Alvers, M. R., Götze, H.-J., Anikiev, D., Plonka, C. (2023): Inversion of potential fields by interactive optimization of 3D subsurface models using spring-based space warping and evolution strategy. - Geophysics, 88, 3, G79-G93.

https://doi.org/10.1190/geo2022-0222.1

Inversion of potential fields by interactive optimization of 3D subsurface models using spring-based space warping and evolution strategy

Michael R. Alvers[*], *Hans-Jürgen Götze*[†], *Denis Anikiev* [‡] *and Christian Plonka*[‡]

ABSTRACT

Rapid development of dynamic data-integrative modeling of geological processes and subsurface structures is an important factor for sustainable utilization of natural resources. One of the current gaps is the interactive construction of realistic three-dimensional models of the Earth's underground. We present here a methodology of 3-D interactive processing of potential fields – gravity and magnetics – as well as their potentials and derivatives, combining forward and inverse modeling. Forward computations are based on the approximation of geological subsurface structures by polyhedra with triangulated surfaces. Inverse computations for the model geometry are performed by means of Covariance-Matrix-Adaptation Evolution Strategy (CMA-ES), which is proved to be efficient in case of a strongly non-linear problem and high-dimensional parameter space, as in potential field models. The main disadvantage is related to triangulation, as certain algorithmic constraints must be applied during approximation of geological body shapes. To avoid topology distortions we use a concept of warping the space containing the model, rather than the model itself. However, the optimized lengths of grid sides are dependent on each other, which degrades the self-adaptation of the CMA-ES. The elegant solution is to introduce a system of virtual elastic springs connecting the grid nodes. We present a numerical formulation of this system and provide a proof of concept rather than an overview of theoretical concepts of inversion schemes. The new workflow is tested on the 3-D SEAM model and applied to a real case study of salt dome modeling in the Northwest German Basin. In the context of the inversion procedure described here we demonstrate how an interpreter can visually control and influence the quality of the inversion on a timeline.

INTRODUCTION

Today's developments in the field of digital data processing make it possible to create increasingly complex images of natural structures and evolutionary processes with ever-increasing computer power. Applied to geoscientific problems, this leads to a better understanding of massive new observations and thus to a better understanding of processes that have led to these situations in nested inter-dependencies. This is of

great importance both for applied geophysics, with its interest in sustained evaluation of natural resources, and for pure geophysics, which is using its methods to penetrate deeper and deeper into the Earth's body.

Geological modelling serves as the basis for dynamic investigations on the global scale, for reservoir evaluations, for deep geothermal energy projects as well as for the evaluation of hydrogeological problems and many others. In the field of dynamic modeling of geological processes and structures as well as the joint inversion of different geoscientific methods from geophysics, geology and petrology, a rapid development has already begun. Examples include the work of Moorkamp et al. (2016) and Wellmann and Caumon (2018) - among many others. A significant contribution to the integration of different independent information in the modeling process is the publicly available software IGMAS+ ("Interactive Gravity and Magnetic Application System", Anikiev et al. (2020), see also Schmidt et al. (2011) and IGMAS+ Team (2022)). It offers intensive work with geoinformation systems (InterOperable GIS) (Breunig et al., 2000). The concept introduced here considers the fact that intuition together with expert knowledge is still the engine of any scientific innovation in Pure and Applied Geophysics. One of the "gaps" that still exist today is the construction of realistic 3D models which meet the requirements of "real word" models and do not reduce geological structures to all too simple geometries. Therefore, among other things, a practical application tool will be described here that allows to convert geological interpretation ideas quickly and reliably into two- and three-dimensional structural density/susceptibility models.

A second aspect is the optimization of physical and geometrical parameters of a model using a nonlinear optimization method. The question of the plausibility of a model can only be answered if it can be checked whether it satisfies known boundary conditions. Specific conditions are synthetically calculated potential fields, which must match measured ones. It should therefore be possible to automatically optimize a model within given plausibility limits, in which a parameter is allowed to move, taking into account certain aspects, such as the fit of calculated to measured data. Therefore a focus of this work will be on the application and testing of a nonlinear optimization technique - the evolution strategy - for parameter optimization. Alvers (1998) had already verified the performance of the evolution strategy by numerically comparing it with other modern optimization techniques and showed its applicability to the extended problems in an interactive modeling process.

An object-oriented program architecture, implemented using the Java programming language (Anikiev et al., 2020), is intended to enable further developments at a later stage and to ensure operational reliability. This concept should also facilitate the programming of interactive modeling procedures with real-time computation and real-time visualization.

A brief statement on forward and backward modeling

Due to the non-uniqueness of the potential methods, the source cannot be directly inferred from a measured gravity and/or magnetic field. Therefore, one needs methods to calculate the potential and its derivatives of a geological model. Saltus and Blakely (2011) describe this in their article "How to extract unique geologic insights from "non-unique" gravity and magnetic interpretation". In general, modeling distinguishes between forward modeling and inversion. An example of forward geophysical modeling is the solution of a particular differential equation (here: Laplace and Poisson equations) assuming boundary values and/or initial conditions in a region whose geometry and physical properties are well known. Forward modeling is always unique. In backward modeling/inversion, the Earth is commonly interpreted as a "data source" (for example, for gravity and magnetic fields) or as a filter that takes in data (e.g., seismic waves), changes the data input, and outputs results of this filtering process. The input and output data in our case are measured geophysical data/fields. The solution of this "inverse problem" incorporates the measured and processed data and derives physical properties and structures in the Earth for the "Earth filter". Mathematically, the inversion of geophysical data is always based on an "ill-posed problem" because it usually suffers from ambiguity. Within the experimental accuracy many different models cause the same data. This ambiguity can - and must be reduced - by restricting additional knowledge (additional information), but it can never be completely eliminated. Therefore, independent data and information should be interpreted in a mutual approach, as described in the article by Saltus and Blakely (2011) mentioned above. Note that many inversion methods are often limited to single physical parameters (density, velocity, electrical conductivity only), as interpretations are limited by hardware and software features; an exception is, for example, the work of Moorkamp et al. (2016).

The public forward modeling tool IGMAS+

The new concept for the automatic interpretation of potential field measurements is embedded in the forward modeling procedure of the gravity and magnetic field and their derivatives used in our workflow is based on studies by Götze (1984) and Götze and Lahmeyer (1988). It assumes that the subsurface is discretized using polyhedra with planar triangulated facets and voxel cubes in a hybrid form; for more information refer to Schmidt et al., 2011. The gravity or magnetic fields caused by such polyhedra at a point on the model surface are expressed by volume integrals, which are transformed into sums of line integrals by the applications of potential theory (Gauss' theorem, also known as Gauss-Ostrogradsky' theorem). The vertical gravity anomaly of the overall model at a point is obtained by summing the contributions of all polyhedra. The procedure and its applications have been published repeatedly (among others, Götze, 1984; Götze and Lahmeyer, 1988; Schmidt et al., 2011). The workflow described therein allows the exact calculation of the gravimetric and magnetic potential, the three field components, and the six gradients in each case. This

was initially programmed as IGAS (Interactive Gravimetric Application System) in the languages ALGOL and FORTRAN, and in the following decades was continuously developed further (using Java) into the actual already mentioned software package IGMAS+, which is cross-platform and is available for operating systems Windows, Linux, and MacOS (Schmidt et al., 2010; Alvers et al., 2014, 2015a,b; Schmidt et al., 2011, 2020; Anikiev et al., 2020, 2021; Götze et al., 2021). Because of the triangulated model structure, the software can handle complex structures (multi Z surfaces) like the overhangs of salt domes very well. It takes into account remnant and induced magnetization of geological bodies and was applied to the interpretation of borehole gravity and magnetics. The software operates ideally in real-time and the model topology remains conserved during user interactions (IGMAS+ Team, 2022). For a long time, the forward modeling software has been one of the fastest 3D modeling techniques in gravity and magnetic (Li and Chouteau, 1998).

METHODS

Basic ideas for the optimization of potential field models

A basic introduction to concepts of optimization will not be given here, because they are the subject of many modern publications. Important works from the last 40 years that give a comprehensive description of the problem of optimization methods can be found, for example, in Goldberg (1989); Rechenberg (1994); Boschetti and Moresi (2001); Shaw and Srivastava (2007); Rao (2009); Sambridge et al. (2013); Pace et al. (2021). According to the recent review article by Pace et al. (2021), global optimization algorithms can be divided into two main groups: Monte Carlo methods belong to the first one. They are based on random selection of the solution search space. The second group includes methods they call "meta-heuristic", meaning that these optimizations are inspired by processes in nature. Here belong optimizations with the evolution strategy and genetic algorithms, both based on the formalization of behavioral evolution, while a third, swarm intelligence models the social behavior of organisms living in groups.

General hints A typical optimization process that satisfies the "metaheuristic concepts" is the interactive modeling of a geophysical model using a suitable graphical user interface and a computer. An introduction to concepts of interactive modeling related to geoscientific problems can be found, for example, in Götze (1984).

The overall goal of this manuscript is to demonstrate automated optimization methods to support the interactive modeling process, which adjust the parameters to be optimized within given limits using an algorithm, so that the quality function takes a minimum, i.e. the differences between measured and calculated values are as small as possible. From this the question arises, under which conditions this iterative process can be terminated. Possible criteria are:

- the reaching of a prior fixed quality,
- the reaching of a prior fixed number of iterations,
- reaching a predefined quality difference between the iterations n and $n-1$, or
- the interactive intervention in the process.

Also the occurrence of errors, e.g. intersections of the model facets during the optimization of coordinates, must be available as a possible criterion for terminating the algorithm, if such errors cannot be corrected.

Requirements for the optimization algorithm In the introduction it was already mentioned that the inversion of potential field data is not unique. Thus, there exist parameter distributions that differ from each other, but cause the same effect on the potential field. Therefore, the quality function has several equal optima. It must be assumed with high probability that sub-optimal solutions exist under these circumstances. Thus, optimization methods used to optimize potential field models should be able to perform a global search, i.e., to search "uniformly" the n-dimensional parameter space spanned by n parameters. This in turn implies that they must not prematurely converge on local optima. Algorithms that accept only improvements in quality after parameter variation often cannot leave sub-optima because the gradient of the quality function at such a point does not point in the direction of the global optimum, and any variation results in a degradation of quality.

In optimization methods, a distinction is made between methods that use local information of the quality function and those that stochastically search the parameter space. While methods which use local information of the quality function, e.g. that of the gradient, run very fast in the direction of an optimum, possibly a local one, there is a risk of landing on a side optimum in case of premature convergence, for the reason just described. This is plausible because the gradient at a local optimum is in the opposite direction to the global optimum. Stochastic search methods do not use local information of the quality function and search the parameter space based on a random process. The risk of premature convergence to local optima is lower for these methods, since they can in principle leave local optima. Since Newton, a wide variety of optimization methods have been developed, all with one goal in mind: finding the global optimum (minimum or maximum) of a given objective function. Especially in the last forty years, many methods have been published that stochastically search the parameter space and accept or reject solutions found according to certain heuristics and continue searching. A very carefully elaborated overview of optimization methods used today has already been mentioned above (Pace et al., 2021).

For this study, Alvers (1998) studied six methods for dealing with "nonlinear" optimization problems: Monte Carlo search (e.g., Metropolis et al., 1953), Hill Climbing (e.g., Rechenberg, 1973), Simulated Annealing (e.g., Kirkpatrick et al., 1983), Deluge-Algorithm (Dueck, 1993), Threshold-Accepting (Dueck and Scheuer, 1990), Downhill-Simplex (Nelder and Mead, 1965), Genetic Algorithms (Holland, 1992; Schöneburg,

1994), and Evolutionary Strategies (Rechenberg, 1994; Schwefel, 1978, 1995; Hansen and Ostermeier, 1997; Hansen, 2006, 2016). The latter will be considered in more detail in the next section because it is particularly well suited to support automated modeling of potential field data (Alvers et al., 2015b).

Evolution strategy as a tool for lattice optimization

The algorithms of evolutionary strategies (ES), which were developed at about the same time as the genetic algorithms (e.g., Rechenberg, 1994; Schwefel, 1995), are also based on the idea of adapting the principles of natural evolution: recombination, mutation and selection; refer also to Weng (2019). Evolutionary strategies belong to the evolutionary algorithms (EA) and are still primarily used for optimization or search. They can also find satisfactory solutions to problems about whose nature little knowledge is available. This is due to the properties of their natural model (among others: Gerdes et al. (2004); De Jong (2006); Weicker (2015)). The main difference with genetic algorithms is that evolutionary strategies do not require transformation of the parameters to be optimized into a binary (or other) representation. While genetic algorithms require bounds in which the parameters can move, this is not mandatory for evolutionary strategies. Rechenberg (1994) was able to derive a mathematical basis for the convergence behavior of the evolution strategies for certain synthetic test functions, which allows predictions about the convergence behavior. Schwefel (1995) also worked out essential theoretical foundations which form the basis of the evolution strategies. Dirks (1995) and Alvers (1998) showed very early applications of the ES in optimization tasks in geophysics.

The $(\mu+\lambda)$-ES with adaptive step size control The notation "$(\mu+\lambda)$-ES" was introduced by Schwefel (1978). The strategy parameters μ and λ denote the number of parents (models) and number of offspring (models), respectively. With them the algorithm of $(\mu+\lambda)$-ES can be described as follows:

- Step 1: Generate μ different starting models: parent population,
- Step 2: create λ copies by randomly choosing μ starting models (offspring population),
- Step 3: mutate the parameters of the individuals of the offspring population by adding normally distributed random numbers,
- Step 4: evaluate the individuals thus generated by evaluating the quality function and selection,
- Step 5 select the μ best individuals from the parent and offspring population as output for the next generation.

