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

1 FUNDAMENTALS

2 Let $\mathbf{d}$ be a $D \times 1$ vector, whose i-th element d_i is the observed potential field at the position (x_i, y_i, z_i),
3 $i \in \{1 : D\}$, of a topocentric Cartesian system with x, y and z axes pointing to north, east and down,
4 respectively. Consider that d_i can be satisfactorily approximated by a harmonic function

$$f_i = \sum_{j=1}^{P} g_{ij}\, p_j\,, \quad i \in \{1 : D\}\,, \tag{1}$$

5 where, p_j represents the scalar physical property of a virtual source (i.e., monopole, dipole, prism) located
6 at (x_j, y_j, z_j), $j \in \{1 : P\}$ and

$$g_{ij} \equiv g(x_i - x_j, y_i - y_j, z_i - z_j)\,, \quad z_i < \min\{z_j\}\,, \quad \forall i \in \{1 : D\}\,, \tag{2}$$

7 is a harmonic function, where $\min\{z_j\}$ denotes the minimum z_j, or the vertical coordinate of the shallowest
8 virtual source. These virtual sources are called *equivalent sources* and they form an *equivalent layer*. In
9 matrix notation, the potential field produced by all equivalent sources at all points (x_i, y_i, z_i), $i \in \{1 : D\}$,
10 is given by:

$$\mathbf{f} = \mathbf{G}\mathbf{p}\,, \tag{3}$$

11 where $\mathbf{p}$ is a $P \times 1$ vector with j-th element p_j representing the scalar physical property of the j-th
12 equivalent source and $\mathbf{G}$ is a $D \times P$ matrix with element g_{ij} given by equation 2.

13   The equivalent-layer technique consists in solving a linear inverse problem to determine a parameter
14 vector $\mathbf{p}$ leading to a predicted data vector $\mathbf{f}$ (equation 3) *sufficiently close to* the observed data vector $\mathbf{d}$,
15 whose i-th element d_i is the observed potential field at (x_i, y_i, z_i). The notion of *closeness* is intrinsically
16 related to the concept of *vector norm* (e.g., Golub and Van Loan, 2013, p. 68) or *measure of length* (e.g.,
17 Menke, 2018, p. 41). Because of that, almost all methods for determining $\mathbf{p}$ actually estimate a parameter
18 vector $\tilde{\mathbf{p}}$ minimizing a length measure of the difference between $\mathbf{f}$ and $\mathbf{d}$ (see subsection 1.3). Given an
19 estimate $\tilde{\mathbf{p}}$, it is then possible to compute a potential field transformation

$$\mathbf{t} = \mathbf{A}\tilde{\mathbf{p}}\,, \tag{4}$$

20 where $\mathbf{t}$ is a $T \times 1$ vector with k-th element t_k representing the transformed potential field at the position
21 (x_k, y_y, z_k), $k \in \{1 : T\}$, and

$$a_{kj} \equiv a(x_k - x_j, y_k - y_j, z_k - z_j)\,, \quad z_k < \min\{z_j\}\,, \quad \forall k \in \{1 : T\}\,, \tag{5}$$

22 is a harmonic function representing the kj-th element of the $T \times P$ matrix $\mathbf{A}$.

1.1 Spatial distribution and total number of equivalent sources

24   There is no well-established criteria to define the optimum number P or the spatial distribution of the
25 equivalent sources. We know that setting an equivalent layer with more (less) sources than potential-field
26 data usually leads to an underdetermined (overdetermined) inverse problem (e.g., Menke, 2018, p. 52–53).
27 Concerning the spatial distribution of the equivalent sources, the only condition is that they must rely on a
28 surface that is located below and does not cross that containing the potential field data. Soler and Uieda
29 (2021) present a practical discussion about this topic.

30   From a theoretical point of view, the equivalent layer reproducing a given potential field data set cannot
31   cross the true gravity or magnetic sources. This condition is a consequence of recognizing that the equivalent
32   layer is essentially an indirect solution of a boundary value problem of potential theory (e.g., Roy, 1962;
33   Zidarov, 1965; Dampney, 1969; Li et al., 2014; Reis et al., 2020). In practical applications, however, there
34   is no guarantee that this condition is satisfied. Actually, its is widely known from practical experience (e.g.,
35   Gonzalez et al., 2022) that the equivalent-layer technique works even for the case in which the layer cross
36   the true sources.

37   Regarding the depth of the equivalent layer, Dampney (1969) proposed a criterion based on horizontal data
38   sampling, suggesting that the equivalent-layer depth should be between two and six times the horizontal grid
39   spacing, considering evenly spaced data. However, when dealing with a survey pattern that has unevenly
40   spaced data, Reis et al. (2020) adopted an alternative empirical criterion. According to their proposal, the
41   depth of the equivalent layer should range from two to three times the spacing between adjacent flight
42   lines. The criteria of Dampney (1969) and Reis et al. (2020) are valid for planar equivalent layers. Cordell
43   (1992) have proposed and an alternative criterion for scattered data that leads to an undulating equivalent
44   layer. This criterion have been slightly modified by Guspí et al. (2004), Guspí and Novara (2009) and
45   Soler and Uieda (2021), for example, and consists in setting one equivalent source below each datum at a
46   depth proportional to the horizontal distance to the nearest neighboring data points. Soler and Uieda (2021)
47   have compared different strategies for defining the equivalent sources depth for the specific problem of
48   interpolating gravity data, but they have not found significant differences between them.

1.2   **Matrix** G

50   Generally, the harmonic function g_{ij} (equation 2) is defined in terms of the inverse distance between the
51   observation point (x_i, y_i, z_i) and the j-th equivalent source at (x_j, y_j, z_j),

$$\frac{1}{r_{ij}} \equiv \frac{1}{\sqrt{(x_i - x_j)^2 + (y_i - y_j)^2 + (z_i - z_j)^2}} \, , \tag{6}$$

52   or by its partial derivatives of first and second orders, respectively given by

$$\partial_\alpha \frac{1}{r_{ij}} \equiv \frac{-(\alpha_i - \alpha_j)}{r_{ij}^3} \, , \quad \alpha \in \{x, y, z\} \, , \tag{7}$$

53   and

$$\partial_{\alpha\beta} \frac{1}{r_{ij}} \equiv \begin{cases} \frac{3(\alpha_i - \alpha_j)^2}{r_{ij}^5} \, , & \alpha = \beta \, , \\ \frac{3(\alpha_i - \alpha_j)(\beta_i - \beta_j)}{r_{ij}^5} - \frac{1}{r_{ij}^3} \, , & \alpha \neq \beta \, , \end{cases} \quad \alpha, \beta \in \{x, y, z\} \, . \tag{8}$$

54   In this case, the equivalent layer is formed by punctual sources representing monopoles or dipoles (e.g.,
55   Dampney, 1969; Emilia, 1973; Leão and Silva, 1989; Cordell, 1992; Oliveira Jr. et al., 2013; Siqueira et al.,
56   2017; Reis et al., 2020; Takahashi et al., 2020; Soler and Uieda, 2021; Takahashi et al., 2022). Another
57   common approach consists in not defining g_{ij} by using equations 6–8, but other harmonic functions
58   obtained by integrating them over the volume of regular prisms (e.g., Li and Oldenburg, 2010; Barnes and
59   Lumley, 2011; Li et al., 2014; Jirigalatu and Ebbing, 2019). There are also some less common approaches
60   defining the harmonic function g_{ij} (equation 2) as the potential field due to plane faces with constant
61   physical property (Hansen and Miyazaki, 1984), doublets (Silva, 1986) or by computing the double
62   integration of the inverse distance function with respect to z (Guspí and Novara, 2009).

A common assumption for most of the equivalent-layer methods is that the harmonic function g_{ij} (equation 2) is independent on the actual physical relationship between the observed potential field and their true sources (e.g., Cordell, 1992; Guspí and Novara, 2009; Li et al., 2014). Hence, g_{ij} can be defined according to the problem. The only condition imposed to this function is that it decays to zero as the observation point (x_i, y_i, z_i) goes away from the position (x_j, y_j, z_j) of the j-th equivalent source. However, several methods use a function g_{ij} that preserves the physical relationship between the observed potential field and their true sources. For the case in which the observed potential field is gravity data, g_{ij} is commonly defined as a component of the gravitational field produced at (x_i, y_i, z_i) by a point mass or prism located at (x_j, y_j, z_j), with unit density. On the other hand, g_{ij} is commonly defined as a component of the magnetic induction field produced at (x_i, y_i, z_i) by a dipole or prism located at (x_j, y_j, z_j), with unit magnetization intensity, when the observed potential field is magnetic data.

The main challenge in the equivalent-layer technique is the computational complexity associated with handling large datasets. This complexity arises because the sensitivity matrix $\mathbf{G}$ (equation 3) is dense regardless of the harmonic function g_{ij} (equation 2) employed. In the case of scattered potential-field data, the structure of $\mathbf{G}$ is not well-defined, regardless of the spatial distribution of the equivalent sources. However, in a specific scenario where (i) each potential-field datum is directly associated with a single equivalent source located directly below it, and (ii) both the data and sources are based on planar and regularly spaced grids, Takahashi et al. (2020, 2022) demonstrate that $\mathbf{G}$ exhibits a block-Toeplitz Toeplitz-block (BTTB) structure. In such cases, the product of $\mathbf{G}$ and an arbitrary vector can be efficiently computed using a 2D fast Fourier transform as a discrete convolution.

