

Article

Gravity Data Inversion with Method of Local Corrections for Finite Elements Models

Petr S. Martyshko *, Igor V. Ladovskii, Denis D. Byzov and Alexander G. Tsidaev 

Institute of Geophysics, Ural Branch of Russian Academy of Science, Amundsena 100-305, 620016 Ekaterinburg, Russia; ila848@mail.ru (I.V.L.); ivanov389@gmail.com (D.D.B.); atsidaev@gmail.com (A.G.T.)

* Correspondence: pmart3@mail.ru

Received: 3 August 2018; Accepted: 3 October 2018; Published: 10 October 2018



Abstract: We present a new method for gravity data inversion for the linear problem (reconstruction of density distribution by given gravity field). This is an iteration algorithm based on the ideas of local minimization (also known as local corrections method). Unlike the gradient methods, it does not require a nonlinear minimization, is easier to implement and has better stability. The algorithm is based on the finite element method. The finite element approach in our study means that the medium (part of a lithosphere) is represented as a set of equal rectangular prisms, each with constant density. We also suggest a time-efficient optimization, which speeds up the inversion process. This optimization is applied on the gravity field calculation stage, which is a part of every inversion iteration. Its idea is to replace multiple calculations of the gravity field for all finite elements in all observation points with a pre-calculated set of uniform fields for all distances between finite element and observation point, which is possible for the current data set. Method is demonstrated on synthetic data and real-world cases. The case study area is located on the Timan-Pechora plate. This region is one of the promising oil- and gas-producing areas in Russia. Note that in this case we create a 3D density model using joint interpretation of seismic and gravity data.

Keywords: inverse problem; 3D density model; joint interpretation

1. Introduction

Our main goal is to construct a stable method of big gravity data inversion which has to be three-dimensional. The usual approach to find a volumetric density distribution is forward gravity modeling. One changes an initial model in interactive way to diminish gravity residuals. However, such a forward modeling approach cannot solve the non-uniqueness and the instability of gravity data inversion.

Our method of inversion follows the procedure of Cordell and Henderson [1]. A solution is calculated from gravity data automatically by successive approximations, without a time-consuming trial-and-error process. We apply a different approach to form a successive approximation. Moreover, we take into account instability of the inverse problem by means of a sort of regularization. The method is based on a fast forward modeling algorithm (to calculate the gravity effect of a volumetric density distribution); the initial 3D density model is deduced from independent geophysical studies to converge towards a solution; the representation of the sought density in the form of multiplicative function; sustainable adaptive algorithm for solving linear inverse problem.

The process for constructing density models based on gravity field anomalies contains two major blocks: forward modeling by forward gravity problem solving, and gravity data inversion (inverse gravity problem solving). High-efficiency algorithms for solving the forward gravity problem are required, not only for forward modeling, but also for successful iterative inverse problem

solving schemes. Gravity inverse problems are a classic example of ill-posed problems: generally, their solutions are neither unique nor sustainable [2]. Stable and informative inverse gravity problem solutions can be obtained by iteratively calculating small deviations to the initial density distribution.

Initial density distribution is a model that we build before the gravity modeling stage. Such models can be constructed using seismic data, such as deep seismic sounding data (DSS). Two-dimensional seismic profiles form the 3D carcass of the model. Space between profiles is filled with the interpolated data using technique, as described by Martyshko et al. [3].

Discretization for the interpolation is done by gridding to regular 3D grid. Each cell is a rectangular prism with constant density. We adopt a finite elements method (FEM) to work with such volume. This means that the gravity field of the whole model is calculated as a sum of gravity fields of all elements. Closed form solution for rectangular prism gravity field is well-known [4,5] Currently, FEM is one of the most used methods for forward modeling. For recent studies one can refer to, e.g., Mostafa [6], who studied a particular case of FEM with cubic elements, and Couder-Castañeda et al. [7], who performed an optimization of method by parallelization on a cluster. However, they use a small grid size: models containing around 10,000–15,000 prisms in each study. For detailed regional models, we need methods that can handle millions of (and sometimes even more than a 1 billion) elements in a reasonable time. In this case, one cannot rely on parallelization only. Modification of algorithm itself is required to lower its complexity. Also, there are recent studies that are targeted at elements of non-equal size: e.g., Dubey and Tiwari [8] approximate surface with a set of vertical laminas; also they provide closed form for full gravity and gravity gradient tensor components for bodies of different shapes. Li and Oldenburg state [9] that FEM can also be a stage in iterative potential field inversion. We adopt the same approach.

The correct 3D gravity problem solution depends on the mass continuation outside the model volume technique. We cannot simply consider that density is zero everywhere outside the study volume and, thus, ignore it. Since we need to solve the inverse problem, a calculated field should have a similar structure as the observed one. The observed field is an anomaly field, which is produced by density increase and decrease relative to a background density value. To make a calculated field structure similar to this, we need to preselect some background density distribution. Without a priori information, the mean density value may be a good approximation for the background density. In this paper, we use background density, which only depends on the depth. We obtain this value by averaging the density values inside every horizontal layer of interpolated model. The approach is similar to the one of Cai and Wang [10]. Each horizontal layer is one FEM cell thick.

2. Methods

2.1. Forward Gravity Calculation for a Model with FEM

Using a right Cartesian coordinate system, let D be the rectangular parallelepiped area filled with masses of density $\rho(x, y, z)$:

$$D = [x_{min}; x_{max}] \times [y_{min}; y_{max}] \times [z_{min}; z_{max}]. \quad (1)$$

Let $\vec{r} = (x, y, z) \in D$ denote a position vector of a point in D . Let $Q = (\xi, \eta, \zeta) \notin D$ be the point of gravity field calculation with position vector $\vec{q} = (\xi, \eta, \zeta)$. In this notation the vertical component of gravity field g calculated at Q is

$$g(Q) = \gamma \frac{\partial}{\partial \zeta} \int_D \frac{\rho(\vec{r})}{|\vec{r} - \vec{q}|} dx dy dz, \quad (2)$$

where γ is the gravitational constant.

A grid-approximation of the parallelepiped density function, $\rho(x, y, z)$, has the 3D grid $\{x_i, y_j, z_k\}$ ($x_i < x_{i+1}, y_j < y_{j+1}, z_k < z_{k+1}, x_0 = x_{min}, x_{N_x} = x_{max}, y_0 = y_{min}, y_{N_y} = y_{max}, z_0 = z_{min}, z_{N_z} = z_{max}$),

which constructs elements $D_{i,j}^k$. Element of this three-dimensional grid and its spatial position is presented on Figure 1.

$$D = \cup_{i,j,k} D_{i,j}^k; i \in \overline{0, N_x - 1}; j \in \overline{0, N_y - 1}; k \in \overline{0, N_z - 1}$$

$$D_{i,j}^k = [x_i; x_{i+1}) \times [y_j; y_{j+1}) \times [z_k; z_{k+1}), \tag{3}$$

so the density is a constant for every element:

$$\rho(x, y, z) = \rho_{i,j}^k, (x, y, z) \in D_{i,j}^k. \tag{4}$$

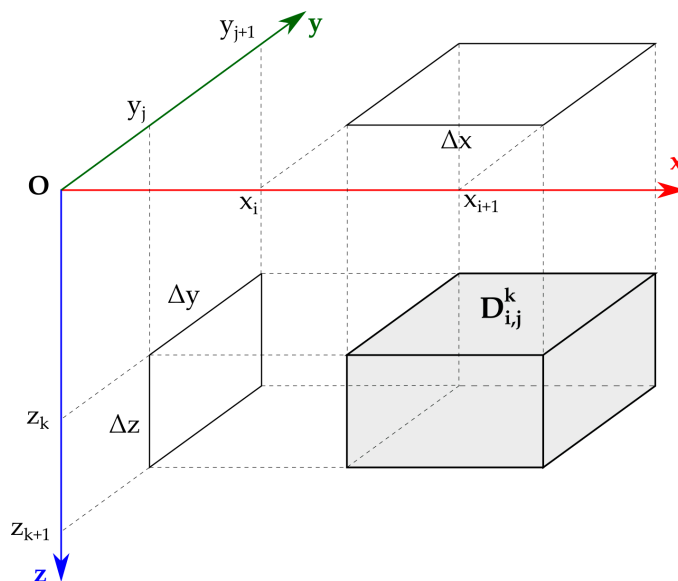


Figure 1. Element of grid approximation $D_{i,j}^k$ and its projections.