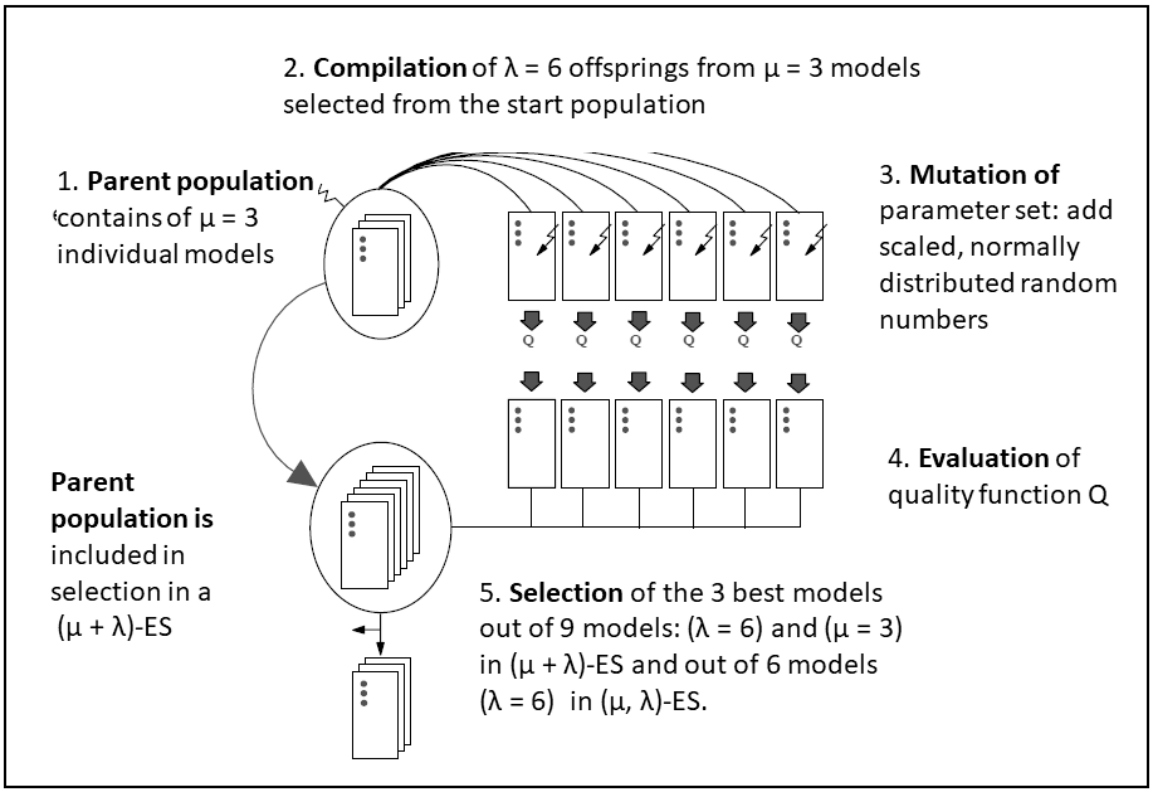


Figure 1: The flow diagram of a (3 + 6)-ES with the symbolism introduced by Rechenberg (1994). With three parent models, six offspring models are generated by random selection. These are mutated and subjected to selection. The three best ones from both the parent and offspring populations become the output for the next generation.

It is essential that the parent population is included in the selection. This is expressed in Figure 1 by the arrow between parent and offspring population.

In Schwefel's notation, the "+" sign symbolizes the inclusion of the parent population in the selection. If the individuals of the parent population are not included in the selection and only the individuals of the progeny population are considered, regressions in the quality landscape are possible in principle. Only the μ best offsprings are then selected, without considering whether they are better or worse than the individuals of the starting population. The $(\mu + \lambda)$-ES can be referred to as parallel hill-climbing. Only the (μ, λ)-ES represents a conceptual extension, since leaving local optima becomes possible in principle. The parent population is then not included in the selection process. For $\mu = \lambda = 1 \rightarrow (1 + 1)$-ES, the evolutionary strategy turns into hill-climbing, because the mutant offspring is accepted only if it has a better quality than the parent vector. The (1, 1)-ES describes a random walk, because every variation is accepted, no matter if it has a worse or better quality.

The mutation of the parameters is done by adding normally distributed random

numbers. This is plausible because variations in biology are often small differences in the expression of different traits of a species that satisfy normal distributions (Sedlag and Weinert, 1987). The introduction of a concept to control the step size allows an adaptation of the algorithm to local conditions of the quality function. The idea is to assign each individual its own "step size" with which the mutation distribution is scaled. In selection, individuals of good quality are chosen and the average of the step sizes of these individuals is taken as the step size for the next generation. Thus, it is possible to adaptively reduce the variations near an optimum in such a way that an optimum can be reached with high accuracy. However, this reduction of the step size is independent of the type of optimum (local or global) and bears the risk of premature convergence. This concept is also a hindrance when dealing with boundary conditions, since step sizes at edges quickly become infinitesimally small and the optimization process stagnates. Even damping the adaptation of step sizes does not solve this problem in principle (Hansen, 2016; Hansen and Ostermeier, 1997). A mutative step width selection proceeds according to the following scheme (Alvers, 1998):

- λ offsprings are generated applying the following rules:
  - Increase (global) step width δ by factor ϵ for $\lambda/3$ individuals,
  - Decrease δ by factor ϵ for $\lambda/3$ individuals,
  - Leave δ untouched for the rest, and
  - Multiply random variations by δ.
- ϵ can be chosen freely and is therefore (besides of μ,ρ,λ) another strategy parameter.
- δ is "global" and can not adapt to different dimensions.

(μ, λ) ES with covariance matrix adaptation (CMA). In our workflow we have considered developments by Hansen and Ostermeier (1997) and Hansen (2016) who extended the concept of evolutionary strategies in such a way that adaptive rotation and scaling of the mutation distribution becomes possible. These algorithms are called CMA-ES algorithms (covariance matrix adaptation). They allow the optimization of scaled quality functions. The algorithm is rather complex to implement, therefore only an explanation of the principle will be given here. The method has been described in detail in online tutorials by Hansen (2016) and Weng (2019). In the following we restrict ourselves to the case of (1, λ)-ES. The algorithm can be easily extended to the (μ, λ)-ES.

A descendant vector $\vec{x}_N^k$ of $k = 1, ..., \lambda$ offsprings is generated as follows:

$$x_N^k = x_E + \delta_E \mathbf{B}_E z_k$$

Where

- $\vec{x} = (x_1, ..., x_n) \in \mathbb{R}^n$ – vector of object variables to be optimized,
- E – index of parent vector,
- N – index of offsprings vector,
- $k = (1, ..., \lambda)$ – index for offsprings,
- $\delta > 0$ – global step size,
- $\vec{z} = (z_1, ..., z_n) \in \mathbb{R}^n$ – vector of normally distributed random numbers,
- $\mathbf{B} = \mathbf{R}n \times n$ – transformation matrix that linearly transforms $\vec{z}$, and
- n – dimension of vector

The main problem is to perform the adaptation of the matrix $\mathbf{B}$ according to the difference vector between parent vector and selected offspring vector. The idea underlying this is the evaluation of successful steps in the parameter space. However, not only the amount of the mutation vector, i.e. the global step size, is considered, but the vector as a whole. Thus, information about the direction of motion in parameter space is also considered (Figure 2). The information of successful mutations is summed and used to adapt the transformation matrix $\mathbf{B}$.

The transformation of the mutation distribution z by $\mathbf{B}$ is independent of the adaptation of the global step size δ. The reason for this is that the global step size can be adapted in a much shorter time than z. To adapt z, $n(n+1)/2$ iterations are required for n parameters, while δ can be adapted in n iterations. The mechanism by which the strategy parameters δ and $\mathbf{B}$ are adapted is very extensive and will not be described here. For a detailed description including implementation notes and tests of the algorithms on synthetic functions, see Hansen (2016) and Weng (2019).

What can we conclude so far? Alvers (1998), Alvers et al. (2013) and Alvers et al. (2014) showed on the basis of extensive test calculations and modeling examples that the evolution strategy with covariance matrix adaptation is an optimization method which, under various constraints (penalty term and geometry limits), can optimize a considerable number of free parameters in a reasonable number of function calls and thus in a reasonable computation time. The algorithms are largely independent of strategy-internal parameters and behave numerically stable to small variations in initial models and the use of sequences of other random numbers. Starting from very different initial situations, they find solutions that are very similar to the best one found and differ slightly in standard deviation. Evolution strategies show satisfactory behavior when it comes to "fine tuning" parameters. Global aspects of the quality function are also well recognized by the adaptation mechanism, which practically always leads to a fast and safe convergence of the algorithm.

The presented algorithms for mutation of the model geometry work satisfactorily with respect to a (1,10)-evolution strategy. However, since the optimization was

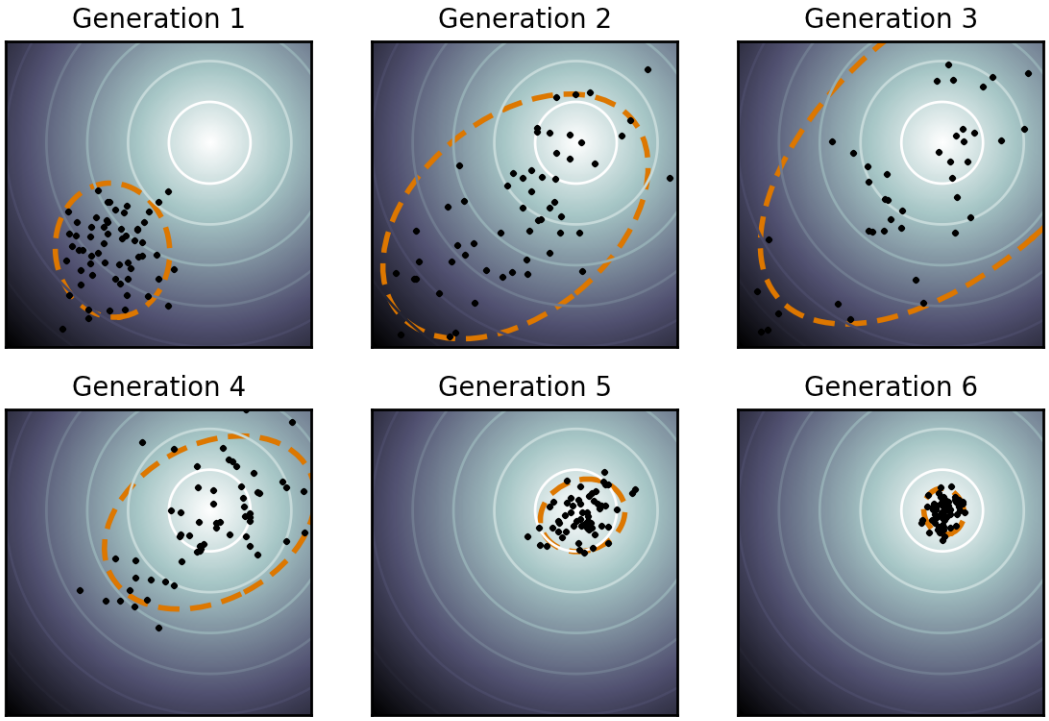


Figure 2: Figure from Wikipedia (2022): "Illustration of an actual optimization run with covariance matrix adaptation on a simple two-dimensional problem. The spherical optimization landscape is depicted with solid lines of equal function-values. The population (dots) is much larger than necessary, but clearly shows how the distribution of the population (dotted line) changes during the optimization. On this simple problem, the population concentrates over the global optimum within a few generations."

repeatedly aborted due to overlapping triangles, the achieved agreement between measured and calculated data is not satisfactory. In other words: CMA-ES' learning/adaptation mechanism is disturbed/destroyed because of too many invalid models coming from bounds checks. To remedy this deficiency, a more suitable mutation procedure must be created that eliminates overlaps a priori. Furthermore, the existing mutation procedure is not suitable for the application of CMA evolution strategies, since an initialization of the strategy has to take place in each generation and thus the adapted directions are lost. A new mutation procedure should therefore be designed to enable the application of CMA processes. Therefore, in the following we present ideas which can overcome the problems described.

The new idea: distort the space, not the model vertices

As already described above, overlaps and/or intersections of individual polygon sides of model triangles, which can arise due to automatic adjustments, form the major problem to be taken very seriously in the geometry optimization of two- and three-dimensional model structures. In principle, this can be prevented by the incorporation of boundary conditions. However, these boundary conditions restrict the parameter space inadmissibly and must be assigned by the user in a complex manner for each individual parameter. For smaller problems, where the number of parameters is not too large, this approach will be feasible with reasonable effort. For more realistic 3D models, however, such an approach is not practical. For example, Hese (2012) describes a combined GOCAD/IGMAS+ density modeling in northern Germany, which involves a model with more than 250 000 triangles and in some cases extremely thin stratigraphic layers (Figure 3).

It must also be considered that the automated treatment of infringed boundary conditions is still far from trivial. Thus, no universal concept for correction of "exceeded" boundary conditions can be given, which is equally suitable for all optimization methods. Also, the automatic generation of boundary conditions is difficult and, if it is possible at all, it would certainly lead to a lasting restriction of the freedom of movement for the individual model vertices, especially if the structural resolution of the model is very high. This is equivalent to the situation when distances between the model faces are very small.

Furthermore, Figure 4 illustrates that a generalization of the structures is often not possible, since even small geological bodies – e.g., the "cap rock" above salt diapirs with an extremely high density have a non-negligible gravity effect.

