associated rest of the matrix are discarded in order to reduce the influence of smaller singular values. However, the difficulty here is to determine the number $t$. For concrete parameter-choosing strategies and the associated problems we refer to the literature (see e.g. Mendivil et al. (2013)).

We tried to calculate $\kappa_G$ explicitly, but since $G$ consists of several millions of entries, it was computationally not suitable. Therefore, we reduced the data set to one year in order to reduce the size of $G$. Here, we found the largest singular value $\sigma_{\max}$ of $G$ to be approximately $10^6$, but the smallest singular value turned out to be 0. We note that, since $G$ has linearly independent columns, this may result from limited machine precision and is more likely to be a very small number such that $\kappa_G$ becomes very large as predicted. The small singular values result from the spline matrix $G_{\mathrm{s}}$ (see also equation (3.8)), which is needed for the construction of the forward operator matrix $G$. Since for every observation date we need three times the same spline values (separately for the $x$-, the $y$-, and the $z$-component), it follows that $G_{\mathrm{s}}$ has repeated rows.

# 3  Application to core field modeling

We apply the LMMAES algorithm, described in subsection 2.5.3, in order to invert Swarm satellite data. Our resulting model is called the "EvoMag model". The first section gives details about the used Swarm satellite data, about the parameterization of the model and the inversion process. Within the second section, we present the model results and compare EvoMag with other established models such as the IGRF.

## 3.1  Derivation of the EvoMag model

### 3.1.1  Data and data pre-processing

The Swarm satellite mission was launched by the European Space Agency (ESA) on 22nd November 2013 and is still operative. It consists of three identical satellites which orbit the Earth in the ionosphere at altitudes of 462 kilometers up to 510 kilometers. The satellites measure the strength and the orientation of the Earth's magnetic field with the aim to provide a high-quality survey of the geomagnetic field and its temporal evolution. For this purpose, every satellite carries a proton magnetometer which measures the scalar field along with a vector field magnetometer which measures the magnetic field vector itself. In addition, three star tracker cameras are used for the attitude information of the satellite in space.

The measured magnetic field data are provided in the sensor coordinate system. However, for our purpose, we need the data to be given in Earth-references north-east-center coordinates (NEC), within a Cartesian, local and geocentric coordinate system. Here, we will use quaternions for the necessary coordinate transformation. A quaternion

$$q = (q_1, q_2, q_3, q_4) \tag{3.1}$$

is a 4-tuple of real numbers which can be thought of as a representation of the axis of rotation and rotation angle in $\mathbb{R}^3$. Quaternions are usually normed in the sense that $q$ in equation (3.1) fulfills

$$1 = q_1^2 + q_2^2 + q_3^2 + q_4^2. \tag{3.2}$$

In general, the rotation of a point can be described by multiplication with an orthogonal rotation matrix $R_q$. The rotation matrix can be expressed with the help of quaternions as

$$R_q = \begin{pmatrix} 1 - 2(q_2^2 + q_3^2) & 2(q_1 q_2 - q_3 q_4) & 2(q_1 q_3 + q_2 q_4) \\ 2(q_1 q_2 + q_3 q_4) & 1 - 2(q_1^2 + q_3^2) & 2(q_2 q_3 - q_1 q_4) \\ 2(q_1 q_3 - q_2 q_4) & 2(q_2 q_3 + q_1 q_4) & 1 - 2(q_1^2 + q_2^2) \end{pmatrix}. \tag{3.3}$$
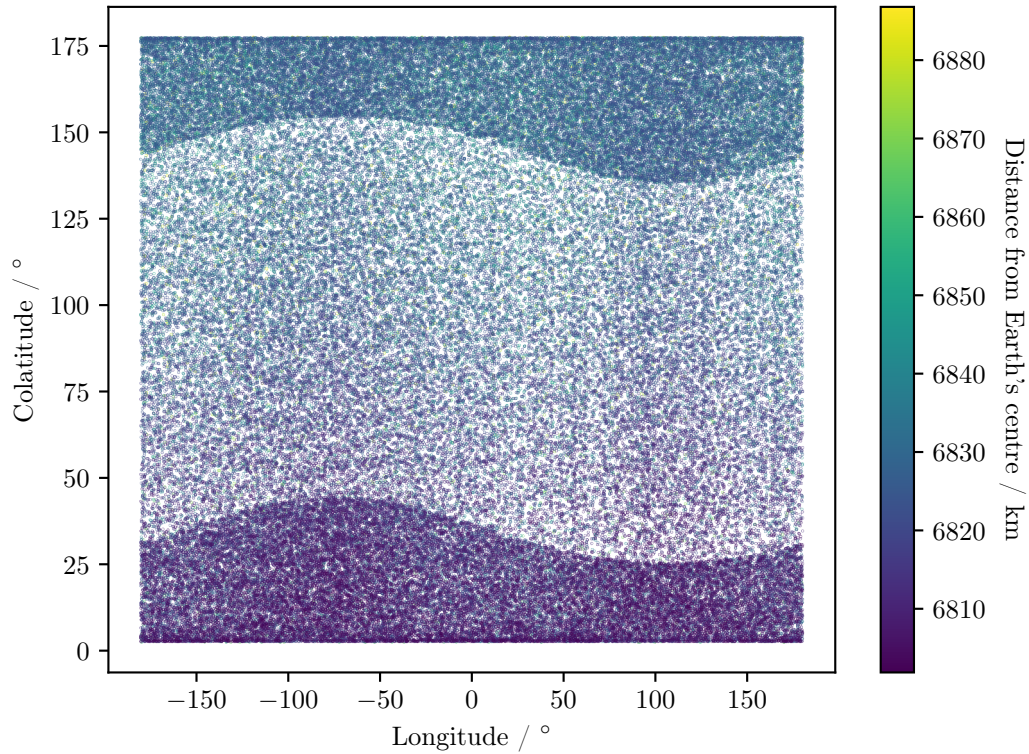
Figure 3.1: Scatter plot for the visualization of the global coverage of the used Swarm
data. The differences in data density at the poles result from the fact that
data selection criteria differed with respect to that region.

We have to apply two rotations, one is static and the other one depends on the position
of the satellite. These quaternions are contained in the data set.

Data selection is necessary for reducing contributions of unmodeled sources, e.g.
ionospheric currents. We used the same criteria as given in the GFZ Mag.numIGRF13
model, as described in subsection 1.3.3. For example, within low-latitude regions be-
tween $\pm 55°$ geomagnetic latitude, we use Swarm satellite nighttime data from 2014 to
2019 (compare also figure 3.1). The usage of nighttime data (23 h until 05 h local time)
reduces ionospheric currents' contributions in the data. It is also desirable to reduce
contributions from the ring current. In this respect, the MMA_SHA_2F index was
used to reduce such contributions. This index is similar to the Dst index and can be
regarded as a measure of the ring current. Finally, the data set which we used consists
of 1,202,003 data points of the Swarm-A satellite. Figure 3.1 depicts the global cov-
erage of the used Swarm data, whereas figure 3.2 shows the measured magnetic field,
separately for the $x$-, the $y$-, and the $z$-component pointing north, east, and down,
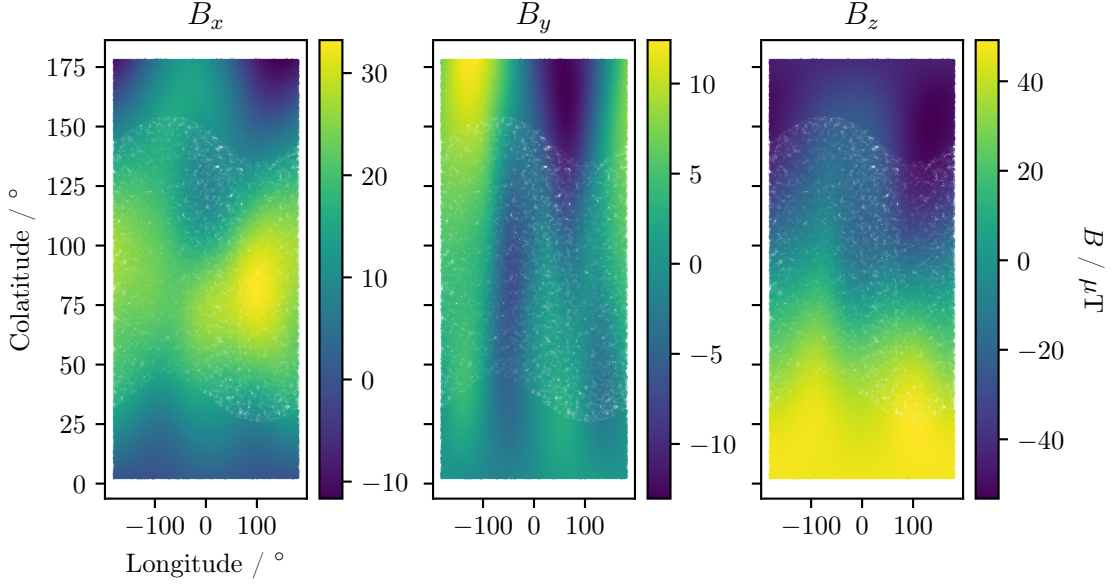respectively.

Figure 3.2: Scatter plot for the visualization of the measured magnetic field.