All $D_{i,j}^k$ parallelepipeds have the same dimensions, so $x_i = x_0 + i\Delta x, y_j = y_0 + j\Delta y, z_k = z_0 + k\Delta z, \Delta x > 0$, and $\Delta y > 0, \Delta z > 0$.

Gravity field of a prism with unit density and up to gravitational constant can be calculated with closed form solution formula [4] as

$$G_{i,j}^k(Q) = -v(\vec{r} - \vec{q}) \begin{vmatrix} x_{i+1} & y_{j+1} & z_{k+1} \\ x_i & y_j & z_k \end{vmatrix}. \tag{5}$$

here $G_{i,j}^k(Q)$ is the gravity effect of the prism $D_{i,j}^k$, calculated at a point Q and $v(\vec{a})$ is an expression $v(\vec{a}) = a_x \ln(a_y + |\vec{a}|) + a_y \ln(a_x + |\vec{a}|) - a_z \arctg(\frac{a_x a_y}{a_z |\vec{a}|})$. In our case, $\vec{a} = \vec{r} - \vec{q}$, so $a_x = x - \xi;$

$$a_y = y - \eta; a_z = z - \zeta, |\vec{a}| = \sqrt{(x - \xi)^2 + (y - \eta)^2 + (z - \zeta)^2}.$$

And the full gravity field of area D is a sum of fields of prisms

$$g(Q) = \gamma \sum_{i=0}^{N_x-1} \sum_{j=0}^{N_y-1} \sum_{k=0}^{N_z-1} \rho_{i,j}^k G_{i,j}^k(Q). \tag{6}$$

here, $\rho_{i,j}^k$ is the density value for the element $D_{i,j}^k$.

Representing the solution in the form of (5) and (6) allows the field calculation algorithm for the layer to be optimized between arbitrary depths.

In practice, it is convenient to calculate gravity field on a flat surface. In this case, all Q points are located on a single plane. Rectangular grid can be given on this surface. Let this grid be oriented similarly to D and parallel to the D upper face plane. We also assume that the grid step between nodes is Δx and Δy . Thus, a set, τ , of all field calculation points can be specified as follows: $\tau = (\vec{q}_m)_{m=0}^{M-1}$, $\vec{q}_m = (\xi_{i1}, \eta_{j1}, \zeta)$, $\xi_{i1} = \xi_0 + i_1 \Delta x$, $\eta_{j1} = \eta_0 + j_1 \Delta y$, $\zeta \notin (z_{min}; z_{max})$, $i_1 \in \overline{0, M_x - 1}$, $j_1 \in \overline{0, M_y - 1}$, $m = i_1 + M_x j_1$, $M = M_x M_y$.

2.2. Inverse Gravity Problem for Layered Media Model (Density Calculations Using Known Field Values)

The 3D density $\rho(x, y, z)$ calculations in an inhomogeneous area, D , based on field values, $g(\xi, \eta, \zeta)$, specified for the point set τ , were implemented by inversion of integral operator on the right hand side of the (Equation (2)); g is a known function as measurements. Mathematically speaking, such a problem is ill posed and its solution depends heavily on small variations in the initial field data, g . However, if we select the density class with only lateral density variations, the determination of the density distribution in the horizontal layer will be stable [11].

We examined the density for an inhomogeneous parallelepiped with vertical thickness H as a product that only depends on a depth function, $\rho_0(z)$ and $\Phi(x, y)$:

$$\rho(x, y, z) = \rho_0(z)\Phi(x, y) \tag{7}$$

We assume $\rho_0(z)$ is known from the logging data or may be approximated by some kind of the initial model analysis. Building a grid analogue (4) of the multiplicative density, $\rho_{i,j}^k$, on the partitioning, $\{x_i, y_j, z_k\}$, yields $\rho_0(z) = \rho_0^k, z \in [z_k; z_{k+1})$; $\Phi(x, y) = \Phi_{i,j}, (x, y) \in [x_i; x_{i+1}) \times [y_j; y_{j+1})$, $i \in \overline{0, N_x - 1}$, $j \in \overline{0, N_y - 1}$, $k \in \overline{0, N_z - 1}$. The field of layered parallelepiped was calculated on a flat surface $\zeta = 0$. We regroup summands with neighbor indices k and $k + 1$ in sum of (Equation (6)) using the primitive (5):

$$g(\xi_{i1}, \eta_{j1}, \zeta) = g_{i1,j1} = \gamma \sum_{i=0}^{N_x-1} \sum_{j=0}^{N_y-1} \Phi_{i,j} \sum_{k=0}^{N_z-1} \Delta\rho_0^k G_{i1,j1,i,j}^k \tag{8}$$

where $\Delta\rho_0^k$ is the difference between the background densities for the k and $k - 1$ horizontal layers: $\Delta\rho_0^k = \rho_0^k - \rho_0^{k-1}$, $k \in \overline{1, N_z - 1}$ and $\Delta\rho_0^0 = \rho_0^0$, $\Delta\rho_0^{N_z} = -\rho_0^{N_z-1}$, $G_{i1,j1,i,j}^k = G_{i,j}^k(\xi_{i1}, \eta_{j1}, \zeta)$.

To lower the number of the indices in the direct problem operators of (Equation (8)) we introduced the continuous indexing of the density parallelepiped vertical columns: $n = i + jN_x$, $n \in \overline{0, (N_x N_y - 1)}$ and field calculation points: $m = i_1 + M_x j_1$, $m \in \overline{0, (M_x M_y - 1)}$. If $N_x = M_x$, $N_y = M_y$, only one field calculation point (m -th) is located above every density column with index n . We suppose these conditions are further satisfied. For the function $\Phi_{i,j} = \Phi_n$ of two variables we have a linear system of equations:

$$\gamma \sum_{n=0}^{M-1} \Phi_n \sum_{k=0}^{N_z-1} G_{m,n}^k \Delta\rho_0^k = g_m \tag{9}$$

where $G_{m,n}^k$ is the 3D integration tensor; $\Delta\rho_0^k$ is the density increment in depth; Φ_n is the unknown lateral density change; $g_m = g(\xi_{i1}, \eta_{j1}, \zeta)$ is observed at a height ζ gravity field value at m -th point.

The value $\gamma \sum_{k=0}^{N_z} G_{m,n}^k \Delta\rho_0^k$ in (Equation (9)) may be regarded as coefficient for the Φ_n . From a physical point of view, it represents the gravity field of a vertical column, D_n , which is constructed from D_n^k elements with densities ρ_0^k :

$$G_{m,n} = \gamma \sum_{k=0}^{N_z} G_{m,n}^k \Delta\rho_0^k \tag{10}$$

The problem of determination of density distribution reduces to a linear equation system [12] $G_{m,n}\Phi_n = g_m$. The $G_{m,n}$ matrix forms from the convolution (10) of the integration tensor with an increment vector and was only calculated once. Any vector of unknowns, Φ_n , reduces the 3D parallelepiped field calculations to a trivial matrix vector production operation:

$$g_m = \sum_{n=0}^{M-1} G_{m,n}\Phi_n. \tag{11}$$

2.3. Inverse Problem Solution Iteration Algorithm

We propose a sustainable adaptive method for solving linear equation systems (11) based on the local corrections method [13–16]. However, the original method was created for the reconstruction of boundary surface position. We use it for the density refinement. The method uses the local one-dimensional density distribution model. In our case this means that gravity field income for m -th point g_m depends only on density distribution in m -th column D_m . Income from all other columns is ignored for this point.

