The computation aspects of the equivalent-layer technique: review and perspective

Diego Takahashi [1,*], André L. A. Reis [2], Vanderlei C. Oliveira Jr. [1] and Valéria C. F. Barbosa [1]

[1] *Observatório Nacional, Department of Geophysics, Rio de Janeiro, Brasil*
[2] *Universidade do Estado do Rio de Janeiro, Department of Applied Geology, Rio de Janeiro, Brasil*

Correspondence*:
Valéria C.F. Barbosa
valcris@on.br

ABSTRACT

Equivalent-layer technique is a powerful tool for processing potential-field data in the space domain. However, the greatest hindrance for using the equivalent-layer technique is its high computational cost for processing massive data sets. The large amount of computer memory usage to store the full sensitivity matrix combined with the computational time required for matrix-vector multiplications and to solve the resulting linear system, are the main drawbacks that made unfeasible the use of the equivalent-layer technique for a long time. More recently, the advances in computational power propelled the development of methods to overcome the heavy computational cost associated with the equivalent-layer technique. We present a comprehensive review of the computation aspects concerning the equivalent-layer technique addressing how previous works have been dealt with the computational cost of this technique. Historically, the high computational cost of the equivalent-layer technique has been overcome by using a variety of strategies such as: moving data-window scheme, equivalent data concept, wavelet compression, lower-dimensional subspace, quadtree discretization, reparametrization of the equivalent layer by a piecewise-polynomial function, iterative scheme without solving a system of linear equations and the convolutional equivalent layer using the concept of block-Toeplitz Toeplitz-block (BTTB) matrices. We compute the number of floating-point operations of some of these strategies adopted in the equivalent layer technique to show their effectiveness in reducing the computational demand. Numerically, we also address the stability of some of these strategies used in the equivalent layer technique by comparing with the stability via the classic equivalent-layer technique with the zeroth-order Tikhonov regularization.

Keywords: equivalent layer, gravimetry, fast algorithms, computational cost, stability analysis

1 INTRODUCTION

In accord with potential theory, a continuous potential-field data (gravity and magnetic data) produced by any source can be exactly reproduced by a continuous and infinite 2D physical-property surface distribution that is called the equivalent layer. The equivalent layer is a mathematical solution of Laplace's equation in the source-free region with the observed potential-field data as the Dirichlet boundary condition (Kellogg, 1929). Grounded on well-established potential theory, the equivalent-layer technique has been used by exploration geophysicists for processing potential-field data since the late 1960s (Dampney, 1969).

Although there was always a great demand for gravity and magnetic data processing, the equivalent-layer technique has not been massively used. This occurs because its high computational cost makes the equivalent-layer technique computationally inefficient for processing massive data sets. In the classic equivalent-layer technique, the continuous problem of the equivalent layer involving integrals is approximated by a discrete form of the equivalent layer. First, a discrete and finite set of equivalent sources (point masses, prisms, magnetic diploes, doublets) is arranged in a layer with finite horizontal dimensions and located below the observation surface. Next, a linear system of equations is set up with a large and full sensitivity matrix. Then, a regularized linear inverse problem is solved to estimate the physical property of each equivalent source within the discrete equivalent layer subject to fitting a discrete set of potential-field observations. Finally, the estimated physical-property distribution within the equivalent layer is used to accomplish the desired processing of the potential-field data (e.g., interpolation, upward/downward continuation, reduction to the pole). The latter step is done by multiplying the matrix of Green's functions associated with the desired transformation by the estimated physical-property distribution.

Beginning in the late 1980s, the equivalent-layer techniques computationally efficient have arose. To our knowledge, the first method towards improving the efficiency was proposed by Leão and Silva (1989) who used an overlapping moving-window scheme spanning the data set. The strategy adopted in Leão and Silva (1989) involves solving several smaller, regularized linear inverse problems instead of one large problem. This strategy uses a small data window and distributes equivalent sources on a small regular grid at a constant depth located below the data surface. Leão and Silva (1989) ensure that sources window extends beyond the boundaries of the data window. For each position of the data window, this scheme consists in computing the processed field at the center of the data window only and the next estimates of the processed field are obtained by shifting the data window across the entire dataset. Recently, Soler and Uieda (2021) developed a computational approach to increase the efficiency of the equivalent-layer techinique by combining two strategies. The fisrt one — the block-averaging source locations — reduces the model parameters and the second strategy — the gradient-boosted algorithm — reduces the size of the linear system to be solved by fitting the equivalent source model iteratively along overlapping windows. Notice that the equivalent-layer strategy of using a moving-window scheme either in Leão and Silva (1989) or in Soler and Uieda (2021) is similar to discrete convolution.

In another approach to reduce computational workload of the equivalent-layer technique Mendonça and Silva (1994) developed an iterative procedure by incorporating one data point at a time and thus selecting a smaller data set. This strategy adopted by Mendonça and Silva (1994) is known as 'equivalent data concept'. Li and Oldenburg (2010) transformed the full sensitivity matrix into a sparse one using the compression of the coefficient matrix via wavelet transforms based on the orthonormal compactly supported wavelets. For jointly processing the components of gravity-gradient data using the equivalent-source processing, Barnes and Lumley (2011) applied the quadtree model discretization to generate a sparse linear system of equations. Davis and Li (2011) adaptively dicretized the model (quadtree model discretization) based on localized anomalies and used wavelet transforms to reduce, reordered the model parameters (Hilbert space-filling curves) and compressed each row of the sensitivity matrix of the reordered parameter set (wavelet transforms). By using the subspace method, Mendonça (2020) reduced the dimension of the linear system of equations to be solved in the equivalent-layer technique. The subspace bases span the parameter-model space and they are constructed by applying the singular value decomposition to the matrix containing the gridded data. These strategies followed by Li and Oldenburg (2010), Barnes and Lumley (2011), Davis and Li (2011) and Mendonça (2020) may be grouped into the strategy of compression approaches to solve large linear system of equations.

Following the strategy of reparametrization of the equivalent layer, Oliveira Jr. et al. (2013) reduced the model parameters by approximating the equivalent-source layer by a piecewise-polynomial function defined on a set of user-defined small equivalent-source windows. The estimated parameters are the polynomial coefficients for each window and they are much smaller than the original number of equivalent sources. Siqueira et al. (2017) developed an iterative solution where the sensitivity matrix is transformed into a diagonal matrix with constant terms through the use of the 'excess mass criterion' and of the positive correlation between the observed gravity data and the masses on the equivalent layer. Jirigalatu and Ebbing (2019) combined the Gauss-fast Fourier transform (FFT) with Landweber's algorithm and proposed a fast equivalent-layer technique for jointly processing two-components of the gravity-gradient data. The Landweber's algorithm has some similarities with with gradient-descent algorithm. The strategies worked out by Siqueira et al. (2017) and Jirigalatu and Ebbing (2019) avoid calculating the Hessian matrix and solving linear system of equations.

Recently, Takahashi et al. (2020, 2022), developed fast and effective equivalent-layer techniques for processing, respectively, gravity and magnetic data by modifying the forward modeling to estimate the physical-property distribution over the layer through a 2D discrete convolution that can be efficiently computed via 2D FFT. These methods took advantage of the Block-Toeplitz Toeplitz-block (BTTB) structure of the sensitivity matrices, allowing them to be calculated by using only their first column. In practice, the forward modeling uses a single equivalent source, which significantly reduces the the required RAM memory. Takahashi et al. (2020, 2022) employed the strategy of the convolutional equivalent layer using the concept of BTTB matrices.

Here, we present a comprehensive review of

2 THE EQUIVALENT-LAYER TECHNIQUE

2.1 Fundamentals

Consider a set of N potential-field observations (gravity or magnetic data) $d_i^o\ (x_i, y_i, z_i)$, $i = 1, \ldots, N$, at the ith observation point (x_i, y_i, z_i) of a Cartesian coordinate system with x-, y- and z-axis pointing to north, east and down, respectively. Physically, the discrete set of potential-field observations is produced by a unknown source distribution in the subsurface. Mathematically, it represents a discrete set of a harmonic function.

A standard way to deal with the classical equivalent-layer technique is approximate the observed potential-field data by the predicted data, which in turn are produced by a fictitious layer of sources, called equivalent layer. The equivalent layer is located below the observation surface, at depth z_0 ($z_0 > z_i$), and with finite horizontal dimensions being composed by a finite discrete set of equivalent sources (e.g., point masses, dipoles, or prisms). Mathematically, this approximation can be written in matrix notation as

$$\mathbf{d} = \mathbf{A}\mathbf{p}\,, \tag{1}$$

where $\mathbf{d}$ is an N-dimensional predicted data vector whose ith element, $d_i\ (x_i, y_i, z_i)$, $i = 1, \ldots, N$, is the predicted potential-field observation, $\mathbf{p}$ is an M-dimensional parameter vector whose jth element p_j can be a physical property of the jth equivalent source and $\mathbf{A}$ is the $N \times M$ sensitivity matrix whose ijth element a_{ij} is a harmonic function.

2.2 Computational strategies

The classical equivalent-layer technique consists of estimating the parameter vector $\mathbf{p}$ from the N-dimensional observed data vector $\mathbf{d^o}$ whose ith element is defined as the $d_i^o\ (x_i, y_i, z_i)$, $i = 1, \ldots, N$. Usually, this estimate can be obtained by a regularized least-squares solution. The estimated parameter is stable, fits the observed data and can be used to yield a desired linear transformation of the data, such as interpolation, upward (or downward) continuation, reduction to the pole, joint processing of gravity gradient data and more. Mathematically, the desired linear transformation of the data can be obtained by

$$\hat{\mathbf{t}} = \mathbf{T}\mathbf{p}^*\,, \tag{2}$$

where $\hat{\mathbf{t}}$ is an N-dimensional transformed data vector, $\mathbf{p}^*$ is an M-dimensional estimated parameter vector and $\mathbf{T}$ is the $N \times M$ matrix of Green's functions whose ijth element is the transformed field at the ith observation point produced by the jth equivalent source.