A way out could be to define so-called "mutation operators", where in principle there can be no "intersections/interpenetrations". However, it is clear that this cannot be achieved by directly changing the geometry parameters (without boundary conditions); this has just been discussed. Therefore, a concept should be sought that optimizes not the model itself, but the (fictitious) space in which it is located. It would then have to be "distorted" in such a way that the geometry of the model

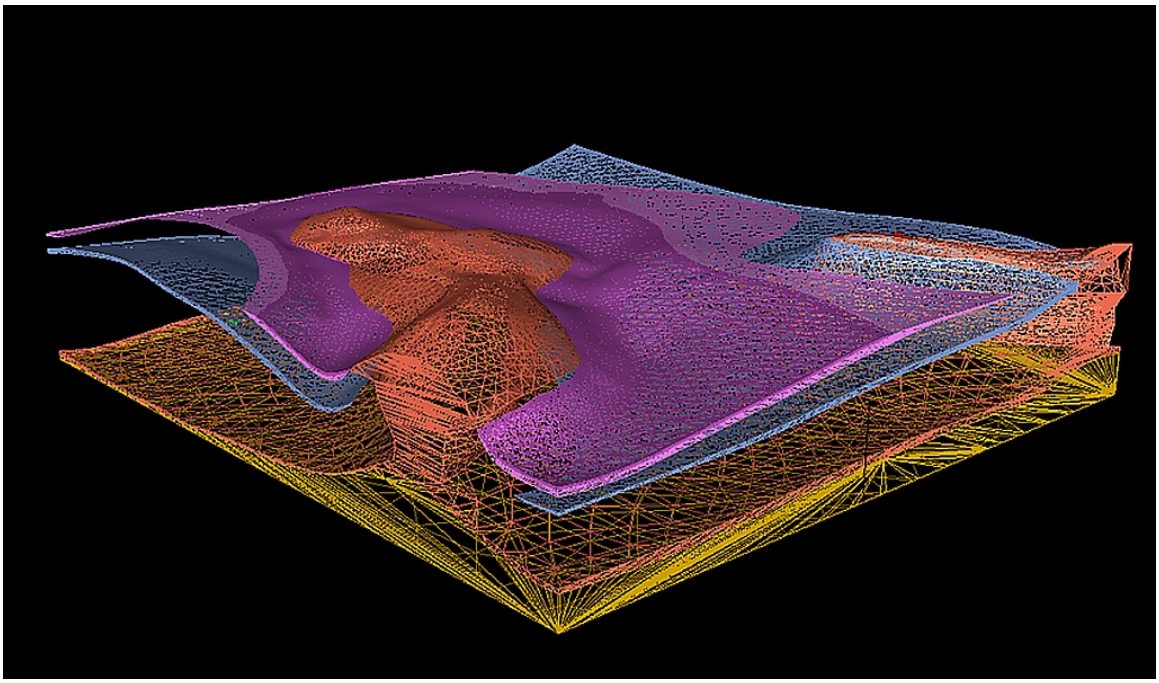


Figure 3: In the PhD thesis of Hese (2012) a salt dome model was constructed which was built on 251 435 triangular facets and partly very small lithological structures.

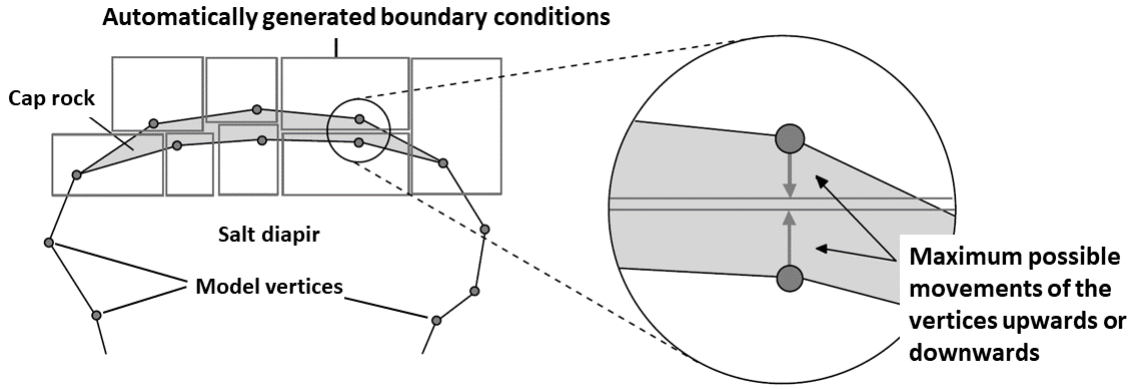


Figure 4: An example of too tight boundary conditions. If the model points are close to each other, boundary conditions strongly restrict the freedom of movement strongly (after Alvers, 1998).

(which in the distorted space also changes its geometry) assumes an optimal form in the sense of the defined quality function.

This "space transformation" should be chosen in such a way that the topology of a structure, e.g., a spanned grid, is preserved. The "parameters" of the transformation procedure are then optimized.

In an early numerical experiment Alvers (1998) implemented so-called manipulators. The "manipulator forces" can be different in all three spatial directions and decrease with the distance to the point according to a given function.

However, the manipulator concept was dropped because when using multiple manipulators the "forces" overlapped and it could no longer be guaranteed that the topology of the model would be preserved.

Space warping A method that arose from the analysis of conformal mappings bypasses the problems described and introduces few new ones. It was also created following methods of rectification of satellite images (among others refers to Lillesand et al. (2015)). Since the eighties of the last century, so-called free-form deformation (FFD) has been applied to the deformation of rigid objects in computer graphics. FFD is a geometric technique and is based on the idea of enclosing an object in a cube or other envelope object and transforming the object within the envelope when the envelope is deformed. Unlike the technique we use, at that time the deformation of the envelope was based on the concept of hyperpatches, the three-dimensional counterparts to parametric curves such as Bézier curves, B-splines, or NURBs. The technique was first described by Sederberg and Parry (1986) and later was extended by Coquillart (1990) to a technique called "extended freeform deformation". The significant difference with these works, however, is that the deformation of space does not serve a graphical representation, but the spring system used is the basis for the inversion of potential fields.

A specific part of the two- or three-dimensional space is enclosed by a regular grid. In the area of this two- or three-dimensional grid/lattice lies the model structure that is to be optimized. Each geometry point of the model has a unique position in this grid relative to a specific grid element (cf. Figure 5a left). The idea is to change the lattice in such a way that the shape of the model located in this enveloped area takes on an optimal shape in terms of the quality function.

After distorting the grid, each point within the respective grid element can be linearly interpolated (Figure 5b). It must be noted that the grid elements do not intersect themselves. One possibility is to vary the grid spacing $(d_{ij})_x$ and $(d_{ij})_z$ – in the three-dimensional case also $(d_{ij})_y$. The grid spacing must not become smaller than zero. A predefined maximum grid width $maxg$ makes sense, since it is known a priori in which interval $[0, maxg]$ a "grid width" $(d_{ij})_k$ with $k = x, z, y$ is allowed to move.

The optimization of the model parameters was thus shifted to the optimization of

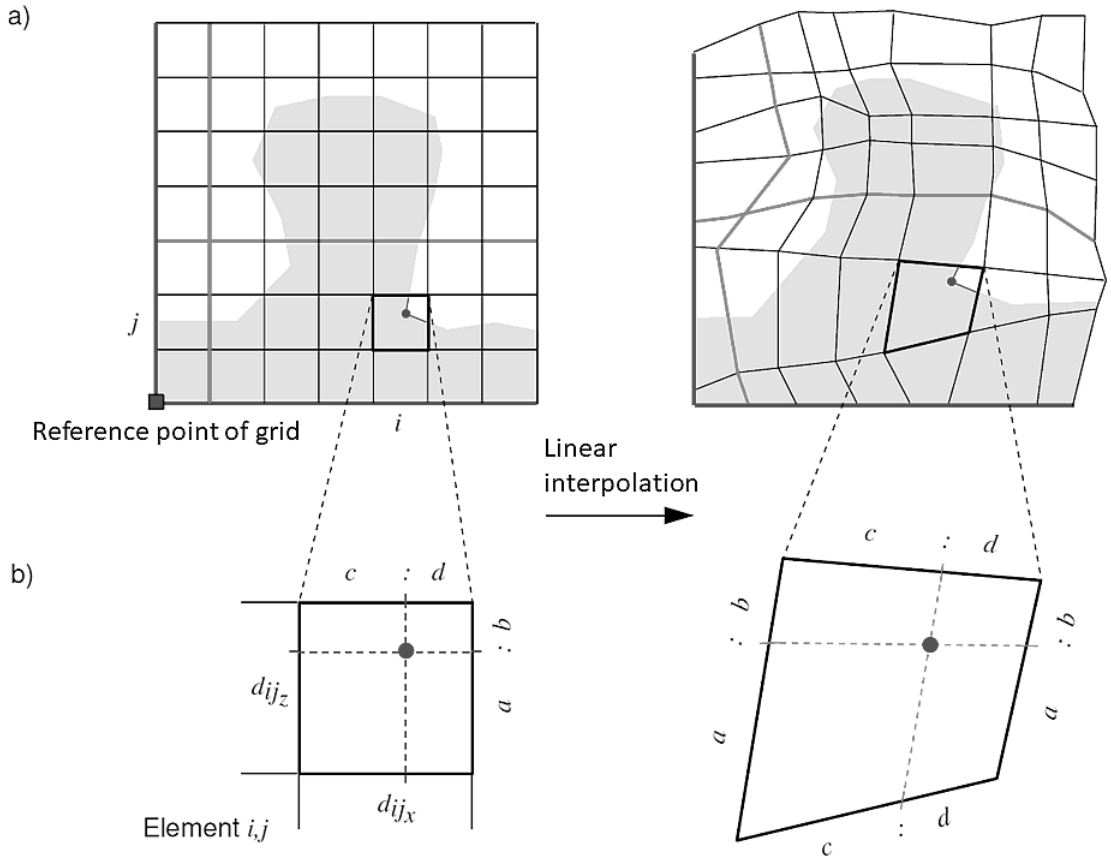


Figure 5: The scheme of space warping. A regular grid is superimposed on the model to be optimized (a, left). Each vertex can be uniquely assigned to a lattice element i, j vertex. (a) shows the original and the distorted lattice. (b) Due to the mutations of the grid spacings $(d_{ij})_x$ and $(d_{ij})_z$ the grid/lattice is distorted. The vertices of the model can be linearly interpolated according to the new vertices of the model element (after Alvers et al., 2013).

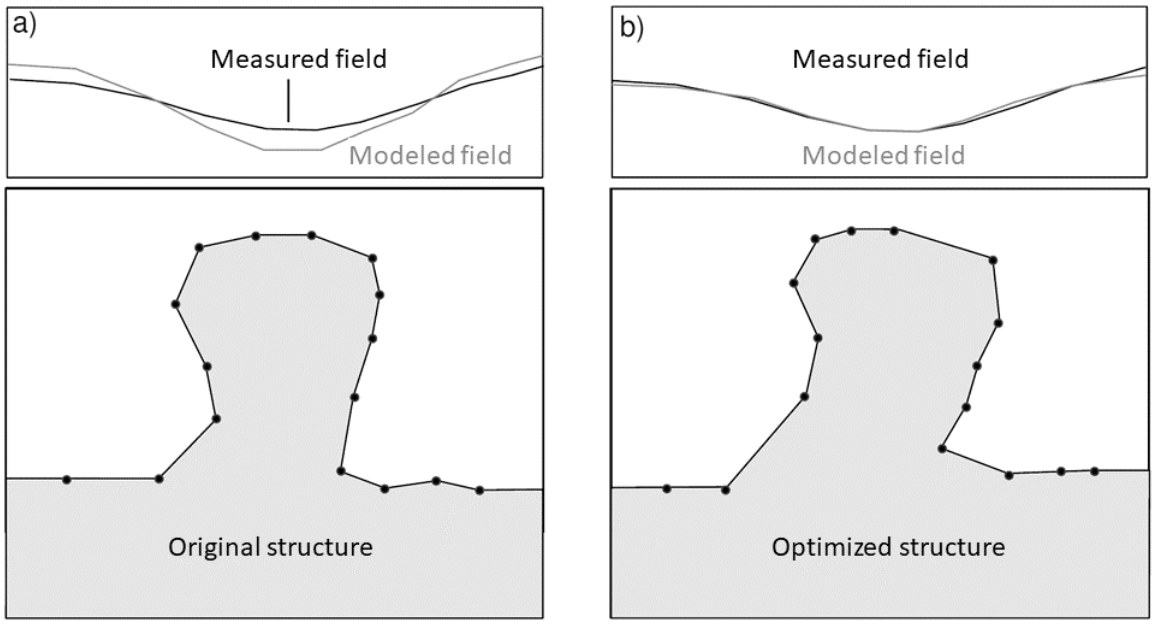


Figure 6: After the optimization process, the fit of the measured and modeled field is optimal (schematic representation and not scaled).

the grid lengths $(d_{ij})_x$, $(d_{ij})_y$, $(d_{ij})_z$. Thereby, the safe preservation of the topology of the lattice and thus also the preservation of the topology of the model was gained. By "preservation of model topology" we mean that in the model to be modified no triangle sides interpenetrate. In this case the used algorithm of Götze and Lahmeyer (1988) would give wrong results for the potential fields and their gradients.

Figure 6shows the result of the space transformation. It should be noted that all grid changes must be relative from one element to the next. Therefore, the order in which each grid spacing is processed is also important. Consequently, the question of a "reference point" in the model arises. This reference point, from which all variations are performed, must be specified interactively by the user in this procedure. In Figure 5a it was arbitrarily placed in the lower left corner of the grid. Such a reference system limits the freedom of the optimization a priori, since it could be that starting from a badly chosen reference point, the searched shape is in principle not constructible. However, the position of this point, i.e., the position of the grid relative to the model, could itself be optimized.