## 3.1.2 Parameterization of the EvoMag model

In our model, we parameterize the magnetic field as the gradient of the scalar magnetic potential

$$V = V_{\text{int}} + V_{\text{ext}} \tag{3.4}$$

with

$$
\begin{aligned}
V_{\text{int}}(r,\theta,\phi,t) &= a \sum_{l=1}^{L_{\text{int}}} \sum_{m=-l}^{l} \left(\frac{a}{r}\right)^{l+1} g_l^m(t) Y_l^m(\theta,\phi), \\
V_{\text{ext}}(r,\theta,\phi) &= a \sum_{l=1}^{L_{\text{ext}}} \sum_{m=-l}^{l} \left(\frac{r}{a}\right)^{l} q_l^m Y_l^m(\theta,\phi).
\end{aligned}
\tag{3.5}
$$

We note that this implies the restriction to a source-free region. The reference radius $a$ was set to 6371.2 km and the maximal expansion degrees $L_{\text{int}}$ and $L_{\text{ext}}$ were chosen to be 13 and 1, respectively. The Gauss coefficients $g_l^m$ of the internal potential represent the core magnetic field. To model secular variation, these need to be time dependent and are expanded in a basis of sixth-order B-splines $B_{i,6}$ such that

$$g_l^m(t) = \sum_{i=1}^{N_{\text{b}}} {}^i g_l^m B_{i,6}(t), \tag{3.6}$$

where we use $N_{\text{b}} = 18$ spline functions to represent each Gauss coefficient and the spline coefficients ${}^i g_l^m$ have to be inverted for. The time dependence is modeled on the interval from November 26th, 2013 up to February 1st, 2020. The internal knots for the
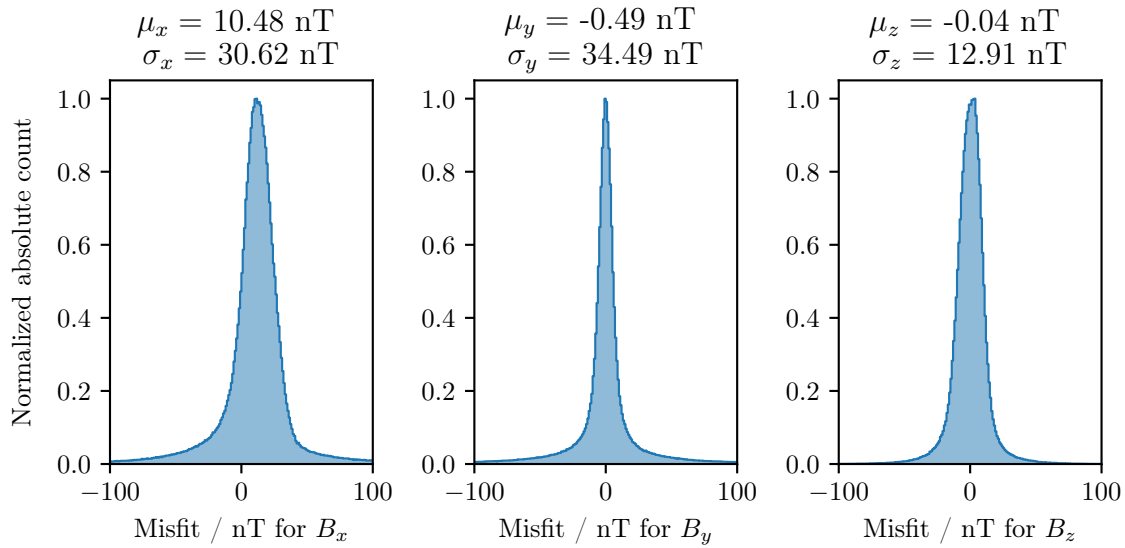
Figure 3.3: Normalized histogram of the residuals of our model where we ignored the external field contributions. Shown are the residuals for the $x$-component of the magnetic field (left), for the $y$-component (middle) and for the $z$-component (right). On top of each subfigure the associated mean value $\mu$ and the standard deviation $\sigma$ are depicted.

spline functions are equidistant and we use five-fold repeated knots at the boundary points of the interval. Further, the three coefficients for the external potential are assumed to be static, and mainly represent the magnetospheric ring current. Finally, this leads to 3513 model parameters which have to be estimated.

We also inverted for a model with internal coefficients $g_l^m$ only, but the obtained results showed the necessity to include the external field as well. Histograms of the residuals for the model without external coefficients are depicted in figure 3.3. Clearly, a shift of the distribution mean is visible for the $x$-component of the magnetic field. This shift results from the magnetic signature of the ring current which is mainly oriented along the $x$-direction. Theoretically, the separation into internal and external sources should work. However, the data are noisy and parts of the external fields can leak into our model. The shift of the distribution disappeared when we also considered the external coefficients, see figure 3.11.

As a further improvement of our model we also tried to estimate the static internal Gauss coefficients for the spherical harmonic degrees $l = 14 - 60$. This corresponds to a consideration of the crustal field. However, requirements for computing memory are challenging. In more detail, every spherical harmonic degree $l$ consists of $(2l+1)$ Gauss coefficients $g_l^m$. Adding 29 new coefficients for $l = 14$ only, corresponds to already 29 columns in our forward operator matrix. As a result, the size of the matrix $G$ increases by $29 \cdot 3 \cdot 1,202,003$ (rows correspond to number of data) elements which have to be kept in storage. With 8-byte float numbers, an additional crustal field with $l = 14 - 60$ would therefore correspond to an additional demand of 94 GB main memory. Hence,

we did not invert for the crustal part of the field. For further discussion of memory efficiency see chapter 4.

### 3.1.3 Inversion

**Applied inversion technique**

We inverted data by solving the linear equation system $Gm = d$ from equation (1.8) in a least-squares sense, i.e. we minimize the function

$$f(m) = \|Gm - d\|_2^2. \tag{3.7}$$

In order to minimize $f$, we applied the LMMAES algorithm with $\lambda = m = 28$ and $\mu = 14$, where $\lambda$ is the number of individuals in each generation, $m$ is the number of used direction vectors, and $\mu$ is the number of parents used for recombination. We did not use any particular model as a start point for the optimization run. The start point $m_0$ was chosen to be the zero vector. We set the termination criterion for the algorithm depending on the function value changes over five generations. If the function value changes only by a maximal value of $10^2$, the algorithm stopped. Please note that $10^2$ is a small value due to the large number of data points $(1,202,003)$. In order to calculate our model vector, we needed around 1.5 million function evaluations.

**Remarks on implementation in Python**

In LMMAES, the calculation of the forward model is crucial. However, straightforward implementation of the needed matrix-vector product $Gm$ (equation (1.8)) is not feasible due to memory reasons, as the complete matrix $G$ with more than 94 GB would be required to be kept in memory. Therefore, we divided the matrix product into smaller parts and implemented the matrix-vector multiplication as a multiplication with submatrices, such that

$$B = (G_\mathrm{d} \otimes G_\mathrm{s})m \tag{3.8}$$

holds, where $B$ is the magnetic field vector, the matrix $G_\mathrm{d}$ contains the values of the spherical harmonics functions, the matrix $G_\mathrm{s}$ contains the spline function values, and $\otimes$ denotes the row-wise outer product of $G_\mathrm{d}$ and $G_\mathrm{s}$. In EvoMag, we use the FieldTools library (Matuschek and Mauerberger, 2019) for the evaluation of $G_\mathrm{d}$. Further, equation (3.8) can be evaluated efficiently by the einsum function of the Python library NumPy.

In each generation, $\lambda = 28$ model vectors have to be evaluated on the objective function. This process can be accelerated by evaluating the objective function for all model vectors at once. For this purpose, we store the whole sample of 28 model vectors into a matrix $M$ (note that $M$ referred to the transformation matrix in subsection 2.5.3), whose rows are the individual model vectors. Using this $M$, the threaded einsumt (Wojciechowski, 2020) function of Python can be applied for an effective evaluation of the corresponding forward model, since einsumt performs better on large arrays.

$$
\begin{array}{ccc}
B_i & \longrightarrow & \mathrm{CPU}_0 \\
B_{i+1} & \longrightarrow & \mathrm{CPU}_1 \\
\vdots & & \vdots \\
B_{i+n} & \longrightarrow & \mathrm{CPU}_n
\end{array}
$$

Figure 3.4: Idea of the parallelization approach. The calculation is split into chunks, where the magnetic field vectors get distributed to different CPUs.

**Remarks on parallelization in Python**

One computational advantage of Evolution Strategies is that every individual step offers room for parallelization. Some approaches in order to parallelize CMAES exist, see e.g. Khan (2018). Here, we present our efforts to parallelize LMMAES using Python.

In general, a purely multithreaded approach (shared memory) is desirable because it does not require additional function calls for splitting and memory allocation. In contrast, a process-based distributed memory approach is only effective, if the ratio of times spent for allocation and computation is small. Compared to memory allocation, however, numerical evaluation of dot-products is usually very fast. Unfortunately, Python does not admit "real" multithreading because it relies on a technique called "Global Interpreter Lock" (GIL). This lock allows only one thread to execute written statements, and results from the fact that Python is an interpreted language.

The evaluation of the objective function can be parallelized as the forward calculation in equation (3.7) is independent for each datum. This idea is illustrated in figure 3.4. In order to circumvent the GIL, we use einsumt (Wojciechowski, 2020) for the matrix multiplication, which relies on the built-in subprocess module of Python. In contrast to multithreading, the subprocess module spawns new processes. The overhead is a function call and reallocation of memory. Hence, we cannot expect the performance which is offered by a real multithreading approach. Nevertheless, we were able to implement the calculations in a way such that 250,000 function evaluations per day are possible on an ordinary workstation featuring eight threads.

First attempts using "real" multithreading are promising and give hope for a speedup factor of 3 (see also chapter 4). There are also attempts to parallelize the individual samples across physical machines using MPI. For the EvoMag model, the number of individual samples in each generation is small ($\lambda = 28$), therefore it must be carefully examined how much latency is lost in the communication overhead. Two to three machines seem realistic, which leads to a speedup of maybe by a factor of two.