The biggest hurdle to use the classical equivalent-layer technique is the computational complexity to handle large datasets because the sensitivity matrix $\mathbf{A}$ (equation 1) is dense. Usually, the estimated parameter vector $\mathbf{p}^*$ requires to solve a large-scale linear inversion which in turn means to deal with some obstacles concerning large computational cost: i) the large computer memory to store large and full matrices; ii) the long computation time to mutiply a matrix by a vector; and iii) the long computation time to solve a large linear system of equations.

Here, we review some strategies for reducing the computational cost of equivalent-layer technique. These strategies are the following:

2.2.1 The moving data-window scheme

Leão and Silva (1989) reduced the total processing time and memory usage of equivalent-layer technique by means of a moving data-window scheme. A small moving data window with N_w observations and a small equivalent layer with M_w equivalent sources ($M_w > N_w$) located below the observations are established. For each position of a moving-data window, Leão and Silva (1989) estimate a stable solution $\mathbf{p_w}^*$ by using a data-space approach with the zeroth-order Tikhonov regularization (Aster et al., 2018), i.e.,

$$\left(\mathbf{A_w}\mathbf{A_w}^\top + \mu\mathbf{I}\right)\mathbf{w} \quad = \quad \mathbf{d_w}^o\,, \tag{3a}$$

$$\mathbf{A_w}^\top\,\mathbf{w} \quad = \quad \mathbf{p_w}^*\,, \tag{3b}$$

where $\mathbf{w}$ is a dummy vector, μ is a regularizing parameter, $\mathbf{d_w}^o$ is an N_w-dimensional vector containing the observed potential-field data, $\mathbf{A_w}$ is an $N_w \times M_w$ sensitivity matrix related to a moving-data window, $\mathbf{I}$ is an identity matrix of order N_w and the superscript $\top$ stands for a transpose. After estimating an $M_w \times 1$ parameter vector $\mathbf{p_w}^*$ (equation 3b) the desired transformation of the data is only calculated at the central point of each moving-data window, i.e.:

$$\hat{t}_k = \mathbf{t}_k^\top\,\mathbf{p_w}^*\,, \tag{4}$$

where $\hat{t}_k$ is the transformed data calculated at the central point k of the data window and $\mathbf{t}_k$ is an $M_w \times 1$ vector whose elements form the kth row of the $N_w \times N_w$ matrix of Green's functions $\mathbf{T}$ (equation 2) of the desired linear transformation of the data.

By shifting the moving-data window with a shift size of one data spacing, a new position of a data window is set up. Next, the aforementioned process (equations 3b and 4) is repeated for each position of a moving-data window, until the entire data have been processed. Hence, instead of solving a large inverse problem, Leão and Silva (1989) solve several much smaller ones.

To reduce the size of the linear system to be solved, Soler and Uieda (2021) adopted the same strategy proposed, originally, by Leão and Silva (1989) of using a small moving-data window sweeping the whole data. In Leão and Silva (1989), a moving-data window slides to the next adjacent data window following a sequential movement, the predicted data is calculated inside the data window and the desired transformation are only calculated at the center of the moving-data window. Unlike Leão and Silva (1989), Soler and Uieda (2021) do not adopt a sequential order of the data windows; rather, they adopt a randomized order of windows in the iterations of the gradient-boosting algorithm (Friedman, 2001 and 2002). The gradient-boosting algorithm in Soler and Uieda (2021) estimates a stable solution using the data and the equivalent sources that fall within a moving-data window; however, it calculates the predicted data and the residual data in the whole survey data. Next, the residual data that fall within a new position of the data window is used as input data to estimate a new stable solution within the data window which in turn is used to calculated a new predicted data and a new residual data in the whole survey data. Finally, unlike Leão and Silva (1989), in Soler and Uieda (2021) neither the data nor the equivalent sources need to be distributed in regular grids. Indeed, Leão and Silva (1989) built their method using regular grids, but in fact regular grids are not necessary. Regarding the equivalent-source layout, Soler and Uieda (2021) proposed the block-averaged sources locations in which the survey area is divided into horizontal blocks and one single equivalent source is assigned to each block. Each single source per block is placed over the layer with its horizontal coordinates given by the average horizontal positions of observation points. According to Soler and Uieda (2021), the block-averaged sources layout reduces the number of equivalent sources significantly and the gradient-boosting algorithm provides even greater efficiency in terms of data fitting.

2.2.2 The equivalent-data concept

To reduced the total processing time and memory usage of equivalent-layer technique, Mendonça and Silva (1994) proposed a strategy called 'equivalent data concept'. The equivalent data concept is grounded on the principle that there is a subset of redundant data that does not contribute to the final solution and thus can be dispensed. Conversely, there is a subset of observations, called equivalent data, that contributes effectively to the final solution and fits the remaining observations (redundant data). Iteractively, Mendonça and Silva (1994) selected the subset of equivalent data that is substantially smaller than the original dataset. This selection is carried out by incorporating one data point at a time.

According to Mendonça and Silva (1994), the number of equivalent data is about one-tenth of the total number of observations. These authors used the equivalent data concept to carry out an interpolation of gravity data. They showed a reduction of the total processing time and memory usage by, at least, two orders of magnitude as opposed to using all observations in the interpolation process via the classical equivalent-layer technique.

2.2.3 The wavelet compression and lower-dimensional subspace

For large data sets, the sensitivity matrix $\mathbf{A}$ (equation 1) is a drawback in applying the equivalent-layer technique because it is a large and dense matrix.

Li and Oldenburg (2010) transformed a large and full sensitivity matrix into a sparse one by using fast wavelet transforms. In the wavelet domain, Li and Oldenburg (2010) applyied a 2D wavelet transform to each row and column of the original sensitivity matrix $\mathbf{A}$ to expand it in the wavelet bases. This operation can be done by premultiplying the original sensitivity matrix $\mathbf{A}$ by a matrix representing the 2D wavelet transform $\mathbf{W_2}$ and then the resulting is postmultiplied by the transpose of $\mathbf{W_2}$ (i.e., $\mathbf{W_2}^\top$).

$$\tilde{\mathbf{A}} = \mathbf{W_2}\,\mathbf{A}\,\mathbf{W_2}^\top\,, \tag{5}$$

where $\tilde{\mathbf{A}}$ is the expanded original sensitivity matrix in the wavelet bases with many elements zero or close to zero. Next, the matrix $\tilde{\mathbf{A}}$ is replaced by its sparse version $\tilde{\mathbf{A}}_\mathbf{s}$ in the wavelet domain which in turn is obtained by retaining only the large elements of the $\tilde{\mathbf{A}}$. Thus, the elements of $\tilde{\mathbf{A}}$ whose amplitudes fall below a relative threshold are discarded. In Li and Oldenburg (2010), the original sensitivity matrix $\mathbf{A}$ is high compressed resulting in a sparce matrix $\tilde{\mathbf{A}}_\mathbf{s}$ with a few percent of nonzero elements and the the inverse problem is solved in the wavelet domain by using $\tilde{\mathbf{A}}_\mathbf{s}$ and a incomplete conjugate gradient least squares, without an explicit regularization parameter and a limited number of iterations. The solution is obtained by solving the following linear system

$$\tilde{\mathbf{A}}_\mathbf{L}^\top\,\tilde{\mathbf{A}}_\mathbf{L}\,\tilde{\mathbf{p}}_\mathbf{L}^*\; = \tilde{\mathbf{A}}_\mathbf{L}^\top\,\tilde{\mathbf{d}}^o\,, \tag{6}$$

where $\tilde{\mathbf{p}}_\mathbf{L}^*$ is obtained by solving the linear system given by equation 6,

$$\tilde{\mathbf{A}}_\mathbf{L} \;=\; \tilde{\mathbf{A}}_\mathbf{s}\,\tilde{\mathbf{L}}^{-\mathbf{1}}, \tag{7a}$$

$$\tilde{\mathbf{p}}_\mathbf{L} \;=\; \tilde{\mathbf{L}}\tilde{\mathbf{p}}, \tag{7b}$$

$$\tilde{\mathbf{d}}^o \;=\; \mathbf{W_2}\,\mathbf{d}^o, \tag{7c}$$

where $\tilde{\mathbf{L}}$ is a diagonal and invertible weighting matrix representing the finite-difference approximation in the wavelet domain. Finally, the distribution over the equivalent layer in the space domain $\mathbf{p}$ is obtained by applying an inverse wavelet transform in two steps, i.e.:

$$\tilde{\mathbf{p}} = \tilde{\mathbf{L}}^{-1}\, \tilde{\mathbf{p}}^{*}_{\mathbf{L}} \,, \tag{8}$$

and

$$\mathbf{p} = \mathbf{W_2}\, \tilde{\mathbf{p}} \,. \tag{9}$$

Although the data misfit quantifying the difference between the observed and predicted data by the equivalent source is calculated in the wavelet domain, we understand that the desired transformation is calculated via equation 2 which uses a full matrix of Green's functions $\mathbf{T}$.

Li and Oldenburg (2010) used the equivalent-layer technique with a wavelet compression to perform an upward continuation of total-field anomaly between uneven surfaces. For regularly spaced grid of data, Li and Oldenburg (2010) reported that high compression ratios are achived with insignificant loss of accuracy. As compared to the upward-continued total-field anomaly by equivalent layer using the dense matrix, Li and Oldenburg's (2010) approach, using the Daubechies wavelet, decreased CPU (central processing unit) time by up to two orders of magnitude.