Intensive 3D testing results in some drawbacks. If one considers (optimizes) only the sides (in Figure 7, legs a and b) it becomes clear that these are not independent of each other. So we want to find a method which is more elegant and does not have these disadvantages (check and dependence of the grid sides).

Therefore, the question remains: how to warp space and keep its topology?

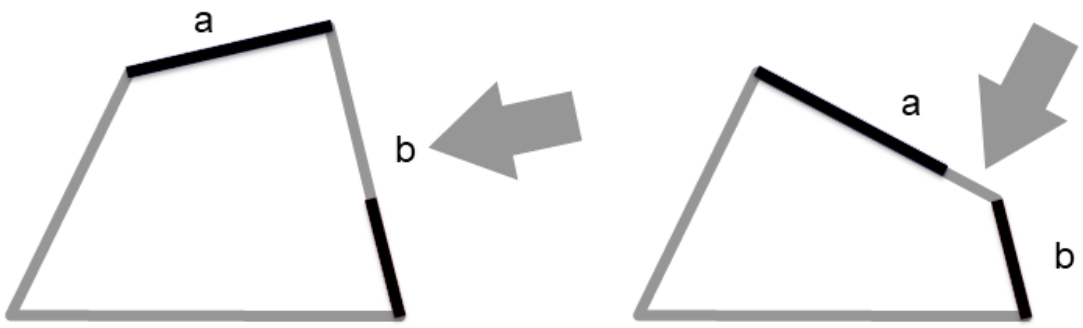


Figure 7: Grid legs a and b are not independent.

Solution: the "spring algorithm"

As has just been shown, rigid connections of the lattice nodes are a drawback if the modeling procedure is to be performed automatically. A possible way out is the idea (Matthias Zschunke, pers. comm.) to connect the space lattice points not by rigid connections but by elastic springs. The forces that then change the lattice with the possibilities and advantages of the evolution strategy presented in the last section are easy to calculate and the resulting system of equations to solve with manageable effort. Such an approach would ensure that no triangles would intersect and to some extent it would also be possible to define boundary conditions for the inversion.

Suggested methodology: springs Previously, we explained that the inversion of 3D regions of the density model is only performed in areas defined/wrapped by a 3D lattice. The model domain is updated by the optimization tool only within this lattice. Figure 8 makes clear that it is not the positions of the vertices describing the model that are changed, but the spanned space. This is a new aspect for the inversion within the modeling software.

The parameterization of the model is done by using the interior points of the lattice to warp the space, as outlined earlier. The grid size, or its resolution, can be freely chosen to work out model details locally or to initiate more regional changes, depending on the boundary conditions or the sensitivity of the model in the corresponding depth ranges. The user can watch the model update live on screen (e.g., Alvers et al., 2015a,b) and stop or restart the process at any time.

To account for non-uniqueness in the interpretation of potential data, the inversion of the model geometry must be constrained by other independent information. For this constraint of the inversions we rely on the fact that the grid used for the parameterization is positioned in a way that model parts are excluded from the inversion (see, for example, Figure 8). But one must be careful: in cases where the model boundaries are not horizontal, this simple option of a limited inversion is not applicable. This results in the fact that a single model interface cannot be excluded

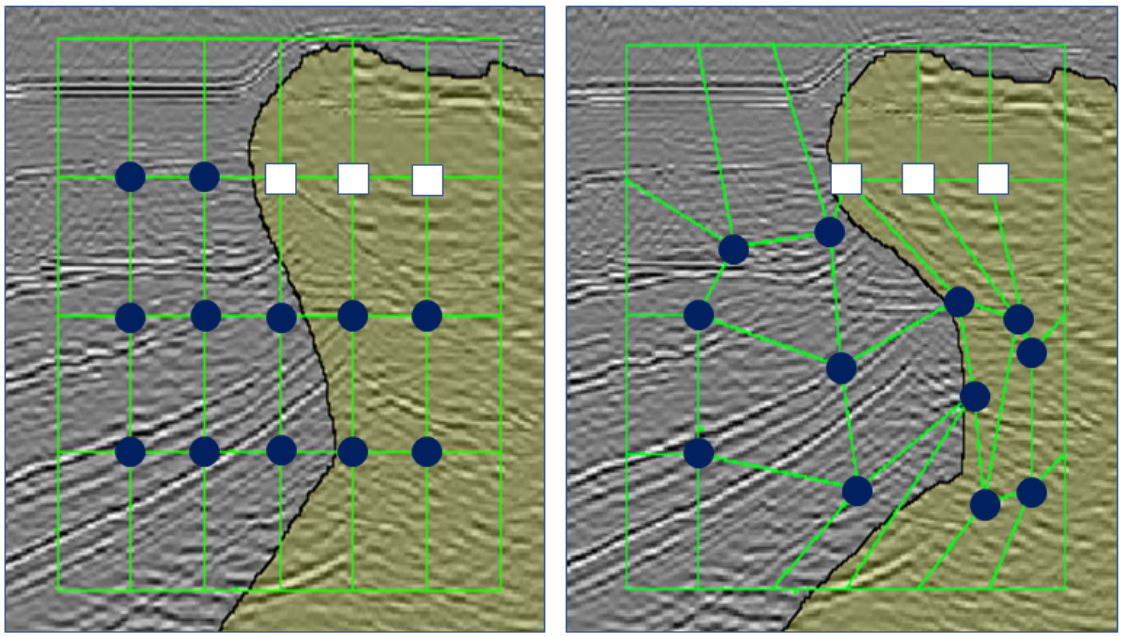


Figure 8: The flank of a salt dome is to be modeled (greenish) against the background of seismic processing results (gray). By defining the position of some points of the grid for the inversion the user can exclude a part of the model from the inversion calculation (white squares), the blue dots define the part of the model at the salt dome flank which is defined for the spatial distortion. Here, we did not invert the upper parts of the salt dome because this area is already well modeled (modified after Alvers et al., 2015b). The right panel shows the situation when the lattice nodes have already been changed (via warping after the mutation step of the ES) and the black line shows the updates of the model vertices by a trilinear interpolation.

from the inversion, because this would make model topology inconsistent in many practical cases.

Preview and numerical background of the idea Let us now derive equations in 2D case ((x, z) space) for a system of springs connecting $(I+1) \times (J+1)$ nodes placed in a form of a lattice with $I+1$ rows and $J+1$ columns (Figure 9). Nodes on the border of the lattice are fixed, i.e. they can't change position, whereas $I \times J$ nodes inside the lattice are free, i.e. are able to move.

A spring $(i,j)-(l,m)$ connects two adjacent nodes $\mathrm{N}_{i,j}$ and $\mathrm{N}_{l,m}$ at positions $\mathbf{p}_{i,j}$ and $\mathbf{p}_{l,m}$, where $i = 0, ..., I+1$, $j = 0, ..., J+1$ and $l = i-1, i+1$, $m = j-1, j+1$. Each spring has a rest length (which is the length in non-deformed state), $\mathrm{r}_{(i,j)-(l,m)}$. If the distance $||\mathbf{p}_{i,j} - \mathbf{p}_{l,m}||$ between the nodes (endpoints) of a spring is equal to $\mathrm{r}_{(i,j)-(l,m)}$, there is no contribution from the spring to the total force acting on either node. If the distance is different, there is a spring force acting on each node. The magnitude of the force $\mathbf{F}_{(l,m)-(i,j)}$ acting on node $\mathrm{N}_{i,j}$ at $\mathbf{p}_{i,j}$, is, according to the Hooke's law:

$$||\mathbf{F}_{(l,m)-(i,j)}|| = \mathrm{k}_{(l,m)-(i,j)} \left(||\mathbf{p}_{l,m} - \mathbf{p}_{i,j}|| - \mathrm{r}_{(l,m)-(i,j)} \right). \tag{1}$$

The force acts along the spring, that is, the line connecting the nodes. The coefficient $\mathrm{k}_{(i,j)-(l,m)}$ in equation 1 is called spring constant or stiffness. Note also that $\mathrm{k}_{(i,j)-(l,m)} = \mathrm{k}_{(l,m)-(i,j)}$ and $\mathrm{r}_{(i,j)-(l,m)} = \mathrm{r}_{(l,m)-(i,j)}$.

In vector form, the Hooke's law can be written as

$$\mathbf{F}_{(l,m)-(i,j)} = \mathrm{k}_{(l,m)-(i,j)} \left(||\mathbf{p}_{l,m} - \mathbf{p}_{i,j}|| - \mathrm{r}_{(l,m)-(i,j)} \right) \frac{\mathbf{p}_{i,j} - \mathbf{p}_{l,m}}{||\mathbf{p}_{l,m} - \mathbf{p}_{i,j}||}. \tag{2}$$

Expression $(\mathbf{p}_{i,j} - \mathbf{p}_{l,m})/||\mathbf{p}_{l,m} - \mathbf{p}_{i,j}||$ in equation 2 is just a unit vector pointing along the spring away from the node $\mathrm{N}_{i,j}$ at position $\mathbf{p}_{i,j}$ with coordinates $(\mathrm{x}_{i,j}, \mathrm{z}_{i,j})$.

The total force $\mathbf{F}_{i,j}$ acting on a free node $\mathrm{N}_{i,j}$ in 2D case consists of four spring forces:

$$\mathbf{F}_{(i,j)} = \mathbf{F}_{(i-1,j)-(i,j)} + \mathbf{F}_{(i+1,j)-(i,j)} + \mathbf{F}_{(i,j-1)-(i,j)} + \mathbf{F}_{(i,j+1)-(i,j)}. \tag{3}$$

Equilibrium condition for the system of springs (Figure 9) is that all forces acting on all moving nodes are zero:

$$\mathbf{F}_{(i,j)} = 0, \quad i = 1, ..., I; \quad j = 1, ..., J \tag{4}$$

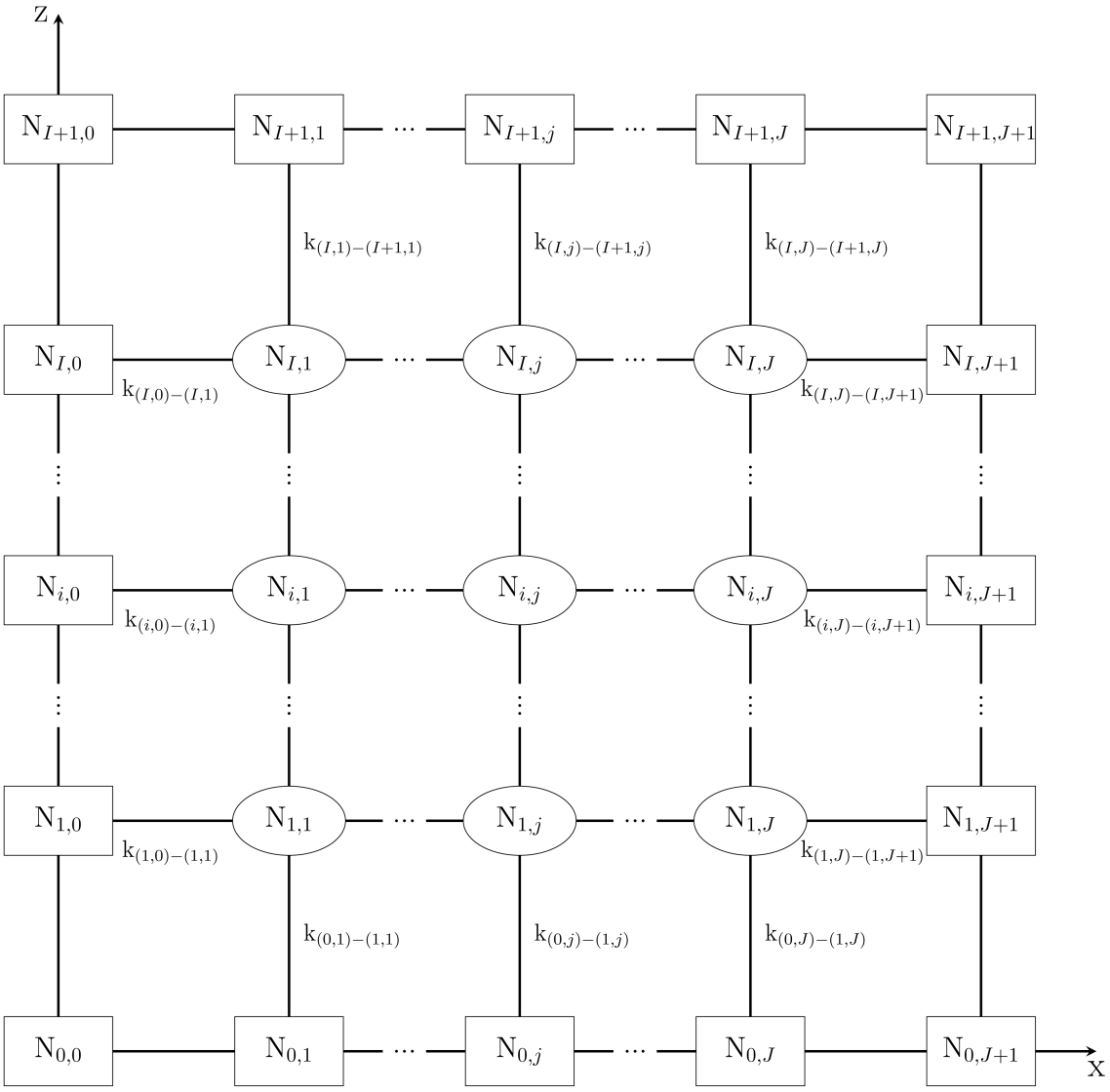


Figure 9: Illustration of a spring system in two dimensions with free nodes (ellipses) inside the lattice and fixed nodes (rectangles) on the borders. $k_{(i,j)-(l,m)}$ denote the spring constant (stiffness) of a spring connecting the nodes $N_{i,j}$ and $N_{l,m}$. Size of the lattice is $I+1$ rows (z direction) by $J+1$ columns (x direction).