Consider Equations (8) and (11) for forward gravity calculation as formulas for model field calculation at the points of set $\tau = (\vec{q}_m)_{m=0}^{M-1}$ on the level of Earth surface: $U(\xi_{i1}, \eta_{j1}, 0) = U_m$. The difference between observed gravity field and the calculated gravity field is the error for inverse problem calculation:

$$\delta g_m = g_m - U_m = g_m - \sum_{n=0}^{M-1} G_{m,n}\Phi_n, \tag{12}$$

where Φ_n is the constant density of the n -th model column. Then we build an iteration algorithm δg_m calculation. Let $\delta g_m^{(\theta)}$ be the difference between fields (12) after θ iterations:

$$\delta g_m^{(\theta)} = \delta g_m^{(\theta-1)} - \delta U_m^{(\theta)} = \delta g_m^{(\theta-1)} - \sum_{n=0}^{M-1} G_{m,n}\delta\Phi_n^{(\theta)}. \tag{13}$$

This algorithm is based on consequential independent reduction of the remaining $\delta g^{(\theta)}$ fields at every field calculation point. This reduction was performed by changing $\delta\Phi_n^{(\theta)}$ for the vertical density column D_n . If we assume that the field in point Q_m was produced by the nearest D_m column mass only and is independent of other columns, then we have only one summand ($n = m$) in a sum in (13):

$$\delta g_m^{(\theta)} = \delta g_m^{(\theta-1)} - G_{m,m}\delta\Phi_m^{(\theta)}. \tag{14}$$

Therefore, the value $\delta\Phi_m^{(\theta)} = \delta g_m^{(\theta-1)} / G_{m,m}$ can be selected as the Φ_m correction for the θ -th iteration to a first approximation.

By $\delta\Phi_m^{(\theta)}$ variation the field variation at every point is calculated

$$\delta U_m^{(\theta)} = U_m^{(\theta)} - U_m^{(\theta-1)} = \sum_{n=0}^{M-1} \frac{G_{m,n}}{G_{n,n}} \delta g_n^{(\theta-1)}. \tag{15}$$

Adding $\Phi_m^{(\theta-1)}$ to $\delta\Phi_m^{(\theta)}$ at every field calculation point we obtain density distribution on θ -th iteration: $\Phi_m^{(\theta)} = \Phi_m^{(\theta-1)} + \delta\Phi_m^{(\theta)}$; model field is calculated as $U_m^{(\theta)} = U_m^{(\theta-1)} + \delta U_m^{(\theta)}$. Difference (12) between observed and model field on θ -th iteration $\delta g_m^\theta = g_m - U_m^\theta$ is the error of the field picking. The stopping condition is the desired accuracy ε achievement:

$$\|\delta g^{(\theta)}\| = \sqrt{\sum_{m=0}^{M-1} (\delta g_m^{(\theta)})^2} < \varepsilon. \tag{16}$$

However, iteration schemes constructed this way generally diverge ($\|\delta g^{(\theta)}\|$ increases) because the ceteris paribus for the contribution of D_m to the model field at Q_m can be less than the final accumulation of gravity fields of all of the other columns, $D \setminus D_m$. Reducing the square $\Delta x \Delta y$ of the partitioning cell in the $0xy$ plane increases the divergence because $\lim_{\Delta x \Delta y \rightarrow 0} \|U(D_m)\| = 0$, and $\|U(D)\|$ does not depend on $\Delta x \Delta y$. Assuming $\delta \Phi_m^{(\theta)} = \delta g_m^{(\theta-1)} / G_{m,m}$, we assigned all of the field $\delta g_m^{(\theta-1)}$ values to column D_m to overestimate the lateral density correction: $\lim_{\Delta x \Delta y \rightarrow 0} |\delta \Phi_m^{(\theta)}| = \infty$. To prevent this overestimate, we assumed $\delta \Phi_m^{(\theta)} = \alpha^{(\theta)} \frac{\delta g_m^{(\theta-1)}}{G_{m,m}} + \beta^{(\theta)}$ for every iteration. $\alpha^{(\theta)}$ and $\beta^{(\theta)}$ are common to all columns $\{D_m\}_{m=0}^{M-1}$. The field discrepancy at each point is

$$\delta g_m^{(\theta)} = \delta g_m^{(\theta-1)} - \alpha^{(\theta)} \delta U_m^{(\theta)} - \beta^{(\theta)} G_m, \tag{17}$$

where $G_m = \sum_{n=0}^{M-1} G_{m,n}$ is the field for the entire parallelepiped D with $\Phi(x, y) \equiv 1$ calculated at \vec{q}_m . $\alpha^{(\theta)}$ and $\beta^{(\theta)}$ were selected from the minimum condition, $\|\delta g^{(\theta)}\|$:

$$\begin{aligned} \alpha^{(\theta)} &= \frac{1}{Q^{(\theta)}} \left(\sum_{m=0}^{M-1} (G_m)^2 \sum_{m=0}^{M-1} \delta g_m^{(\theta-1)} \delta U_m^{(\theta)} - \sum_{m=0}^{M-1} G_m \delta U_m^{(\theta)} \sum_{m=0}^{M-1} G_m \delta g_m^{(\theta-1)} \right) \\ \beta^{(\theta)} &= \frac{1}{Q^{(\theta)}} \left(\sum_{m=0}^{M-1} (\delta U_m^{(\theta)})^2 \sum_{m=0}^{M-1} G_m \delta g_m^{(\theta-1)} - \sum_{m=0}^{M-1} G_m \delta U_m^{(\theta)} \sum_{m=0}^{M-1} \delta g_m^{(\theta-1)} \delta U_m^{(\theta)} \right) \\ Q^{(\theta)} &= \sum_{m=0}^{M-1} (G_m)^2 \sum_{m=0}^{M-1} (\delta U_m^{(\theta)})^2 - \left(\sum_{m=0}^{M-1} G_m \delta U_m^{(\theta)} \right)^2 \end{aligned} \tag{18}$$

In summary, we rewrote the main iteration algorithm stages for selecting $\{\Phi_m\}_{m=0}^{M-1}$ values to minimize the discrepancy between the observed and model fields $\|\delta g\|$. First, the model field, $U_m^{(0)}$, and “remaining” field, $\delta g_m^{(0)} = g_m - U_m^{(0)}$, are calculated from the initial approximation model. Assuming $\Phi_m^{(0)} = 0$, cyclically repeat the following steps for $\theta \geq 1$ iterations:

1. Calculate $\delta U_m^{(\theta)}$ by Formula (15).
2. Calculate $\alpha^{(\theta)}$ and $\beta^{(\theta)}$ using Formula (18).
3. Calculate $\delta g_m^{(\theta)}$ from (17).
4. Calculate $\delta \Phi_m^{(\theta)} = \delta \Phi_m^{(\theta-1)} + \beta^{(\theta)} + \alpha^{(\theta)} \frac{\delta g_m^{(\theta-1)}}{G_{m,m}}$.
5. Check the stopping conditions: when $\|\delta g^{(\theta)}\| = \sqrt{\sum_{m=0}^{M-1} (\delta g_m^{(\theta)})^2} < \varepsilon$ (i.e., the desired accuracy of ε achieved) or $\delta g^{(\theta)}$ is constant (because the selection has stabilized); if no condition is met, continue with the next iteration, by incrementation $\theta + 1$.

$$\rho_0(z) = \rho_0^k, z \in [z_k; z_{k+1}); \Phi(x, y) = \Phi_{i,j}, (x, y) \in [x_i; x_{i+1}) \times [y_j; y_{j+1}), i \in \overline{0, N_x - 1}, j \in \overline{0, N_y - 1}, k \in \overline{0, N_z - 1}$$

After executing the density distribution cycle, $\rho(x, y, z) = \rho_0^k \Phi_{i,j}^{(\theta)}, z \in [z_k; z_{k+1}); \Phi(x, y) = \Phi_{i,j}, (x, y, z) \in [x_i; x_{i+1}) \times [y_j; y_{j+1}) \times [z_k; z_{k+1}), i \in \overline{0, N_x - 1}, j \in \overline{0, N_y - 1}, k \in \overline{0, N_z - 1}$, approximates (up to a constant) the difference, $\delta g^{(\theta)}$, between the observed and initial modeled field. Therefore, adding this distribution to the initial model yields a density model with a field that is closer to the observation with an error of $\|\delta g^{(\theta)}\|$.