1.3 General formulation

A general formulation for almost all equivalent-layer methods can be achieved by first considering that the $P \times 1$ parameter vector $\mathbf{p}$ (equation 3) can be reparameterized into a $Q \times 1$ vector $\mathbf{q}$ according to:

$$\mathbf{p} = \mathbf{H}\,\mathbf{q}\,, \tag{9}$$

where $\mathbf{H}$ is a $P \times Q$ matrix. The predicted data vector $\mathbf{f}$ (equation 3) can then be rewritten as follows:

$$\mathbf{f} = \mathbf{G}\,\mathbf{H}\,\mathbf{q}\,. \tag{10}$$

Note that the original parameter vector $\mathbf{p}$ is defined in a P-dimensional space whereas the reparameterized parameter vector $\mathbf{q}$ (equation 9) lies in a Q-dimensional space. For convenience, we use the terms P-space and Q-space to designate them.

In this case, the problem of estimating a parameter vector $\tilde{\mathbf{p}}$ minimizing a length measure of the difference between $\mathbf{f}$ (equation 3) and $\mathbf{d}$ is replaced by that of estimating an auxiliary vector $\tilde{\mathbf{q}}$ minimizing the goal function

$$\Gamma(\mathbf{q}) = \Phi(\mathbf{q}) + \mu\,\Theta(\mathbf{q})\,, \tag{11}$$

which is a combination of particular measures of length given by

$$\Phi(\mathbf{q}) = (\mathbf{d} - \mathbf{f})^\top \mathbf{W}_d\,(\mathbf{d} - \mathbf{f})\,, \tag{12}$$

and

$$\Theta(\mathbf{q}) = (\mathbf{q} - \bar{\mathbf{q}})^\top \mathbf{W}_q\,(\mathbf{q} - \bar{\mathbf{q}})\,, \tag{13}$$

where the regularization parameter μ is a positive scalar controlling the trade-off between the data-misfit function $\Phi(\mathbf{q})$ and the regularization function $\Theta(\mathbf{q})$; $\mathbf{W}_d$ is a $D \times D$ symmetric matrix defining the relative importance of each observed datum d_i; $\mathbf{W}_q$ is a $Q \times Q$ symmetric matrix imposing prior information on $\mathbf{q}$; and $\bar{\mathbf{q}}$ is a $Q \times 1$ vector of reference values for $\mathbf{q}$ that satisfies

$$\bar{\mathbf{p}} = \mathbf{H} \, \bar{\mathbf{q}} \,, \tag{14}$$

where $\bar{\mathbf{p}}$ is a $P \times 1$ vector containing reference values for the original parameter vector $\mathbf{p}$.

After obtaining an estimate $\tilde{\mathbf{q}}$ for the reparameterized parameter vector $\mathbf{q}$ (equation 9), the estimate $\tilde{\mathbf{p}}$ for the original parameter vector (equation 3) is computed by

$$\tilde{\mathbf{p}} = \mathbf{H} \, \tilde{\mathbf{q}} \,. \tag{15}$$

The reparameterized vector $\tilde{\mathbf{q}}$ is obtained by first computing the gradient of $\Gamma(\mathbf{q})$,

$$\boldsymbol{\nabla}\Gamma(\mathbf{q}) = -2\,\mathbf{H}^\top \mathbf{G}^\top \mathbf{W}_d \,(\mathbf{d} - \mathbf{f}) + 2\,\mu\,\mathbf{W}_q\,(\mathbf{q} - \bar{\mathbf{q}}) \,. \tag{16}$$

Then, by considering that $\boldsymbol{\nabla}\Gamma(\tilde{\mathbf{q}}) = \mathbf{0}$ (equation 16), where $\mathbf{0}$ is a vector of zeros, as well as adding and subtracting the term $\left(\mathbf{H}^\top \mathbf{G}^\top \mathbf{W}_d \mathbf{G}\,\mathbf{H}\right)\bar{\mathbf{q}}$, we obtain

$$\tilde{\boldsymbol{\delta}}_q = \mathbf{B}\,\boldsymbol{\delta}_d \,, \tag{17}$$

where

$$\tilde{\boldsymbol{\delta}}_q = \tilde{\mathbf{q}} - \bar{\mathbf{q}} \,, \tag{18}$$

$$\boldsymbol{\delta}_d = \mathbf{d} - \mathbf{G}\,\mathbf{H}\,\bar{\mathbf{q}} \,, \tag{19}$$

$$\mathbf{B} = \left(\mathbf{H}^\top \mathbf{G}^\top \mathbf{W}_d\,\mathbf{G}\,\mathbf{H} + \mu\,\mathbf{W}_q\right)^{-1} \mathbf{H}^\top \mathbf{G}^\top \mathbf{W}_d \,, \tag{20}$$

or, equivalently (Menke, 2018, p. 62),

$$\mathbf{B} = \mathbf{W}_q^{-1}\,\mathbf{H}^\top \mathbf{G}^\top \left(\mathbf{G}\,\mathbf{H}\,\mathbf{W}_q^{-1}\,\mathbf{H}^\top \mathbf{G}^\top + \mu \mathbf{W}_d^{-1}\right)^{-1} \,. \tag{21}$$

Evidently, we have considered that all inverses exist in equations 20 and 21.

The $Q \times D$ matrix $\mathbf{B}$ defined by equation 20 is commonly used for the case in which $D > Q$, i.e., when there are more data than parameters (overdetermined problems). In this case, we consider that the estimate $\tilde{\mathbf{q}}$ is obtained by solving the following linear system for $\tilde{\boldsymbol{\delta}}_q$ (equation 18):

$$\left(\mathbf{H}^\top \mathbf{G}^\top \mathbf{W}_d\,\mathbf{G}\,\mathbf{H} + \mu\,\mathbf{W}_q\right)\tilde{\boldsymbol{\delta}}_q = \mathbf{H}^\top \mathbf{G}^\top \mathbf{W}_d\,\boldsymbol{\delta}_d \,. \tag{22}$$

On the other hand, for the cases in which $D < Q$ (underdetermined problems), matrix $\mathbf{B}$ is usually defined according to equation 21. In this case, the general approach involves estimating $\tilde{\mathbf{q}}$ in two steps. The first consists in solving a linear system for a dummy vector, which is subsequently used to compute $\tilde{\mathbf{q}}$ by a

116   matrix-vector product as follows:

$$\left(\mathbf{G\,H\,W}_q^{-1}\,\mathbf{H}^\top\mathbf{G}^\top + \mu\mathbf{W}_d^{-1} \right)\mathbf{u} = \boldsymbol{\delta}_d$$
$$\tilde{\boldsymbol{\delta}}_q = \mathbf{W}_q^{-1}\,\mathbf{H}^\top\mathbf{G}^\top\mathbf{u} \tag{23}$$

117   where $\mathbf{u}$ is a dummy vector. After obtaining $\tilde{\boldsymbol{\delta}}_q$ (equations 22 and 23), the estimate $\tilde{\mathbf{q}}$ is computed with
118   equation 18.

1.3.1   Optional normalization strategy

120   Setting the regularization parameter μ (equation 11) can be very difficult due to scale differences between
121    $\mathbf{G}$ and $\mathbf{p}$ (equation 3) or $\mathbf{GH}$ and $\mathbf{q}$ (equation 9. When faced with this scenario, a popular strategy (e.g., Li
122   and Oldenburg, 2010; Soler and Uieda, 2021) involves creating the linear system (equations 22 and 23) by
123   substituting $\mathbf{GH}$ and $\mathbf{q}$ with

$$\mathbf{G}_n = \mathbf{G\,H\,N}, \quad \mathbf{q}_n = \mathbf{N}^{-1}\mathbf{q}, \tag{24}$$

124   and then finding the solution $\tilde{\mathbf{q}}_n$ for the normalized parameter vector $\mathbf{q}_n$. The estimate $\tilde{\mathbf{q}}$ for the
125   reparameterized parameter vector $\mathbf{q}$ (equation 9) is subsequently obtained by removing the normalization
126   as follows:

$$\tilde{\mathbf{q}} = \mathbf{N}\,\tilde{\mathbf{q}}_n, \tag{25}$$

127   where $\mathbf{N}$ is an invertible normalization matrix. This strategy usually constrains the practical range of the
128   regularization parameter μ (equation 11).

1.3.2   Classical approach

130   The classical approach in the equivalent-layer technique consists in using $\mathbf{H} = \mathbf{I}_P$, so that $P = Q$, $\mathbf{p} = \mathbf{q}$
131   (equation 9), $\bar{\mathbf{p}} = \bar{\mathbf{q}}$ (equation 14), $\tilde{\mathbf{p}} = \tilde{\mathbf{q}}$ (equation 15), $\mathbf{p}_n = \mathbf{q}_n$ (equation 24) and $\tilde{\mathbf{p}}_n = \tilde{\mathbf{q}}_n$ (equation
132   25). In this case, the linear system (equations 22 and 23) is directly solved for

$$\tilde{\boldsymbol{\delta}}_p = \tilde{\mathbf{p}} - \bar{\mathbf{p}}, \tag{26}$$

133   instead of $\tilde{\boldsymbol{\delta}}_q$ (equation 18).