Together equation 3 and equation 4 give:

$$\begin{aligned}
&\mathrm{k}_{(i-1,j)-(i,j)}\left(||\mathbf{p}_{i-1,j}-\mathbf{p}_{i,j}||-\mathrm{r}_{(i-1,j)-(i,j)}\right)\frac{\mathbf{p}_{i,j}-\mathbf{p}_{i-1,j}}{||\mathbf{p}_{i-1,j}-\mathbf{p}_{i,j}||}+\\
&\mathrm{k}_{(i+1,j)-(i,j)}\left(||\mathbf{p}_{i+1,j}-\mathbf{p}_{i,j}||-\mathrm{r}_{(i+1,j)-(i,j)}\right)\frac{\mathbf{p}_{i,j}-\mathbf{p}_{i+1,j}}{||\mathbf{p}_{i+1,j}-\mathbf{p}_{i,j}||}+\\
&\mathrm{k}_{(i,j-1)-(i,j)}\left(||\mathbf{p}_{i,j-1}-\mathbf{p}_{i,j}||-\mathrm{r}_{(i,j-1)-(i,j)}\right)\frac{\mathbf{p}_{i,j}-\mathbf{p}_{i,j-1}}{||\mathbf{p}_{i,j-1}-\mathbf{p}_{i,j}||}+\\
&\mathrm{k}_{(i,j+1)-(i,j)}\left(||\mathbf{p}_{i,j+1}-\mathbf{p}_{i,j}||-\mathrm{r}_{(i,j+1)-(i,j)}\right)\frac{\mathbf{p}_{i,j}-\mathbf{p}_{i,j+1}}{||\mathbf{p}_{i,j+1}-\mathbf{p}_{i,j}||}=0,\\
&\qquad\qquad i=1,...,I;\quad j=1,...,J
\end{aligned}\tag{5}$$

Spring constants and rest lengths in equations 5 are known, however we are free to choose arbitrary rest lengths (except that they must be positive and must not be all equal). Initially we assume a value of 1 for the spring constants so that the system is in equilibrium, and later the spring constants are updated by the evolution strategy (mutation) on the basis of the quality function. At the same time, we are free to choose the rest lengths after each update of the spring constants. In order to simplify equations 5 we can assume that rest length of each spring is twice as big as its length when it is deformed:

$$\begin{aligned}
\mathrm{r}_{(i-1,j)-(i,j)}&=2||\mathbf{p}_{i-1,j}-\mathbf{p}_{i,j}||\\
\mathrm{r}_{(i+1,j)-(i,j)}&=2||\mathbf{p}_{i+1,j}-\mathbf{p}_{i,j}||\\
\mathrm{r}_{(i,j-1)-(i,j)}&=2||\mathbf{p}_{i,j-1}-\mathbf{p}_{i,j}||\\
\mathrm{r}_{(i,j+1)-(i,j)}&=2||\mathbf{p}_{i,j+1}-\mathbf{p}_{i,j}||\\
&\qquad i=1,...,I;\quad j=1,...,J.
\end{aligned}\tag{6}$$

Assumptions 6 allow us to rewrite equations 5:

$$\begin{aligned}
&\mathrm{k}_{(i-1,j)-(i,j)}\left(\mathbf{p}_{i,j}-\mathbf{p}_{i-1,j}\right)+\mathrm{k}_{(i+1,j)-(i,j)}\left(\mathbf{p}_{i,j}-\mathbf{p}_{i+1,j}\right)+\\
&\mathrm{k}_{(i,j-1)-(i,j)}\left(\mathbf{p}_{i,j}-\mathbf{p}_{i,j-1}\right)+\mathrm{k}_{(i,j+1)-(i,j)}\left(\mathbf{p}_{i,j}-\mathbf{p}_{i,j+1}\right)=0,\\
&\qquad\qquad i=1,...,I;\quad j=1,...,J
\end{aligned}\tag{7}$$

Projection of the set of equations 7 to axis x then gives:

$$\begin{aligned}
&\mathrm{k}_{(i-1,j)-(i,j)}\left(\mathrm{x}_{i,j}-\mathrm{x}_{i-1,j}\right)+\mathrm{k}_{(i+1,j)-(i,j)}\left(\mathrm{x}_{i,j}-\mathrm{x}_{i+1,j}\right)+\\
&\mathrm{k}_{(i,j-1)-(i,j)}\left(\mathrm{x}_{i,j}-\mathrm{x}_{i,j-1}\right)+\mathrm{k}_{(i,j+1)-(i,j)}\left(\mathrm{x}_{i,j}-\mathrm{x}_{i,j+1}\right)=0,\\
&\qquad\qquad i=1,...,I;\quad j=1,...,J,
\end{aligned}\tag{8}$$

and projection to z-axis gives:

$$\begin{aligned}
&\mathrm{k}_{(i-1,j)-(i,j)}\left(\mathrm{z}_{i,j}-\mathrm{z}_{i-1,j}\right)+\mathrm{k}_{(i+1,j)-(i,j)}\left(\mathrm{z}_{i,j}-\mathrm{z}_{i+1,j}\right)+\\
&\mathrm{k}_{(i,j-1)-(i,j)}\left(\mathrm{z}_{i,j}-\mathrm{z}_{i,j-1}\right)+\mathrm{k}_{(i,j+1)-(i,j)}\left(\mathrm{z}_{i,j}-\mathrm{z}_{i,j+1}\right)=0,\\
&\qquad\qquad i=1,...,I;\quad j=1,...,J.
\end{aligned}\tag{9}$$

Set of equations 8 transform into the following linear system of equations:

$$A\mathbf{x} = \mathbf{b}_{\mathrm{x}}, \tag{10}$$

where vector $\mathbf{x}$ consists of x-coordinates of the free nodes to be found:

$$\mathbf{x} = \begin{pmatrix} \mathrm{x}_{1,1} \\ \mathrm{x}_{2,1} \\ \dots \\ \mathrm{x}_{I,1} \\ \mathrm{x}_{1,2} \\ \mathrm{x}_{2,2} \\ \dots \\ \mathrm{x}_{I-1,J} \\ \mathrm{x}_{I,J} \end{pmatrix},$$

and vector $\mathbf{b}_{\mathrm{x}}$ contains x-coordinates of the fixed lattice nodes:

$$\mathbf{b}_{\mathrm{x}} = \begin{pmatrix} \mathrm{k}_{(0,1)-(1,1)}\mathrm{x}_{0,1} + \mathrm{k}_{(1,0)-(1,1)}\mathrm{x}_{1,0} \\ \mathrm{k}_{(2,0)-(1,1)}\mathrm{x}_{2,0} \\ \dots \\ \mathrm{k}_{(I+1,1)-(I,1)}\mathrm{x}_{I+1,1} + \mathrm{k}_{(I,0)-(I,1)}\mathrm{x}_{I,0} \\ \mathrm{k}_{(0,2)-(1,2)}\mathrm{x}_{0,2} \\ 0 \\ \dots \\ \mathrm{k}_{(I+1,J)-(I,J)}\mathrm{x}_{I+1,J} \\ \mathrm{k}_{(I+1,J)-(I,J)}\mathrm{x}_{I+1,J} + \mathrm{k}_{(I,J+1)-(I,J)}\mathrm{x}_{I,J+1} \end{pmatrix}.$$

Elements of matrix A in equation 10 consist of the known spring constants:

$$A = \begin{pmatrix} \mathrm{d}_1 & -\mathrm{k}_{(2,1)-(1,1)} & \dots & 0 & -\mathrm{k}_{(1,2)-(1,1)} & 0 & \dots & 0 & 0 \\ -\mathrm{k}_{(1,1)-(2,1)} & \mathrm{d}_2 & \dots & 0 & 0 & -\mathrm{k}_{(2,2)-(2,1)} & \dots & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & \mathrm{d}_I & -\mathrm{k}_{(I+1,1)-(I,1)} & 0 & \dots & 0 & 0 \\ -\mathrm{k}_{(1,1)-(1,2)} & 0 & \dots & -\mathrm{k}_{(1,1)-(I+1,1)} & \mathrm{d}_{I+1} & -\mathrm{k}_{(2,2)-(1,2)} & \dots & 0 & 0 \\ 0 & -\mathrm{k}_{(2,1)-(2,2)} & \dots & 0 & -\mathrm{k}_{(1,2)-(2,2)} & \mathrm{d}_{I+2} & \dots & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & 0 & 0 & 0 & \dots & \mathrm{d}_{IJ-1} & -\mathrm{k}_{(I,J)-(I-1,J)} \\ 0 & 0 & \dots & 0 & 0 & 0 & \dots & -\mathrm{k}_{(I-1,J)-(I,J)} & \mathrm{d}_{IJ} \end{pmatrix},$$

where the terms on the diagonal read:

$$\mathrm{d}_{i+I(j-1)} = \mathrm{k}_{(i-1,j)-(i,j)} + \mathrm{k}_{(i+1,j)-(i,j)} + \mathrm{k}_{(i,j-1)-(i,j)} + \mathrm{k}_{(i,j+1)-(i,j)}, \quad i = 1, ..., I;\ j = 1, ..., J.$$

A linear system similar to equation 10 can be easily written for z-coordinates:

$$A\mathbf{z} = \mathbf{b}_{\mathrm{z}}.$$

Note that matrix A remains unchanged, and vector $\mathbf{b}_z$ reads:

$$\mathbf{b}_z = \begin{pmatrix} k_{(0,1)-(1,1)}z_{0,1} + k_{(1,0)-(1,1)}z_{1,0} \\ k_{(2,0)-(1,1)}z_{2,0} \\ \dots \\ k_{(I+1,1)-(I,1)}z_{I+1,1} + k_{(I,0)-(I,1)}z_{I,0} \\ k_{(0,2)-(1,2)}z_{0,2} \\ 0 \\ \dots \\ k_{(I+1,J)-(I,J)}z_{I+1,J} \\ k_{(I+1,J)-(I,J)}z_{I+1,J} + k_{(I,J+1)-(I,J)}z_{I,J+1} \end{pmatrix}.$$

A system like equation 10 is solved through LU decomposition (lower-upper composition) by using a combination of forward and back substitution. Knowing the LU decomposition for matrix A we can substitute it with:

$$LU\mathbf{x} = \mathbf{b}_x,$$

where L is a lower triangular matrix and U is an upper triangular matrix.

With the equations:

$$\begin{aligned} U\mathbf{x} &= L^{-1}\mathbf{b}_x; \\ \mathbf{x} &= U^{-1}(L^{-1}\mathbf{b}_x), \end{aligned}$$

where we solve $L^{-1}\mathbf{b}_x$ using the forward substitution and $\mathbf{x} = U^{-1}(L^{-1}\mathbf{b}_x)$ using backward substitution.

After finding solutions for the systems of equations for x-coordinates and z-coordinates (in 3D case a similar system will appear for y-coordinates), the lattice free node positions are adjusted according to the new coordinates and than a trilinear interpolation is used to calculate the new coordinates of the "wrapping" mesh.

As already explained earlier, the basic idea behind the new method is to warp space by deforming a lattice which encloses the structure to be optimized.

Here is the algorithm:
START

(1) The initial model is separated into the individual non-transformed lattice cells.

(2) In the **mutation** step, new spring constants are determined and updated,

- the original lattice is warped by the solution of equation 5,
- a trilinear interpolation is applied to the initial model, resulting in a "new" model,
- the potential field component(s) is/are calculated and

(3) In the **evaluation** step the fitness functions Q are calculated.

(4) Finally, the **selection step** chooses the best models following the "plus" CMA-ES. If the stop criterion has not been reached yet, the process starts over again with step (1).

END
This procedure must be followed for all "offspring" models.

Rewind the results of the interactive inversion The derived procedure for inversion of potential data using evolution strategy and space warping gives the user the ability to interactively intervene at any stage of the inversion process (Figure 10). This is possible by an intervention in the convergence diagram of the inversion. In this interaction with the process, the user can return to any modeling step that was created during the automatic inversion. Within a few seconds, the responses are recalculated for the selected model. Refer also to the example SEAM model in the next section to get more information on the inversion process parameters: a (12,24) CMA-ES was applied, and standard deviation for spring constants were set to 1 and stop fitness is set to 10^{-12} mGal. The user is able to redefine the parameters and start a new inversion process.

EXAMPLES

So far we have shown that a geometry inversion of the models is supported by two methodological components: (i) the CMA-ES and (ii) the newly proposed space warping tool explained before. These two components have been integrated into the used software and are in beta testing. The new method will prevail in any case where geological models of the subsurface are approximated with triangular facets - e.g. "SKUA-GOCAD models" (`https://www.pdgm.com/products/skua-gocad`). Now, two practical adaptations will be presented, one based on the SEAM test model (SEG Advanced Modeling Program) of SEG and EAGE (Pangman, 2007), and the second based on a processing of salt structures in the North German Basin (unpublished industry project and gravity data of HJG).

SEAM model

The synthetic 3D SEAM model was originally developed to test new seismic software tools and to verify the quality of seismic modeling techniques. It is based on a complex salt structure which is similar to a realistic structure found in the Gulf of Mexico. Therefore, the model is based on seismic velocities defined in 24.6 million voxels and about 260 000 triangles (Figure 11).