Mendonça (2020) overcame the solution of intractable large-scale equivalent-layer problem by using the subspace method (e.g., Skilling and Bryan, 1984; Kennett et al., 1988; Oldenburg et al., 1993; Barbosa et al., 1997). The subspace method reduces the dimension of the linear system of equations to be solved. Given a higher-dimensional space (e.g., M-dimensional model space, $\mathbb{R}^M$), there exists many lower-dimensional subspaces (e.g., Q-dimensional subspace) of $\mathbb{R}^M$. The linear inverse problem related to the equivalent-layer technique consists in finding an M-dimension parameter vector $\mathbf{p} \in \mathbb{R}^M$ which adequately fits the potential-field data. The subspace method looks for a parameter vector who lies in a Q-dimensional subspace of $\mathbb{R}^M$ which, in turn, is spanned by a set of Q vectors $\mathbf{v}_i = 1, ..., Q$, where $\mathbf{v}_i \in \mathbb{R}^M$ In matrix notation, the parameter vector in the subspace method can be written as

$$\mathbf{p} = \mathbf{V}\, \boldsymbol{\alpha} \,, \tag{10}$$

where $\mathbf{V}$ is an $M \times Q$ matrix whose columns $\mathbf{v}_i = 1, ..., Q$ form a basis vectors for a subspace Q of $\mathbb{R}^M$. In equation 10, the parameter vector $\mathbf{p}$ is defined as a linear combination in the space spanned by Q basis vectors $\mathbf{v}_i = 1, ..., Q$ and $\boldsymbol{\alpha}$ is a Q-dimensional unknown vector to be determined. The main advantage of the subspace method is that the linear system of M equations in M unknowns to be originally solved is reduced to a new linear system of Q equations in Q unknowns which requires much less computational effort since $Q << M$, i.e.:

$$\mathbf{V}^{\top}\, \mathbf{A}^{\top}\, \mathbf{A}\, \mathbf{V}\, \boldsymbol{\alpha}^{*} \;=\; \mathbf{V}^{\top}\, \mathbf{d}^{\boldsymbol{o}} \,. \tag{11}$$

To avoid the storage of matrices $\mathbf{A}$ and $\mathbf{V}$, Mendonça (2020) evaluates an element of the matrix $\mathbf{AV}$ by calculating the dot product between the row of matrix $\mathbf{A}$ and the column of the matrix $\mathbf{B}$. After estimating $\boldsymbol{\alpha}^{*}$ (equation 11) belonging to a Q-dimensional subspace of $\mathbb{R}^M$, the distribution over the equivalent layer $\mathbf{p}$ in the $\mathbb{R}^M$ is obtained by applying equation 10. The choice of the Q basis vectors $\mathbf{v}_i = 1, ..., Q$ (equation 10) in the subspace method is not strict. Mendonça (2020), for example, chose the eigenvectors yielded by applying the singular value decomposition of the matrix containing the gridded data set. The number of eigenvectors used to form basis vectors will depend on the singular values.

The proposed subspace method for solving large-scale equivalent-layer problem by Mendonça (2020) was applied to estimate the mass excess or deficiency caused by causative gravity sources.

2.2.4 The quadtree discretization

To make the equivalent-layer technique tractable, Barnes and Lumley (2011) also transformed the dense sensitivity matrix $\mathbf{A}$ (equation 1) into a sparse matrix. In Barnes and Lumley (2011), a sparce version of the sensitivity matrix is achived by grouping equivalent sources (e.g., they used prisms) distant from an observation point together to form a larger prism or larger block. Each larger block has averaged physical properties and averaged top- and bottom-surfaces of the grouped smaller prisms (equivalent sources) that are encompassed by the larger block. The authors called it the 'larger averaged block' and the essence of their method is the reduction in the number of equivalent sources, which means a reduction in the number of parameters to be estimated implying in model dimension reduction.

The key of the Barnes and Lumley's (2011) method is the algorithm for deciding how to group the smaller prisms. In practice, these authors used a recursive bisection process that results in a quadtree discretization of the equivalent-layer model.

By using the quadtree discretization, Barnes and Lumley (2011) were able to jointly process multiple components of airborne gravity-gradient data using a single layer of equivalent sources. To our knowledge, Barnes and Lumley (2011) are the pioneers on processing full-tensor gravity-gradient data jointly. In addition to computational feasibility, Barnes and Lumley's (2011) method reduces low-frequency noise and can also remove the drift in time-domain from the survey data. Those authors stressed that the $G_{zz}-$component calculated through the single estimated equivalent-layer model projected on a grid at a constant elevation by inverting full gravity-gradient data has the low-frequency error reduced by a factor of 2.4 as compared to the inversion of an individual component of the gravity-gradient data.

2.2.5 The reparametrization of the equivalent layer

Oliveira Jr. et al. (2013) reparametrized the whole equivalent-layer model by a piecewise bivariate-polynomial function defined on a set of Q equivalent-source windows. In Oliveira Jr. et al.'s (2013) approach, named polynomial equivalent layer (PEL), the parameter vector within the kth equivalent-source window $\mathbf{p}^k$ can be written in matrix notation as

$$\mathbf{p}^k = \mathbf{B}^k \, \mathbf{c}^k \, , \qquad k \; = \; 1 \ldots Q \, , \tag{12}$$

where $\mathbf{p}^k$ is an M_w-dimensional vector containing the physical-property distribution within the kth equivalent-source window, $\mathbf{c}^k$ is a P-dimensional vector whose lth element is the lth coefficient of the αth-order polynomial function and $\mathbf{B}^k$ is an $M_w \times P$ matrix containing the first-order derivative of the αth-order polynomial function with respect to one of the P coefficients.

By using a regularized potential-field inversion, Oliveira Jr. et al. (2013) estimates the polynomial coefficients for each equivalent-source window by solving the following linear system

$$\left(\mathbf{B}^\top \, \mathbf{A}^\top \, \mathbf{A} \, \mathbf{B} \, + \, \mu \mathbf{I} \right) \, \mathbf{c}^* \; = \; \mathbf{B}^\top \, \mathbf{A}^\top \, \mathbf{d}^o \, , \tag{13}$$

where μ is a regularizing parameter, $\mathbf{c}^*$ is an estimated H-dimensional vector containing all coefficients describing all polynomial functions within all equivalent-source windows which compose the entire equivalent layer, $\mathbf{I}$ is an identity matrix of order $H (H = P\dot{Q})$ and $\mathbf{B}$ is an $M \times H$ block diagonal matrix such that the main-diagonal blocks are $\mathbf{B}^k$ matrices (equation 12) and all off-diagonal blocks are zero matrices. For ease of the explanation of equation 13, we keep only the zeroth-order Tikhonov regularization and omitting the first-order Tikhonov regularization (Aster et al., 2018) which was also used by Oliveira Jr. et al. (2013).

The main advantage of the PEL is solve H-dimensional system of equations (equation 13), where H totalizes the number of polynomial coefficients composing all equivalent-source windows, requiring a lower computational effort since $H <<< N$. To avoid the storage of matrices $\mathbf{A}$ and $\mathbf{B}$, Oliveira Jr. et al. (2013) evaluate an element of the matrix $\mathbf{AB}$ by calculating the dot product between the row of matrix $\mathbf{A}$ and the column of the matrix $\mathbf{B}$. After estimating all polynomial coefficients of all windows, the estimated coefficients ($\mathbf{c}^*$ in equation 13) are transformed into a single physical-property distribution encompassing the entire equivalent layer.

As stated by Oliveira Jr. et al. (2013), the computational efficiency of PEL approach stems from the fact that the total number of polynomial coefficients H required to depict the physical-property distribution within the equivalent layer is generally much smaller than the number of equivalent sources. Consequently, this leads to a considerably smaller linear system that needs to be solved. Hence, the main strategy of polynomial equivalent layer is the model dimension reduction.

The polynomial equivalent layer was applied to perform upward continuations of gravity and magnetic data and reduction to the pole of magnetic data.

2.2.6 The iterative scheme without solving a linear system

There exists a class of methods that iteratively estimate the distribution of physical properties within an equivalent layer without the need to solve linear systems. The method initially introduced by Cordell (1992) and later expanded upon by Guspí and Novara (2009) updates the physical property of sources, located beneath each potential-field data, by removing the maximum residual between the observed and fitted data. In addition, Xia and Sprowl (1991) and Xia et al. (1993) have developed efficient iterative algorithms for updating the distribution of physical properties within the equivalent layer in the wavenumber and space domains, respectively. Specifically, in Xia and Sprowl's (1991) method the physical-property distribution is updated by using the ratio between the squared depth to the equivalent source and the gravitational constant multiplied by the residual between the observed and predicted observation at the measurement station. Neither of these methods solve linear systems.

Following this class of methods of iterative equivalent-layer technique that does not solve linear systems, Siqueira et al. (2017) developed a fast iterative equivalent-layer technique for processing gravity data in which the sensitivity matrix $\mathbf{A}$ (equation 1) is replaced by a diagonal matrix $N \times N$, i.e.:

$$\tilde{\tilde{\mathbf{A}}} = 2\,\pi\,\gamma\,\mathbf{\Delta S^{-1}}\,, \tag{14}$$

where γ is Newton's gravitational constant and $\mathbf{\Delta S^{-1}}$ is a diagonal matrix of order N whose diagonal elements Δs_i, $i = 1, ..., N$ are the element of area centered at the ith horizontal coordinates of the ith observation point. The physical foundations of Siqueira et al.'s (2017) method rely on two constraints: i) the excess of mass; and ii) the positive correlation between the gravity observations and the mass distribution over the equivalent layer.

Although Siqueira et al.'s (2017) method does not solve any linear system of equations, it can be theoretically explained by solving the following linear system at the kth iteration:

$$\tilde{\tilde{\mathbf{A}}}^{\top}\,\tilde{\tilde{\mathbf{A}}}\mathbf{\Delta}\,\hat{\mathbf{p}}^k \;=\; \tilde{\tilde{\mathbf{A}}}^{\top}\,\mathbf{r}^k\,, \tag{15}$$

where $\mathbf{r}^k$ is an N-dimensional residual vector whose ith element is calculated by subtracting the ith observed data d_i^o from the ith fitted data d_i^k at the kth iteration, i.e.,

$$r_i^k = d_i^o \; - \; d_i^k \; . \tag{16}$$

and $\boldsymbol{\Delta}\,\hat{\mathbf{p}}^k$ is an estimated N-dimensional vector of parameter correction.

Because $\tilde{\tilde{\mathbf{A}}}$, in equation 15, is a diagonal matrix (equation 14), the parameter correction estimate is directly calculated without solving system of linear equations, and thus, an ith element of $\boldsymbol{\Delta}\,\hat{\mathbf{p}}^k$ is directly calculated by

$$\Delta\hat{p}^k{}_i = \frac{\Delta s_i \, r_i^k}{2\,\pi\,\gamma} \; . \tag{17}$$

The mass distribution over the equivalent layer is updated by:

$$\hat{p}_i^{k+1} = \hat{p}_i^k \; + \; \Delta\hat{p}^k{}_i \; . \tag{18}$$

Siqueira et al.'s (2017) method starts from a mass distribution on the equivalent layer, whose ith mass p_i^o is proportional to the ith observed data d_i^o, i.e.,

$$p_i^o = \frac{\Delta s_i \, d_i^o}{2\,\pi\,\gamma} \; . \tag{19}$$

Siqueira et al. (2017) applied their fast iterative equivalent-layer technique to interpolate, calculate the horizontal components, and continue upward (or downward) gravity data.