2   NOTATION FOR SUBVECTORS AND SUBMATRICES

Here, we use a notation inspired on that presented by (Van Loan, 1992, p. 4) to represent subvectors and
submatrices. Subvectors of $\mathbf{d}$, for example, are specified by $\mathbf{d[i]}$, where $\mathbf{i}$ is a list of integer numbers that
"pick out" the elements of $\mathbf{d}$ forming the subvector $\mathbf{d[i]}$. For example, $\mathbf{i} = (1, 6, 4, 6)$ gives the subvector
$\mathbf{d[i]} = [d_1 \; d_6 \; d_4 \; d_6]^\top$. Note that the list $\mathbf{i}$ of indices may be sorted or not and it may also have repeated
indices. For the particular case in which the list has a single element $\mathbf{i} = (i)$, then it can be used to extract
the i-th element $d_i \equiv \mathbf{d}[i]$ of $\mathbf{d}$. Regular lists can be represented by using the colon notation. For example,

$$\mathbf{i} = (3 : 8) \Leftrightarrow \mathbf{d[i]} = [d_3 \; d_4 \; \ldots \; d_8]^\top$$
$$\mathbf{i} = (: 8) \Leftrightarrow \mathbf{d[i]} = [d_1 \; d_2 \; \ldots \; d_8]^\top \quad ,$$
$$\mathbf{i} = (3 :) \Leftrightarrow \mathbf{d[i]} = [d_3 \; d_4 \; \ldots \; d_D]^\top$$

134   where D is the number of elements forming $\mathbf{d}$.

The notation above can also be used to define submatrices of the $D \times P$ matrix $\mathbf{G}$. For example, $\mathbf{i} = (2, 7, 4, 6)$ and $\mathbf{j} = (1, 3, 8)$ lead to the submatrix

$$\mathbf{G[i, j]} = \begin{bmatrix} g_{21} & g_{23} & g_{28} \\ g_{71} & g_{73} & g_{78} \\ g_{41} & g_{43} & g_{48} \\ g_{61} & g_{63} & g_{68} \end{bmatrix} .$$

Note that, in this case, the lists $\mathbf{i}$ and $\mathbf{j}$ "pick out", respectively, the rows and columns of $\mathbf{G}$ that form the submatrix $\mathbf{G[i, j]}$. The i-th row of $\mathbf{G}$ is given by the $1 \times P$ vector $\mathbf{G}[i, :]$. Similarly, the $D \times 1$ vector $\mathbf{G}[:, j]$ represents the j-th column. Finally, we may use the colon notation to define the following submatrix:

$$\mathbf{i} = (2 : 5), \mathbf{j} = (3 : 7) \Leftrightarrow \mathbf{G[i, j]} = \begin{bmatrix} g_{23} & g_{24} & g_{25} & g_{26} & g_{27} \\ g_{33} & g_{34} & g_{35} & g_{36} & g_{37} \\ g_{43} & g_{44} & g_{45} & g_{46} & g_{47} \\ g_{53} & g_{54} & g_{55} & g_{56} & g_{57} \end{bmatrix} ,$$

135 which contains the contiguous elements of $\mathbf{G}$ from rows 2 to 5 and from columns 3 to 7.

3 COMPUTATIONAL STRATEGIES

136 Here, we review some strategies for reducing the computational cost of equivalent-layer technique.
137 Typically, estimating a parameter vector $\tilde{\mathbf{p}}$ or $\tilde{\mathbf{q}}$ requires to solve a large-scale linear inversion (equations
138 22 and 23). This, in turn, means to deal with some obstacles concerning large computational cost:

139 (i) the large computer memory to store large and full matrices;

140 (ii) the long computation time to multiply a matrix by a vector; and

141 (iii) the long computation time to solve a large linear system of equations.

142 Two important factors affecting the efficiency of a given matrix algorithm are the storage and amount of
143 required arithmetic. Here, we quantify this last factor by counting *flops*, which are floating point additions,
144 subtractions, multiplications or divisions (Golub and Van Loan, 2013, p. 12–14). We focus on the overall
145 strategies used by the selected methods.

3.1 Moving window

147 The initial approach to enhance the computational efficiency of the equivalent-layer technique is
148 commonly denoted *moving window* and involves first splitting the observed data d_i, $i \in \{1 : D\}$, into
149 M overlapping subsets (or data windows) formed by D^m data each, $m \in \{1 : M\}$. The data inside the
150 m-th window are usually adjacent to each other and have indices defined by an integer list $\mathbf{i}^m$ having
151 D^m elements. The number of data D^m forming the data windows are not necessarily equal to each other.
152 Each data window has a $D^m \times 1$ observed data vector $\mathbf{d}^m \equiv \mathbf{d}[\mathbf{i}^m]$. The second step consists in defining
153 a set of P equivalent sources with scalar physical property p_j, $j \in \{1 : P\}$, and also split them into M
154 overlapping subsets (or source windows) formed by P^m data each, $m \in \{1 : M\}$. The sources inside the
155 m-th window have indices defined by an integer list $\mathbf{j}^m$ having P^m elements. Each source window has a
156 $P^m \times 1$ parameter vector $\mathbf{p}^m$ and is located right below the corresponding m-th data window. Then, each

157 $\mathbf{d}^m \equiv \mathbf{d}[\mathbf{i}^m]$ is approximated by

$$\mathbf{f}^m = \mathbf{G}^m \mathbf{p}^m \,, \tag{27}$$

158 where $\mathbf{G}^m \equiv \mathbf{G}[\mathbf{i}^m, \mathbf{j}^m]$ is a submatrix of $\mathbf{G}$ (equation 3) formed by the elements computed with equation
159 2 using only the data and equivalent sources located inside the window m-th. The main idea of the moving-
160 window approach is using the $\tilde{\mathbf{p}}^m$ estimated for each window to obtain (i) an estimate $\tilde{\mathbf{p}}$ of the parameter
161 vector for the entire equivalent layer or (ii) a given potential-field transformation $\mathbf{t}$ (equation 4). The main
162 advantages of this approach is that (i) the estimated parameter vector $\tilde{\mathbf{p}}$ or transformed potential field are
163 not obtained by solving the full, but smaller linear systems and (ii) the full matrix $\mathbf{G}$ (equation 3) is never
164 stored.

165 Leão and Silva (1989) presented a pioneer work using the moving-window approach. Their method
166 requires a regularly-spaced grid of observed data on a horizontal plane z_0. The data windows are defined by
167 square local grids of $\sqrt{D'} \times \sqrt{D'}$ adjacent points, all of them having the same number of points D'. The
168 equivalent sources in the m-th data window are located below the observation plane, at a constant vertical
169 distance Δz_0. They are arranged on a regular grid of $\sqrt{P'} \times \sqrt{P'}$ adjacent points following the same
170 grid pattern of the observed data. The local grid of sources for all data windows have the same number
171 of elements P'. Besides, they are vertically aligned, but expands the limits of their corresponding data
172 windows, so that $D' < P'$. Because of this spatial configuration of observed data and equivalent sources,
173 we have that $\mathbf{G}^m = \mathbf{G}'$ (equation 27) for all data windows (i.e., $\forall\, m \in \{1 : M\}$), where $\mathbf{G}'$ is a $D' \times P'$
174 constant matrix.

175 By omitting the normalization strategy used by Leão and Silva (1989), their method consists in directly
176 computing the transformed potential field t_c^m at the central point $(x_c^m, y_c^m, z_0 + \Delta z_0)$ of each data window
177 as follows:

$$t_c^m = \left(\mathbf{G}'\mathbf{a}'\right)^\top \left[\mathbf{G}'\left(\mathbf{G}'\right)^\top + \mu\,\mathbf{I}_{D'}\right]^{-1} \mathbf{d}^m \,, \quad m \in \{1 : M\} \,, \tag{28}$$

178 where $\mathbf{I}_{D'}$ is the identity matrix of order D' and $\mathbf{a}'$ is a $P' \times 1$ vector with elements computed by equation
179 5 by using all equivalent sources in the m-th subset and only the coordinate of the central point in the m-th
180 data window. Due to the presumed spatial configuration of the observed data and equivalent sources, $\mathbf{a}'$ and
181 $\mathbf{G}'$ are the same for all data windows. Hence, only the data vector $\mathbf{d}^m$ is modified according to the position
182 of the data window. Note that equation 28 combines the potential-field transformation (equation 4) with
183 the solution of the undetermined problem (equation 23) for the particular case in which $\mathbf{H} = \mathbf{W}_q = \mathbf{I}_{P'}$
184 (equations 9 and 13), $\mathbf{W}_d = \mathbf{I}_{D'}$ (equation 12), $\bar{\mathbf{p}} = \mathbf{0}$ (equation 14), where $\mathbf{I}_{P'}$ and $\mathbf{I}_{D'}$ are identity
185 matrices of order P' and D', respectively, and $\mathbf{0}$ is a vector of zeros.

186 The method proposed by Leão and Silva (1989) can be outlined by the Algorithm 1. Note that Leão and
187 Silva (1989) directly compute the transformed potential t_c^m at the central point of each data window without
188 explicitly computing and storing an estimated for $\mathbf{p}^m$ (equation 27). It means that their method allows
189 computing a single potential-field transformation. A different transformation or the same one evaluated at
190 different points require running their moving-data window method again.

191 Soler and Uieda (2021) generalized the method proposed by Leão and Silva (1989) for irregularly spaced
192 data on an undulating surface. A direct consequence of this generalization is that a different submatrix
193 $\mathbf{G}^m \equiv \mathbf{G}[\mathbf{i}^m, \mathbf{j}^m]$ (equation 27) must be computed for each window. Differently from Leão and Silva
194 (1989), Soler and Uieda (2021) store the computed $\tilde{\mathbf{p}}^m$ for all windows and subsequently use them to obtain
195 a desired potential-field transformation (equation 4) as the superposed effect of all windows. The estimated
196 $\tilde{\mathbf{p}}^m$ for all windows are combined to form a single $P \times 1$ vector $\tilde{\mathbf{p}}$, which is an estimate for original

---

Algorithm 1: Generic pseudo-code for the method proposed by Leão and Silva (1989).