## 3.2 Results and model validation

### 3.2.1 Model surface magnetic field

In figure 3.5 we show the magnetic field predicted by our model at an altitude of $R = 6371.2$ km for the year 2015, separately for the $x$-, $y$-, and $z$-component of $B$. In addition, the bottom right subfigure shows the total field $F = (B_x^2 + B_y^2 + B_z^2)^{1/2}$.
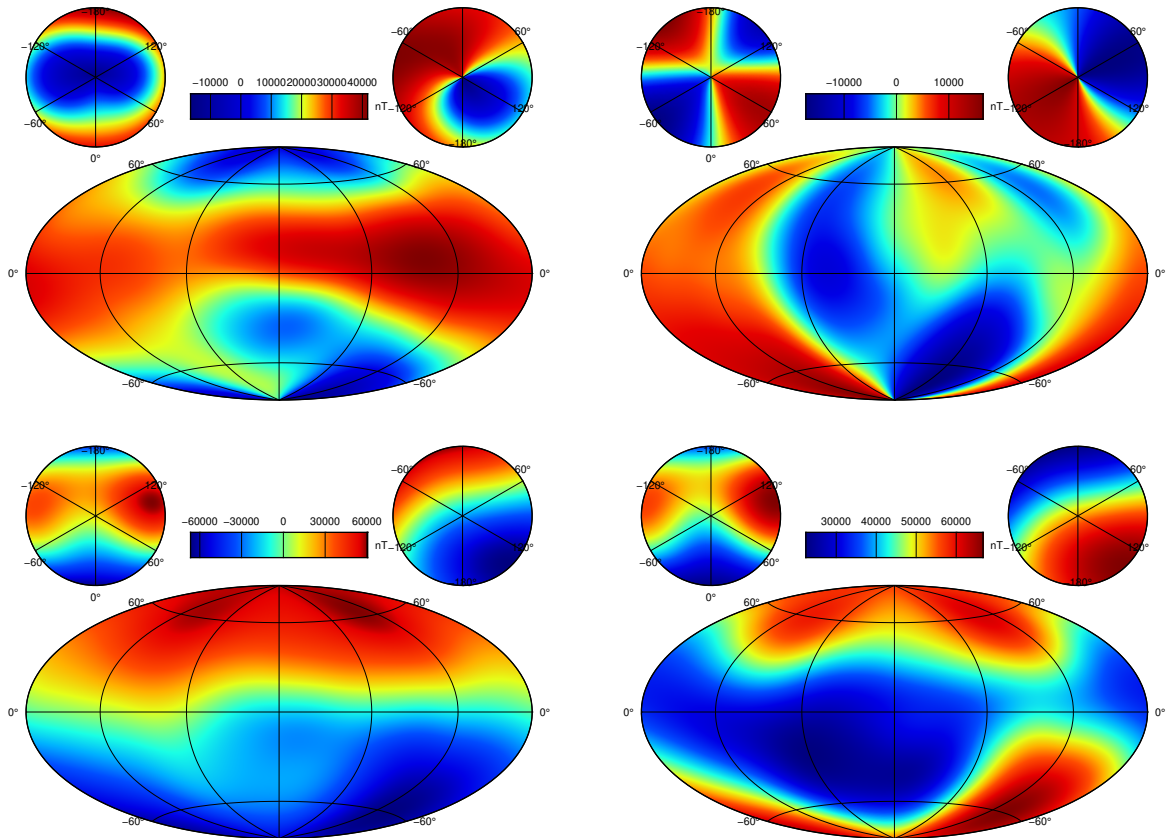
Figure 3.5: Model prediction of the magnetic field $B$ for 2015; $B_x$ (top left), $B_y$ (top right), $B_z$ (bottom left), and total field $F$ (bottom right) at Earth's mean radius $R = 6371.2$ km.

All subfigures show the expected main characteristics of the Earth's core magnetic field. For example, the magnetic field for the $z$-component shows the typical dipole dominance, with high values above the equator and low values below the equator. The figure also shows that the the equator is bent within the region of the South Atlantic Anomaly. This anomaly is also well depicted in the figure with the total field, where the minimal value $F_{\min} = 22,388$ nT is reached. This value is reasonable since the Mag.num.IGRF13 model predicts 22,250 nT (Rother, Korte, et al., 2020). Furthermore, the total field is strong at the north pole and within a region over Russia and Canada. Here, the two "wobbles" are characteristically and responsible for the recent fast pole wandering.

## 3.2.2 Time dependence of Gauss coefficients

In order to investigate the secular variation of the core field, it is possible to have a look at the temporal behavior of certain Gauss coefficients. For example, the three coefficients $g_1^{-1}, g_1^0$ and $g_1^1$ describe a dipole magnetic field. In figure 3.6 we show the temporal evolution of the leading Gauss coefficient $g_1^0$ as given by our model.
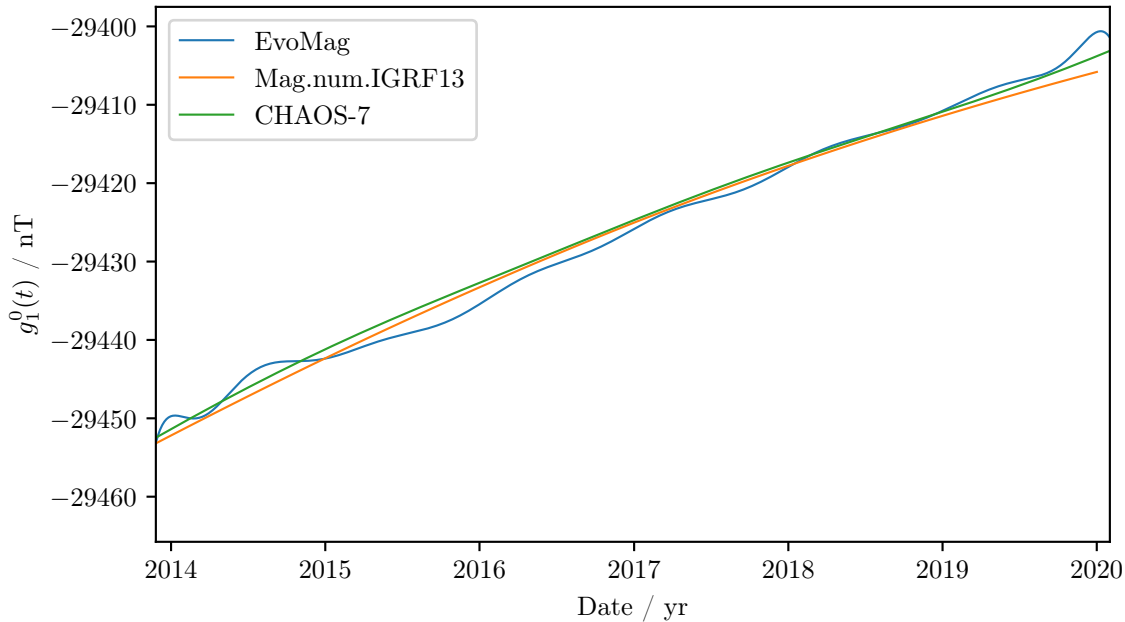
Figure 3.6: Time dependence of the Gauss coefficient $g_1^0$ for our model, compared with the results of the Mag.num.IGRF13 model and CHAOS-7. The time interval spans from the 26th November 2013 up to the first February 2020, these are the boundary points of our model time span.

Furthermore, we compare our results with the Mag.num.IGRF13 model since we apply the same data selection criteria, and we also compare our results with the established CHAOS-7 model.

On one hand, figure 3.6 depicts that our results lie in the same range as the established models. On the other hand, we recognize several oscillations near the end points of the interval. We want to check whether they are systematically and therefore have a look at two other Gauss coefficients of higher degree, namely $g_4^{-4}$ (figure 3.7) and $g_7^7$ (figure 3.8). Compared to our other coefficients, $g_4^{-4}$ is a typical example for low oscillations and a good agreement with CHAOS, and $g_7^7$ is a typical example for higher oscillations and a worse agreement with CHAOS. For the temporal behavior of $g_4^{-4}$ we see that our model agrees very well with the CHAOS-7 model and with the Mag.num.IGRF13 model, except for some differences at the end point of the model time span. Here, the differences between our model and CHAOS are very small. Moreover, the Mag.num model predicts a slightly different value for 2020. This is not surprising because in the Mag.num model, the temporal evolution is more strongly damped, and we use an undamped model.

Regarding $g_7^7$, figure 3.8 also shows oscillations near the end points, and they are stronger than the ones of the lower degree coefficients. The oscillations result from the absence of regularization terms in our model. An additional minimization of terms (see e.g. equation 1.21) as given in the Mag.num.IGRF13 model described in subsection 1.3.3 should stabilize the results. The stronger oscillations of the high degree coefficient
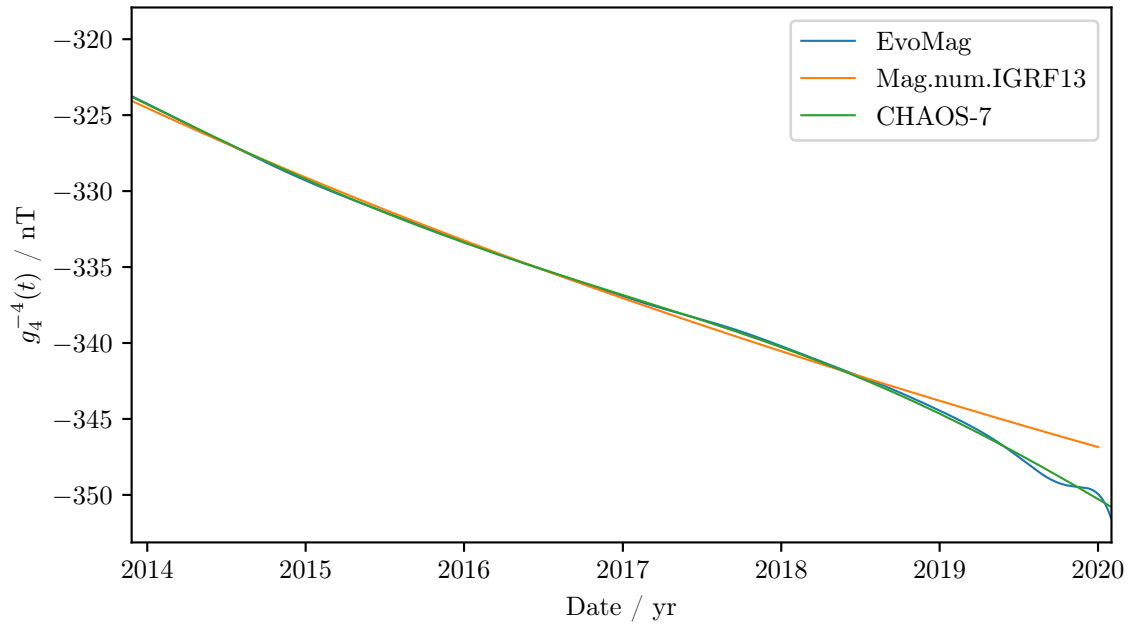
Figure 3.7: Time dependence of the Gauss coefficient $g_4^{-4}$ for our model, compared with the results of the Mag.num.IGRF13 model and CHAOS-7. The time interval spans from the 26th November 2013 up to the first February 2020, these are the boundary points of our model time span.
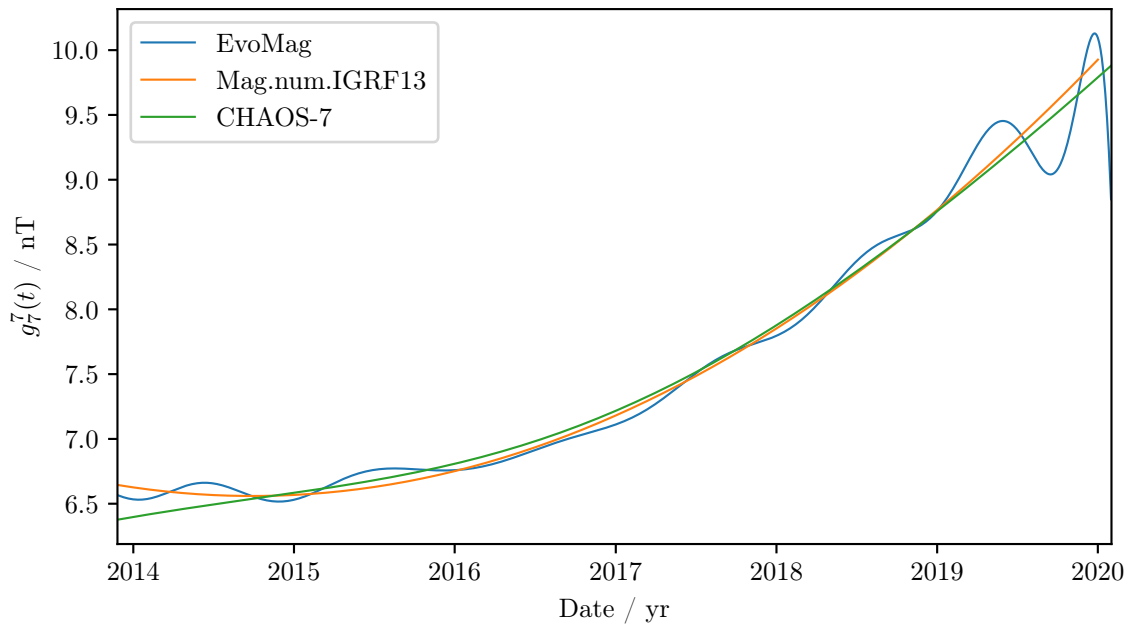