For jointly process two gravity gradient components, Jirigalatu and Ebbing (2019) used the Gauss-FFT for forward calculation of potential fields in the wavenumber domain combined with Landweber's iteration coupled with a mask matrix $\mathbf{M}$ to reduce the edge effects without increasing the computation cost. The mask matrix $\mathbf{M}$ is defined in the following way: if the corresponding pixel does not contain the original data, the element of $\mathbf{M}$ is set to zero; otherwise, it is set to one. The kth Landweber iteration is given by

$$\mathbf{p}_{k+1} = \mathbf{p}_k + \omega \left[\mathbf{A_1}^\top (\mathbf{d_1} - \mathbf{M}\mathbf{A_1}\mathbf{p}_k) + \mathbf{A_2}^\top (\mathbf{d_2} - \mathbf{M}\mathbf{A_2}\mathbf{p}_k) \right] \; , \tag{20}$$

where ω is a relaxation factor, $\mathbf{d_1}$ and $\mathbf{d_2}$ are the two gravity gradient components and $\mathbf{A_1}$ and $\mathbf{A_2}$ are the corresponding gravity gradient kernels. Jirigalatu and Ebbing (2019) applied their method for processing two horizontal curvature components of Falcon airborne gravity gradient.

2.2.7 The convolutional equivalent layer with BTTB matrices

Takahashi et al. (2020, 2022) introduced the convolutional equivalent layer for gravimetric and magnetic data processing, respectively.

Takahashi et al. (2020) demonstrated that the sensitivity matrix $\mathbf{A}$ (equation 1) associated with a planar equivalent layer formed by a set of point masses, each one directly beneath each observation point and considering a regular grid of observation points at a constant height has a symmetric block-Toeplitz Toeplitz-block (BTTB) structure. A symmetric BTTB matrix has, at least, two attractive properties. The first one is that it can be defined by using only the elements forming its first column (or row). The second attractive property is that any BTTB matrix can be embedded into a symmetric Block-Circulat Circulant-Block

(BCCB) matrix. This means that the full sensitivity matrix $\mathbf{A}$ (equation 1) can be completely reconstruct by using the first column of the BCCB matrix only. In what follows, Takahashi et al. (2020) computed the forward modeling by using only a single equivalent source. Specifically, it is done by calculating the eigenvalues of the BCCB matrix that can be efficiently computed by using only the first column of the BCCB matrix via 2D fast Fourier transform (2D FFT). By comparing with the classic approach in the Fourier domain, the convolutional equivalent layer for gravimetric data processing proposed by Takahashi et al. (2020) performed upward- and downward-continue gravity data with a very small border effects and noise amplification.

By using the original idea of the convolutional equivalent layer proposed by Takahashi et al. (2020) for gravimetric data processing, Takahashi et al. (2022) developed the convolutional equivalent layer for magnetic data processing. By assuming a regularly spaced grid of magnetic data at a constant height and a planar equivalent layer of dipoles, Takahashi et al. (2022) proved that the sensitivity matrix linked with this layer possess a BTTB structure in the specific scenario where each dipole is exactly beneath each observed magnetic data point. Takahashi et al. (2022) used a conjugate gradient least-squares (CGLS) algorithm which does not require an inverse matrix or matrix-matrix multiplication. Rather, it only requires matrix-vector multiplications per iteration, which can be effectively computed using the 2D FFT as a discrete convolution. The matrix-vector product only uses the elements that constitute the first column of the associated BTTB matrix, resulting in computational time and memory savings. Takahashi et al. (2022) showed the robustness of the convolutional equivalent layer in processing magnetic survey that violates the requirement of regular grids in the horizontal directions and flat observation surfaces.

The matrix-vector product in Takahashi et al. (2020, 2022) (e.g., $\mathbf{d} = \mathbf{A}\mathbf{p}$, such as in equation 1) is the main issue to be solved. To solve it efficiently, these authors involked the auxiliary linear system

$$\mathbf{w} = \mathbf{C}\mathbf{v} \, , \tag{21}$$

where $\mathbf{w}$ and $\mathbf{v}$ are, respectively, vectors of data and parameters completed by zeros and $\mathbf{C}$ is a BCCB matrix formed by $2Q \times 2Q$ blocks, where each block $\mathbf{C}_q$, $q = 0, \ldots, Q-1$, is a $2P \times 2P$ circulant matrix. The first column of $\mathbf{C}$ is obtained by rearranging the first column of the sensitivity matrix $\mathbf{A}$ (equation 1). Because a BCCB matrix is diagonalized by the 2D unitary discrete Fourier transform (DFT), $\mathbf{C}$ can be written as

$$\mathbf{C} = \left(\mathbf{F}_{2Q} \otimes \mathbf{F}_{2P}\right)^{*} \boldsymbol{\Lambda} \left(\mathbf{F}_{2Q} \otimes \mathbf{F}_{2P}\right) \, , \tag{22}$$

where the symbol "$\otimes$" denotes the Kronecker product (**?**), $\mathbf{F}_{2Q}$ and $\mathbf{F}_{2P}$ are the $2Q \times 2Q$ and $2P \times 2P$ unitary DFT matrices (**?**, p. 31), respectively, the superscritpt "$*$" denotes the complex conjugate and $\boldsymbol{\Lambda}$ is a $4QP \times 4QP$ diagonal matrix containing the eigenvalues of $\mathbf{C}$. Due to the diagonalization of the matrix $\mathbf{C}$, the auxiliary system (equation 21) can be rewritten by using equation 22 and premultiplying both sides of the result by $\left(\mathbf{F}_{2Q} \otimes \mathbf{F}_{2P}\right)$, i.e.,

$$\boldsymbol{\Lambda} \left(\mathbf{F}_{2Q} \otimes \mathbf{F}_{2P}\right) \mathbf{v} = \left(\mathbf{F}_{2Q} \otimes \mathbf{F}_{2P}\right) \mathbf{w} \, . \tag{23}$$

By applying the vec-operator (Takahashi et al., 2020) to both sides of equation 23, by premultiplying both sides of the result by $\mathbf{F}^{*}_{2Q}$ and then postmultiplying both sides of the result by $\mathbf{F}^{*}_{2P}$

$$\mathbf{F}^{*}_{2Q} \left[\mathbf{L} \circ \left(\mathbf{F}_{2Q} \, \mathbf{V} \, \mathbf{F}_{2P}\right)\right] \mathbf{F}^{*}_{2P} = \mathbf{W} \, , \tag{24}$$

where "$\circ$" denotes the Hadamard product (?, p. 298) and $\mathbf{L}$, $\mathbf{V}$ and $\mathbf{W}$ are $2Q \times 2P$ matrices obtained by rearranging, along their rows, the elements forming the diagonal of matrix $\mathbf{\Lambda}$, vector $\mathbf{v}$ and vector $\mathbf{w}$, respectively. The left side of equation 24 contains the 2D Inverse Discrete Fourier Transform (IDFT) of the term in brackets, which in turn represents the Hadamard product of matrix $\mathbf{L}$ and the 2D DFT of matrix $\mathbf{V}$. Matrix $\mathbf{L}$ contains the eigenvalues of $\mathbf{\Lambda}$ (equation 22) and can be efficiently computed by using only the first column of the BCCB matrix $\mathbf{C}$ (equation 21).

Actually, in Takahashi et al. (2020, 2022) a fast 2D discrete circular convolution (?) is used to process very large gravity and magnetic datasets efficiently. The convolutional equivalent layer was applied to perform upward continuation of large magnetic datasets. Compared to the classical Fourier approach, Takahashi et al.'s (2022) method produces smaller border effects without using any padding scheme.

Without taking advantage of the symmetric BTTB structure of the sensitivity matrix (Takahashi et al., 2020) that arises when gravimetric observations are measured on a horizontally regular grid, on a flat surface and considering a regular grid of equivalent sources whithin a horizontal layer, Mendonça (2020) explored the symmetry of the gravity kernel to reduce the number of forward model evaluations. By exploting the symmetries of the gravity kernels and redundancies in the forward model evaluations on a regular grid and combining the subspace solution based on eigenvectors of the gridded dataset, Mendonça (2020) estimated the mass excess or deficiency produced by anomalous sources with positive or negative density contrast.

2.2.8 The deconvolutional equivalent layer with BTTB matrices

To avoid the iterations of the conjugate gradient method in Takahashi et al. (2022), we can employ the deconvolution process. Equation 24 shows that estimate the matrix $\mathbf{V}$, containing the elements of parameter vector $\mathbf{p}$, is a inverse problem that could be solved by deconvolution. From equation 24, the matrix $\mathbf{V}$ can be obtain by deconvolution, i.e.

$$\mathbf{V} \ = \ \mathbf{F}_{2Q}^{*} \left[\frac{\left(\mathbf{F}_{2Q} \mathbf{W} \, \mathbf{F}_{2P}\right)}{\mathbf{L}} \right] \mathbf{F}_{2P}^{*} \, . \tag{25}$$

Equation 25 shows that the parameter vector (in matrix $\mathbf{V}$) can be theoretically obtain by dividing each potential-field observations (in matrix $\mathbf{W}$) by each eigenvalues (in matrix $\mathbf{L}$). Hence, the parameter vector is constructed by element-by-element division of data by eigenvalues.

However, the deconvolution often is extremely unstable. This means that a small change in data can lead to an enormous change in the estimated parameter. Hence, equation 25 requires regularization to be useful. We usede wiener deconvolution to obtain a stable solution, i.e.,

$$\mathbf{V} \ = \ \mathbf{F}_{2Q}^{*} \left[\left(\mathbf{F}_{2Q} \mathbf{W} \, \mathbf{F}_{2P}\right) \frac{\mathbf{L}^{*}}{\left(\mathbf{L} \, \mathbf{L}^{*} + \mu\right)} \right] \mathbf{F}_{2P}^{*} \, , \tag{26}$$

where the matrix $\mathbf{L}^{*}$ contains the complex conjugate eigenvalues and μ is a parameter that controls the degree of stabilization.