2.4. Speed Optimization of Calculations with Forward Problem Formula

Idea of optimization is to replace absolute coordinate values in formulas with relative shifts. To do this we transform Equation (6) into a vector form. The difference $\vec{r} - \vec{q}$ represents a vector between

observation and calculation points. It does not contain information about absolute coordinate values anymore. Denote $\vec{r}_{i,j}^k = (x_i, y_j, z_k)$ and reveal in (6) $G_{i,j}^k$ using Formula (5):

$$g(Q) = -\gamma \sum_{i=0}^{N_x-1} \sum_{j=0}^{N_y-1} \sum_{k=0}^{N_z-1} \rho_{i,j}^k \left(v(\vec{r}_{i+1,j+1}^{k+1} - \vec{q}) - v(\vec{r}_{i,j+1}^{k+1} - \vec{q}) - v(\vec{r}_{i+1,j}^{k+1} - \vec{q}) + v(\vec{r}_{i,j}^{k+1} - \vec{q}) - v(\vec{r}_{i+1,j+1}^k - \vec{q}) + v(\vec{r}_{i,j+1}^k - \vec{q}) + v(\vec{r}_{i+1,j}^k - \vec{q}) - v(\vec{r}_{i,j}^k - \vec{q}) \right) \tag{19}$$

Combining the summands with indices $i, i + 1; j, j + 1$ and $k, k + 1$ at $\vec{r}_{i,j}^k$ yields (by the analogy with the derivation of Equation (8))

$$g(Q) = -\gamma \sum_{i=0}^{N_x} \sum_{j=0}^{N_y} \sum_{k=0}^{N_z} v(\vec{r}_{i,j}^k - \vec{q}) (\rho_{i-1,j-1}^{k-1} - \rho_{i-1,j}^{k-1} - \rho_{i,j-1}^{k-1} + \rho_{i,j}^{k-1} - \rho_{i-1,j-1}^k + \rho_{i-1,j}^k + \rho_{i,j-1}^k - \rho_{i,j}^k) = \gamma \sum_{i=0}^{N_x} \sum_{j=0}^{N_y} \sum_{k=0}^{N_z} \bar{\rho}_{i,j}^k v(\vec{r}_{i,j}^k - \vec{q}), \tag{20}$$

assuming $\rho_{i,j}^k = 0$ when $i = -1 \vee i = N_x \vee j = -1 \vee j = N_y \vee k = -1 \vee k = N_z$.

Formula (19) requires v to be calculated $8N_x N_y N_z$ times, while Formula (20) only requires $(N_x + 1)(N_y + 1)(N_z + 1)$ calculations, which provides an almost eightfold calculation time reduction for a sufficiently large dimension $N = N_x \times N_y \times N_z$.

For the next optimization, we can write a formula for T set of g values calculated using τ set points located in the nodes of the uniform rectangular 2D grid of $M = M_x \times M_y$ size.

$$T = \left(g(\vec{q}_m) \right)_{m=0}^{M-1} = \gamma \left(\sum_{i=0}^{N_x-1} \sum_{j=0}^{N_y-1} \sum_{k=0}^{N_z-1} \rho_{i,j}^k G_{i,j}^k(\vec{q}_m) \right)_{m=0}^{M-1} \tag{21}$$

Calculating T using Formula (21) requires v be calculated $8MN$ times. Applying optimization (20) yields:

$$T = \gamma \left(\sum_{i=0}^{N_x} \sum_{j=0}^{N_y} \sum_{k=0}^{N_z} \bar{\rho}_{i,j}^k v(\vec{r}_{i,j}^k - \vec{q}_m) \right)_{m=0}^{M-1}. \tag{22}$$

Using this formula, the v values only need to be calculated $M(N_x + 1)(N_y + 1)(N_z + 1)$ times. However, the specified τ in the vector set $\vec{r}_{i,j}^k - \vec{q}_m$ exhibits significant overlap and only a single v calculation is needed: $\vec{r}_{i,j}^k - \vec{q}_m = (x_0 + i\Delta x - (\xi_0 + i_1\Delta x), y_0 + j\Delta y - (\eta_0 + j_1\Delta y), z_k - \zeta) = (x_0 - \xi_0 + (i - i_1)\Delta x, y_0 - \eta_0 + (j - j_1)\Delta y, z_k - \zeta)$. Introducing new indices $i_2 = i - i_1, i_2 \in \overline{1 - M_x, N_x}, j_2 = j - j_1, j_2 \in \overline{1 - M_y, N_y}$ and denoting $v_{i_2,j_2}^k = v(x_0 - \xi_0 + i_2\Delta x, y_0 - \eta_0 + j_2\Delta y, z_k - \zeta)$ allows (22) to be rewritten as

$$T = \gamma \sum_{k=0}^{N_z} \left(\sum_{i_2=-i_1}^{N_x-i_1} \sum_{j_2=-j_1}^{N_y-j_1} \bar{\rho}_{i_1+i_2,j_1+j_2}^k v_{i_2,j_2}^k \right)_{m=0}^{M-1}. \tag{23}$$

So v only needs be calculated for $(N_x + M_x)(N_y + M_y)(N_z + 1)$ points, which is two orders of magnitude less than using (21) and (22). Notably, there is a possible program implementation for (23) that does not require storing the set $\{v_{i_2,j_2}^k | i_2 \in \overline{1 - M_x, N_x}, j_2 \in \overline{1 - M_y, N_y}, k \in \overline{0, N_z}\}$ in memory. On-the-fly calculations of these elements reduce the memory usage without affecting the performance.

The offered direct problem solution method has two advantages compared to the parallelepiped-based one (Equation (6)): (1) the symmetry of the τ set with respect to partition D is not required (i.e., the conditions $x_0 - \xi_0 = \frac{\Delta x}{2} l_x, y_0 - \eta_0 = \frac{\Delta y}{2} l_y, l_x \in \mathbb{Z}, l_y \in \mathbb{Z}, N_x = M_x, N_y = M_y$ are not needed); (2) Equation (23) is two times faster than using Equation (6) even theoretically

(and practice indicates larger N values yield larger speed advantages) because the “1st step” of Equation (6) calculates the set $\{G_{0,0}^k(\xi_{i1}, \eta_{j1}, \zeta) | i_1 \in \overline{0, N_x - 1}, j_1 \in \overline{0, N_y - 1}, k \in \overline{0, N_z - 1}\}$ (described in 1) conditions, which requires $8N_x N_y N_z$ calculations of v , while (23) under the same conditions requires only $4N_x N_y (N_z + 1)$ calculations.

3. Results

3.1. Forward Gravity Problem

To evaluate the speed improvement when using the optimized method (23) in comparison with the unoptimized calculation using formula for the gravity field of prism (6), a series of field calculations were performed for 3D density grid models with different element counts, grid sizes and field point calculation counts. The partitioning parameters and calculation times are presented in Table 1. The extra RAM column shows the amount of memory, which is needed to store 64-bit floating point v values for all possible shifts. The same memory area may be reused when precalculating values for the next layer, therefore its total amount is relatively small. Calculation time is presented for a single core of Intel Xeon E5-2620 CPU (2.5 GHz). The corresponding graphs are shown in Figure 2. As can be seen, even for small discretization $50 \times 50 \times 50$, the difference between the non-optimized and optimized methods is two orders of magnitude. This difference tends to become larger with discretization elements increase.

Table 1. Dependence between the calculation time and partitioning parameters.