Figure 3.8: Time dependence of the Gauss coefficient $g_7^7$ for our model, compared with the results of the Mag.num.IGRF13 model and CHAOS-7. The time interval spans from the 26th November 2013 up to the first February 2020, these are the boundary points of our model time span.
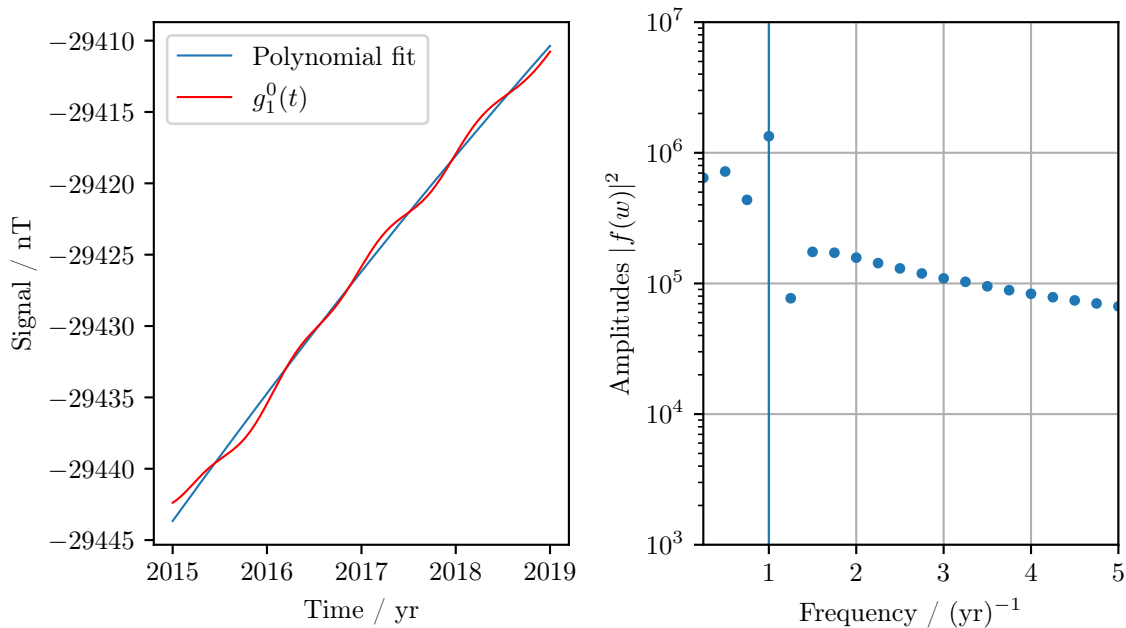
Figure 3.9: Fourier analysis of the fluctuations of $g_1^0$. The polynomial $p$ in the left hand figure was subtracted from $g_1^0$. Left: $g_1^0(t)$ (red) and $p$ (blue). Right: Amplitudes and their corresponding frequencies. The frequency of the maximal amplitude corresponds to the wavelength which we seek for.

could result from a decrease of the signal-to-noise ratio, also see subsection 3.2.3.

Both figures 3.6 and 3.8 also reveal smaller fluctuations away from the boundaries of our model. The established models do not show such oscillations, and we want to investigate whether they are in some sense physically meaningful. Therefore, we apply discrete Fourier transform with the goal to calculate the wavelength of the intermediate oscillations. Since we want to study the oscillations, we take a second degree polynomial fit of $g_1^0$ and subtract it from the original $g_1^0$ signal. We apply the discrete Fourier Transform to the resulting function $f$. The original $g_1^0$ signal and the polynomial fit are shown in figure 3.9, left. Moreover, the results of the Fourier Transform are shown on the right hand side plot in figure 3.9. There, the amplitudes corresponding to $|f|^2$ are shown, together with their associated frequency. They reveal that the wavelength of the inner fluctuations is one year. For this reason, we conclude that the inner oscillations may result from seasonal variations of the external field. This conclusion is underlined by our observation that the same results for this wavelength holds for the other oscillating coefficients.

### 3.2.3 Correlation coefficients

A measure for the similarity of two models is given by their degree correlation coefficients (Arkani-Hamed, 2001). Suppose that $\{g_{lm}, h_{lm}\}$ are the Gauss coefficients of the first model and $\{g'_{lm}, h'_{lm}\}$ are the Gauss coefficients of the second model. Then, the
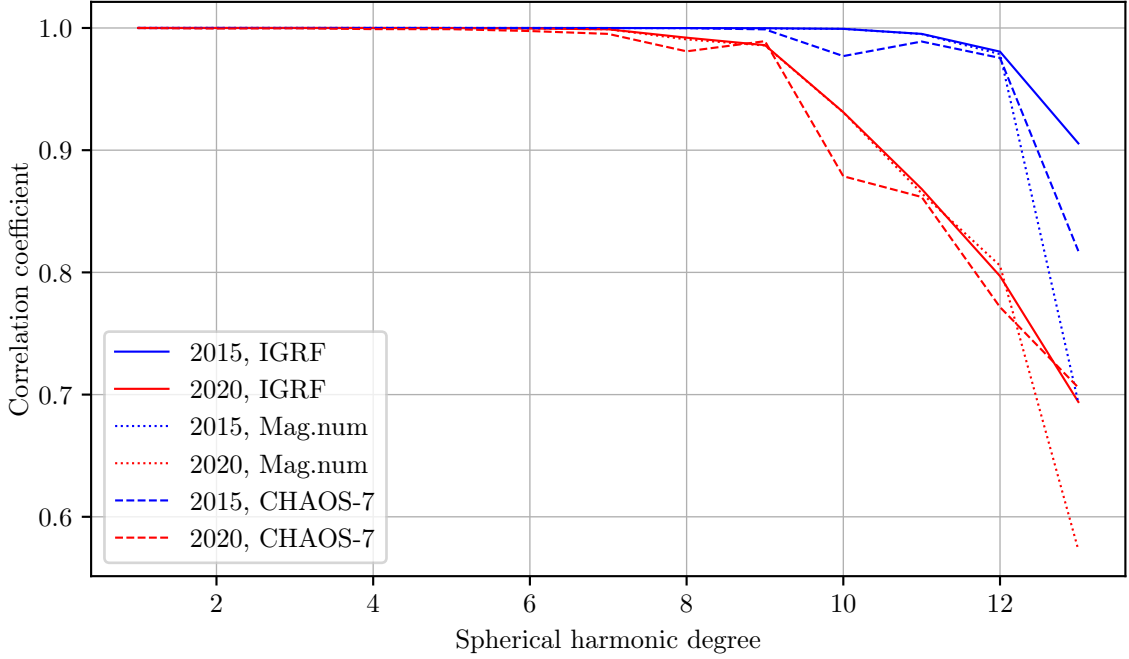
Figure 3.10: Degree correlation coefficients between different models. The blue curves
correspond to 2015 and the red curves are associated to 2020. Solid lines
show the coefficients between EvoMag and the IGRF, dotted lines be-
tween our model and Mag.num, and dashed lines between our model and
CHAOS-7.

degree-$l$-correlation-coefficient $\eta_l$ is given by

$$\eta_l = \frac{\sum_{m=0}^{l}(g_{lm}g'_{lm} + h_{lm}h'_{lm})}{[\sum_{m=0}^{l}(g_{lm}^2 + h_{lm}^2) \times \sum_{m=0}^{l}(g_{lm}'^2 + h_{lm}'^2)]^{1/2}}. \tag{3.9}$$

We calculated the degree correlation coefficients between the internal part of EvoMag
and the IGRF, CHAOS-7 and Mag.num, respectively, for the years 2015 and 2020.
The results are shown in figure 3.10. Figure 3.10 depicts a decrease of the correlation
coefficients with increasing spherical harmonic degree. This results from a decrease of
the signal-to-noise ratio at higher degrees. This situation can be explained by having a
sharper look at the equation (3.5) for the internal potential. Here, the term $(a/r)^{l+1}$ is
crucial. This term shows that with increasing $r$, the Gauss coefficients decrease more
rapidly with larger $l$. The noise stays the same over all wavelengths, but the signal
decreases with increasing wavelength. This behavior explains why the results become
worse with increasing spherical harmonic degree. On the other hand, comparing the
blue and the red curves in figure 3.10 shows that the results for 2015 are better than
the ones for 2020. The reason therefore is that 2020 is a boundary of our model and
that less data are available at this boundary (see also subsection 3.2.2).