2.3 Solution stability

The solution stability of the equivalent-layer methods is rarely addressed. Here, we follow the numerical stability analysis presented in Siqueira et al. (2017).

Let us assume noise-free potential-field data $\mathbf{d}$, we estimate a physical-property distribution $\mathbf{p}$ (estimated solution) within the equivalent layer. Then, the noise-free data $\mathbf{d}$ are contaminated with additive D different sequences of pseudorandom Gaussian noise, creating different noise-corrupted potential-field data $\mathbf{d}^{\mathbf{o}}_{\ell}$, $\ell = 1, ..., D$. From each $\mathbf{d}^{\mathbf{o}}_{\ell}$, we estimate a physical-property distribution $\hat{\mathbf{p}}_{\ell}$ within the equivalent layer.

Next, for each noise-corrupted data $\mathbf{d}^{\mathbf{o}}_{\ell}$ and estimated solution $\hat{\mathbf{p}}_{\ell}$, the ℓth model perturbation δp_{ℓ} and the ℓth data perturbation δd_{ℓ} are, respectively, evaluated by

$$\delta p_{\ell} = \frac{\| \hat{\mathbf{p}}_{\ell} - \mathbf{p} \|_2}{\| \mathbf{p} \|_2}, \quad \ell = 1, ..., D, \tag{27}$$

and

$$\delta d_{\ell} = \frac{\| \mathbf{d}^{\mathbf{o}}_{\ell} - \mathbf{d} \|_2}{\| \mathbf{d} \|_2}, \quad \ell = 1, ..., D. \tag{28}$$

Regardless of the particular method used, the following inequality (Aster et al., 2018, p. 66) is applicable:

$$\delta p_{\ell} \leq \kappa \, \delta d_{\ell}, \quad \ell = 1, ..., D, \tag{29}$$

where κ is the constant of proportionality between the model perturbation δp_{ℓ} (equation 27) and the data perturbation δd_{ℓ} (equation 28). The constant κ acts as the condition number of an invertible matrix in a given inversion, and thus measures the instability of the solution. The larger (smaller) the value of κ the more unstable (stable) is the estimated solution.

Equation 29 shows a linear relationship between the model perturbation and the data perturbation. By plotting δp_{ℓ} (equation 27) against δd_{ℓ} (equation 28) produced by a set of D estimated solution obtained by applying a given equivalent-layer method, we obtain a straight line behaviour described by equation 29. By applying a linear regression, we obtain a fitted straight line whose estimated slope (κ in equation 29) quantifies the solution stability.

Here, the analysis of solution stability is numerically conducted by applying the classical equivalent-layer technique with zeroth-order Tikhonov regularization, the convolutional method for gravimetric and magnetic data, the deconvolutional method (equation 25) and the deconvolutional method with different values for the Wiener stabilization (equation 26).

3 NUMERICAL SIMULATIONS

We investigated different computational algorithms for inverting gravity disturbances and total-field anomalies. To test the capability of the fast equivalent-layer technique for processing that potential field data we measure of the computational effort by counting the number of floating-point operations (*flops*), such as additions, subtractions, multiplications, and divisions (Golub and Loan, 2013) for different number of observation points, ranging from $10,000$ up to $1,000,000$. The results generated when using iterative methods are set to $it = 50$ for the number of iterations.

3.1 Floating-point operations calculation

To measure the computational effort of the different algorithms to solve the equivalent layer linear system, a non-hardware dependent method can be useful because allow us to do direct comparison between them. Counting the floating-point operations (*flops*), i.e., additions, subtractions, multiplications and divisions is a good way to quantify the amount of work of a given algorithm (Golub and Loan, 2013). For example, the number of *flops* necessary to multiply two vectors $\mathbb{R}^N$ is $2N$. A common matrix-vector multiplication with dimension $\mathbb{R}^{N \times N}$ and $\mathbb{R}^N$, respectively, is $2N^2$ and a multiplication of two matrices $\mathbb{R}^{N \times N}$ is $2N^3$. Figure 1 shows the total flops count for the different methods presented in this review with a crescent number of data, ranging from $10,000$ to $1,000,000$ for the gravity equivalent layer and figure 2 for magnetic data.

3.1.1 Normal equations using Cholesky decomposition

The equivalent sources can be estimated directly from solving the normal equations **??**. In this work we will use the Cholesky decompositions method to calculate the necessary *flops*. In this method it is calculated the lower triangule of $\mathbf{A}^T\mathbf{A}$ $(1/2N^3)$, the Cholesky factor $(1/3N^3)$, a matrix-vector multiplication $(2N^2)$ and finally solving the triangular system $(2N^2)$, totalizing

$$f_{classical} = \frac{5}{6}N^3 + 4N^2 \tag{30}$$

3.1.2 Window method (Leão and Silva, 1989)

The moving data-window scheme (Leão and Silva, 1989) solve N linear systems with much smaller sizes (equation 3b). For our results we are considering a data-window of the same size of wich the authors presented in theirs work ($N_w = 49$) and the same number of equivalent sources ($M_w = 225$). We are doing this process for all the other techniques to standardize the resolution of our problem. Using the Cholesky decomposition with this method the *flops* are

$$f_{window} = N\frac{5}{6}M_w N_w^2 + 4N_w M_w \tag{31}$$

3.1.3 PEL method (Oliveira Jr. et al., 2013)

The polynomial equivalent layer uses a simliar approach od moving windows from Leão and Silva (1989). For this operations calculation (equation 13) we used a first degree polynomial (two variables) and each window contains $N_s = 1,000$ observed data and $M_s = 1,000$ equivalent sources. Following the steps given in (Oliveira Jr. et al., 2013) the total *flops* becomes

$$f_{pel} = \frac{1}{3}H^3 + 2H^2 + 2NM_sH + H^2N + 2HN + 2NP \tag{32}$$

where H is the number of constant coefficients for the first degree polynomial ($P = 3$) times the number of windows ($P \times N/N_s$).

3.1.4 Conjugate gradient least square (CGLS)

The CGLS method is a very stable and fast algorithm for solving linear systems iteratively. Its computational complexity envolves a matrix-vector product outside the loop ($2N^2$), two matrix-vector products inside the loop ($4N^2$) and six vector products inside the loop ($12N$) (Aster et al., 2018)

$$f_{cgls} = 2N^2 + it(4N^2 + 12N) \tag{33}$$

3.1.5 Wavelet compression method with CGLS (Li and Oldenburg, 2010)

For the wavelet method (equation 6) we have calculated a coompression rate of 98% ($C_r = 0.02$) for the threshold as the authors used in Li and Oldenburg (2010) and the wavelet transformation requiring $\log_2(N)$ *flops* each (equations 5 and 7c), with its inverse also using the same number of operations (equation 9). Combined with the conjugate gradient least square necessary steps and iterations, the number of *flops* are

$$f_{wavelet} = 2NC_r + 4N\log_2(N) + it(4N\log_2(N) + 4NC_r + 12C_r) \tag{34}$$

3.1.6 Fast equivalent layer for gravity data (Siqueira et al., 2017)

The fast equivalent layer from Siqueira et al. (2017) solves the linear system in *it* iterations. The main cost of this method (equations 15,16, 17 and 18)is the matrix-vector multiplication to asses the predicted data ($2N^2$) and three simply element by element vector sum, subtraction and division ($3N$ total)

$$f_{siqueira} = it(3N + 2N^2) \tag{35}$$

3.1.7 Convolutional equivalent layer for gravity data (Takahashi et al., 2020)

This methods replaces the matrix-vector multiplication of the iterative fast-equivalent technique (Siqueira et al., 2017) by three steps, involving a Fourier transform, an inverse Fourier transform, and a Hadamard product of matrices (equation 24). Considering that the first column of our BCCB matrix has $4N$ elements, the flops count of this method is

$$f_{convgrav} = \kappa 4N\log_2(4N) + it(27N + \kappa 8N\log_2(4N)) \tag{36}$$

In the resultant count we considered a *radix-2* algorithm for the fast Fourier transform and its inverse, which has a κ equals to 5 and requires $\kappa 4N\log_2(4N)$ flops each. The Hadarmard product of two matrices of $4N$ elements with complex numbers takes $24N$ flops. Note that equation 36 is different from the one presented in Takahashi et al. (2020) because we also added the flops necessary to calculate the eigenvalues in this form. It does not differentiate much in order of magnitude because the iterative part is the most costful.

3.1.8 Convolutional equivalent layer for magnetic data (Takahashi et al., 2022)

The convolutional equivalent layer for magnetic data uses the same flops count of the main operations as in the gravimetric case (equation 24), the difference is the use of the conjugate gradient algorithm to solve the inverse problem. It requires a Hadamard product outside of the iterative loop and the matrix-vector and vector-vector multiplications inside the loop as seem in equation 33.

$$f_{convmag} = \kappa 16N \log_2(4N) + 24N + it(\kappa 16N \log_2(4N) + 60N) \tag{37}$$

3.1.9 Deconvolutional method

The deconvolution method does not require an iterative algorithm, rather it solves the estimative of the physical properties in a single step using the $4N$ eigenvalues of the BCCB matrix as in the convolutional method. From equation 25 it is possible to deduce this method requires two fast Fourier transform ($\kappa 4N \log_2(4N)$), one for the eigenvalues and another for the data transformation, a element by element division ($24N$) and finally, a fast inverse Fourier transform for the final estimative ($\kappa 4N \log_2(4N)$).

$$f_{deconv} = \kappa 12N \log_2(4N) + 24N \tag{38}$$

Using the deconvolutional method with a Wiener stabilization adds two multiplications of complex elements of the conjugates eigenvalues ($24N$ each) and the sum of $4N$ elements with the stabilization parameter μ as shown in equation 26

$$f_{deconvwiener} = \kappa 12N \log_2(4N) + 76N \tag{39}$$

4 SYNTHETIC DATA SIMULATIONS

For all applications, we generate a model composed by two spheres and a poligonal prism in a regular spaced grid of 50×50. The upper left sphere has a density contrast of $600\,kg/m^3$, the right upper sphere a negative contrast of $-500\,kg/m^3$ and the bottom prism is equal to $550\,kg/m^3$. To generate the magnetic data, the bodies are in the same position and all of them have the same magnetization intensity and direction ($3.46\,A/m$ intensity, $35.26°$ inclination and $45.0°$ declination) within a simulated geomagnetic field direction of $20.0°$ inclination and $35.0°$ declination. These synthetic datas are shown in figures 4 and 7, respectively.