$N_x = M_x$	$N_y = M_y$	N_z	$N = N_x * N_y * N_z$	Calculation Time Using Formula (6), Seconds	Calculation Time Using Formula (23), Seconds	Extra RAM Used for (23), Kb
50	50	50	125,000	227	10	80
75	75	75	421,875	1725	36	178
100	100	100	1,000,000	7269	84	316
125	125	125	1,953,125	22,183	165	492
150	150	150	3,375,000	55,197	284	708
175	175	175	5,359,375	119,304	451	963
200	200	200	8,000,000	232,602	673	1256
250	250	250	15,625,000	709,844	1313	1961

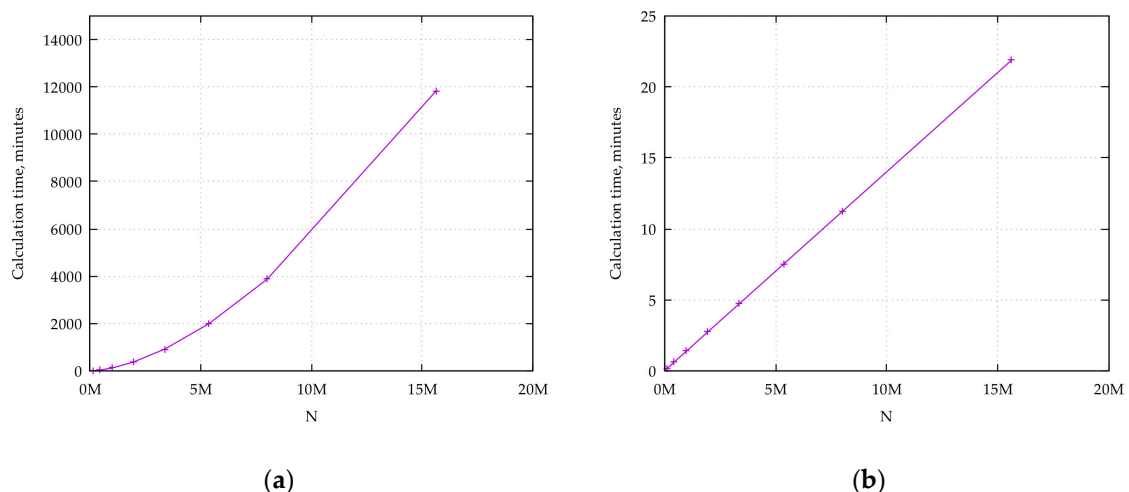


Figure 2. Dependence of calculation time on the partitioning elements count. (a) without optimizations—using Formula (6); (b) with optimizations—using Formula (23).

3.2. Linear Inverse Problem Algorithm Test

To evaluate the linear inverse problem solution algorithm iteration efficiency, we considered an example of an inhomogeneous density distribution model. The D area is the parallelepiped

$D = 50 \times 50 \times 10 \text{ km}^3$ with two objects: a centered, near-surface piece, $D_1 = 20 \times 20 \times 2 \text{ km}^3$, with a density of $\rho_1 = -1.0 \text{ g/cm}^3$, and a deeper piece, D_2 , with the same size and a density of $\rho_2 = 2.0 \text{ g/cm}^3$. The upper face for D_1 is located at a depth of 2 km; the upper face for D_2 is 6 km. The model and field calculation point partitioning grid were selected using the conditions $N_x = N_y = N_z = M_x = M_y = 50$; therefore, $\Delta x = \Delta y = 1 \text{ km}$ and $\Delta z = 0.2 \text{ km}$. Points of field calculations are located above the center points of partition elements: $x_i - \xi_i = \Delta x/2$, and $y_j - \eta_j = \Delta y/2$. As the “observed” field g_m we took as the calculated field of the model. Direct calculation using Formula (6) was performed. Mean value was subtracted from the field, so it was centered at zero for convenience (Figure 3). The average density distribution for the model D is shown in Figure 4. Averaging is performed for all horizontal layers of model area D . We used the result of this averaging as the known $\rho_0(z)$.

Figure 5 shows the solution for the linear inverse problem (density calculation for a given field). Figure 5a was obtained using a priori density distribution information (Figure 4); Figure 5b was calculated for $\rho_0(z) \equiv 1$. Both solutions satisfy the field g_m with relative errors below 1% and were stable for the selecting parameter. The calculation time for the 50×50 grid was below 2 min after 12 iterations.

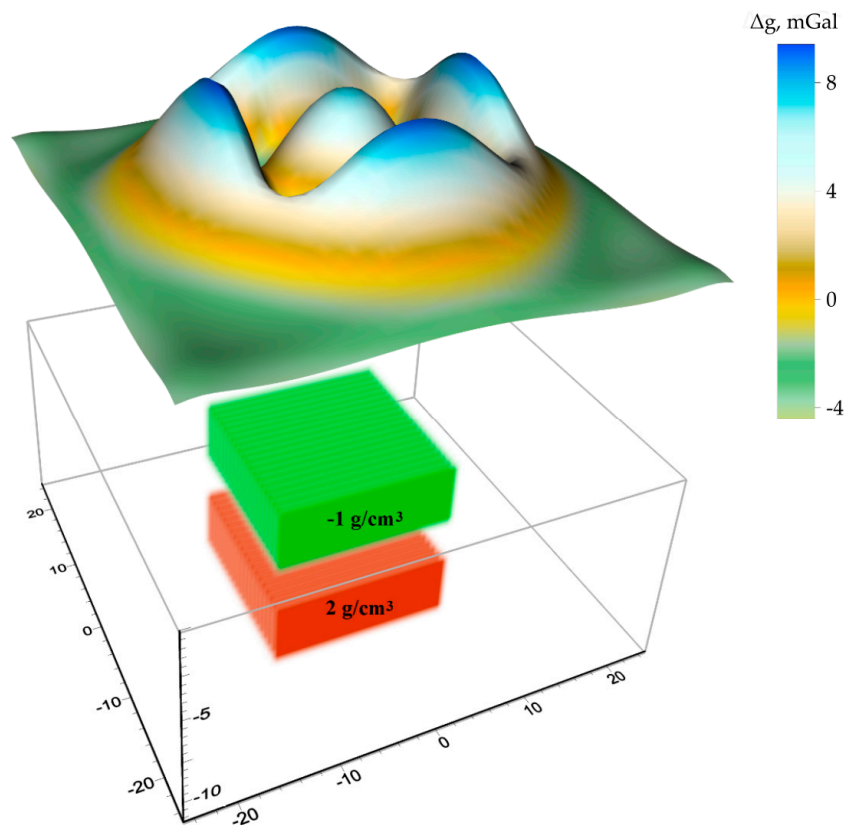


Figure 3. Direct gravity problem for inhomogeneous density parallelepiped: the 3D density distribution model $\rho(x, y, z)$ and calculated model field.

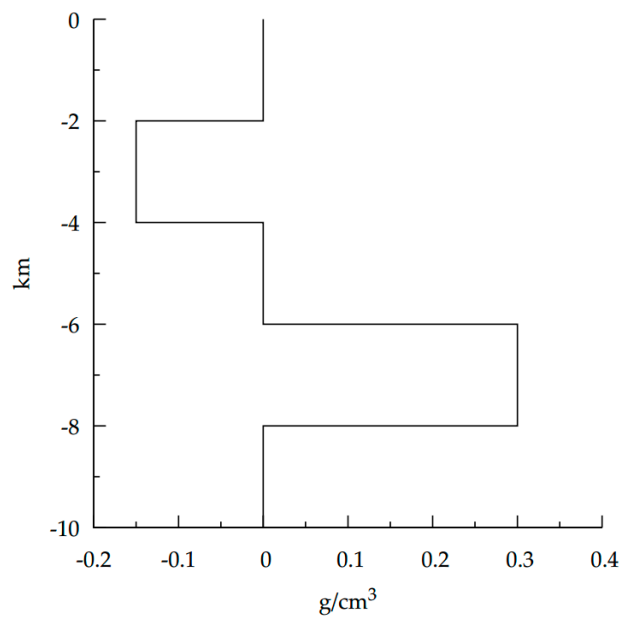


Figure 4. Initial change in model density with depth.