Initialization :
1. Set the indices $\mathbf{i}^m$ for each data window, $m \in \{1 : M\}$;
2. Set the indices $\mathbf{j}^m$ for each source window, $m \in \{1 : M\}$;
3. Set the constant depth $z_0 + \Delta z_0$ for all equivalent sources ;
4. Compute the vector $\mathbf{a}'$ associated with the desired potential-field transformation ;
5. Compute the matrix $\mathbf{G}'$;
6. Compute $(\mathbf{G}'\mathbf{a}')^\top \left[\mathbf{G}' \left(\mathbf{G}' \right)^\top + \mu\, \mathbf{I}_{D'} \right]^{-1}$;
7. $m = 1$;
8. **while** $m < M$ **do**
9.   Compute t_c^m (equation 28) ;
10. $m \leftarrow m + 1$;
11. **end**

---

parameter vector $\mathbf{p}$ (equation 3). For each data window, Soler and Uieda (2021) solve an overdetermined problem (equation 22) for $\tilde{\mathbf{p}}^m$ by using $\mathbf{H} = \mathbf{W}_q = \mathbf{I}_{P^m}$ (equations 9 and 13), $\mathbf{W}_d^m$ (equation 12) equal to a diagonal matrix of weights for the data inside the m-th window and $\bar{p} = \mathbf{0}$ (equation 14), so that

$$\left[(\mathbf{G}^m)^\top \mathbf{W}_d^m \mathbf{G}^m + \mu\, \mathbf{I}_{P'} \right] \tilde{\mathbf{p}}^m = (\mathbf{G}^m)^\top \mathbf{W}_d^m \mathbf{d}^m . \tag{29}$$

It is important to stress here that Soler and Uieda (2021) used the normalization strategy described in section 1.3.1, but we have conveniently omitted here. The overall steps of their method are defined by the Algorithm 2. Note that this algorithm starts with a residuals vector $\mathbf{r}$ that is iteratively updated. At each iteration, the potential field predicted a source window is computed at all observation points and removed from the residuals vector $\mathbf{r}$.

Algorithm 2: Generic pseudo-code for the method proposed by Soler and Uieda (2021).

Initialization :
1. Set the indices $\mathbf{i}^m$ for each data window, $m \in \{1 : M\}$;
2. Set the indices $\mathbf{j}^m$ for each source window, $m \in \{1 : M\}$;
3. Set the depth of all equivalent sources ;
4. Set a $D \times 1$ residuals vector $\mathbf{r} = \mathbf{d}$;
5. Set a $P \times 1$ vector $\tilde{\mathbf{p}} = \mathbf{0}$;
6. $m = 1$;
7. **while** $m < M$ **do**
8.   Set the matrix $\mathbf{W}_d^m$;
9.   Compute the matrix $\mathbf{G}^m$;
10.   Compute $\tilde{\mathbf{p}}^m$ (equation 29) ;
11. $\tilde{\mathbf{p}}[\mathbf{j}^m] \leftarrow \tilde{\mathbf{p}}[\mathbf{j}^m] + \tilde{\mathbf{p}}^m$;
12. $\mathbf{r} \leftarrow \mathbf{r} - \mathbf{G}[:, \mathbf{j}^m]\, \tilde{\mathbf{p}}^m$;
13. $m \leftarrow m + 1$;
14. **end**

3.2 Column update

Cordell (1992) proposed a computational strategy that was later used by Guspí and Novara (2009) and relies on first defining one equivalent source located right below each observed data d_i, $i \in \{1 : D\}$, at

---

208 a vertical coordinate $z_i + \Delta z_i$, where Δz_i is proportional to the distance from the i-th observation point
209 (x_i, y_i, z_i) to its closest neighbor. The second step consists in updating the physical property p_j of a given
210 equivalent source, $j \in \{1 : D\}$ and remove its predicted potential field from the observed data vector $\mathbf{d}$,
211 producing a residuals vector $\mathbf{r}$. Then, the same procedure is repeated for other sources with the purpose of
212 iteratively updating $\mathbf{r}$ and the $D \times 1$ parameter vector $\mathbf{p}$ containing the physical property of all equivalent
213 sources. At the end, the algorithm produces an estimate $\tilde{\mathbf{p}}$ for the parameter vector yielding a predicted
214 potential field $\mathbf{f}$ (equation 3) satisfactorily fitting the observed data $\mathbf{d}$ according to a given criterion. Note
215 that the method proposed by Cordell (1992) iteratively solves the linear $\mathbf{G}\tilde{\mathbf{p}} \approx \mathbf{d}$ with a $D \times D$ matrix $\mathbf{G}$.
216 At each iteration, only a single column of $\mathbf{G}$ (equation 3) is used. An advantage of this *column-update*
217 *approach* is that the full matrix $\mathbf{G}$ is never stored.

218   Algorithm 3 delineates the Cordell's method. Note that a single column $\mathbf{G}[:, i_{\texttt{max}}]$ of the $D \times D$ matrix $\mathbf{G}$
219 (equation 3) is used per iteration, where $i_{\texttt{max}}$ is the index of the maximum absolute value in $\mathbf{r}$. As pointed out
220 by Cordell (1992), the method does not necessarily decrease monotonically along the iterations. Besides,
221 the method may not converge depending on how the vertical distances Δz_i, $i \in \{1 : D\}$, controlling the
222 depths of the equivalent sources are set. According to Cordell (1992), the maximum absolute value $r_{\texttt{max}}$
223 in $\mathbf{r}$ decreases robustly at the beginning and oscillates within a narrowing envelope for the subsequent
224 iterations.

Algorithm 3: Generic pseudo-code for the method proposed by Cordell (1992).

  Initialization :
1 Compute a $D \times 1$ vector $\mathbf{\Delta z}$ whose i-th element Δz_i is a vertical distance controlling the depth of
    the i-th equivalent source, $i \in \{1 : D\}$;
2 Set a tolerance ϵ ;
3 Set a maximum number of iteration `ITMAX` ;
4 Set a $D \times 1$ residuals vector $\mathbf{r} = \mathbf{d}$;
5 Set a $D \times 1$ vector $\tilde{\mathbf{p}} = \mathbf{0}$;
6 Define the maximum absolute value $r_{\texttt{max}}$ in $\mathbf{r}$;
7 $m = 1$;
8 **while** $(r_{\texttt{max}} > \epsilon)$ **and** $(m < \texttt{ITMAX})$ **do**
9   Define the coordinates $(x_{\texttt{max}}, y_{\texttt{max}}, z_{\texttt{max}})$ and index $i_{\texttt{max}}$ of the observation point associated with
       $r_{\texttt{max}}$;
10     $\tilde{\mathbf{p}}[i_{\texttt{max}}] \leftarrow \tilde{\mathbf{p}}[i_{\texttt{max}}] + (a_{\texttt{max}} \mathbf{\Delta z}[i_{\texttt{max}}])$;
11     $\mathbf{r} \leftarrow \mathbf{r} - (\mathbf{G}[:, i_{\texttt{max}}] \tilde{\mathbf{p}}[i_{\texttt{max}}])$;
12   Define the new $r_{\texttt{max}}$ in $\mathbf{r}$;
13     $m \leftarrow m + 1$;
14 **end**

3.3   Row update

226   Mendonça and Silva (1994) proposes an algebraic reconstruction technique (ART) (e.g., van der Sluis
227 and van der Vorst, 1987, p. 58) to estimate a parameter vector $\tilde{\mathbf{p}}$ for a regular grid of P equivalent sources
228 on a horizontal plane z_0. Such methods iterate on the linear system rows to estimate corrections for the
229 parameter vector, which may substantially save computer time and memory required to compute and store
230 the full linear system matrix along the iterations. The convergence of such *row-update methods* depends on
231 the linear system condition. The main advantage of such methods is not computing and storing the full
232 linear system matrix, but iteratively using its rows. The particular ART method proposed by Mendonça and

Silva (1994) considers that

$$\mathbf{d} = \begin{bmatrix} \mathbf{d}_e \\ \mathbf{d}_r \end{bmatrix} , \quad \mathbf{G} = \begin{bmatrix} \mathbf{G}_e \\ \mathbf{R}_r \end{bmatrix} , \tag{30}$$

where $\mathbf{d}_e$ and $\mathbf{d}_r$ are $D_e \times 1$ and $D_r \times 1$ vectors and $\mathbf{G}_e$ and $\mathbf{G}_r$ are $D_e \times P$ and $D_r \times P$ matrices, respectively. Mendonça and Silva (1994) designate $\mathbf{d}_e$ and $\mathbf{d}_r$ as, respectively, *equivalent* and *redundant* data. With the exception of a normalization strategy, Mendonça and Silva (1994) calculate a $P \times 1$ estimated parameter vector $\tilde{\mathbf{p}}$ by solving an underdetermined problem (equation 23) involving only the equivalent data $\mathbf{d}_e$ (equation 30) for the particular case in which $\mathbf{H} = \mathbf{W}_p = \mathbf{I}_P$ (equations 9 and 13), $\mathbf{W}_d = \mathbf{I}_{D_e}$ (equation 12) and $\bar{p} = \mathbf{0}$ (equation 14), which results in

$$\begin{aligned} (\mathbf{F} + \mu\, \mathbf{I}_{D_e})\, \mathbf{u} &= \mathbf{d}_e \\ \tilde{\mathbf{p}} &= \mathbf{G}_e^\top \mathbf{u} \end{aligned} , \tag{31}$$

where $\mathbf{F}$ is a $P \times P$ matrix that replaces $\mathbf{G}_e\, \mathbf{G}_e^\top$. Mendonça and Silva (1994) presume that the estimated parameter vector $\tilde{\mathbf{p}}$ obtained from equation 31 leads to a $D_r \times 1$ residuals vector

$$\mathbf{r} = \mathbf{d}_r - \mathbf{G}_r \tilde{\mathbf{p}} \tag{32}$$

having a maximum absolute value $r_{\texttt{max}} \leq \epsilon$, where ϵ is a predefined tolerance.