## 3.2.4 Residuals

The residuals of a model are an important tool for investigating the quality of a regression. They are given by the vector $Gm - d$, where $m$ is our model vector. The vector $Gm$ is also referred to as the predicted data, and $d$ is the vector of the observed data. We look at the residuals always separately for the $x$-, $y$-, and $z$-component. In figure 3.11 normalized histograms of the residuals of the EvoMag model are depicted. Clearly visible is that the shift of the distribution mean for the $x$-component (see figure 3.3) disappeared after including the three additional external field Gauss coefficients. It is also visible that all three distributions largely correspond to a normal distribution, as expected. Still, some outliers are present as compared to a normal distribution (longer tails). For comparison, figure 3.12 shows normalized histograms of the residuals of EvoMag and Mag.num. In this figure, only the core field part of the models was considered, i.e. without external field contributions. The largest difference between the histograms can be seen in the $z$-component. Here, the Mag.num distribution is bimodal, whereas the EvoMag distribution is not. Probably, the reason is that the Mag.num parameterization also includes the induced part of the ring current, which is not part of our model. Therefore, the signal of the induced ring current leaks into our core field coefficients. Hence, our model fits the data better, as the data also include the signal of the ring current. This cannot be seen within the diagram for the $y$-components (where the two models agree very well), as the ring current is largely dipolar and therefore rotationally symmetric (there is no signal induced by the ring current, assuming a 1-D electrical conductivity profile for Earth's interior).

Figure 3.13 shows a color-coded scatter plot of the residuals. A closer investigation of the right hand panel in figure 3.13 may indicate lithospheric field contributions. For a more detailed examination of any lithospheric field structures, in figure 3.15, the EvoMag residuals of the field intensity are shown. These residuals are large on the poles due to field-aligned currents. Moreover, the Bangui anomaly is clearly visible north from the equator and east of the central meridian. The Bangui anomaly (see figure 3.14) in Central Africa is the "largest lithospheric magnetic field anomaly on Earth at low latitudes" (Ouabego et al., 2013). Furthermore, other anomalies are visible, e.g. in Europe. These anomalies are also predicted by the LCS-1 model (compare figure 3.14). However, some north-south-elongated systematic spatial patterns are also visible in the residuals, which are not related to any crustal anomalies as predicted by LCS-1. These patterns can result from temporal differences in adjacent satellite tracks (Lesur, Rother, Vervelidou, et al., 2013). Additionally, there seem to be some systematic residuals, which are not resulting from lithospheric field contributions, but are due to the fact that the model is not perfect and that some signals are not included in the parameterization.
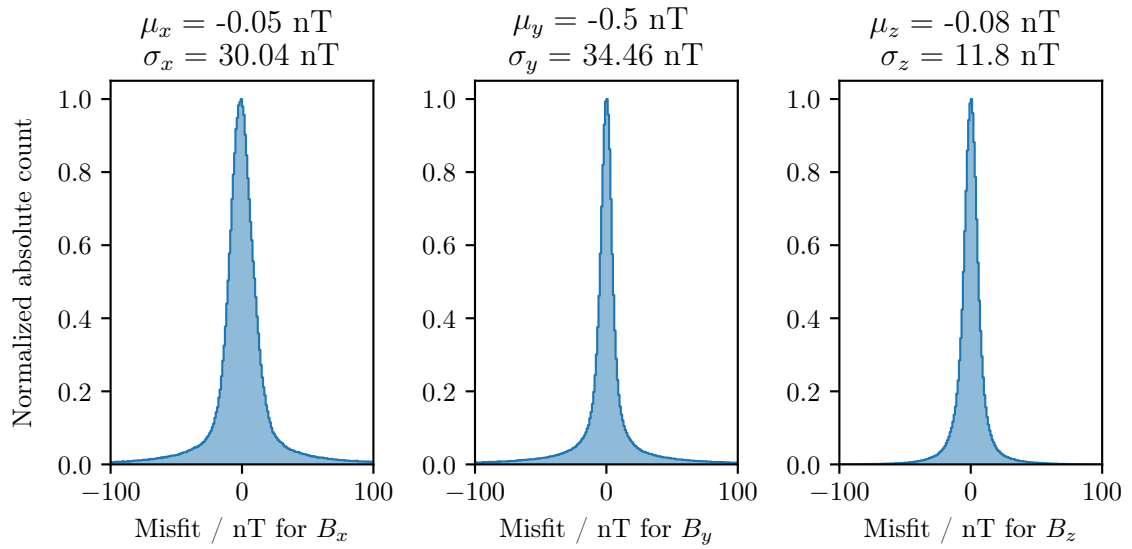
$\mu_x = -0.05$ nT
$\sigma_x = 30.04$ nT

$\mu_y = -0.5$ nT
$\sigma_y = 34.46$ nT

$\mu_z = -0.08$ nT
$\sigma_z = 11.8$ nT

Figure 3.11: Normalized histogram of the residuals of the EvoMag model. Shown are the residuals for the $x$-component of the magnetic field (left), for the $y$-component (middle) and for the $z$-component (right). On the top of each subfigure the associated mean value $\mu$ and the standard deviation $\sigma$ are depicted.
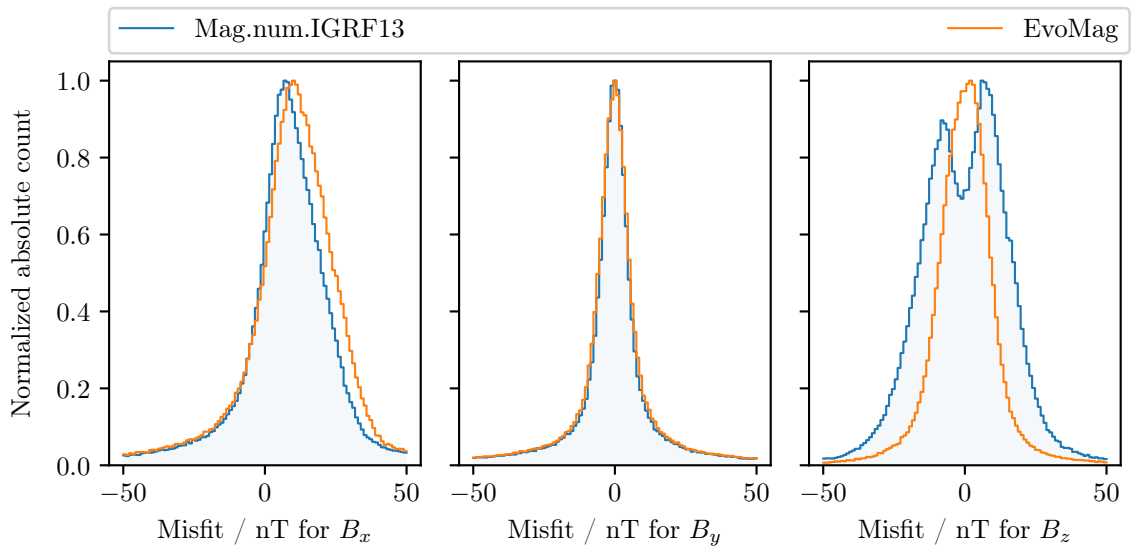
Figure 3.12: Normalized histograms of the EvoMag and Mag.num core field residuals. Shown are the residuals for the $x$-component of the magnetic field (left), for the $y$-component (middle) and for the $z$-component (right) for Mag.num (blue curves) and EvoMag (orange curves).
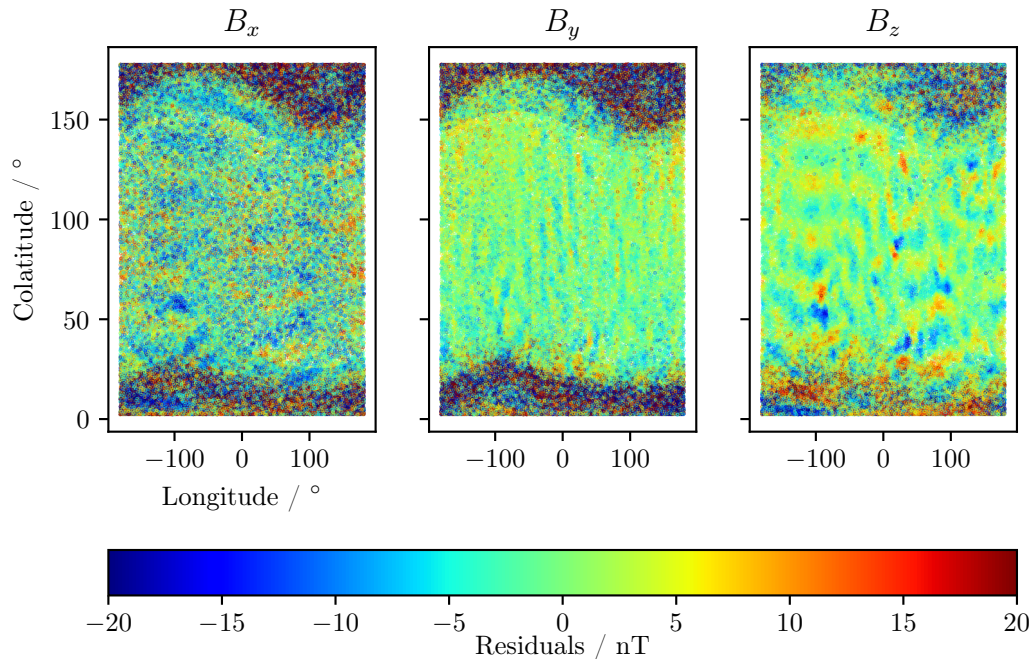
Figure 3.13: Scatter plot of the residuals of the EvoMag model. Shown are the residuals for the $x$-component of the magnetic field (left), for the $y$-component (middle) and for the $z$-component (right).