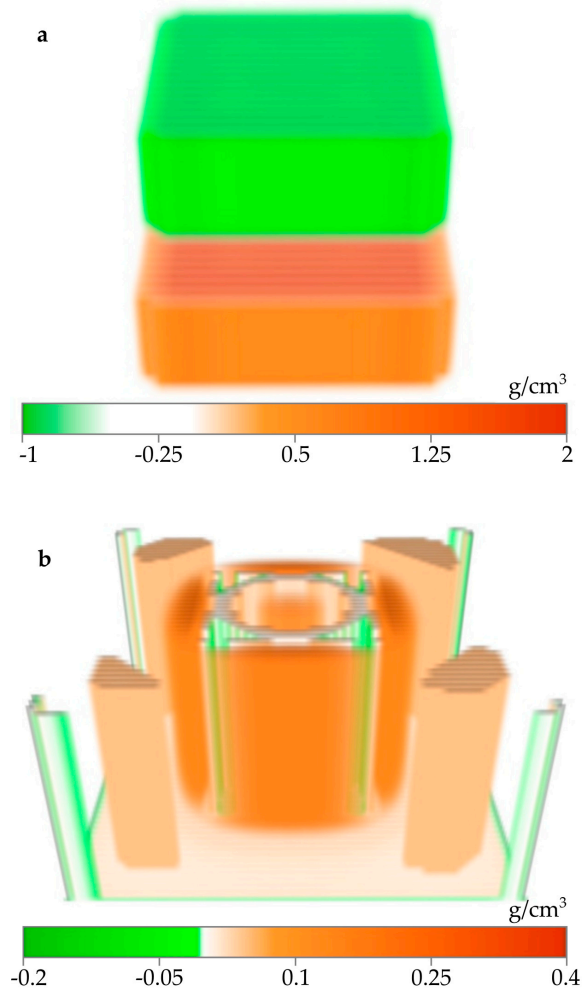


Figure 5. Linear inverse problem solution (a) with a $\rho_0(z)$ dependence for the initial model (Figure 4) and (b) with $\rho_0(z) \equiv 1$.

The solution shown in Figure 5a agrees well with the initial 3D model (Figure 3). The recovery error was 10% for the near-surface piece, D_1 , and 15% for the lower piece, D_2 . The solution in Figure 5b was not acceptable because the obtained density was a different order of magnitude than the initial model density and contained both positive and negative densities at all depths.

Now we demonstrate how the solution is changed when different $\rho_0(z)$ functions are used. The following two examples show what happens with the structure of the result if average density value for the layer, which contains one of the objects, is not set.

Figure 6a presents background density change, in which the lower object is absent. The result of inverse problem solution in this case contains only one object. Despite its geometry is, in general, close to the geometry of the upper object of the model, the density values differ considerably. Both positive and negative density values were obtained. These density values belong to an interval $(-1, 0.5) \text{ g/cm}^3$.

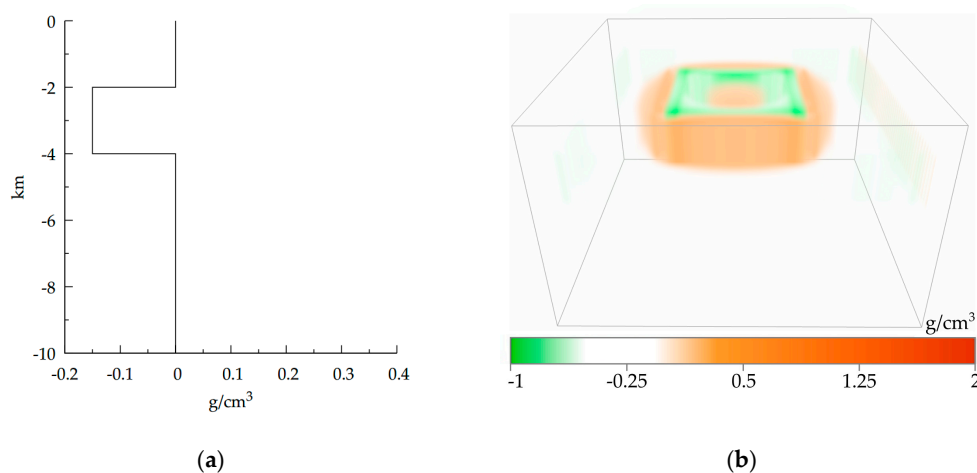


Figure 6. Synthetic example modification #1: (a) Background density distribution $\rho_0(z)$ and (b) Linear problem inverse solution, which is based on it.

In Figure 7a the background density distribution with removed information on the upper object is shown. The result of the inverse problem solution (Figure 7b) is obtained as mutually compensating masses with positive and negative density values, which are changed abruptly. This happens because the algorithm inverts the highly variable model field at relatively large depth. Density values are contained in an interval $(-56.8, 34.1) \text{ g/cm}^3$ range, which is physically meaningless.

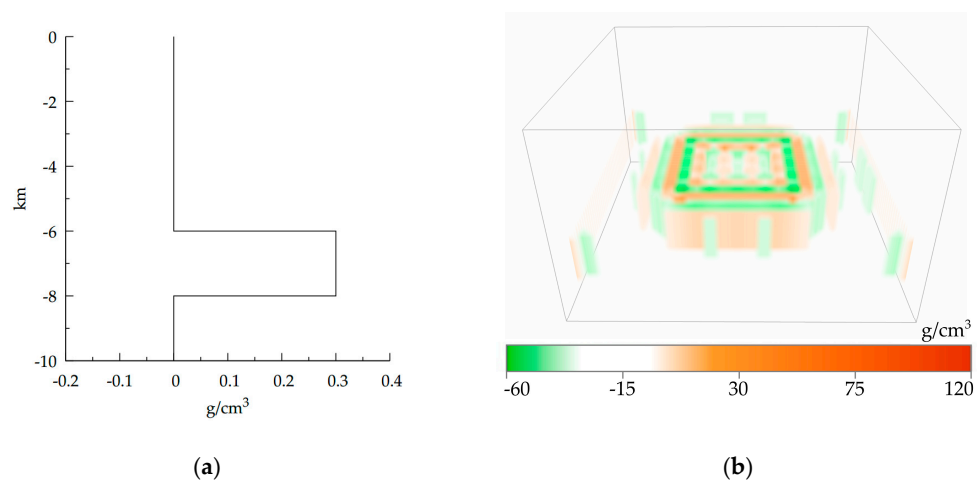


Figure 7. Synthetic example modification #2: (a) Background density distribution $\rho_0(z)$ and (b) Linear problem inverse solution, which is based on it.

The following two background density distributions contain information on both objects, but the position of these objects is changed. In first example, initial density change $\rho_0(z)$ defines objects as being located without any space between them (Figure 8a). The inversion result is shown in Figure 8b. It is visible that objects are located closely one to another. However, to compensate an elevation of masses with positive density value, the additional masses with negative densities appeared. The density values in resulting model belong to an interval $(-2.1, 2.2) \text{ g/cm}^3$.

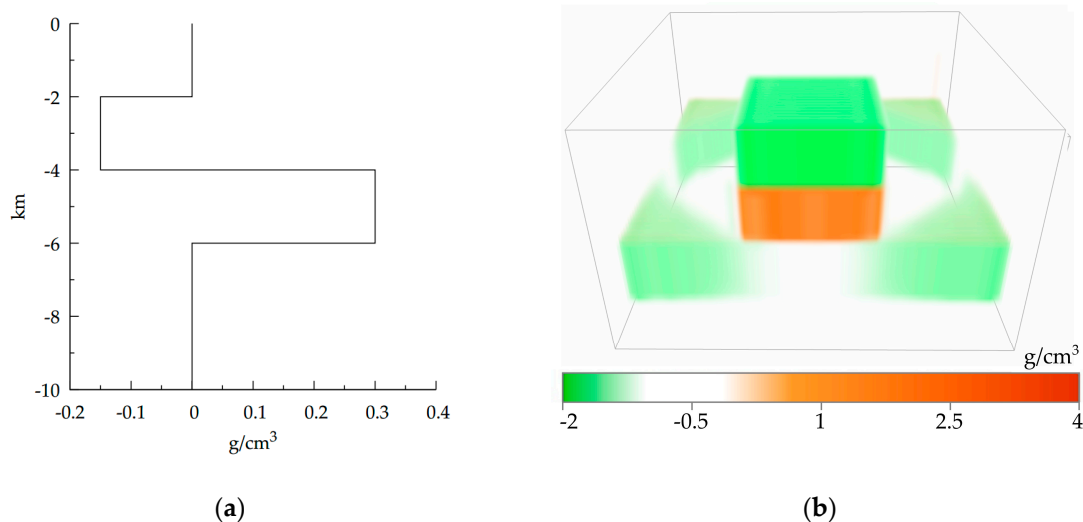


Figure 8. Synthetic example modification #3: (a) Background density distribution $\rho_0(z)$ and (b) Linear problem inverse solution, which is based on it.