The overall method of Mendonça and Silva (1994) is defined by Algorithm 4. It is important noting that the number D_e of equivalent data in $\mathbf{d}_e$ increases by one per iteration, which means that the order of the linear system in equation 31 also increases by one at each iteration. Those authors also propose a computational strategy based on Cholesky factorization (e.g., Golub and Van Loan, 2013, p. 163) for efficiently updating $(\mathbf{F} + \mu \mathbf{I}_{D_e})$ at a given iteration (line 16 in Algorithm 4) by computing only its new elements with respect to those computed in the previous iteration.

3.4 Reparameterization

Another approach for improving the computational performance of equivalent-layer technique consists in setting a $P \times Q$ reparameterization matrix $\mathbf{H}$ (equation 9) with $Q << P$. This strategy has been used in applied geophysics for decades (e.g., Skilling and Bryan, 1984; Kennett et al., 1988; Oldenburg et al., 1993; Barbosa et al., 1997) and is known as *subspace method*. The main idea relies in reducing the linear system dimension from the original P-space to a lower-dimensional subspace (the Q-space). An estimate $\tilde{\mathbf{q}}$ for the reparameterized parameter vector $\mathbf{q}$ is obtained in the Q-space and subsequently used to obtain an estimate $\tilde{\mathbf{p}}$ for the parameter vector $\mathbf{p}$ (equation 3) in the P-space by using equation 9. Hence, the key aspect of this *reparameterization approach* is solving an appreciably smaller linear inverse problem for $\tilde{\mathbf{q}}$ than that for the original parameter vector $\tilde{\mathbf{p}}$ (equation 3).

Oliveira Jr. et al. (2013) have used this approach to describe the physical property distribution on the equivalent layer in terms of piecewise bivariate polynomials. Specifically, their method consists in splitting a regular grid of equivalent sources into source windows inside which the physical-property distribution is described by bivariate polynomial functions. The key aspect of their method relies on the fact that the total number of coefficients required to define the bivariate polynomials is considerably smaller than the original number of equivalent sources. Hence, they formulate a linear inverse problem for estimating the polynomial coefficients and use them later to compute the physical property distribution on the equivalent layer.

Algorithm 4: Generic pseudo-code for the method proposed by Mendonça and Silva (1994).

Initialization :
1   Set a regular grid of P equivalent sources at a horizontal plane z_0 ;
2   Set a tolerance ϵ ;
3   Set a $D \times 1$ residuals vector $\mathbf{r} = \mathbf{d}$;
4   Define the maximum absolute value $r_{\mathtt{max}}$ in $\mathbf{r}$;
5   Define the index $i_{\mathtt{max}}$ of $r_{\mathtt{max}}$;
6   Define the list of indices $\mathbf{i}_r$ of the remaining data in $\mathbf{r}$;
7   Define $\mathbf{d}_e = \mathbf{d}[i_{\mathtt{max}}]$;
8   Compute $(\mathbf{F} + \mu\, \mathbf{I}_{D_e})$ and $\mathbf{G}_e$;
9   Compute $\tilde{\mathbf{p}}$ (equation 31) ;
10   Compute $\mathbf{r} = \mathbf{d}[\mathbf{i}_r] - \mathbf{G}[\mathbf{i}_r, :]\, \tilde{\mathbf{p}}$;
11   Define the maximum absolute value $r_{\mathtt{max}}$ in $\mathbf{r}$;
12   **while** $(r_{\mathtt{max}} > \epsilon)$ **do**
13   Define the index $i_{\mathtt{max}}$ of $r_{\mathtt{max}}$;
14   Define the list of indices $\mathbf{i}_r$ of the remaining elements in $\mathbf{r}$;
15     $\mathbf{d}_e \leftarrow \begin{bmatrix} \mathbf{d}_e \\ \mathbf{d}[i_{\mathtt{max}}] \end{bmatrix}$;
16   Update $(\mathbf{F} + \mu\, \mathbf{I}_{D_e})$ and $\mathbf{G}_e$;
17   Update $\tilde{\mathbf{p}}$ (equation 31) ;
18   Update $\mathbf{r} = \mathbf{d}[\mathbf{i}_r] - \mathbf{G}[\mathbf{i}_r, :]\, \tilde{\mathbf{p}}$;
19   Define the maximum absolute value $r_{\mathtt{max}}$ in $\mathbf{r}$;
20   **end**

The method proposed by Oliveira Jr. et al. (2013) consists in solving an overdetermined problem (equation 22) for estimating the polynomial coefficients $\tilde{\mathbf{q}}$ with $\mathbf{W}_d = \mathbf{I}_D$ (equation 12) and $\bar{q} = \mathbf{0}$ (equation 14), so that

$$\left(\mathbf{H}^\top \mathbf{G}^\top \mathbf{G}\, \mathbf{H} + \mu\, \mathbf{W}_q\right) \tilde{\mathbf{q}} = \mathbf{H}^\top \mathbf{G}^\top\, \mathbf{d}\,, \tag{33}$$

where $\mathbf{W}_q = \mathbf{H}^\top \mathbf{W}_p\, \mathbf{H}$ is defined by a matrix $\mathbf{W}_p$ representing the zeroth- and first-order Tikhonov regularization (e.g., Aster et al., 2019, p. 103). Note that, in this case, the prior information is defined in the P-space for the original parameter vector $\mathbf{p}$ and then transformed to the Q-space. Another characteristic of their method is that it is valid for processing irregularly-spaced data on an undulating surface.

Mendonça (2020) also proposed a reparameterization approach for the equivalent-layer technique. Their approach, however, consists in setting $\mathbf{H}$ as a truncated singular value decomposition (SVD) (e.g., Aster et al., 2019, p. 55) of the observed potential field. Differently from Oliveira Jr. et al. (2013), however, the method of Mendonça (2020) requires a regular grid of potential-field data on horizontal plane. Another difference is that these authors uses $\mathbf{W}_q = \mathbf{I}_Q$ (equation 13), which means that the regularization is defined directly in the Q-space.

Before Oliveira Jr. et al. (2013) and Mendonça (2020), Barnes and Lumley (2011) also proposed a computationally efficient method for equivalent-layer technique based on reparameterization. A key difference, however, is that Barnes and Lumley (2011) did not set a $P \times Q$ reparameterization matrix $\mathbf{H}$ (equation 9) with $Q << P$. Instead, they used a matrix $\mathbf{H}$ with $Q \approx 1.7\, P$. Their central idea is setting a reparameterization scheme that groups distant equivalent sources into blocks by using a bisection process. This scheme leads to a quadtree representation of the physical-property distribution on the equivalent layer, so that matrix $\mathbf{GH}$ (equation 10) is notably sparse. Barnes and Lumley (2011) explore this sparsity in

287 solving the overdetermined problem for $\tilde{\mathbf{q}}$ (equation 33) via conjugate-gradient method (e.g., Golub and
288 Van Loan, 2013, sec. 11.3).

3.5 Wavelet compression

290   Previously to Barnes and Lumley (2011), the idea of transforming the dense matrix $\mathbf{G}$ (equation 3) into a
291 sparse one has already been used in the context of equivalent-layer technique. Li and Oldenburg (2010)
292 proposed a method that applies the discrete wavelet transform to introduce sparsity into the original dense
293 matrix $\mathbf{G}$. Those authors approximate a planar grid of potential-field data by a regularly-spaced grid of
294 equivalent sources, so that the number of data D and sources P is the same , i.e., $D = P$. Specifically, Li
295 and Oldenburg (2010) proposed a method that applies the wavelet transform to the original dense matrix $\mathbf{G}$
296 and sets to zero the small coefficients that are below a given threshold, which results in an approximating
297 sparse representation of $\mathbf{G}$ in the wavelet domain. They first consider the following approximation

$$\mathbf{d}_w \approx \mathbf{G}_s\,\mathbf{p}_w\,, \tag{34}$$

298 where

$$\mathbf{d}_w = \boldsymbol{\mathcal{W}}\,\mathbf{d}\,, \quad \mathbf{p}_w = \boldsymbol{\mathcal{W}}\,\mathbf{p}\,, \tag{35}$$

299 are the observed data and parameter vector in the wavelet domain; $\boldsymbol{\mathcal{W}}$ is a $D \times D$ orthogonal matrix
300 defining a discrete wavelet transform; and $\mathbf{G}_s$ is a sparse matrix obtained by setting to zero the elements of

$$\mathbf{G}_w = \boldsymbol{\mathcal{W}}\,\mathbf{G}\,\boldsymbol{\mathcal{W}} \tag{36}$$

301 with absolute value smaller than a given threshold.