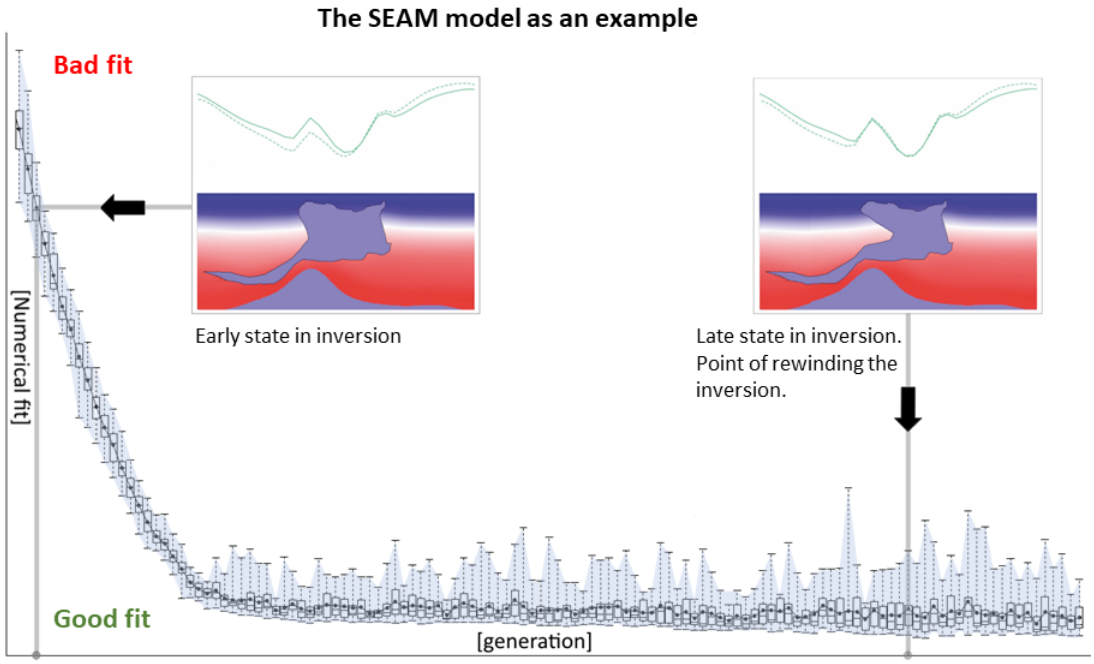


Figure 10: The convergence diagram of the inversion, in which the user can interactively intervene (after Alvers et al., 2015b). Example shows the inversion of the SEAM model (Pangman, 2007)) at an early stage of the inversion and at a very late stage (thick arrow). The colors in the SEAM model represent model densities; "red" symbolizes higher and "blue" lower densities compared to a density of the background model (white color). For each iteration a box-whisker plot provides the user with information on the maximum, minimum, mean and median values with its lower and upper quartiles.

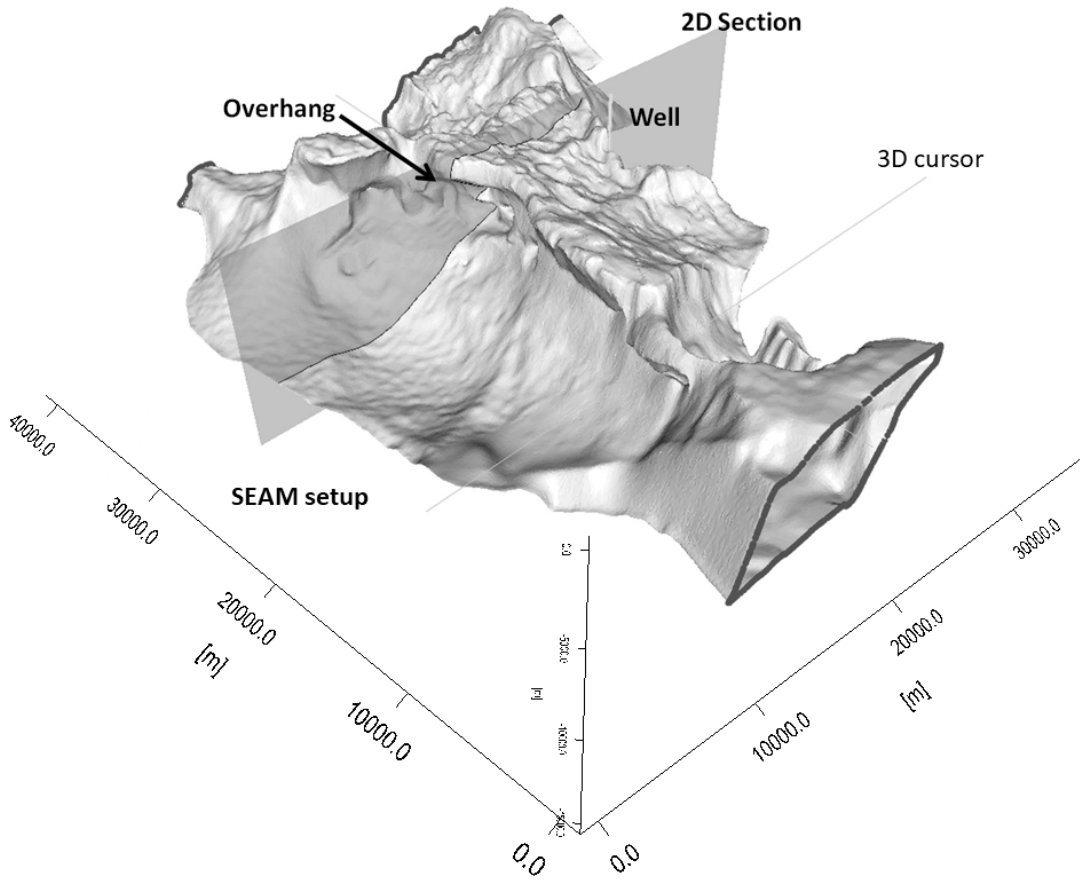


Figure 11: A view of the SEAM model (Pangman, 2007) to test the IGMAS+ inversion procedure. The model geometry consists of 24.6 million voxels and 256 331 triangles. There is also a virtual well available, the position of the 3D cursor and the 2D section shows the vertical model cross section of Figure 12.

We converted these velocities into densities using the "Gardner's relation" (Gardner et al., 1974) and used them as a test bed for our new inversion. The result is high-resolution stratigraphic bedding in the vicinity of an intricately built-up salt pillow and an upwarping of the salt horizon (Zechstein) (blue color in Figure 12). The gravity effect and vertical gradient of gravity of this initial model was calculated by forward modeling and, in the absence of "measured" fields, used later in comparison with the inversion results. By hand, we then modified the salt flank of the overhang to simulate the effects of incorrect seismic imaging.

This incorrect salt model was inserted into a simplified sediment model described by a voxel cube. It was derived from a smoothed velocity model and converted to density. The remaining anomaly in the vertical gradient of gravity is explained by the spurious salt model (modified overhang) and the simplified sediment model. The interactive inversion was applied to the area of the salt overhang, assuming that the top and bottom of the salt are known (Figure 12a). The inversion was able to recover most of the original salt geometry (see Figure 12c). Here a (12,24) CMA-ES was applied, the "comma strategy" in which the parent population is not considered in the selection process. For the result, 6000 generations with 17 offsprings per generation were calculated. Thus, about 100 000 calculation steps had to be performed for the triangle and voxel calculation. Standard deviation for spring constants were set to 1 and the stop fitness is set to 10^{-12} mGal. The recovered model is not perfect, of course, but would provide a good basis for the next imaging step. Assuming that the basis of the salt is uncertain, the solution found is of course different and reflects the ambiguity of the potential field data. In our example, the vertical gravity gradient was used for the inversion. Other components of the gravity field and/or magnetic measurements can also be included in the inversion.

In Figure 13 the grid for the space warping is shown together with the salt some flank. The yellow dots inside the grid are the positions where the grid and the space are bent. In the additional materials the Movie 1 of the active space wrapping can be seen. In the small map on the right side, the quality of the fit changes with each iteration. The residual field of measured and modeled vertical gradient is shown. Figure 14 is a zoomed in version of the small residual map in Figure 13 (right). The standard deviation of 1.18 E was found before optimizing the modified flank of the salt pillow. After six hours of computation (3-Quade 6600, 2014), it is then 0.14 E and thus reduced in size by 20 percent. The model part that was inverted had the following spatial dimensions: 10 km x 10 km x 6 km. The voxel size in the SEAM model is 120 m x 120 m x 60 m and this results in the sum of the voxels to be inverted of about 700 000 voxels; in addition there are about 50 000 triangular facets that were included in the inversion. It should be considered that the vertical gradients are not really measured, but synthetic in nature and therefore, no errors can be given for the reference field. In the absence of real errors, the execution time was arbitrarily set at 6 hours to achieve a small deviation from the comparison field. The deficits at the edges (in blue) can only be resolved by enlarging the grid - they are local effects.

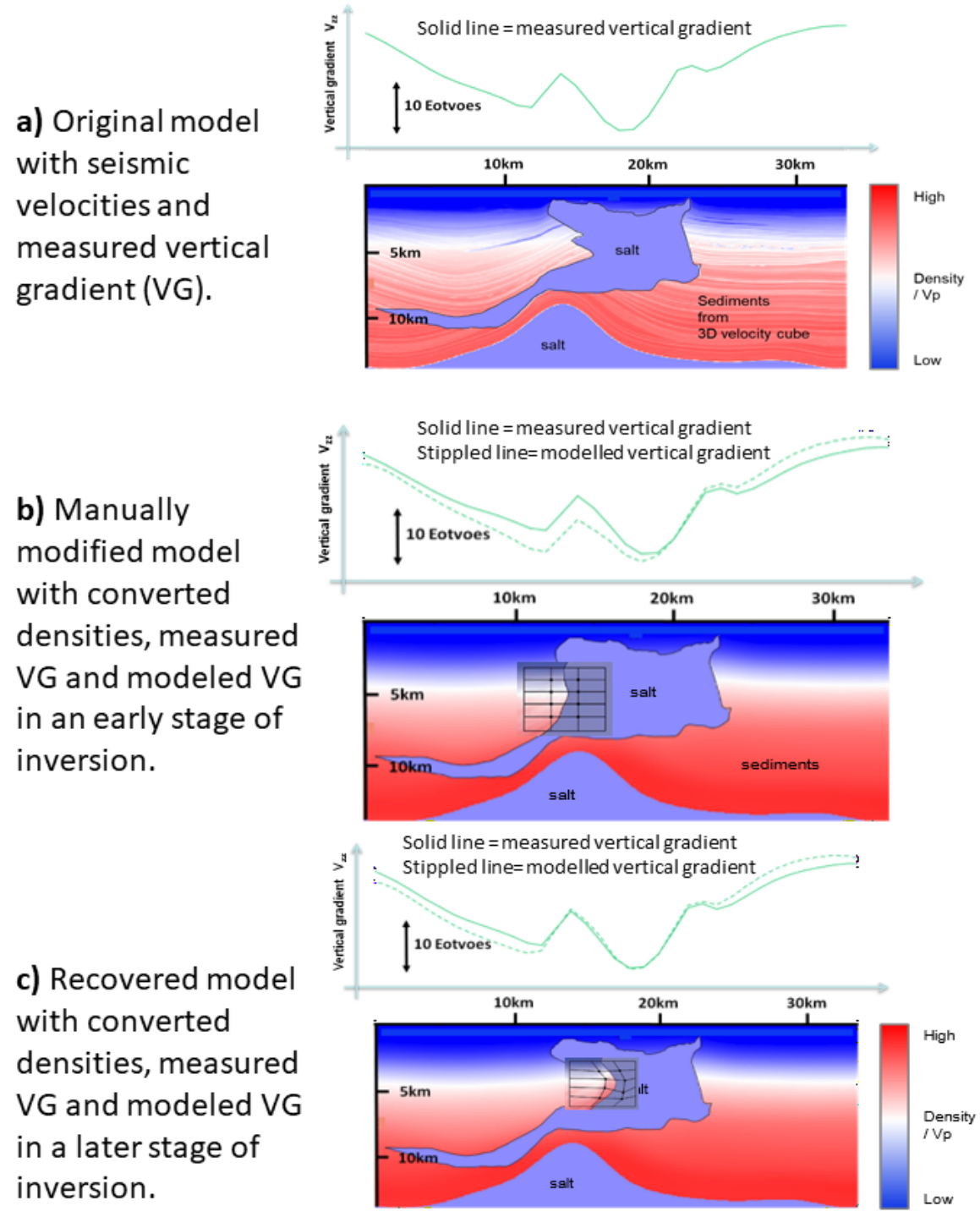


Figure 12: Inversion of the flank of the SEAM salt structure visualized for the vertical section in Figure 11. Vertical gradients of gravity are used for inversion. (a) shows the original shape of the left salt pillow flank and the correct sediment model. In (b) the flank was changed by us and the grid was fixed for optimization. (c) then shows the result, which resembles the original model.