For the last synthetic example modification, we increase the depth of objects in the initial density change by 5 km (Figure 9a). The result of inversion (Figure 9b) has features that are similar to modification #2 (Figure 7). The effect of mutual compensation of the masses is visible in both resulting objects. The density values are changed in an interval $(-50.3, 79.5) \text{ g/cm}^3$. This differs significantly from the density change of the initial model.

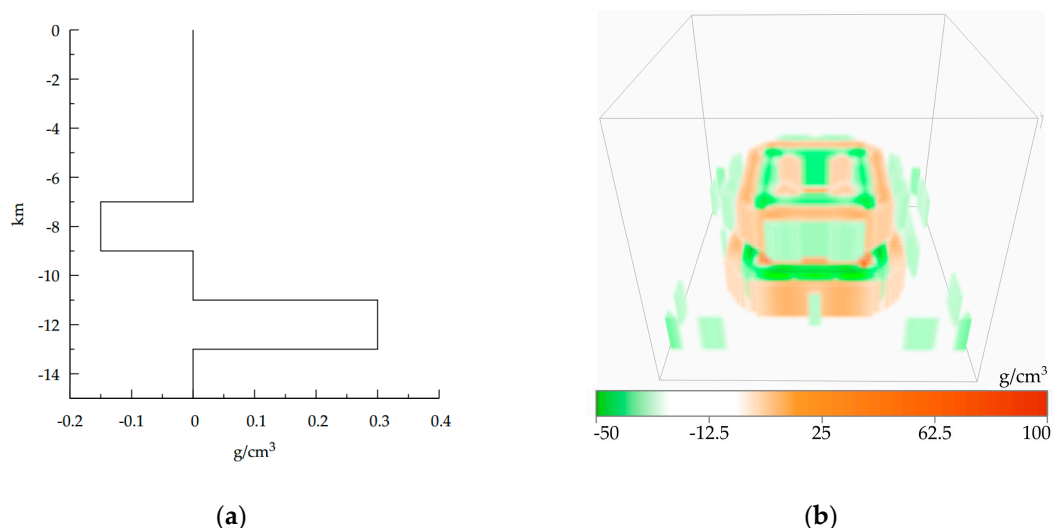


Figure 9. Synthetic example modification #4: (a) Background density distribution $\rho_0(z)$ and (b) Linear problem inverse solution, which is based on it.

Thus, invalid selection of the initial localization of the masses dramatically reduces stability of the inverse problem solution. The wrong initial model can lead to the unstable solution with abrupt

density change, which generates mutually compensating gravity anomalies. This is not a feature of our inversion method only. Such an effect is defined by the ill-posed nature of inversion problem formulation. This is why the gravity modeling process should be preceded with the selection of reasonable limitations, which help to stabilize the solution and keep the physical and geological meaning of the model. Definition of the background density $\rho_0(z)$ is one such limitation. Our goal was to demonstrate that even if the gravity field of the model has a complex morphology produced by objects superposition, the good selection of the background density still allows one to restore the density distribution in a form, which is close to the initial model.

3.3. Case Study

In order to test the feasibility and efficiency of our original approach, we apply it to the Timan-Pechora region's crust and upper mantle. The interpretation of the results is beyond the goal of this work and will be held in a future article.

The study area is located at the boundaries 60–67° N and 48–62° E with a depth down to 80 km from the Earth surface level. Figure 10 shows the observed gravity field in Bouguer reduction. Position of deep seismic sounding (DSS) profiles is drawn in red color. These profiles were used to construct the initial density model approximation. Space between profiles was filled with interpolated data [3] (Figure 11a).

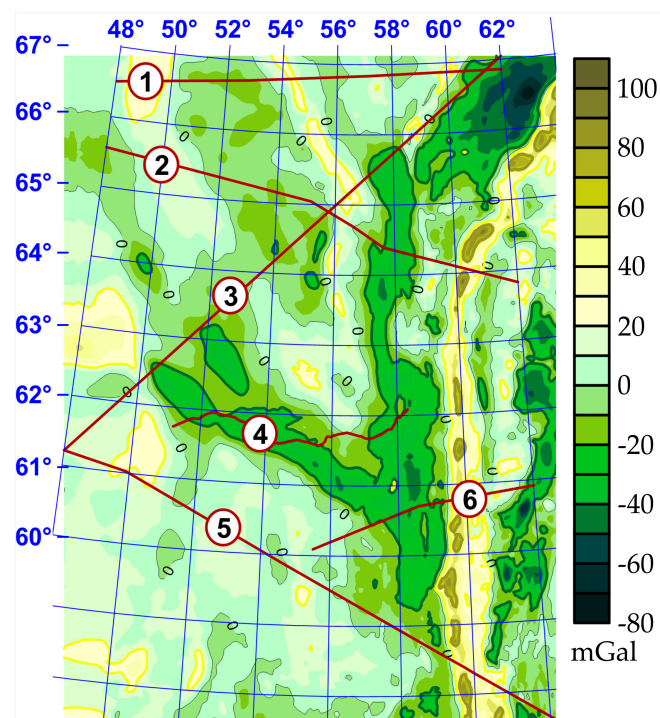


Figure 10. DSS profile scheme with a map of a gravity field anomaly in the Bouguer reduction. Profiles: (1) Agat-2, (2) Quartz, (3) Globus, (4) Syktyvkarskii, (5) Rubin-1, and (6) Krasnoleninskiy.

The model was discretized on a regular $256 \times 256 \times 80$ grid for a field discretization of 256×256 (using linear interpolation of 1:1,000,000 scale maps). The field discrepancy of the observed and initial model fields was 18.5 mGal in terms of standard deviation. The $\rho_0(z)$ distribution using this method obtained from the modeled average density based on depth (Figure 12). The selection process ended after 76 iterations, which took 304 s on NVidia GTX 980 using the CUDA-optimized implementation of the algorithm, and the resultant density distribution model was obtained (Figure 11b). The observed and modeled field discrepancy was 0.1 mGal, as chosen for the stopping condition.

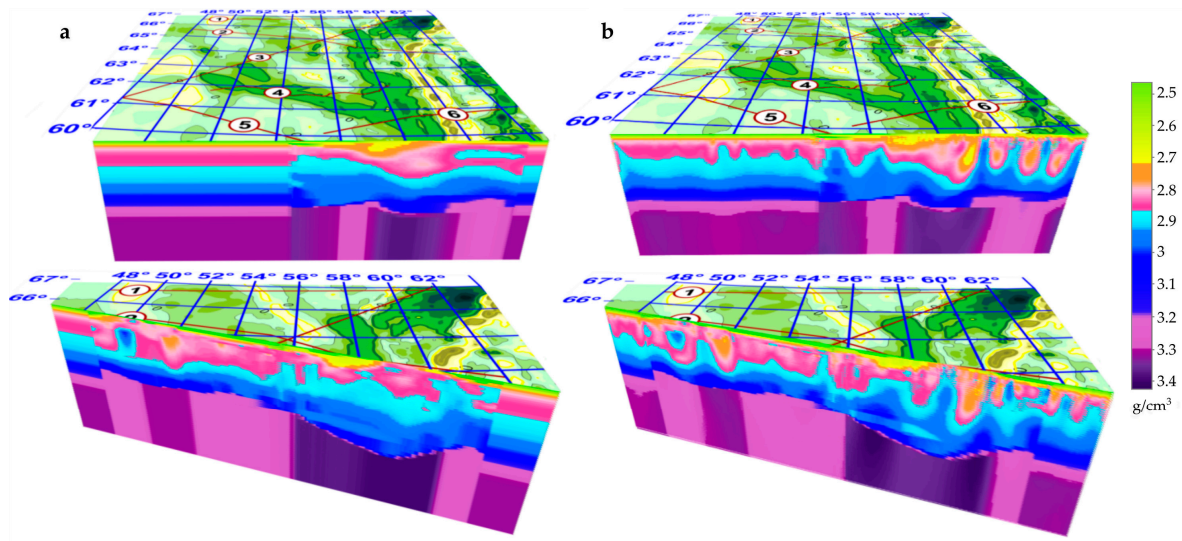


Figure 11. Initial (a) and fitted (b) 3D density model for the Timan-Pechora plate. The bottom image cuts along the shown Quartz profile.