302   Li and Oldenburg (2010) apply the normalization strategy defined in section 1.3.1 to equation 34 by using
303 a diagonal normalization matrix $\mathbf{N}$ (equation 24) and then formulate an overdetermined problem (equation
304 22) with $\mathbf{H} = \mathbf{I}_P$ (equations 9), $\mu = 0$ (equation 11), $\mathbf{W}_d = \mathbf{I}_D$ (equation 12) and $\bar{p} = \mathbf{0}$ (equation 14) so
305 that

$$\left(\mathbf{G}_n^{\top}\mathbf{G}_n\right)\tilde{\mathbf{p}}_n = \mathbf{G}_n^{\top}\mathbf{d}_w\,, \tag{37}$$

306 where $\mathbf{G}_n$ and $\tilde{\mathbf{p}}_n$ are defined according to equation 24 in terms of the sparse matrix $\mathbf{G}_s$ (equation 34).
307 They solve this linear system (equation 37) with the conjugate-gradient method (e.g., Golub and Van Loan,
308 2013, sec. 11.3) and use it to obtain an estimate $\tilde{\mathbf{p}}$ for the parameter vector given by

$$\tilde{\mathbf{p}} = \boldsymbol{\mathcal{W}}^{\top}\left(\mathbf{N}\,\tilde{\mathbf{p}}_n\right)\,, \tag{38}$$

309 where the term within parentheses removes the normalization (equation 25) and matrix $\boldsymbol{\mathcal{W}}^{\top}$ applies an
310 inverse wavelet transform.

3.6 Iterative methods using the full matrix $\mathbf{G}$

312   In the context of equivalent-layer technique, there are methods (e.g., Li and Oldenburg, 2010; Barnes
313 and Lumley, 2011) that modify the original linear system and then iteratively solve this modified system
314 by using the well-established conjugate gradient method (e.g., Golub and Van Loan, 2013, sec. 11.3).
315 Xia and Sprowl (1991) also proposed an iterative method for estimating a parameter vector $\tilde{\mathbf{p}}$ (equation
316 3). Their method, however, uses the original matrix $\mathbf{G}$ without previously computing a compression
317 or reparameterization, for example. More than two decades later, Siqueira et al. (2017) have deduced

318 essentially the same method presented by Xia and Sprowl (1991), but by following a more theoretical
319 approach based on the Gauss' theorem (e.g., Kellogg, 1967, p. 43) and the total excess of mass (e.g.,
320 Blakely, 1996, p. 60). Besides, Siqueira et al. (2017) have shown that this method produces very stable
321 solutions, even for noise-corrupted potential-field data.

322   The iterative method proposed by Xia and Sprowl (1991) and Siqueira et al. (2017) is outlined in
323 Algorithm 5, presumes an equivalent layer formed by monopoles (point masses) and can be applied to
324 irregularly-spaced data on an undulating surface. Note that the residuals $\mathbf{r}$ are used to compute a correction
325 $\mathbf{\Delta p}$ for the parameter vector at each iteration (line 11), which requires a matrix-vector product involving
326 the full matrix $\mathbf{G}$. Interestingly, this approach for estimating the physical property distribution on an
327 equivalent layer is the same originally proposed by Bott (1960) for estimating the basement relief under
328 sedimentary basins. The method of Xia and Sprowl (1991) and Siqueira et al. (2017) was originally
329 proposed for processing gravity data, but can be potentially applied to any harmonic function because it
330 actually represents an iterative solution of the classical *Dirichlet's problem* or the *first boundary value*
331 *problem of potential theory* (Kellogg, 1967, p. 236) on a plane.

332   Recently, Jirigalatu and Ebbing (2019) presented another iterative method for estimating a parameter
333 vector $\tilde{\mathbf{p}}$ (equation 3). With the purpose of combining different potential-field data, their method basically
334 modifies that shown in Algorithm 5 by changing the initial approximation and the iterative correction for
335 the parameter vector. Specifically, Jirigalatu and Ebbing (2019) replace line 3 by $\tilde{\mathbf{p}} = \mathbf{0}$, where $\mathbf{0}$ is a vector
336 of zeros, and line 8 by $\mathbf{\Delta p} = \omega\, \mathbf{G}^\top \mathbf{r}$, where ω is a positive scalar defined by trial and error. Note that
337 this modified approach requires two matrix-vector products involving the full matrix $\mathbf{G}$ per iteration. To
338 overcome the high computational cost of these two products, Jirigalatu and Ebbing (2019) set an equivalent
339 layer formed by prisms and compute their predicted potential field in the wavenumber domain by using the
340 Gauss-FFT technique Zhao et al. (2018).

341   INCLUIMOS Xia et al. (1993) - FOURIER - OU NÃO?

Algorithm 5: Generic pseudo-code for the iterative methods proposed by Xia and Sprowl (1991)
and Siqueira et al. (2017). The symbol "∘" denotes the entrywise or Hadamard product (e.g., Horn
and Johnson, 1991, p. 298) and $\mathbf{s}$ is a $P \times 1$ vector whose j-th element is a predefined element of
area centered at the j-th equivalent source.

Initialization :
1 Set P equivalent sources on a horizontal plane z_0 ;
2 Set the auxiliary vector $\mathbf{s}$;
3 Set a tolerance ϵ ;
4 Compute $\tilde{\mathbf{p}} = \mathbf{s} \circ \mathbf{d}$;
5 Compute $\mathbf{G}$;
6 Compute $\mathbf{r} = \mathbf{d} - \mathbf{G}\,\tilde{\mathbf{p}}$;
7 Compute $\|\mathbf{r}\|$;
8 **while** $(\|\mathbf{r}\| > \epsilon)$ **do**
9 $\quad$ $\mathbf{\Delta p} = \mathbf{s} \circ \mathbf{r}$;
10 $\quad$ $\tilde{\mathbf{p}} \leftarrow \tilde{\mathbf{p}} + \mathbf{\Delta p}$;
11 $\quad$ $\mathbf{r} \leftarrow \mathbf{r} - \mathbf{G}\,\mathbf{\Delta p}$;
12 $\quad$ Compute $\|\mathbf{r}\|$;
13 **end**

3.7 Discrete convolution

Recently, Takahashi et al. (2020, 2022) have shown that $\mathbf{G}$ assumes a *doubly block Toeplitz* (Jain, 1989, p. 28) or *block-Toeplitz Toeplitz-block* (BTTB) (Chan and Jin, 2007, p. 67) structure for the particular case in which (i) there is a single equivalent source right below each potential-field datum and (ii) both data and sources rely on planar and regularly spaced grids. This structure allows formulating the product of $\mathbf{G}$ and an arbitrary vector as a *fast discrete convolution* (Van Loan, 1992, section 4.2).

Takahashi et al. (2020, 2022) consider a regular grid of D potential-field data at points (x_i, y_i, z_0), $i \in \{1 : D\}$, on a horizontal plane z_0. The data indices i may be ordered along the $x-$ or $y-$direction, which results in an $x-$ or $y-$oriented grid, respectively. They also consider a single equivalent source located right below each datum, at a constant vertical coordinate $z_0 + \Delta z$, $\Delta z > 0$. In this case, the number of data and equivalent sources are equal to each other (i.e., $D = P$) and $\mathbf{G}$ (equation 3) is a $D \times D$ matrix formed by $D_B \times D_B$ blocks, where each block has $D_b \times D_b$ elements, with $D = D_B\,D_b$.

Let $\boldsymbol{\mathcal{D}}$ be a $2D_B \times 2D_b$ matrix formed by four blocks of $D_B \times D_b$ elements. All blocks have null elements, except the first block in the first quadrant, which has the observed data d_i, $i \in \{1 : D\}$ arranged along its rows or columns if the data grid has a $x-$ or $y-$oriented pattern, respectively. Let us also consider a $2D_B \times 2D_b$ matrix $\boldsymbol{\mathcal{P}}$ having the same structure of $\boldsymbol{\mathcal{D}}$, but containing the elements p_j of the parameter vector $\mathbf{p}$, $j \in \{1 : P\}$.

The first block (in the first quadrant) contains

The conjugate gradient (CG) is a very popular iterative method for solving linear systems. This method was originally developed to solve systems having a square and positive definite matrix. There are two adapted versions of the CG method. The first is called *conjugate gradient normal equation residual* (CGNR) Golub and Van Loan (2013, sec. 11.3) or *conjugate gradient least squares* (CGLS) (Aster et al., 2019, p. 165) and is used to solve overdetermined problem (equation 22). The second is called *conjugate gradient normal equation error* (CGNE) method Golub and Van Loan (2013, sec. 11.3) and is used to solve the underdetermined problem (equation 23).

Takahashi et al. (2020)

Takahashi et al. (2022)

4 TEXTO ANTIGO

4.0.1 The wavelet compression and lower-dimensional subspace

For large data sets, the sensitivity matrix $\mathbf{A}$ (equation 3) is a drawback in applying the equivalent-layer technique because it is a large and dense matrix.