## 3.2.5  Changes in the declination

As an application of our model, we predict the temporal variation of the magnetic declination and compare our predictions with observations. The declination $D$ is defined as the eastward angle between geographic north and magnetic north. If a spherical coordinate system is used with $B_N$ pointing to geographic north and $B_E$ to geographic east, then $D$ can be calculated as (Campbell, 2003)

$$D = \arctan\left(\frac{B_E}{B_N}\right).$$
(3.10)

We use EvoMag to predict $D$ in daily steps between 2015 and 2019 at the location of the Niemegk geomagnetic observatory ($r = 6364.95$ km, $\theta = 38.120°$, and $\varphi = 12.683°$) and compare the results with the observations. The results are shown in figure 3.16, where daily means of the observations (turquoise line) are plotted along with our daily model predictions (red line). The observed daily means show oscillations during the whole time span, whereas our predictions do not. Observations include information about disturbances of the magnetic field due to sources (currents) in the ionosphere and magnetosphere. Such disturbances are due to varying ionospheric and magnetic currents or geomagnetic storms and substorms. Both impact the declination. Additionally, an offset is visible. The static part of this offset results from unmodeled crustal field contributions, the dynamic part of the offset may result from unmodeled core field contributions. The offset increases at the end of the modeled time span.
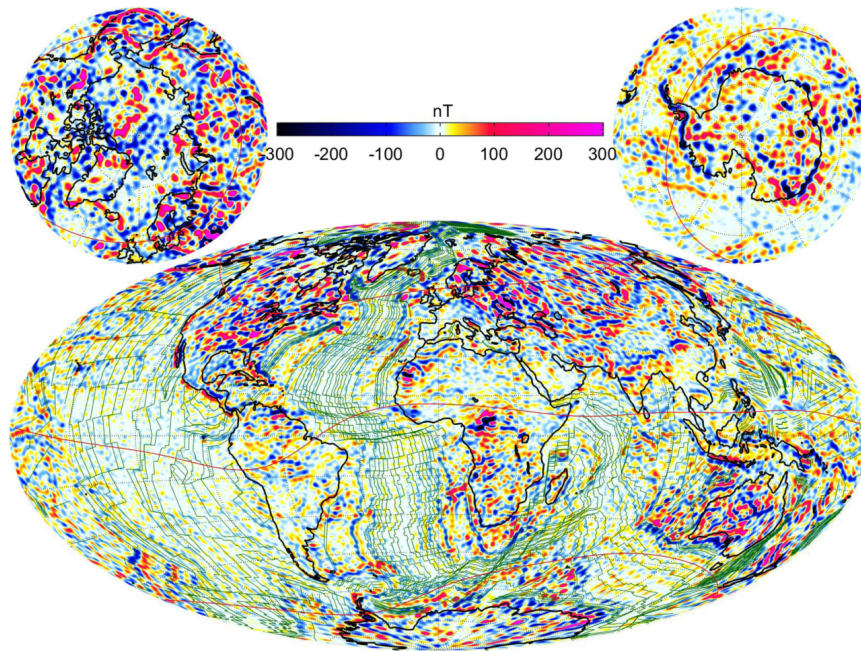
Figure 3.14: Lithospheric field of scalar anomaly $F$ at Earth's surface from the LCS-1 model. The Bangui anomaly in Central Africa is a roughly elliptical anomaly centered at Bangui. The figure was taken from Olsen, Ravat, et al. (2017, figure 7).
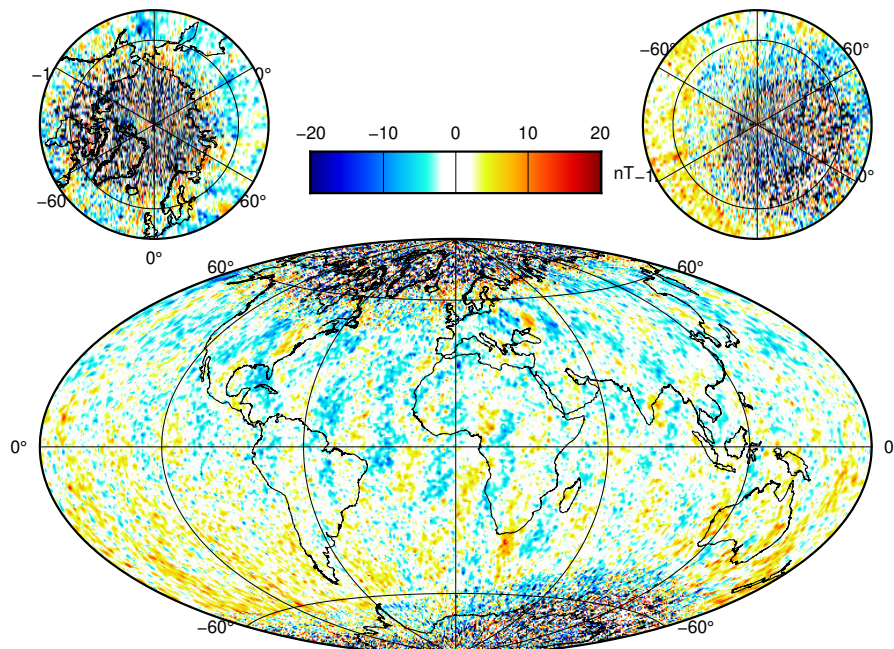


Figure 3.15: EvoMag-residuals of the field intensity. All residuals with absolute value larger than 20 nT are shown by dark red and blue colors and occur at the poles.
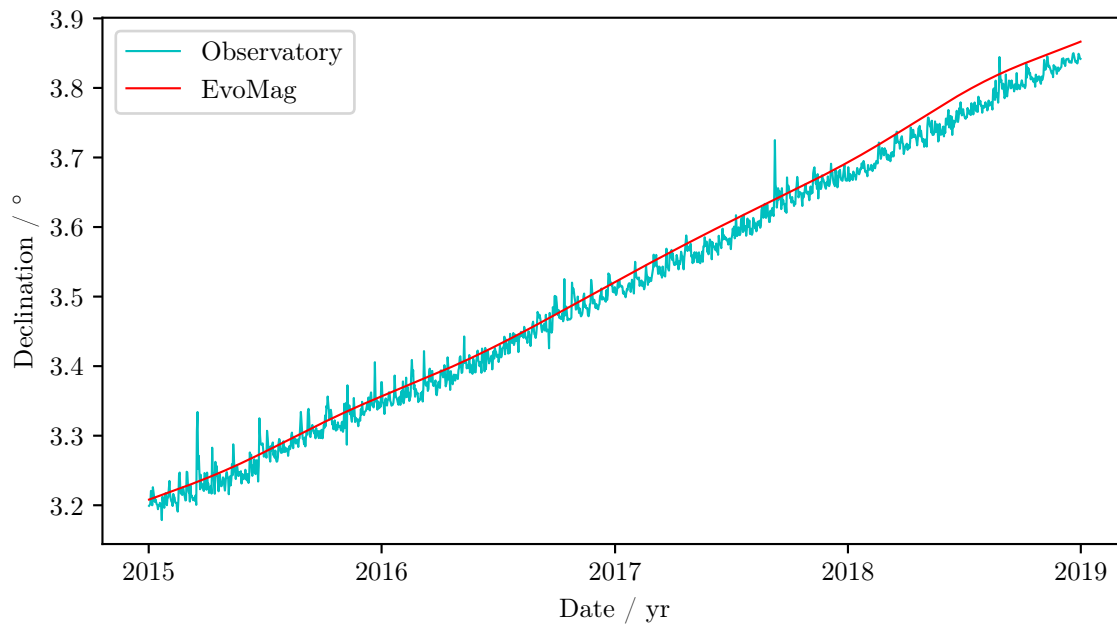
Figure 3.16: Declination of the magnetic field at Niemegk ($r = 6364.95$ km, $\theta = 38.120°$, and $\varphi = 12.683°$). The turquoise line depicts the values measured by the observatory (daily means), the red line corresponds to the values which were obtained with EvoMag.

# 4 Summary and outlook

Here, we summarize our work and discuss the observed advantages and disadvantages of stochastic optimization for core field modeling. We also mention unsolved problems and sketch some ideas for their solution.

Models of the Earth's magnetic field lay the foundation for several scientific studies and provide a physically consistent description of the magnetic field, or at least specific parts thereof. In this work, we presented the theoretical basis of core field modeling and set up the parameterization of a basic core field model which we call the EvoMag model. In order to invert for the model coefficients, we used the stochastic optimization algorithm LMMAES, which is a member of the CMAES family. We explained the theoretical foundations of these algorithms and implemented the inversion within a Python environment. The obtained EvoMag model is based on six years of Swarm A satellite data, and was discussed in detail before comparing it with results from established models such as CHAOS and the IGRF. On one hand, these results clearly show the missing external field part in EvoMag, but on the other hand, they often show very good agreement with established models. For example, the predicted minimum value of the magnetic field within the SAA differs only by 0.62% (138 nT) from the value given by the Mag.num model. The coefficients with the minimal and maximal root mean square difference between EvoMag and CHAOS-7 are $g_7^{-5}$ (predicted by EvoMag as $-2.91$ nT for 2015.0) and $g_1^0$ (predicted by EvoMag as $-29442.39$ nT for 2015.0), respectively. The maximal differences over the six-year model time span between $g_7^{-5}$ and $g_1^0$ are 0.39 nT and 3.26 nT, respectively, and the minimal differences are $7.21 \cdot 10^{-6}$ nT and $8.99 \cdot 10^{-5}$ nT, respectively. The dipole moment of EvoMag evaluates to $1.45124 \cdot 10^{-6}$ nT/km$^3$ and this value differs by only 0.04% from the CHAOS value. This implies the applicability of Evolution Strategies for magnetic field modeling. Also, as explained in subsection (3.1.3) and below, advanced implementations should give rise to shorter calculation times as compared to alternative strategies such as Iteratively Reweighted Least Squares (IRLS).

One of the advantages of Evolution Strategies is their flexible applicability to a large variety of problems and the low number of iterations which are required to approach the optimum, as compared to, e.g. IRLS. Another positive point is that stochastic optimization algorithms are able to find a global minimum almost independently of their initial value. This is not the case for derivative-based methods and for IRLS, since these algorithms do not converge in many cases, or the convergence is only assured if the starting point is close enough to the solution (e.g. Adil, Peng, and Sachdeva (2019)). In addition, it is often not known whether the obtained solution is only a local minimum. In contrast, Evolution Strategies effectively sample the parameter space with the help of a high population size, and hence find the global minimum (if it exists) and avoid

local minima, even if the initial solution is next to a local minimum. However, for these situations it should be also kept in mind that stochastic optimization methods are not determinable and that, in the worst case, results are not exactly reproducible. This problem could be solved by initializing several runs and testing the reproducibility. Furthermore, Evolution Strategies are not only less sensitive to the chosen starting value, but also to other choices of a-priori search parameters. Even if obviously "wrong" parameters were chosen within the initialization of the algorithm, the self-adaptation mechanisms regulate the parameters in a way such that they are best suited for the given search space.

On the downside, the usual ill-posedness of an underlying geophysical problem gives rise to a lot of function calls which are needed for the calculation of the model vector solution. Hence, the efficiency of the algorithm relies in particular on a very fast implementation of the forward model function calls. With our implementation, $250,000$ function calls per day were possible and the calculation took approximately six days on a single working station with eight cores. Although we did not compare this time directly with a classical algorithm, for more complex problems (than $L_2$) a runtime advantage is expectable due to aforementioned reasons (number of iterations, local minima). As explained in subsection 3.1.2, another challenge arises as the memory requirements for the calculations are very high in the case of geomagnetic field modeling, due to the large amount of data and the basis spline function values. More precisely, the complete forward operator matrix $G$ (see equation (1.8)) needs to be kept in memory for efficient calculation of the forward problem. However, modern workstations on high performance machines can have several hundred GB of RAM accessible and therefore are able to store the forward operator accordingly. For the purpose of this work, a machine with 256 GB of RAM would be suitable. An even better possibility for circumventing the memory problems is to parallelize the function calls themselves using shared memory and Message Passing Interface. For our forward model, this is easily realizable since the matrix vector multiplication can be split into multiplications with submatrices, and these smaller parts can be shared to different computers.