4.1 Stability analysis

For the stability analysis we show the comparison of the normal equations solution (equation 2) with zeroth-order Tikhonov regularization (Aster et al., 2018), the convolutional method for gravimetric and magnetic data (equation 24), the deconvolutional method (equation 25) and the deconvolutional method with different values for the Wiener stabilization (equation 26). We create 21 data sets adding a crescent pseudo-random noise to the original data, which varies from 0% to 10% of the maximum anomaly value, in intervals of 0.5%. These noises has mean equal to zero and a Gaussian distribution.

Figure 3 shows how the residual between the predicted data and the noise-free data changes as the level of the noise is increased for the gravimetric data. We can see that for all methods, a linear tendency can be observed as it is expected. The inclination of the straight line is a indicative of the stability of each method. As show in the graph the deconvolutional method is very unstable and it is really necessary to use a stabilization method to have a good parameter estimative. In contrast, a correct value of the stabilization parameter is necessary to not overshoot the smootheness of the solution as it is the case for the well-known zeroth-order Tikhonov regularization as well. Using this gravimetric data, the optimal value for the Wiener stabilization parameter is $\mu = 10^{-20}$.

Figure 5 shows the comparison of the predicted data for each method with the original data (figure 4) using the most noised-corrupted data from the set of the stability analysis. The classical with zeroth-order Tikhonov regularization and the convolutional methods (figures 5a and 5b) yield very similar results for the predicted data confirming its similarities with the stabilization despite the bid difference in floating-point operations. Figure 5c shows the deconvolutional method without a stabilization and demonstrates the necessity to use it for this method. Figure 5d shows the deconvolutional method with Wiener stabilization $\mu = 10^{-15}$ which is too high, demonstrating the over smoothness of the predicted data. Figures 5e and 5f shows the predicted data for an optimal value of the Wiener parameter $\mu = 10^{-20}$ and a low value $\mu = 10^{-25}$, respectively.

For the magnetic data, figure 6 shows a very similar behavior of the stability as the previous case. The Wiener parameter seems to have the best solution for $\mu = 10^{-13}$. For both types of data the best Wiener parameter seems to be one that produces a low slope for the straight line in the stability analysis, discordant from the classical and convolutional methods.

Figure 8 shows the comparison of the predicted data for each method with the original magnetic data in figure 7 using the most noised-corrupted data modeled from the stability analyis. As the previous case the classical (figure 8a) and the convolutional (figure 8b) methods have very similar predicted data but estimated with less orders of magnitude in floating-point operations. The deconvoutional (figure 8c) have have a strong disagreement with the observed data showing the nedd for a stabilization method. Figure 8d has a value of $\mu = 10^{-10}$ and the predicted data became to smooth by it. The optimal value of the

Wiener parameter is shown in figure 8e with $\mu = 10^{-13}$ and figure 8f shows a predicted data with a low stablization value with $\mu = 10^{-16}$.

5 REAL DATA APPLICATION

Gridded data of 1000x500 (500000 observed points) for both grav and mag. Data is at -900m.

Grav equivalent layer depth is 300 m and 50 iterations of the cgls method was used. Mag equivalent layer depth is 0 m and 200 iterations of the cgls method was used.

On an Intel Core i7 7700HQ@2.8 GHz processor in single processing and single-threading modes the gravimetric equivalent layer took 9.19 seconds to estimate the equivalent sources with the convolutional method and 0.51 seconds with the deconvolutional method.

The magnetic equivalent layer took 82 seconds to estimate the equivalent sources with the convolutional method and 0.84 seconds with the deconvolutional method.

As Carajás area is very large different values of the magnetic main field can be considered. The main field declination was calculated using the tool in the website (for the date 01/01/2014): https://www.ngdc.noaa.gov/geomag/calculators/magcalc.shtml For this application I considered an approximated mid location of the area (latitude $-6.55°$ and longitude $-50.75°$). The declination is $-19.865°$ and the inclination $-7.43915°$. As the source magnetization is unknown inclination and declination equal to the main field is being used for all the equivalent sources.

Gravimetric case:

Means

0.0005096975472675431 (convolutional method)

0.4582999511463665 (deconvolutional method with wiener $\mu = 10^{-22}$)

Standart deviations

0.15492798729938298 (convolutional method)

1.229507199000529 (deconvolutional method with wiener $\mu = 10^{-22}$)

Magnetic case:

Means

-0.06404347121632468 (convolutional method)

18.992921718679344 (deconvolutional method with wiener $\mu = 10^{-16}$)

Standart deviations

1.9687559764381535 (convolutional method)

33.641199020925924 (deconvolutional method with wiener $\mu = 10^{-16}$)

6 DISCUSSION AND CONCLUSION

CONFLICT OF INTEREST STATEMENT

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

AUTHOR CONTRIBUTIONS

The Author Contributions section is mandatory for all articles, including articles by sole authors. If an appropriate statement is not provided on submission, a standard one will be inserted during the production process. The Author Contributions statement must describe the contributions of individual authors referred to by their initials and, in doing so, all authors agree to be accountable for the content of the work. Please see here for full authorship criteria.

FUNDING

Diego Takahashi was supported by a Post-doctoral scholarship from CNPq (grant 300809/2022-0) Valéria C.F. Barbosa was supported by fellowships from CNPq (grant 309624/2021-5) and FAPERJ (grant 26/202.582/2019). Vanderlei C. Oliveira Jr. was supported by fellowships from CNPq (grant 315768/2020-7) and FAPERJ (grant E-26/202.729/2018).

ACKNOWLEDGMENTS

We thank the brazillian federal agencies CAPES, CNPq, state agency FAPERJ and Observatório Nacional research institute and Universidade do Estado do Rio de Janeiro.

DATA AVAILABILITY STATEMENT

The datasets generated for this study can be found in the frontiers-paper Github repository link: https://github.com/DiegoTaka/frontiers-paper.

REFERENCES

Aster, R. C., Borchers, B., and Thurber, C. H. (2018). *Parameter estimation and inverse problems* (Elsevier)

Barbosa, V. C. F., Silva, J. B., and Medeiros, W. E. (1997). Gravity inversion of basement relief using approximate equality constraints on depths. *Geophysics* 62, 1745–1757

Barnes, G. and Lumley, J. (2011). Processing gravity gradient data. *Geophysics* 76, I33–I47. doi:10.1190/1.3548548

Cordell, L. (1992). A scattered equivalent-source method for interpolation and gridding of potential-field data in three dimensions. *Geophysics* 57, 629–636

Dampney, C. N. G. (1969). The equivalent source technique. *Geophysics* 34, 39–53. doi:10.1190/1.1439996

Davis, K. and Li, Y. (2011). Fast solution of geophysical inversion using adaptive mesh, space-filling curves and wavelet compression. *Geophysical Journal International* 185, 157–166. doi:10.1111/j.1365-246X.2011.04929.x

Friedman, J. H. (2001). Greedy function approximation: a gradient boosting machine. *Annals of statistics* , 1189–1232

Friedman, J. H. (2002). Stochastic gradient boosting. *Computational statistics and data analysis* 38, 367–378

Golub, G. H. and Loan, C. F. V. (2013). *Matrix Computations (Johns Hopkins Studies in the Mathematical Sciences)* (Johns Hopkins University Press), 4 edn.

Guspí, F. and Novara, I. (2009). Reduction to the pole and transformations of scattered magnetic data using newtonian equivalent sources. *Geophysics* 74, L67–L73

Jirigalatu, J. and Ebbing (2019). A fast equivalent source method for airborne gravity gradient data. *Geophysics* 84, G75–G82. doi:10.1190/GEO2018-0366.1

Kellogg, O. D. (1929). *Foundations of Potential Theory* (Frederick Ungar Publishing Company)

Kennett, B., Sambridge, M., and Williamson, P. (1988). Subspace methods for large inverse problems with multiple parameter classes. *Geophysical Journal International* 94, 237–247

Leão, J. W. D. and Silva, J. B. C. (1989). Discrete linear transformations of potential field data. *Geophysics* 54, 497–507. doi:10.1190/1.1442676

Li, Y. and Oldenburg, D. W. (2010). Rapid construction of equivalent sources using wavelets. *Geophysics* 75, L51–L59. doi:10.1190/1.3378764

Mendonça, C. A. (2020). Subspace method for solving large-scale equivalent layer and density mapping problems. *Geophysics* 85, G57–G68. doi:10.1190/geo2019-0302.1

Mendonça, C. A. and Silva, J. B. C. (1994). The equivalent data concept applied to the interpolation of potential field data. *Geophysics* 59, 722–732. doi:10.1190/1.1443630

Oldenburg, D., McGillivray, P., and Ellis, R. (1993). Generalized subspace methods for large-scale inverse problems. *Geophysical Journal International* 114, 12–20

Oliveira Jr., V. C., Barbosa, V. C. F., and Uieda, L. (2013). Polynomial equivalent layer. *Geophysics* 78, G1–G13. doi:10.1190/geo2012-0196.1

Siqueira, F. C., Oliveira Jr, V. C., and Barbosa, V. C. (2017). Fast iterative equivalent-layer technique for gravity data processing: A method grounded on excess mass constraint. *Geophysics* 82, G57–G69

Skilling, J. and Bryan, R. (1984). Maximum entropy image reconstruction-general algorithm. *Monthly Notices of the Royal Astronomical Society, Vol. 211, NO. 1, P. 111, 1984* 211, 111

Soler, S. R. and Uieda, L. (2021). Gradient-boosted equivalent sources. *Geophysical Journal International* 227, 1768–1783. doi:10.1093/gji/ggab297