? transformed a large and full sensitivity matrix into a sparse one by using fast wavelet transforms. In the wavelet domain, ? applied 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., $\mathbf{W_2}^\top$).

$$\tilde{\mathbf{A}} = \mathbf{W_2}\,\mathbf{A}\,\mathbf{W_2}^\top\,, \tag{39}$$

377 where $\tilde{\mathbf{A}}$ is the expanded original sensitivity matrix in the wavelet bases with many elements zero or close
378 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
379 is obtained by retaining only the large elements of the $\tilde{\mathbf{A}}$. Thus, the elements of $\tilde{\mathbf{A}}$ whose amplitudes
380 fall below a relative threshold are discarded. In **?**, the original sensitivity matrix $\mathbf{A}$ is high compressed
381 resulting in a sparce matrix $\tilde{\mathbf{A}}_\mathbf{s}$ with a few percent of nonzero elements and the the inverse problem is
382 solved in the wavelet domain by using $\tilde{\mathbf{A}}_\mathbf{s}$ and a incomplete conjugate gradient least squares, without an
383 explicit regularization parameter and a limited number of iterations. The solution is obtained by solving the
384 following linear system

$$\tilde{\mathbf{A}}_\mathbf{L}^\top \, \tilde{\mathbf{A}}_\mathbf{L} \, \tilde{\mathbf{p}}_\mathbf{L}^* \; = \tilde{\mathbf{A}}_\mathbf{L}^\top \, \tilde{\mathbf{d}}^o \, , \tag{40}$$

385 where $\tilde{\mathbf{p}}_\mathbf{L}^*$ is obtained by solving the linear system given by equation 40,

$$\tilde{\mathbf{A}}_\mathbf{L} \;\; = \;\; \tilde{\mathbf{A}}_\mathbf{s} \, \tilde{\mathbf{L}}^{-1}, \tag{41a}$$

$$\tilde{\mathbf{p}}_\mathbf{L} \;\; = \;\; \tilde{\mathbf{L}}\tilde{\mathbf{p}}, \tag{41b}$$

$$\tilde{\mathbf{d}}^o \;\; = \;\; \mathbf{W_2} \, \mathbf{d}^o, \tag{41c}$$

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

$$\tilde{\mathbf{p}} = \tilde{\mathbf{L}}^{-1} \, \tilde{\mathbf{p}}_\mathbf{L}^* \, , \tag{42}$$

389 and

$$\mathbf{p} = \mathbf{W_2} \, \tilde{\mathbf{p}} \, . \tag{43}$$

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

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

398   **?** overcame the solution of intractable large-scale equivalent-layer problem by using the subspace method
399 (e.g., Skilling and Bryan, 1984; Kennett et al., 1988; Oldenburg et al., 1993; Barbosa et al., 1997). The
400 subspace method reduces the dimension of the linear system of equations to be solved. Given a higher-
401 dimensional space (e.g., M-dimensional model space, $\mathbb{R}^M$), there exists many lower-dimensional subspaces
402 (e.g., Q-dimensional subspace) of $\mathbb{R}^M$. The linear inverse problem related to the equivalent-layer technique
403 consists in finding an M-dimension parameter vector $\mathbf{p} \in \mathbb{R}^M$ which adequately fits the potential-field
404 data. The subspace method looks for a parameter vector who lies in a Q-dimensional subspace of $\mathbb{R}^M$
405 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
406 parameter vector in the subspace method can be written as

$$\mathbf{p} = \mathbf{V} \, \boldsymbol{\alpha} \, , \tag{44}$$

407 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 $\mathbb{R}^M$.
408 In equation 44, the parameter vector $\mathbf{p}$ is defined as a linear combination in the space spanned by Q basis
409 vectors $\mathbf{v}_i = 1, ..., Q$ and $\boldsymbol{\alpha}$ is a Q-dimensional unknown vector to be determined. The main advantage of
410 the subspace method is that the linear system of M equations in M unknowns to be originally solved is
411 reduced to a new linear system of Q equations in Q unknowns which requires much less computational
412 effort since $Q << M$, i.e.:

$$\mathbf{V}^\top \mathbf{A}^\top \mathbf{A} \mathbf{V} \boldsymbol{\alpha}^* = \mathbf{V}^\top \mathbf{d}^o . \tag{45}$$

413 To avoid the storage of matrices $\mathbf{A}$ and $\mathbf{V}$, **?** evaluates an element of the matrix $\mathbf{AV}$ by calculating the dot
414 product between the row of matrix $\mathbf{A}$ and the column of the matrix $\mathbf{B}$. After estimating $\boldsymbol{\alpha}^*$ (equation 45)
415 belonging to a Q-dimensional subspace of $\mathbb{R}^M$, the distribution over the equivalent layer $\mathbf{p}$ in the $\mathbb{R}^M$ is
416 obtained by applying equation 44. The choice of the Q basis vectors $\mathbf{v}_i = 1, ..., Q$ (equation 44) in the
417 subspace method is not strict. **?**, for example, chose the eigenvectors yielded by applying the singular value
418 decomposition of the matrix containing the gridded data set. The number of eigenvectors used to form
419 basis vectors will depend on the singular values.

420 The proposed subspace method for solving large-scale equivalent-layer problem by **?** was applied to
421 estimate the mass excess or deficiency caused by causative gravity sources.

4.0.2   The quadtree discretization

423 To make the equivalent-layer technique tractable, **?** also transformed the dense sensitivity matrix $\mathbf{A}$
424 (equation 3) into a sparse matrix. In **?**, a sparce version of the sensitivity matrix is achived by grouping
425 equivalent sources (e.g., they used prisms) distant from an observation point together to form a larger prism
426 or larger block. Each larger block has averaged physical properties and averaged top- and bottom-surfaces
427 of the grouped smaller prisms (equivalent sources) that are encompassed by the larger block. The authors
428 called it the 'larger averaged block' and the essence of their method is the reduction in the number of
429 equivalent sources, which means a reduction in the number of parameters to be estimated implying in
430 model dimension reduction.

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

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

4.0.3   The reparametrization of the equivalent layer

443 Oliveira Jr. et al. (2013) reparametrized the whole equivalent-layer model by a piecewise bivariate-
444 polynomial function defined on a set of Q equivalent-source windows. In Oliveira Jr. et al.'s (2013)
445 approach, named polynomial equivalent layer (PEL), the parameter vector within the kth equivalent-source

446   window $\mathbf{p}^k$ can be written in matrix notation as

$$\mathbf{p}^k = \mathbf{B}^k \, \mathbf{c}^k \,, \qquad k \,=\, 1 \ldots Q \,, \tag{46}$$

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

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

$$\left(\mathbf{B}^\top \, \mathbf{A}^\top \, \mathbf{A} \, \mathbf{B} \, + \, \mu \mathbf{I} \right) \, \mathbf{c}^* \,=\, \mathbf{B}^\top \, \mathbf{A}^\top \, \mathbf{d}^o \,, \tag{47}$$

453   where μ is a regularizing parameter, $\mathbf{c}^*$ is an estimated H-dimensional vector containing all coefficients
454   describing all polynomial functions within all equivalent-source windows which compose the entire
455   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
456   such that the main-diagonal blocks are $\mathbf{B}^k$ matrices (equation 46) and all off-diagonal blocks are zero
457   matrices. For ease of the explanation of equation 47, we keep only the zeroth-order Tikhonov regularization
458   and omitting the first-order Tikhonov regularization (**?**) which was also used by Oliveira Jr. et al. (2013).

459   The main advantage of the PEL is solve H-dimensional system of equations (equation 47), where H
460   totalizes the number of polynomial coefficients composing all equivalent-source windows, requiring a
461   lower computational effort since $H <<< N$. To avoid the storage of matrices $\mathbf{A}$ and $\mathbf{B}$, Oliveira Jr. et al.
462   (2013) evaluate an element of the matrix $\mathbf{AB}$ by calculating the dot product between the row of matrix $\mathbf{A}$
463   and the column of the matrix $\mathbf{B}$. After estimating all polynomial coefficients of all windows, the estimated
464   coefficients ($\mathbf{c}^*$ in equation 47) are transformed into a single physical-property distribution encompassing
465   the entire equivalent layer.

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

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

4.0.4 The iterative scheme without solving a linear system

474   There exists a class of methods that iteratively estimate the distribution of physical properties within an
475   equivalent layer without the need to solve linear systems. The method initially introduced by Cordell (1992)
476   and later expanded upon by Guspí and Novara (2009) updates the physical property of sources, located
477   beneath each potential-field data, by removing the maximum residual between the observed and fitted data.
478   In addition, Xia and Sprowl (1991) and Xia et al. (1993) have developed efficient iterative algorithms for
479   updating the distribution of physical properties within the equivalent layer in the wavenumber and space
480   domains, respectively. Specifically, in Xia and Sprowl's (1991) method the physical-property distribution is
481   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 3) is replaced by a diagonal matrix $N \times N$, i.e.:

$$\tilde{\tilde{\mathbf{A}}} = 2\,\pi\,\gamma\,\mathbf{\Delta S^{-1}}\,, \tag{48}$$

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:

$$\tilde{\tilde{\mathbf{A}}}^{\top}\,\tilde{\tilde{\mathbf{A}}}\mathbf{\Delta}\,\hat{\mathbf{p}}^{k}\ =\ \tilde{\tilde{\mathbf{A}}}^{\top}\,\mathbf{r}^{k}\,, \tag{49}$$

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.,

$$r_i^k = d_i^o\ -\ d_i^k\,. \tag{50}$$

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

Because $\tilde{\tilde{\mathbf{A}}}$, in equation 49, is a diagonal matrix (equation 48), the parameter correction estimate is directly calculated without solving system of linear equations, and thus, an ith element of $\mathbf{\Delta}\,\hat{\mathbf{p}}^{k}$ is directly calculated by

$$\Delta\hat{p}^k{}_i = \frac{\Delta s_i\,r_i^k}{2\,\pi\,\gamma}\,. \tag{51}$$

The mass distribution over the equivalent layer is updated by:

$$\hat{p}_i^{k+1} = \hat{p}_i^{k}\ +\ \Delta\hat{p}^k{}_i\,. \tag{52}$$

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.,

$$p_i^o = \frac{\Delta s_i\,d_i^o}{2\,\pi\,\gamma}\,. \tag{53}$$

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

508   mask matrix $\mathbf{M}$ is defined in the following way: if the corresponding pixel does not contain the original
509   data, the element of $\mathbf{M}$ is set to zero; otherwise, it is set to one. The kth Landweber iteration is given by

$$\mathbf{p}_{k+1} = \mathbf{p}_k + \omega \left[\mathbf{A_1}^\top (\mathbf{d_1} - \mathbf{M}\mathbf{A_1}\mathbf{p}_k) + \mathbf{A_2}^\top (\mathbf{d_2} - \mathbf{M}\mathbf{A_2}\mathbf{p}_k) \right] , \tag{54}$$

510   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
511   corresponding gravity gradient kernels. Jirigalatu and Ebbing (2019) applied their method for processing
512   two horizontal curvature components of Falcon airborne gravity gradient.