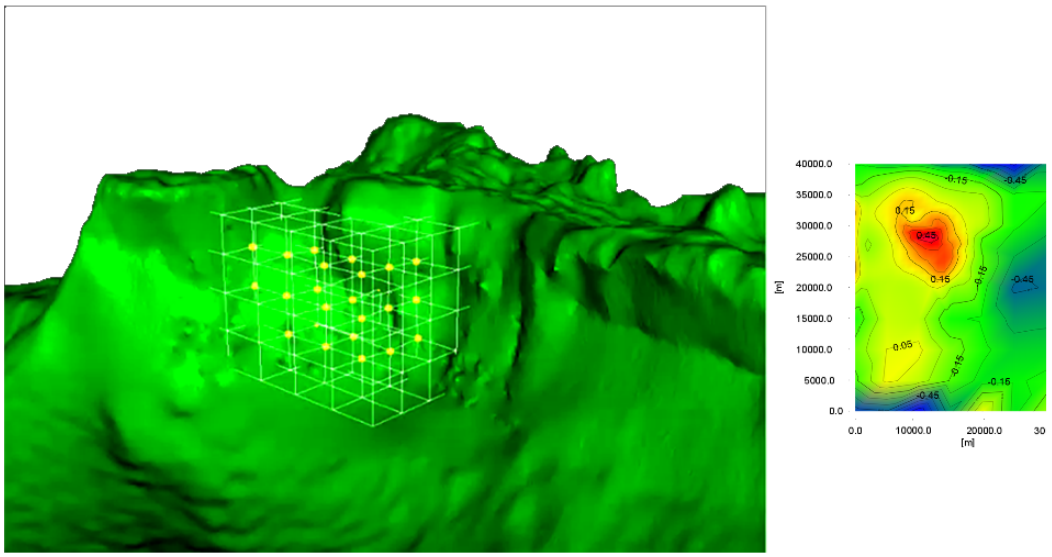


Figure 13: A trial – the flank of the SEAM salt structure with the grid for the optimization of the salt flank. The yellow dots mark space position in the lattice to optimize. To the right, the quality of the fit to the measured field is shown. Reddish colors mark positive residuals and blueish negative. Compare also to Figure 14 which shows a higher resolution of measured and modeled fields.

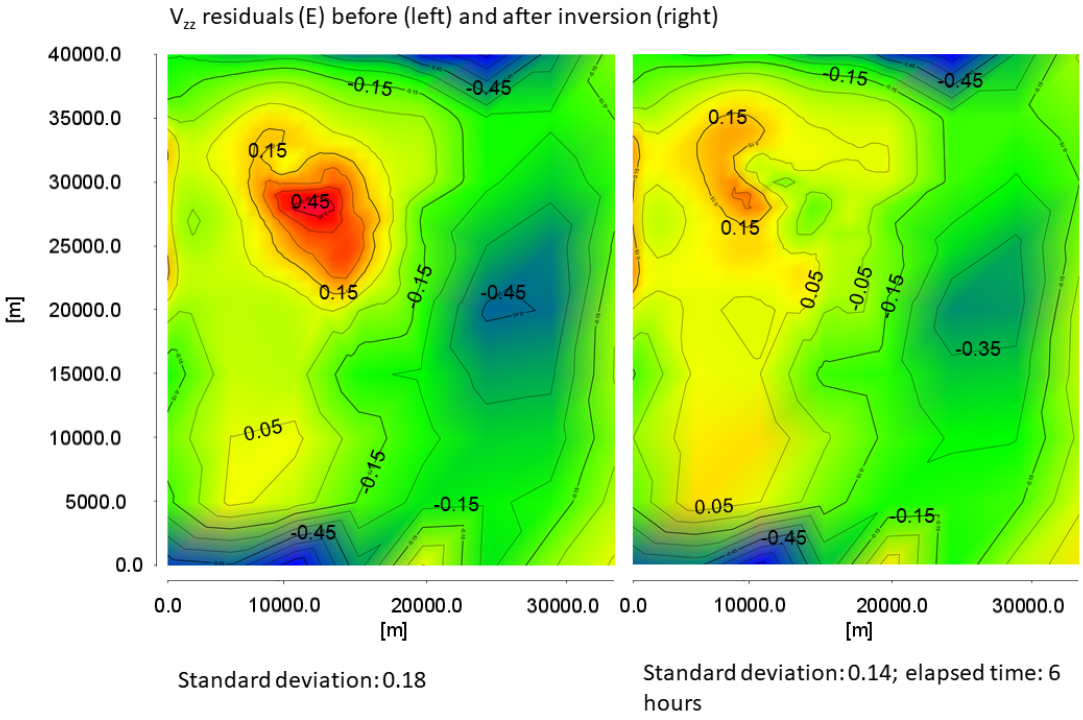


Figure 14: The residuals of modeled and measured gravity gradient before optimization process started (left) and after a time span of six hours (right). Units are given in Eötvös. For more information refer to the text.

Salt domes in the North Sea

Germany's largest oil deposit is located in Schleswig-Holstein state in the north. Of great interest is the exploration of the wider subsurface around the deposits. The challenge in the investigation with geophysical methods is to make geological structures visible in a 3-dimensional computer model in this complex subsurface. The project included the three-dimensional interdisciplinary interpretation of the subsurface in the vicinity of the Büsum and Oldenswort salt domes mentioned above in an area of about 50 km × 50 km. The goal was to explore a high-resolution structural density model between the surface and the Zechstein base at a depth of about 5 km using different geophysical methods.

For demonstration purposes we show the 3D inversion of the salt dome Büsum at the North Sea coast of Schleswig-Holstein (Figure 15). The modeling followed the representations in the Geotectonic Atlas of Northern Germany – GANG (Baldschuhn, 2001).

As part of a study for an industry partner to test to what extent the subsurface structures shown in the Geotectonic Atlas are realistic, since these are mostly based on old refraction and reflection lines, some of which were laid out long before the end of the last century; refer to (Baldschuhn, 2001) for more information. Second, the study should show how well evolutionary strategy and space warping perform together in inversion (Figure 16). In Figure 16a we show the starting phase of the inversion process, whereas Figure 16b displays the final result. For visualization of the full process please refer to the Movie 2 in the additional materials).

During optimization of the structure of the central dome (the Büsum salt dome in Figure 15), we defined a box around it (rectangular block in Figure 16a) and interactively inverted the structure inside the box by application of the CMA-ES and the spring algorithm.

The density of salt and other densities remain untouched. In this example we do not visualize the grid positions of the spatial points that are changed.

Here, a (12,24) CMA-ES was applied, and after 37 min of execution time the result in Figure 16b and in Movie 2 (refer to the additional materials) was obtained. The standard deviation of the model fit is 0.09 mGal and the obtained accuracy is 0.51 mGal after 119 "generations". This is rather close to the accuracy of the measured field, which we can not be shown here due to confidentiality; the stop quality(the difference between the results of two generations) for the standard deviation was set to 10^{-4} mGal. A total of 119 generations with 24 offsprings per generation had to be calculated, i.e. a total of 2856 models on a 3-Quade 6600 machine from the year 2014. Within the available seismic constraints and information of older exploration ventures in this area, the salt dome Büsum has been well mapped. Within the assumed accuracy of the underlying Bouguer gravity field of 0.5 mGal, a satisfactory result was achieved in the automatic interpretation. With this study, we have shown that the new method is able to reconstruct the subsurface structures with satisfac-

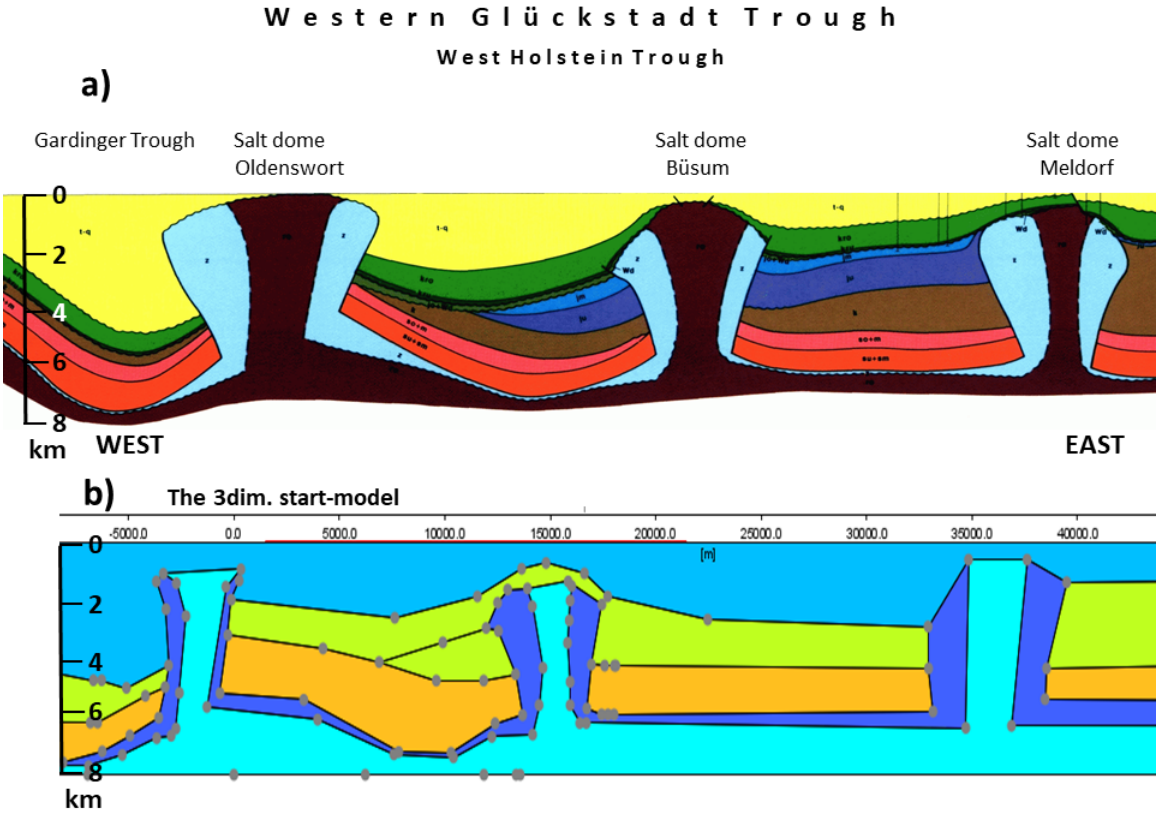
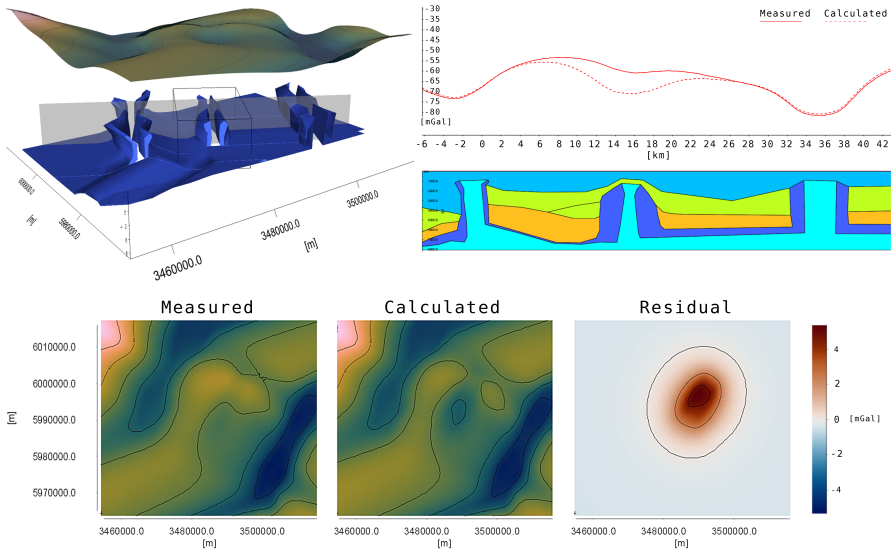
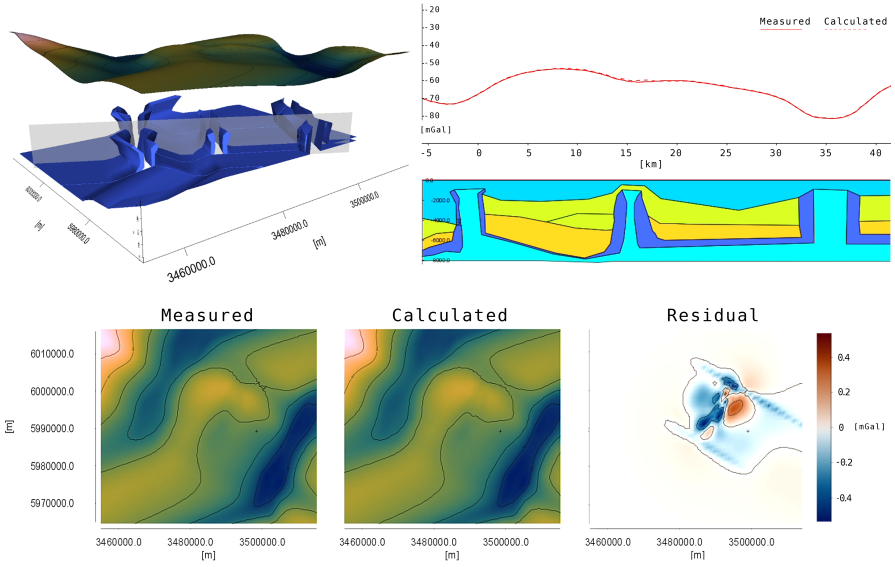


Figure 15: (a) The three salt domes offshore the west coast of Schleswig-Holstein (Northern Germany) from west to east Oldenswort - Büsum - Meldorf (profile 14, after the GANG, Baldschuhn (2001)). The salt cores (Rotliegendes salt) are illustrated by dark brownish colors and the flanks are drawn in light blue (Zechstein salt), Tertiary sediments are shown in yellow, Cretaceous layers in green, Jurassic layers in blue, and Zechstein formations in red-brown. (b) shows the salt domes in the same order as in (a). Small gray dots mark model vertices. In this start model the Rotliegend salt is represented by light blue color, the Zechstein salt by dark blue. For simplicity, the Mesozoic strata are grouped into three stratigraphic units: Permian/Triassic (orange), Jurassic/Cretaceous (light green) and Tertiary (medium blue). The model segment to be inverted is the middle salt dome and its immediate surroundings (Figure 16 and Movie 2 in the additional materials).