Takahashi, D., Oliveira, V. C., and Barbosa, V. C. (2022). Convolutional equivalent layer for magnetic data processing. *Geophysics* 87, 1–59

Takahashi, D., Oliveira Jr, V. C., and Barbosa, V. C. (2020). Convolutional equivalent layer for gravity data processing. *Geophysics* 85, G129–G141

Xia, J. and Sprowl, D. R. (1991). Correction of topographic distortion in gravity data. *Geophysics* 56, 537–541

Xia, J., Sprowl, D. R., and Adkins-Heljeson, D. (1993). Correction of topographic distortions in potential-field data; a fast and accurate approach. *Geophysics* 58, 515–523. doi:10.1190/1.1443434

7 SUPPLEMENTARY TABLES AND FIGURES

7.1 Figures

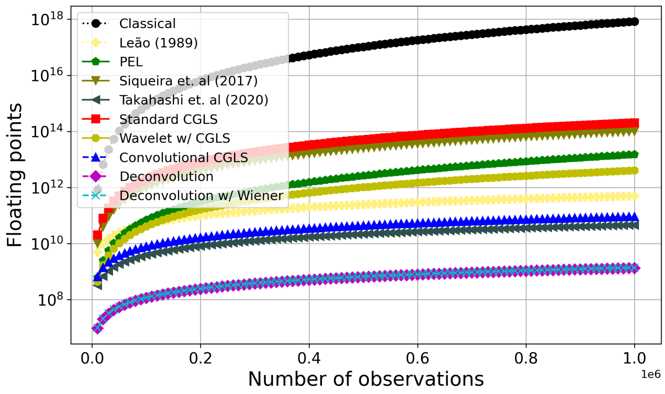

Figure 1. Number of *flops* for many of the methods described in this work to estimate the equivalent sources using gravity data. The range of observations varies from $10,000$ to $1,000,000$.

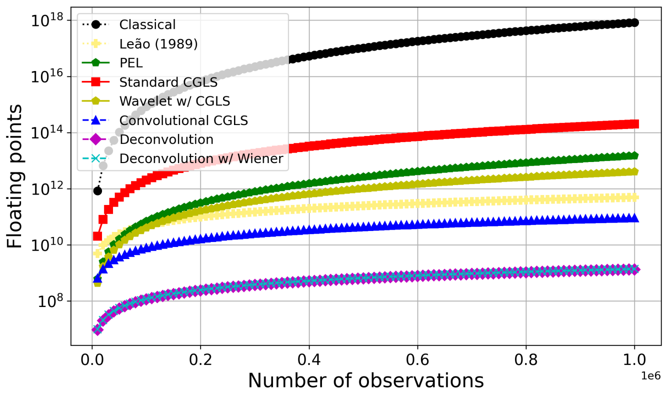

Figure 2. Number of *flops* for many of the methods described in this work to estimate the equivalent sources using magnetic data. The range of observations varies from $10,000$ to $1,000,000$.

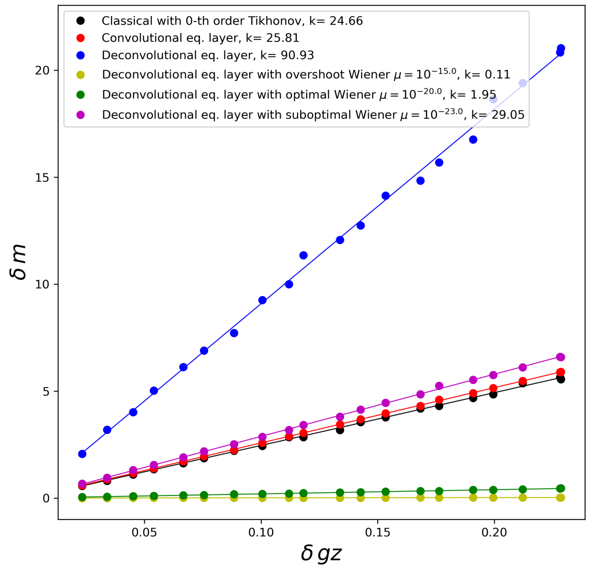

Figure 3. Stability analysis of some of the equivalent layer methods of the gravimetric case.

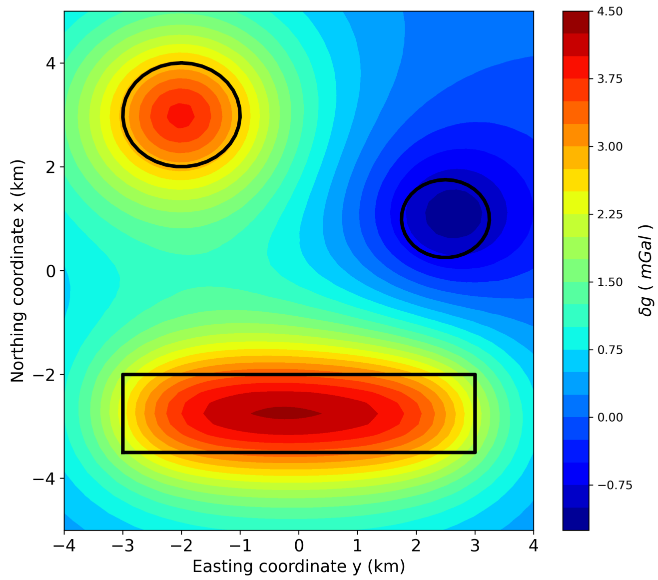

Figure 4. Synthetic noise-free data of the gravimetric case. The observations points and equivalent sources are placed in a regular grid of 50×50.

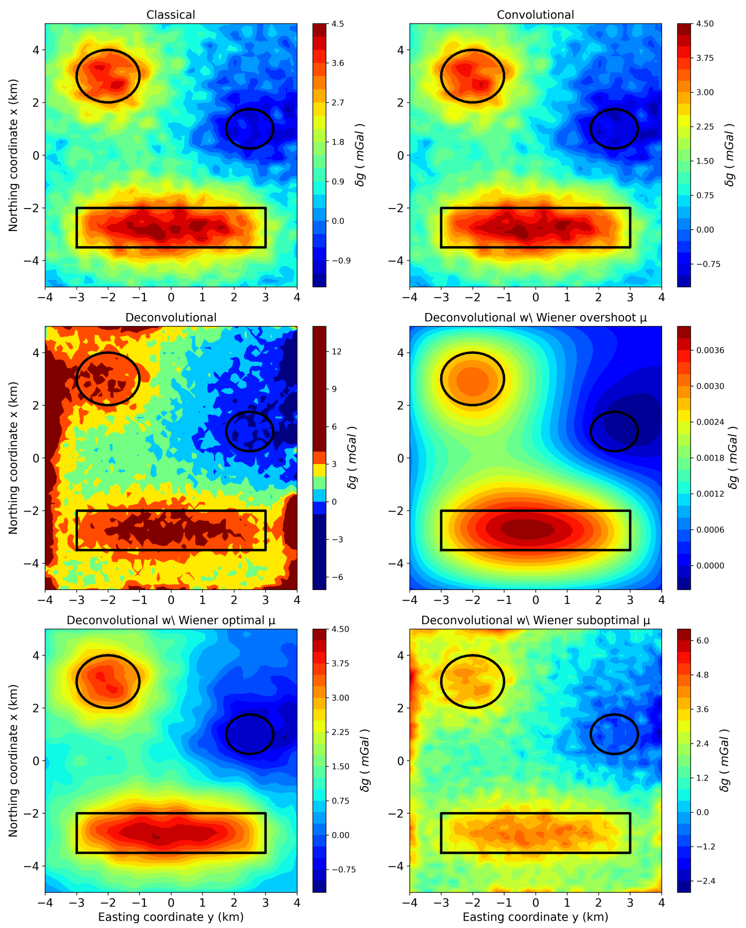

Figure 5. Predicted gravity data for different methods of the equivalent layer with maximum level of noise. Panel **(A)** is the classical method, **(B)** is the convolutional, **(C)** is the deconvolutional, **(D)** is the deconvolutional method using Wiener stabilization with a too high value for μ, **(E)** is the deconvolutional method using Wiener stabilization with a optimal value for μ and **(F)** is the deconvolutional method using Wiener stabilization with a too low value for μ.

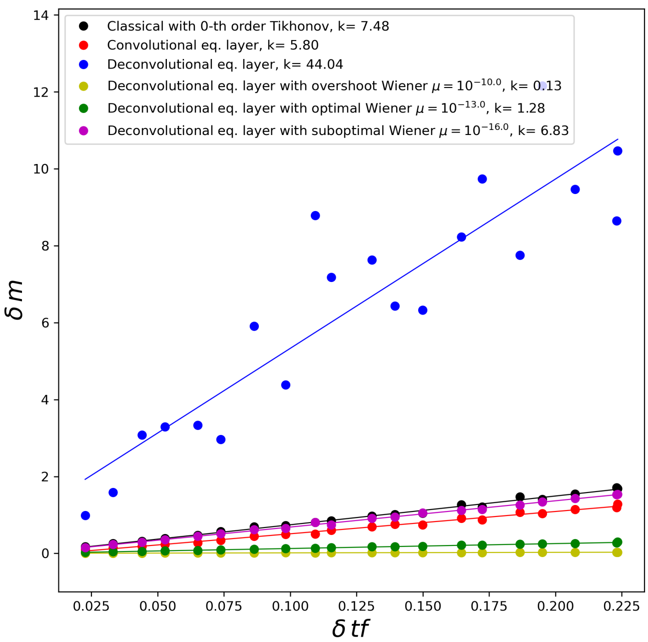

Figure 6. Stability analysis of some of the equivalent layer methods of the magnetic case.