For this work, the LMMAES algorithm was used for the minimization of an objective function which uses a simple $L_2$ norm as a measure of misfit, and a linear forward operator. As a first improvement of the EvoMag model, regularization terms (see e.g. equations (1.21) and (2.3)) should be considered. The absence of such terms was obvious in e.g. the oscillations of the temporal behavior of $g_7^7$ in figure 3.8. Additionally, the usage of such regularization terms is not only important for stabilizing the model results, but also to better condition the objective function. Most importantly, this improvement could result in less function evaluations when calculating the solution. In more sophisticated models the forward operator will become nonlinear, for example, if Euler angles for the adjustment of the sensor coordinates are estimated. Also, in such situations the number of unknowns as well as data points increases. In order to get a serious benefit from the investigated algorithms and to adapt them to more complex models, it is a promising idea to embed the LMMAES algorithm into the present FORTRAN implementation of the Mag.num model. The usage of FORTRAN in this setting is highly desirable because this language is very optimized for linear algebra calcula-

tions and offers extensive possibilities for the usage of multithreading. Furthermore, there already exists a FORTRAN 90 library for CMAES which additionally provides parallelization (Müller et al., 2009). A parallelized implementation of LMMAES will be straightforward to adapt from the currently available implementation. First attempts using "real" multithreading in C++ are promising and may result in a speedup factor 3.

Another idea to investigate is whether it is possible to exploit the covariance matrix for model error estimations. These are usually not provided with models, and difficult to obtain, as both the data errors need to be estimated, and their propagation to model coefficients needs to be understood, especially when non-linearities and regularization are introduced. Although the transformation matrix $M$ in the LMMAES algorithm is only an approximation of $C^{1/2}$, the transformation matrix of the $i$-th generation is still connected to a covariance matrix $C$. An interesting question to investigate is whether in some sense there exists a limit covariance matrix $\Sigma$, i.e. that

$$\lim_{i \to \infty} \Sigma_i = \Sigma \qquad (4.1)$$

holds, where $i$ denotes the $i$-th generation. Equation (4.1) means that the final result of the algorithm has a "meaningful" covariance matrix associated with it. If this would be the case, then the variance could be interpreted as a standard error and it would open the possibility for direct parameter error estimations (see also Grayver and Kuvshinov (2016)).

All in all, we recommend the application of Evolution Strategies for the usage of modeling the geomagnetic field of Earth's core. An efficient implementation of such an algorithm allows a fast inversion even for non-linear problems, for example when Euler angles are considered. This is particularly advantageous as core field models become more complex in the future.

# Bibliography

Adil, Deeksha, Peng, Richard, and Sachdeva, Sushant (2019). "Fast, Provably convergent IRLS Algorithm for p-norm Linear Regression". In: *CoRR* abs/1907.07167. arXiv: 1907.07167. URL: http://arxiv.org/abs/1907.07167.

Arkani-Hamed, Jafar (2001). "A 50-degree spherical harmonic model of the magnetic field of Mars". In: *Journal of Geophysical Research: Planets* 106.E10, pp. 23197–23208. DOI: https://doi.org/10.1029/2000JE001365. eprint: https://agupubs.onlinelibrary.wiley.com/doi/pdf/10.1029/2000JE001365. URL: https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2000JE001365.

Aster, Richard C., Borchers, Brian, and Thurber, Clifford H. (2019). *Parameter Estimation and Inverse Problems (Third Edition)*. Third edition. Elsevier. ISBN: 978-0-12-804651-7. DOI: https://doi.org/10.1016/B978-0-12-804651-7.00010-9.

Auger, A. et al. (2009). "Experimental Comparisons of Derivative Free Optimization Algorithms". In: *Experimental Algorithms*. Ed. by Jan Vahrenhold. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 3–15. ISBN: 978-3-642-02011-7.

Bäck, T., Foussette, C., and Krause, P. (2013). *Contemporary Evolution Strategies*. Natural Computing Series. Springer Berlin Heidelberg. ISBN: 9783642401374. URL: https://books.google.de/books?id=GPy8BAAAQBAJ.

Bäck, Thomas and Hoffmeister, Frank (1994). "Basic aspects of evolution strategies". In: *Statistics and Computing* 4.2, pp. 51–63. ISSN: 1573-1375. URL: https://doi.org/10.1007/BF00175353.

Beyer, H.G. (2001). *The Theory of Evolution Strategies*. Natural Computing Series. Springer. ISBN: 9783540672975. URL: https://books.google.de/books?id=8tbInLufkTMC.

Beyer, Hans-Georg and Sendhoff, Bernhard (Mar. 2017). "Simplify Your Covariance Matrix Adaptation Evolution Strategy". In: *IEEE Transactions on Evolutionary Computation* PP, pp. 1–1. DOI: 10.1109/TEVC.2017.2680320.

Bloxham, Jeremy and Jackson, Andrew (1992). "Time-dependent mapping of the magnetic field at the core-mantle boundary". In: *Journal of Geophysical Research: Solid Earth* 97.B13, pp. 19537–19563. DOI: 10.1029/92JB01591. eprint: https://agupubs.onlinelibrary.wiley.com/doi/pdf/10.1029/92JB01591. URL: https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/92JB01591.

Borsic, Andrea and Adler, A (Sept. 2012). "A primal–dual interior-point framework for using the L1 or L2 norm on the data and regularization terms of inverse problems". In: *Inverse Problems - INVERSE PROBL* 28. DOI: 10.1088/0266-5611/28/9/095011.

Campbell, Wallace H. (2003). *Introduction to Geomagnetic Fields*. 2nd ed. Cambridge: Cambridge University Press. DOI: 10.1017/cbo9781139165136. URL: https://

`www . cambridge . org / core / books / introduction - to - geomagnetic - fields /`
`3BB56187054FBC8B56554BED30DA2F7A`.

Coello Coello, Carlos A. (2005). "An Introduction to Evolutionary Algorithms and Their Applications". In: *Advanced Distributed Systems*. Ed. by Félix F. Ramos, Victor Larios Rosillo, and Herwig Unger. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 425–442. ISBN: 978-3-540-31674-9.

De Boor, Carl (Jan. 1978). *A Practical Guide to Splines*. Vol. Volume 27. DOI: `10 . 2307/2006241`.

Engl, H. W., Hanke, M., and Neubauer, A. (2000). *Regularization of Inverse Problems*. Mathematics and Its Applications. Springer Netherlands. ISBN: 9780792361404. URL: `https://books.google.de/books?id=VuEV-Gj1GZcC`.

Farquharson, Colin G. and Oldenburg, Douglas W. (July 1998). "Non-linear inversion using general measures of data misfit and model structure". In: *Geophysical Journal International* 134.1, pp. 213–227. ISSN: 0956-540X. DOI: `10 . 1046 / j . 1365 - 246x . 1998 . 00555 . x`. eprint: `https : // academic . oup . com / gji / article - pdf / 134 / 1 / 213 / 1764248 / 134 - 1 - 213 . pdf`. URL: `https : // doi . org / 10 . 1046 / j . 1365 - 246x.1998.00555.x`.

Finlay, C. C., Lesur, V., et al. (2017). "Challenges Handling Magnetospheric and Ionospheric Signals in Internal Geomagnetic Field Modelling". In: *Space Science Reviews* 206.1, pp. 157–189. ISSN: 1572-9672. URL: `https://doi.org/10.1007/s11214-016-0285-9`.

Finlay, Christopher C., Olsen, N., et al. (2016). "Recent geomagnetic secular variation from Swarm and ground observatories as estimated in the CHAOS-6 geomagnetic field model". In: *Earth, Planets and Space* 68.1, p. 112. ISSN: 1880-5981. URL: `https://doi.org/10.1186/s40623-016-0486-1`.

Fogel, David B. (1997). "Practical advantages of evolutionary computation". In: *Applications of Soft Computing*. Ed. by Bruno Bosacchi, James C. Bezdek, and David B. Fogel. Vol. 3165. International Society for Optics and Photonics. SPIE, pp. 14–22. DOI: `10.1117/12.279591`. URL: `https://doi.org/10.1117/12.279591`.

Grayver, Alexander V. and Kuvshinov, Alexey V. (Feb. 2016). "Exploring equivalence domain in nonlinear inverse problems using Covariance Matrix Adaption Evolution Strategy (CMAES) and random sampling". In: *Geophysical Journal International* 205.2, pp. 971–987. ISSN: 0956-540X. DOI: `10 . 1093 / gji / ggw063`. eprint: `https : // academic . oup . com / gji / article - pdf / 205 / 2 / 971 / 2016096 / ggw063 . pdf`. URL: `https://doi.org/10.1093/gji/ggw063`.

Gubbins, D. and Herrero-Bervera, E. (2007). *Encyclopedia of Geomagnetism and Paleomagnetism*. Encyclopedia of Geomagnetism and Paleomagnetism. Springer Netherlands. URL: `https://books.google.de/books?id=O-wA0ocxAiIC`.

Hansen, N. (1998). *Verallgemeinerte individuelle Schrittweitenregelung in der Evolutionsstrategie - eine Untersuchung zur entstochastisierten, koordinatensystemunabhängigen Adaptation der Mutationsverteilung*. Berlin: Mensch und Buch Verlag. ISBN: 3-933346-29-0.

Hansen, N., Arnold, D., and Auger, A. (2015). "Evolution Strategies". In: *Handbook of Computational Intelligence*.

Hansen, N. and Ostermeier, A. (June 2001). "Completely Derandomized Self-Adaptation in Evolution Strategies". In: *Evolutionary Computation* 9.2, pp. 159–195. ISSN: 1063-6560. DOI: 10.1162/106365601750190398.

Hansen, N., Ros, Raymond, et al. (2011). "Impacts of invariance in search: When CMA-ES and PSO face ill-conditioned and non-separable problems". In: *Appl. Soft Comput.* 11, pp. 5755–5769.

Hansen, Nikolaus (2016). "The CMA Evolution Strategy: A Tutorial". In: *CoRR* abs/1604.00772. arXiv: 1604.00772. URL: http://arxiv.org/abs/1604.00772.