(a)



(b)

Figure 16: The starting phase (a) and the final result (b) of 3D modeling of the three salt domes Oldeswort (left) - Büsum (center) and Meldorf. In each subfigure, the upper panel shows the 3D model and the gravity field in a perspective view (left) and a vertical cross section along the gray plane (right). The solid red curve marks the measured gravity field, the stippled curve the optimized. The lower panels in both subfigures displays the optimization status with three maps: the measured gravity (left), the modeled gravity (center) and the residual field (right).

tory accuracy. That is, the deviations from the comparison field correspond to the modeling error.

CONCLUSIONS

For the optimization of the geometry of complicated geological models we have searched for new possibilities of potential field data inversion. These new possibilities consist of a combination of the evolution strategy and a new way of warping the model space, which then adequately changes the 3D model as well. The evolution strategy with covariance matrix adaptation (CMA-ES) requires the fewest function calls on average and achieves very good qualities. The CMA-ES is able to overcome typical problems of evolution strategy methods and is stable against numerical variations. However the mentioned evolution strategy is sensitive to the quality of selection and fails when optimization leads to improper changes of the triangulated model geometry. This aspect has been solved by applying warping of the model space and using the additional "springs" approach suggesting that model vertices are connected with elastic springs. We tested the new approach on the SEAM model and on an example from salt domes off the coast of Northern Germany. In the case of our numerical experiments with the SEAM synthetic model, we were able to show that the inversion of the salt-cushion flank with the method presented here was able to restore the original geometry well. Also in the second example, in which a regional structure offshore northern Germany was inverted, we could verify that the combination of the CMA-ES and spatial warping allows an optimal fit of the modeled gravity to the measured field even for large structures.

Finally, we can summarize that the new method will prevail in any case where geological models of the subsurface are approximated with triangular facets – the aforementioned "SKUA-GOCAD" models. Our method focuses on the problem of optimizing huge models with millions of triangles and voxels with the CMA-ES and we have shown that the CMA-ES results in satisfying and realistic underground models. Nevertheless, optimizing the geometry directly is prohibitive, since we run into many problems: a) geological plausible "smoothness" will be destroyed since smoothness can not easily be formulated as a constraint; and b) checking for topology violations is first prohibitively time consuming (in 3D) and second - more importantly - it leads to dying out of populations were at the end only a few "crippled populations" may survive. The aim was to find a method to overcome these limitations in general and specifically when geological models are built by triangular facets. The intrinsic guaranteed conservation of topology and – depending on the coarseness of the grid – a guaranteed smoothness of the final model.

ACKNOWLEDGMENTS

We acknowledge and thank all the colleagues who contributed to the development of the presented inversion tool. We are grateful to Dr. Günter Schmidt who inspired and helped us during the preparation of this manuscript. We also thank Dr. E. Biegert, two anonymous reviewers and editors of GEOPHYSICS for their valuable and constructive comments and discussions which were of great help in improving the manuscript.

REFERENCES

Alvers, M., H. Götze, S. Schmidt, L. Barrio-Alvers, C. Plonka, C. Bodor, and B. Lahmeyer, 2015a, Practical aspects of 3D interactive potential field modelling and inversion: Presented at the Proceedings, EAGE Publications BV.

Alvers, M., C. Plonka, H.-J. Götze, S. Schmidt, and L. Barrio-Alvers, 2013, Semi-automated geometry optimization of voxelized triangle models by means of covariance-matrix-adoption, evolution-strategies, and user interaction: Presented at the DGG 2013.

Alvers, M. R., 1998, Zur anwendung von optimierungsstrategien auf potentialfeldmodelle: Thesis, Freie Universität Berlin.

Alvers, M. R., L. Barrio-Alvers, C. Bodor, H.-J. Götze, B. Lahmeyer, C. Plonka, and S. Schmidt, 2015b, Quo vadis inversión?: First break, **33**; doi: 0.3997/1365-2397.33.4.79746.

Alvers, M. R., H.-J. Götze, L. Barrio-Alvers, S. Schmidt, B. Lahmeyer, and C. Plonka, 2014, A novel warped-space concept for interactive 3D-geometry-inversion to improve seismic imaging: First Break, **32**; doi: 0.3997/1365-2397.32.4.74375.

Anikiev, D., H.-J. Götze, J. Bott, A. M. Gómez-García, M. L. Gomez Dacal, C. Meeßen, C. Spooner, C. Rodriguez Piceda, C. Plonka, S. Schmidt, et al., 2021, Interdisciplinary data-constrained 3-D potential field modelling with IGMAS+: EGU General Assembly Conference Abstracts, Copernicus GmbH, EGU21–2964.

Anikiev, D., H.-J. Götze, C. Meeßen, C. Plonka, M. Scheck-Wenderoth, and S. Schmidt, 2020, IGMAS+: Interactive Gravity and Magnetic Application System. v. 1.3.

Baldschuhn, R., 2001, Geotektonischer atlas von nordwest-deutschland und dem deutschen nordsee-sektor strukturen, strukturentwicklung, paläogeographie: Schweizerbart. (Bilingual digital atlas of the geotectonic structure of Northwestern Germany and the German North-Sea sector).

Boschetti, F., and L. Moresi, 2001, Interactive inversion in geosciences: GEOPHYSICS, **66**, 1226–1234; doi: 10.1190/1.1487069.

Breunig, M., A. Cremers, H. Götze, S. Schmidt, R. Seidemann, S. Shumilov, and A. Siehl, 2000, Geological mapping based on 3d models using an interoperable gis: GIS-HEIDELBERG, **13**, 12–18.

Coquillart, S., 1990, Extended free-form deformation: a sculpturing tool for 3d geometric modeling: Computer Graphics, **24**, 187–196; doi: 10.1145/97880.97900.

De Jong, K. A., 2006, Evolutionary computation : a unified approach: MIT Press.

Dirks, V., 1995, Travel-time curve inversion for transversely isotropic media with the help of an evolution strategy: 57th EAGE Conference and Exhibition, European Association of Geoscientists & Engineers, cp–90.

Dueck, G., 1993, New optimization heuristics: Journal of Computational Physics, **104**, 86–92; doi: 10.1006/jcph.1993.1010.

Dueck, G., and T. Scheuer, 1990, Threshold accepting: A general purpose optimization algorithm appearing superior to simulated annealing: Journal of Computational Physics, **90**, 161–175; doi: 10.1016/0021-9991(90)90201-b.

Gardner, G. H. F., L. W. Gardner, and A. R. Gregory, 1974, Formation velocity and

density—the diagnostic basics for stratigraphic traps: GEOPHYSICS, **39**, 770–780; doi: 10.1190/1.1440465.

Gerdes, I., F. Klawonn, and R. Kruse, 2004, Evolutionäre algorithmen: Vieweg+Teubner Verlag.

Goldberg, D., 1989, Genetic algorithms in search, optimization, and machine learning: Addison-Wesley Publishing Company.

Götze, H.-J., 1984, Über den Einsatz interaktiver Computergraphik im Rahmen 3-dimensionaler Interpretationstechniken in Gravimetrie und Magnetik: Habilitationsschrift, Techn. Univ. Clausthal.

Götze, H.-J., J. Bott, M. L. Gómez Dacal, A. M. Gomez Garcia, C. Rodriguez Piceda, C. Meeßen, C. Plonka, C. Spooner, M. Scheck-Wenderoth, S. Schmidt, and D. Anikiev, 2021, Interdisciplinary 3D potential field modelling of complex lithospheric structures by IGMAS+: Presented at the DGG 2021, FID GEO.

Götze, H.-J., and B. Lahmeyer, 1988, Application of three-dimensional interactive modeling in gravity and magnetics: GEOPHYSICS, **53**, 1096–1108; doi: 10.1190/1.1442546.

Hansen, N., 2006, The CMA evolution strategy: A comparing review, *in* Towards a New Evolutionary Computation: Springer Berlin Heidelberg, 75–102.

———, 2016, The CMA evolution strategy: A tutorial: arXiv preprint arXiv:1604.00772.

Hansen, N., and A. Ostermeier, 1997, Convergence properties of evolution strategies with the derandomized covariance matrix adaptation: the ($\mu/\mu_$l, λ)-cma-es: Proceedings of the Fifth European Congress on Intelligent Techniques and Soft Computing (EUFIT'97), Aachen, Germany, 650–654.

Hese, F., 2012, 3d modellierungen und visualisierung von untergrundstrukturen für die nutzung des unterirdischen raumes in schleswig-holstein: Thesis, Christian-Albrechts-Universität Kiel.

Holland, J., 1992, Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence: MIT Press.

IGMAS+ Team, 2022, IGMAS+ website, `https://www.gfz-potsdam.de/igmas`, accessed April 12, 2022.

Kirkpatrick, S., C. D. Gelatt, and M. P. Vecchi, 1983, Optimization by simulated annealing: Science, **220**, 671–680; doi: 10.1126/science.220.4598.671.

Li, X., and M. Chouteau, 1998, Three-dimensional gravity modeling in all spaces: Surveys in Geophysics, 229–368; doi: 10.1023/A:1006554408567.

Lillesand, T., R. W. Kiefer, and J. Chipman, 2015, Remote sensing and image interpretation: John Wiley & Sons.

Metropolis, N., A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller, 1953, Equation of state calculations by fast computing machines: The Journal of Chemical Physics, **21**, 1087–1092; doi: 10.1063/1.1699114.

Moorkamp, M., B. Heincke, M. Jegen, R. W. Hobbs, and A. W. Roberts, 2016, Joint inversion in hydrocarbon exploration, *in* Integrated Imaging of the Earth: John Wiley & Sons, Inc, 167–189.

Nelder, J. A., and R. Mead, 1965, A simplex method for function minimization: The Computer Journal, **7**, 308–313; doi: 10.1093/comjnl/7.4.308.

Pace, F., A. Santilano, and A. Godio, 2021, A review of geophysical modeling based on particle swarm optimization: Surveys in Geophysics, **42**, 505–549; doi: 10.1007/s10712-021-09638-4.

Pangman, P., 2007, SEAM launched in march: The Leading Edge, **26**, 718–720; doi: 10.1190/tle26060718.1.

Rao, S., 2009, Engineering optimization: theory and practice: John Wiley & Sons.

Rechenberg, I., 1973, Evolutionsstrategie, optimierung technischer systeme nach prinzipien der biologischen evolution: Frommann-Holzboog Verlag.

———, 1994, Evolutionsstrategie'94: Frommann-Holzboog Verlag.

Saltus, R. W., and R. J. Blakely, 2011, Unique geologic insights from "non-unique" gravity and magnetic interpretation: GSA Today, **21**, 4–10; doi: 10.1130/g136a.1.

Sambridge, M., T. Bodin, K. Gallagher, and H. Tkalčić, 2013, Transdimensional inference in the geosciences: Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences, **371**, 20110547; doi: 10.1098/rsta.2011.0547.

Schmidt, S., D. Anikiev, H.-J. Götze, A. Gomez Garcia, M. L. Gomez Dacal, C. Meeßen, C. Plonka, C. Rodriguez Piceda, C. Spooner, and M. Scheck-Wenderoth, 2020, IGMAS+ – a tool for interdisciplinary 3D potential field modelling of complex geological structures: EGU General Assembly 2020, EGU2020–8383.

Schmidt, S., H.-J. Götze, C. Fichler, and M. R. Alvers, 2010, IGMAS+ a new 3D Gravity, FTG and Magnetic Modeling Software: Geoinformatik, 57–63.

Schmidt, S., C. Plonka, H.-J. Götze, and B. Lahmeyer, 2011, Hybrid modelling of gravity, gravity gradients and magnetic fields: Geophysical Prospecting, **59**, 1046–1051; doi: 10.1111/j.1365-2478.2011.00999.x.

Schöneburg, E., 1994, Genetische Algorithmen und Evolutionsstrategien: eine Einführung in Theorie und Praxis der simulierten Evolution: Addison-Wesley Pub. Co.

Schwefel, H.-P., 1978, Optimierung von simulationsmodellen mit der evolutionsstrategie, *in* Medizinische Informatik und Statistik: Springer Berlin Heidelberg, 115–129.

———, 1995, Evolution and optimum seeking: Wiley.

Sederberg, T. W., and S. R. Parry, 1986, Free-form deformation of solid geometric models: Computer Graphics, **20**, 151–160; doi: 10.1145/15886.15903.

Sedlag, U., and E. Weinert, 1987, Biogeographie, Artbildung, Evolution: Gustav Fischer Verlag.

Shaw, R., and S. Srivastava, 2007, Particle swarm optimization: A new tool to invert geophysical data: GEOPHYSICS, **72**, F75–F83; doi: 10.1190/1.2432481.

Weicker, K., 2015, Evolutionäre algorithmen: Springer Fachmedien Wiesbaden.

Wellmann, F., and G. Caumon, 2018, 3-D structural geological models: Concepts, methods, and uncertainties, *in* Advances in Geophysics: Elsevier, 1–121.

Weng, L., 2019, Evolution Strategies, `https://lilianweng.github.io/posts/2019-09-05-evolutio` accessed February 7, 2022. (https://lilianweng.github.io).

Wikipedia, 2022, CMA-ES, `https://en.wikipedia.org/wiki/CMA-ES`, accessed February 16, 2022.