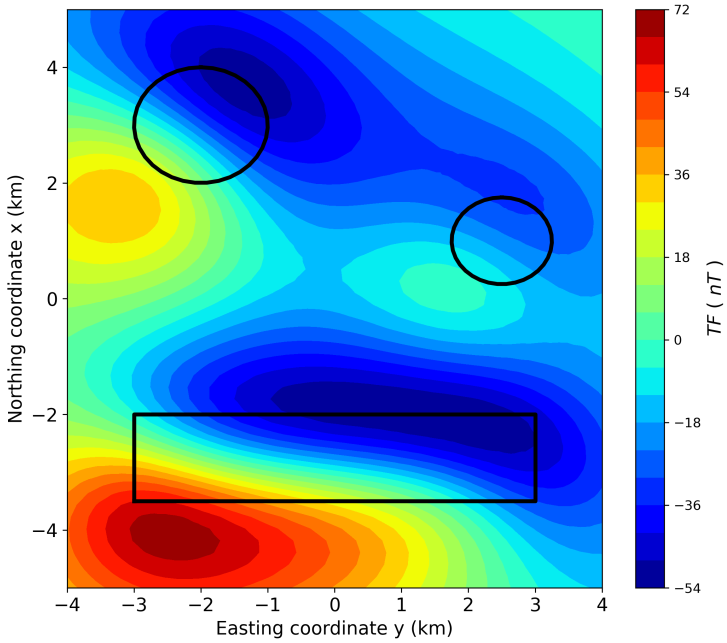

Figure 7. Synthetic noise-free data of the magnetic case. The observations points and equivalent sources are placed in a regular grid of 50×50.

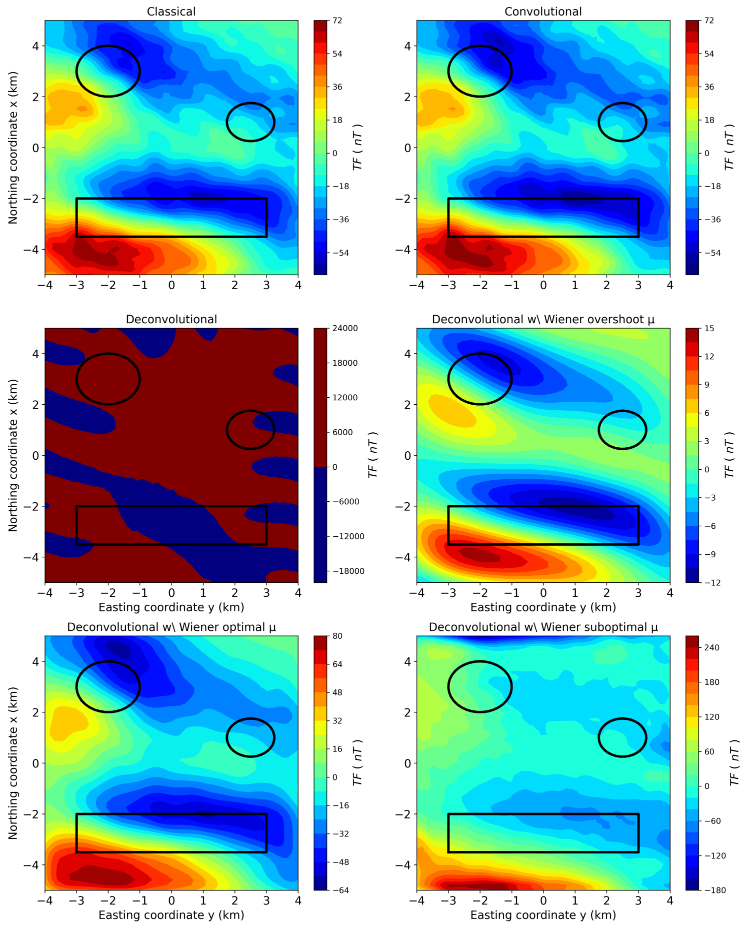

Figure 8. Predicted magnetic data for different methods of the equivalent layer with maximum level of noise. Panel **(A)** is the classical method, **(B)** is the convolutional, **(C)** is the deconvolutional, **(D)** is the deconvolutional method using Wiener stabilization with a too high value for μ, **(E)** is the deconvolutional method using Wiener stabilization with a optimal value for μ and **(F)** is the deconvolutional method using Wiener stabilization with a too low value for μ.

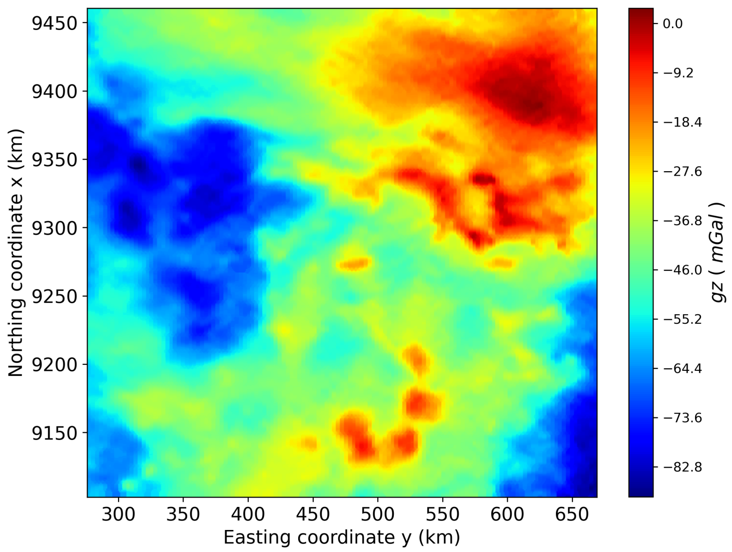

Figure 9. Gridded real aerogravimetric data from Carajás, Brazil. A regular grid of $10,000 \times 500$ is being used, totalizing $N, M = 5,000,000$ obsevation points and equivalent sources.

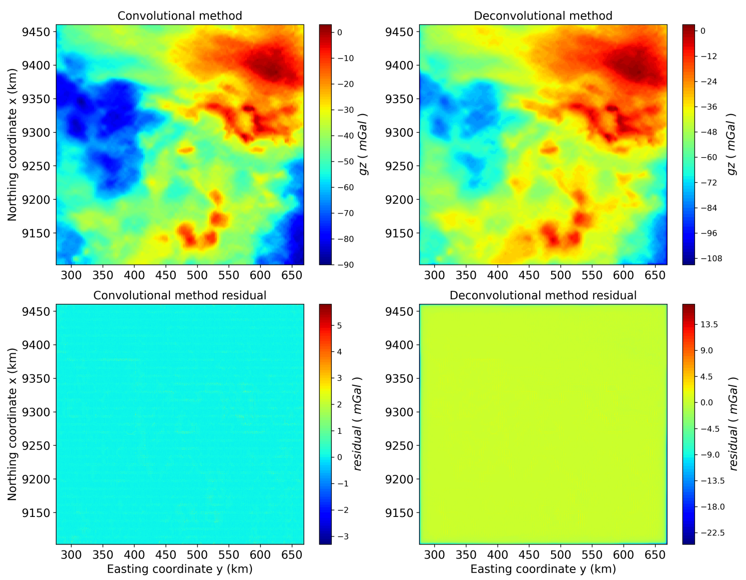

Figure 10. Panel **(A)** shows the predicted data from convolutional equivalent layer method. Panel **(B)** shows the residual from the convolutional equivalent layer method. Panel **(C)** shows the predicted data from deconvolutional equivalent layer method. Panel **(D)** shows the residual from the deconvolutional equivalent layer method.

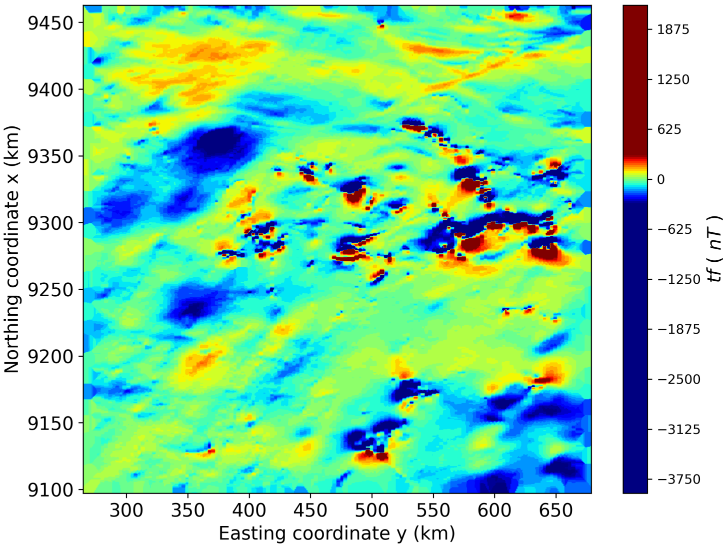

Figure 11. Gridded real aeromagnetic data from Carajás, Brazil. A regular grid of $10,000 \times 500$ is being used, totalizing $N, M = 5,000,000$ obsevation points and equivalent sources.

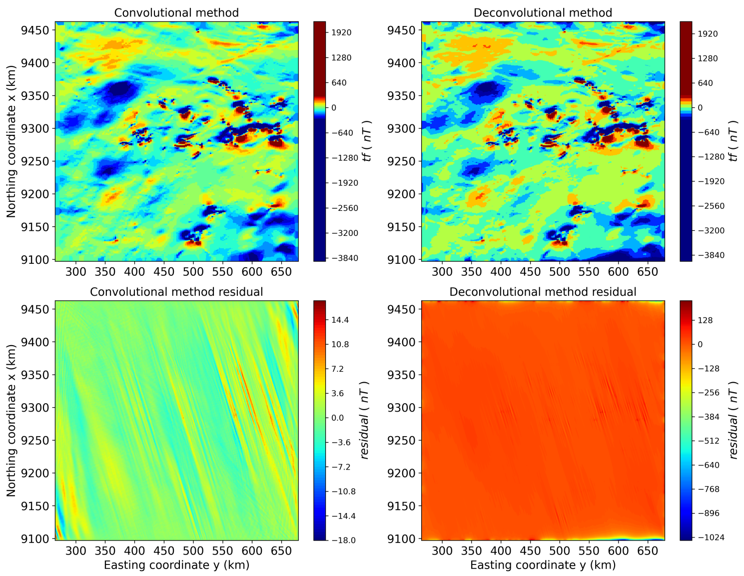

Figure 12. Panel **(A)** shows the predicted data from convolutional equivalent layer method. Panel **(B)** shows the residual from the convolutional equivalent layer method. Panel **(C)** shows the predicted data from deconvolutional equivalent layer method. Panel **(D)** shows the residual from the deconvolutional equivalent layer method.