Hansen, Nikolaus, Müller, Sibylle D., and Koumoutsakos, Petros (2003). "Reducing the Time Complexity of the Derandomized Evolution Strategy with Covariance Matrix Adaptation (CMA-ES)". In: *Evol. Comput.* 11.1, pp. 1–18. DOI: 10.1162/106365603321828970. URL: https://doi.org/10.1162/106365603321828970.

Khan, Najeeb (2018). "A parallel implementation of the covariance matrix adaptation evolution strategy". In: *CoRR* abs/1805.11201. arXiv: 1805.11201. URL: http://arxiv.org/abs/1805.11201.

Langel, R. A. and Estes, R. H. (1982). "A geomagnetic field spectrum". In: *Geophysical Research Letters* 9.4, pp. 250–253. DOI: 10.1029/GL009i004p00250.

Lesur, V., Rother, M., Vervelidou, F., et al. (2013). "Post-processing scheme for modelling the lithospheric magnetic field". In: *Solid Earth* 4.1, pp. 105–118. DOI: 10.5194/se-4-105-2013. URL: https://se.copernicus.org/articles/4/105/2013/.

Lesur, V., Wardinski, I., et al. (2010). "The second generation of the GFZ Reference Internal Magnetic Model: GRIMM-2". In: *Earth, Planets and Space* 62.10, p. 6. DOI: 10.5047/eps.2010.07.007.

Lesur, Vincent, Rother, Martin, Wardinski, Ingo, et al. (2015). "Parent magnetic field models for the IGRF-12GFZ-candidates". In: *Earth, Planets and Space* 67.1, p. 87. ISSN: 1880-5981. URL: https://doi.org/10.1186/s40623-015-0239-6.

Loshchilov, Ilya (Apr. 2014). "A Computationally Efficient Limited Memory CMA-ES for Large Scale Optimization". In: *Proceeding of the sixteenth annual conference on Genetic and Evolutionary Computation Conference (GECCO'2014)*. DOI: 10.1145/2576768.2598294.

– (Oct. 2015). "LM-CMA: An alternative to L-BFGS for large-scale black Box optimization". In: *Evolutionary computation* 25. DOI: 10.1162/EVCO_a_00168.

Loshchilov, Ilya, Glasmachers, Tobias, and Beyer, Hans-Georg (2017). "Limited-Memory Matrix Adaptation for Large Scale Black-box Optimization". In: *CoRR* abs/1705.06693. arXiv: 1705.06693. URL: http://arxiv.org/abs/1705.06693.

Macmillan, Susan and Finlay, Christopher (Oct. 2010). "The International Geomagnetic Reference Field". In: pp. 265–276. DOI: 10.1007/978-90-481-9858-0_10.

Mandea, M. and Korte, M. (Jan. 2011). *Geomagnetic Observations and Models*. ISBN: 978-90-481-9857-3. DOI: 10.1007/978-90-481-9858-0.

Matuschek, Hannes and Mauerberger, Stefan (2019). *FieldTools: Toolbox for manipulating vector fields on the sphere. V. 0.1.3.* GFZ Data Services. DOI: https://doi.org/10.5880/fidgeo.2019.033.

Maus, S. et al. (2006). "Third generation of the Potsdam Magnetic Model of the Earth (POMME)". In: *Geochemistry, Geophysics, Geosystems* 7.7. DOI: `10.1029/2006GC001269`.

Mendivil, Franklin et al. (2013). "A New Method for TSVD Regularization Truncated Parameter Selection". In: *Mathematical Problems in Engineering* 2013, p. 161834. ISSN: 1024-123X. URL: `https://doi.org/10.1155/2013/161834`.

Miyazawa, Hidekazu and Akimoto, Youhei (2017). "Effect of the Mean Vector Learning Rate in CMA-ES". In: *Proceedings of the Genetic and Evolutionary Computation Conference.* GECCO '17. Berlin, Germany: Association for Computing Machinery, pp. 721–728. ISBN: 9781450349208. DOI: `10.1145/3071178.3071203`. URL: `https://doi.org/10.1145/3071178.3071203`.

Morschhauser, A., Lesur, V., and Grott, M. (2014). "A spherical harmonic model of the lithospheric magnetic field of Mars". In: *Journal of Geophysical Research: Planets* 119.6, pp. 1162–1188. DOI: `10.1002/2013JE004555`. eprint: `https://agupubs.onlinelibrary.wiley.com/doi/pdf/10.1002/2013JE004555`. URL: `https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1002/2013JE004555`.

Morschhauser, Achim (2016). "A model of the crustal magnetic field of mars". PhD thesis. Westfälische Wilhelms-Universität Münster. URL: `https://miami.uni-muenster.de/Record/5a51faf0-8b39-4edc-bf67-11c205bc9ccb` (visited on 09/13/2020).

Müller, Christian et al. (Jan. 2009). "pCMALib: A parallel Fortran 90 library for the evolution strategy with covariance matrix adaptation". In: pp. 1411–1418. DOI: `10.1145/1569901.1570090`.

Olsen, N., Lühr, H., et al. (Mar. 2014). "The CHAOS-4 geomagnetic field model". In: *Geophysical Journal International* 197.2, pp. 815–827. ISSN: 0956-540X. DOI: `10.1093/gji/ggu033`.

Olsen, Nils, Ravat, Dhananjay, et al. (Sept. 2017). "LCS-1: a high-resolution global model of the lithospheric magnetic field derived from CHAMP and Swarm satellite observations". In: *Geophysical Journal International* 211.3, pp. 1461–1477. ISSN: 0956-540X. DOI: `10.1093/gji/ggx381`. eprint: `https://academic.oup.com/gji/article-pdf/211/3/1461/21308773/ggx381.pdf`. URL: `https://doi.org/10.1093/gji/ggx381`.

Olsen, Nils and Stolle, Claudia (May 2012). "Satellite Geomagnetism". In: *Annual Review of Earth and Planetary Sciences* 40, pp. 441–465. DOI: `10.1146/annurev-earth-042711-105540`.

Ouabego, Mariane et al. (Nov. 2013). "Rock magnetic investigation of possible sources of the Bangui magnetic anomaly". In: *Physics of the Earth and Planetary Interiors* 224, pp. 11–20. DOI: `10.1016/j.pepi.2013.09.003`.

Rechenberg, I. (1973). *Evolutionsstrategie: Optimierung technischer Systeme nach Prinzipien der biologischen Evolution.* Frommann-Holzboog. ISBN: 9783772803741. URL: `https://books.google.de/books?id=iw2hcQAACAAJ`.

Rother, Martin, Korte, Monika, et al. (Mar. 2020). "Earth's core magnetic field model Mag.num and the IGRF 13 candidate". In: DOI: `10.5194/egusphere-egu2020-8072`. URL: `https://doi.org/10.5194%2Fegusphere-egu2020-8072`.

Rother, Martin, Lesur, Vincent, and Schachtschneider, Reyko (2013). "An algorithm for deriving core magnetic field models from the Swarm data set". In: *Earth, Planets and Space* 65.11, p. 3. ISSN: 1880-5981. URL: `https://doi.org/10.5047/eps.2013.07.005`.

Rudolph, Günter (2012). "Evolutionary Strategies". In: *Handbook of Natural Computing*. Ed. by Grzegorz Rozenberg, Thomas Bäck, and Joost N. Kok. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 673–698. ISBN: 978-3-540-92910-9. DOI: `10.1007/978-3-540-92910-9_22`. URL: `https://doi.org/10.1007/978-3-540-92910-9_22`.

Sabaka, T. J. et al. (2018). "A comprehensive model of Earth's magnetic field determined from 4 years of Swarm satellite observations". In: *Earth, Planets and Space* 70.1, p. 130. ISSN: 1880-5981. URL: `https://doi.org/10.1186/s40623-018-0896-3`.

Scherrmann, Alexander (Feb. 2020). "Evolutionary Strategies. Essay for the Module exam PHY-755". Potsdam.

Sen, Mrinal K. and Stoffa, Paul L. (2013). *Global Optimization Methods in Geophysical Inversion*. Cambridge: Cambridge University Press. DOI: `10.1017/cbo9780511997570`. URL: `https://www.cambridge.org/core/books/global-optimization-methods-in-geophysical-inversion/C2B23286E6BCC2177117431CB568101C`.

Suttorp, Thorsten, Hansen, Nikolaus, and Igel, Christian (2009). "Efficient covariance matrix update for variable metric evolution strategies". In: *Machine Learning* 75.2, pp. 167–197. ISSN: 1573-0565. URL: `https://doi.org/10.1007/s10994-009-5102-1`.

Thébault, Erwan et al. (2015). "International Geomagnetic Reference Field: the 12th generation". In: *Earth, Planets and Space* 67.1, p. 79. ISSN: 1880-5981. URL: `https://doi.org/10.1186/s40623-015-0228-9`.

Tong, Y.L. (1990). *The Multivariate Normal Distribution*. Springer Series in Statistics. Springer New York. ISBN: 9781461396550. URL: `https://books.google.de/books?id=FtHgBwAAQBAJ`.

Wardinski, I. (2005). "Core Surface Flow Models from Decadal and Subdecadal Secular Variation of the Main Geomagnetic Field". Dissertation. URL: `http://dx.doi.org/10.17169/refubium-8325`.

Whaler, K. A. and Gubbins, D. (June 1981). "Spherical harmonic analysis of the geomagnetic field: an example of a linear inverse problem". In: *Geophysical Journal International* 65.3, pp. 645–693. ISSN: 0956-540X. DOI: `10.1111/j.1365-246X.1981.tb04877.x`. eprint: `https://academic.oup.com/gji/article-pdf/65/3/645/1537193/65-3-645.pdf`. URL: `https://doi.org/10.1111/j.1365-246X.1981.tb04877.x`.

Wojciechowski, Marek (Sept. 22, 2020). *GitHub Repository of the einsumt function. Multithreaded version of numpy.einsum*. URL: `https://github.com/mrkwjc/einsumt` (visited on 01/28/2021).

# Selbstständigkeitserklärung

Ich versichere, dass ich die von mir vorgelegte Arbeit selbstständig verfasst habe, die verwendeten Quellen, Internet-Quellen und Hilfsmittel vollständig angegeben habe und alle Stellen der Arbeit – einschließlich Abbildungen –, die anderen Werken oder dem Internet im Wortlaut oder dem Sinn nach entnommen sind, in jedem Fall unter Angabe der Quelle als Entlehnung kenntlich gemacht habe. Die Arbeit wurde bisher im Rahmen keiner anderen Prüfung vorgelegt.

Netzen, den 31.01.2021

_____
Jan Möhring