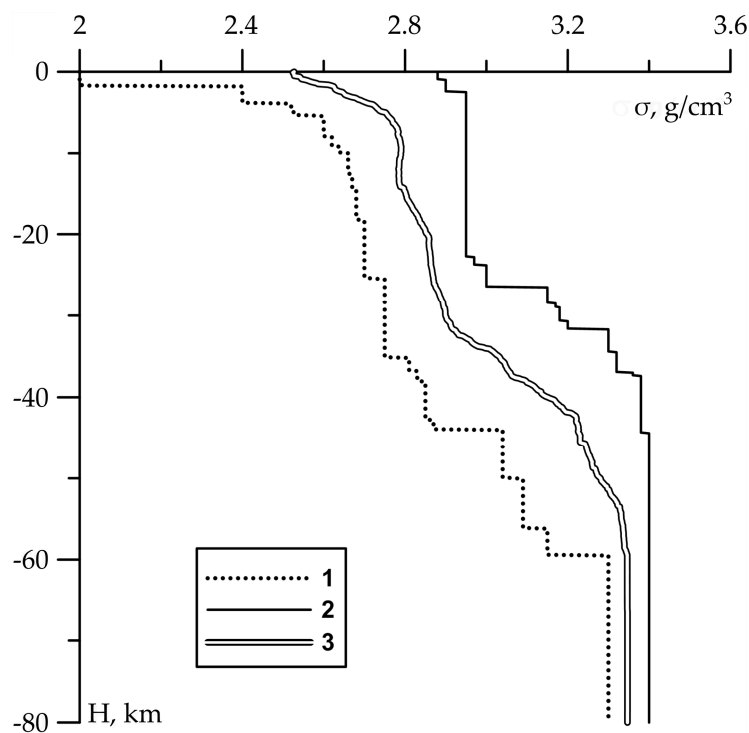


Figure 12. Dependence of variance on depth for the initial model (double line), the minimal (dotted line) and maximal (solid line) density values versus depth.

4. Conclusions

A new method has been suggested to invert the 3D gravity data. The method is based on a rapid algorithm for calculating the gravity effect of a volumetric density distribution, the initial 3D density model deduced from additional geophysical information, the selection of the density class at which the solution is determined stably, and the rapid stable algorithm for solving linear inverse problem.

The following conclusions are drawn. A new rapid algorithm has been suggested to calculate the gravity effect of a volumetric density distribution. The method of local corrections is developed to solve a linear gravity inverse problem. We solve an integral equation for the multiplicative function

determining a volumetric density distribution. We take into account the instability of the inverse problem by means of a form of regularization.

This method is favorable to analogues because of its lower CPU and memory resource usage. The performance was increased due to the use of a single antiderivative computation using equal arguments, which are repetitive because the field and grid have regular structures. Using the idea of local minimizations with the new iteration algorithm allowed for an adaptive regularization to be developed for sustainably solving 3D linear inverse gravity problems. Unlike gradient methods, this technique does not require a nonlinear minimization, is resistant to observed field fluctuations, is easier to implement and reaches the target discrepancy in less time. The presented methods were integrated into a computer geophysical data interpretation system and used to solve actual seismic and density modeling problems. The 3D density model was obtained automatically without interactive forward modeling, which dramatically diminishes the time expenditure. The linear inversion algorithm efficiency was demonstrated using test and practical examples, a geophysical model of the Earth's crust and upper mantle was constructed for extensive portions of the Timan-Pechora plate. We emphasize that these models were based on joint interpretation of geophysical data.

Author Contributions: Project Administration, P.S.M.; Formal Analysis, P.S.M.; Conceptualization, P.S.M., I.V.L. and D.D.B.; Software, D.D.B.; Writing—Original Draft Preparation, D.D.B., A.G.T.; Writing—Review and Editing, P.S.M., A.G.T. and I.V.L.

Funding: This research received no external funding.

Conflicts of Interest: The authors declare no conflict of interest

References

1. Cordell, L.; Henderson, R.G. Iterative three-dimensional solution of gravity anomaly data using a digital computer. *Geophysics* **1968**, *33*, 596–601. [[CrossRef](#)]
2. Sampietro, D.; Sanso, F. Uniqueness theorems for inverse gravimetric problems. In *VII Hotine-Marussi Symposium on Mathematical Geodesy*; Sneeuw, N., Novák, P., Crespi, M., Sansò, F., Eds.; Springer: Berlin/Heidelberg, Germany, 2012; Volume 137, pp. 111–115.
3. Martyshko, P.; Ladovskiy, I.; Tsidaev, A.; Byzov, D.; Kolmogorova, V. Grid joint seismic and density modeling technique for upper lithosphere. In *Proceedings of the XI-th International Conference on Geoinformatics—Theoretical and Applied Aspects*, Kiev, Ukraine, 14–17 May 2012.
4. Nagy, D. Gravitational attraction of a right rectangular prism. *Geophysics* **1966**, *31*, 362–371. [[CrossRef](#)]
5. Baneergee, B.; Gupta, S.P.D. Gravitational attraction of a rectangular parallelepiped. *Geophysics* **1977**, *42*, 1053–1055. [[CrossRef](#)]
6. Mostafa, M.E. Finite cube elements method for calculating gravity anomaly and structural index of solid and fractal bodies with defined boundaries. *Geophys. J. Int.* **2008**, *172*, 887–902. [[CrossRef](#)]
7. Couder-Castañeda, C.; Ortiz-Alemán, J.C.; Orozco-del-Castillo, M.G.; Nava-Flores, M. Forward modeling of gravitational fields on hybrid multi-threaded cluster. *Geofísica Int.* **2015**, *54*, 31–48. [[CrossRef](#)]
8. Dubey, C.P.; Tiwari, V.M. Computation of the gravity field and its gradient: Some applications. *Comput. Geosci.* **2016**, *88*, 83–96. [[CrossRef](#)]
9. Li, Y.; Oldenburg, D.W. 3-D inversion of gravity data. *Geophysics* **1998**, *63*, 109–119. [[CrossRef](#)]
10. Cai, Y.; Wang, C. Fast finite element calculation of gravity anomaly in complex geological regions. *Geophys. J. Int.* **2005**, *162*, 696–708. [[CrossRef](#)]
11. Novoselitskii, V.M. On the Theory of Determining Density Variations in a Horizontal Layer from Gravity Anomaly Data. *Izv. Akad. Nauk SSSR Fiz. Zemli* **1965**, *5*, 25–32.
12. Blakely, R.J. *Potential Theory in Gravity and Magnetic Applications*; Cambridge University Press: Cambridge, UK, 1995.
13. Prutkin, I.L. The Solution of Three-Dimensional Inverse Gravimetric Problem in the Class of Contact Surfaces by the Method of Local Corrections. *Proc. Acad. Sci. USSR Phys. Earth (Engl. Transl.)* **1986**, *22*, 49–60.
14. Martyshko, P.S.; Prutkin, I.L. Technology of depth distribution of gravity field sources. *Geofizicheskiy Zhurnal (Geophys. J.)* **2003**, *25*, 159–165. (In Russian)

15. Prutkin, I.; Vajda, P.; Tenzer, R.; Bielik, M. 3D inversion of gravity data by separation of sources and the method of local corrections: Kolarovo gravity high case study. *J. Appl. Geophys.* **2011**, *75*, 472–478. [[CrossRef](#)]
16. Martyshko, P.S.; Ladovskiy, I.V.; Tsidaev, A.G. Construction of Regional Geophysical Models Based on the Joint Interpretation of Gravimetry and Seismic Data. *Izvestiya Phys. Solid Earth* **2010**, *46*, 931–942. [[CrossRef](#)]



© 2018 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).