513 4.0.5 The convolutional equivalent layer with BTTB matrices

514   **?** (**?**, **?**) introduced the convolutional equivalent layer for gravimetric and magnetic data processing,
515   respectively.

516   **?** demonstrated that the sensitivity matrix $\mathbf{A}$ (equation 3) associated with a planar equivalent layer formed
517   by a set of point masses, each one directly beneath each observation point and considering a regular grid of
518   observation points at a constant height has a symmetric block-Toeplitz Toeplitz-block (BTTB) structure. A
519   symmetric BTTB matrix has, at least, two attractive properties. The first one is that it can be defined by
520   using only the elements forming its first column (or row). The second attractive property is that any BTTB
521   matrix can be embedded into a symmetric Block-Circulat Circulant-Block (BCCB) matrix. This means that
522   the full sensitivity matrix $\mathbf{A}$ (equation 3) can be completely reconstruct by using the first column of the
523   BCCB matrix only. In what follows, **?** computed the forward modeling by using only a single equivalent
524   source. Specifically, it is done by calculating the eigenvalues of the BCCB matrix that can be efficiently
525   computed by using only the first column of the BCCB matrix via 2D fast Fourier transform (2D FFT).
526   By comparing with the classic approach in the Fourier domain, the convolutional equivalent layer for
527   gravimetric data processing proposed by **?** performed upward- and downward-continue gravity data with a
528   very small border effects and noise amplification.

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

540   The matrix-vector product in **?** (**?**, **?**) (e.g., $\mathbf{d} = \mathbf{A}\mathbf{p}$, such as in equation 3) is the main issue to be solved.
541   To solve it efficiently, these authors involked the auxiliary linear system

$$\mathbf{w} = \mathbf{C}\mathbf{v} , \tag{55}$$

542   where $\mathbf{w}$ and $\mathbf{v}$ are, respectively, vectors of data and parameters completed by zeros and $\mathbf{C}$ is a BCCB
543   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.
544   The first column of $\mathbf{C}$ is obtained by rearranging the first column of the sensitivity matrix $\mathbf{A}$ (equation 3).

Because a BCCB matrix is diagonalized by the 2D unitary discrete Fourier transform (DFT), $\mathbf{C}$ can be written as

$$\mathbf{C} = \left(\mathbf{F}_{2Q} \otimes \mathbf{F}_{2P}\right)^{*} \mathbf{\Lambda} \left(\mathbf{F}_{2Q} \otimes \mathbf{F}_{2P}\right) , \tag{56}$$

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 $\mathbf{\Lambda}$ 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 55) can be rewritten by using equation 56 and premultiplying both sides of the result by $\left(\mathbf{F}_{2Q} \otimes \mathbf{F}_{2P}\right)$, i.e.,

$$\mathbf{\Lambda} \left(\mathbf{F}_{2Q} \otimes \mathbf{F}_{2P}\right) \mathbf{v} = \left(\mathbf{F}_{2Q} \otimes \mathbf{F}_{2P}\right) \mathbf{w} . \tag{57}$$

By applying the vec-operator (**?**) to both sides of equation 57, by premultiplying both sides of the result by $\mathbf{F}_{2Q}^{*}$ and then postmultiplying both sides of the result by $\mathbf{F}_{2P}^{*}$

$$\mathbf{F}_{2Q}^{*} \left[\mathbf{L} \circ \left(\mathbf{F}_{2Q} \mathbf{V} \mathbf{F}_{2P}\right)\right] \mathbf{F}_{2P}^{*} = \mathbf{W} , \tag{58}$$

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 58 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 56) and can be efficiently computed by using only the first column of the BCCB matrix $\mathbf{C}$ (equation 55).

Actually, in **?** (**?**, **?**) a fast 2D discrete circular convolution (Van Loan, 1992) 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, **?**'s (**?**) method produces smaller border effects without using any padding scheme.

Without taking advantage of the symmetric BTTB structure of the sensitivity matrix (**?**, **?**) that arises when gravimetric observations are measured on a horizontally regular grid, on a flat surface and considering a regular grid of equivalent sources whitin a horizontal layer, **?** explored the symmetry of the gravity kernel to reduce the number of forward model evaluations. By exploiting 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, **?** estimated the mass excess or deficiency produced by anomalous sources with positive or negative density contrast.

4.0.6 The deconvolutional equivalent layer with BTTB matrices

To avoid the iterations of the conjugate gradient method in **?**, we can employ the deconvolution process. Equation 58 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 58, the matrix $\mathbf{V}$ can be obtain by deconvolution, i.e.

$$\mathbf{V} = \mathbf{F}_{2Q}^{*} \left[\frac{\left(\mathbf{F}_{2Q} \mathbf{W} \mathbf{F}_{2P}\right)}{\mathbf{L}}\right] \mathbf{F}_{2P}^{*} . \tag{59}$$

Equation 59 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 59 requires regularization to be useful. We usede wiener deconvolution to obtain a stable solution, i.e.,

$$\mathbf{V} \;=\; \mathbf{F}_{2Q}^{*} \left[\left(\mathbf{F}_{2Q}\mathbf{W}\,\mathbf{F}_{2P} \right) \frac{\mathbf{L}^{*}}{\left(\mathbf{L}\,\mathbf{L}^{*} + \mu \right)} \right] \mathbf{F}_{2P}^{*} \,, \tag{60}$$

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

4.1 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 $\mathbf{d}_{\ell}^{\mathbf{o}}$, $\ell = 1, ..., D$. From each $\mathbf{d}_{\ell}^{\mathbf{o}}$, we estimate a physical-property distribution $\hat{\mathbf{p}}_{\ell}$ within the equivalent layer.

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

$$\delta p_{\ell} = \frac{\parallel \hat{\mathbf{p}}_{\ell} - \mathbf{p} \parallel_{2}}{\parallel \mathbf{p} \parallel_{2}}, \quad \ell = 1, ..., D, \tag{61}$$

and

$$\delta d_{\ell} = \frac{\parallel \mathbf{d}_{\ell}^{\mathbf{o}} - \mathbf{d} \parallel_{2}}{\parallel \mathbf{d} \parallel_{2}}, \quad \ell = 1, ..., D. \tag{62}$$

Regardless of the particular method used, the following inequality (**?**, p. 66) is applicable:

$$\delta p_{\ell} \leq \kappa \, \delta d_{\ell}, \quad \ell = 1, ..., D, \tag{63}$$

where κ is the constant of proportionality between the model perturbation δp_{ℓ} (equation 61) and the data perturbation δd_{ℓ} (equation 62). 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 63 shows a linear relationship between the model perturbation and the data perturbation. By plotting δp_{ℓ} (equation 61) against δd_{ℓ} (equation 62) 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 63. By applying a linear regression, we obtain a fitted straight line whose estimated slope (κ in equation 63) 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 59) and the deconvolutional method with different values for the Wiener stabilization (equation 60).

5 NUMERICAL SIMULATIONS

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

5.1 Floating-point operations calculation

615 To measure the computational effort of the different algorithms to solve the equivalent layer linear system,
616 a non-hardware dependent method can be useful because allow us to do direct comparison between them.
617 Counting the floating-point operations (*flops*), i.e., additions, subtractions, multiplications and divisions
618 is a good way to quantify the amount of work of a given algorithm (Golub and Van Loan, 2013). For
619 example, the number of *flops* necessary to multiply two vectors $\mathbb{R}^N$ is $2N$. A common matrix-vector
620 multiplication with dimension $\mathbb{R}^{N \times N}$ and $\mathbb{R}^N$, respectively, is $2N^2$ and a multiplication of two matrices
621 $\mathbb{R}^{N \times N}$ is $2N^3$. Figure **??** shows the total flops count for the different methods presented in this review
622 with a crescent number of data, ranging from $10,000$ to $1,000,000$ for the gravity equivalent layer and
623 figure **??** for magnetic data.

5.1.1 Normal equations using Cholesky decomposition

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

$$f_{classical} = \frac{5}{6}N^3 + 4N^2 \tag{64}$$

5.1.2 Window method (Leão and Silva, 1989)

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

$$f_{window} = N\frac{5}{6}M_wN_w^2 + 4N_wM_w \tag{65}$$

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

636 The polynomial equivalent layer uses a simliar approach od moving windows from Leão and Silva (1989).
637 For this operations calculation (equation 47) we used a first degree polynomial (two variables) and each
638 window contains $N_s = 1,000$ observed data and $M_s = 1,000$ equivalent sources. Following the steps
639 given in (Oliveira Jr. et al., 2013) the total *flops* becomes

$$f_{pel} = \frac{1}{3}H^3 + 2H^2 + 2NM_sH + H^2N + 2HN + 2NP \tag{66}$$

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

5.1.4 Conjugate gradient least square (CGLS)

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

$$f_{cgls} = 2N^2 + it(4N^2 + 12N) \tag{67}$$

5.1.5 Wavelet compression method with CGLS (?)

647 For the wavelet method (equation 40) we have calculated a coompression rate of 98% ($C_r = 0.02$)
648 for the threshold as the authors used in **?** and the wavelet transformation requiring $\log_2(N)$ *flops* each
649 (equations 39 and 41c), with its inverse also using the same number of operations (equation 43). Combined
650 with the conjugate gradient least square necessary steps and iterations, the number of *flops* are

$$f_{wavelet} = 2NC_r + 4N\log_2(N) + it(4N\log_2(N) + 4NC_r + 12C_r) \tag{68}$$

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

652 The fast equivalent layer from Siqueira et al. (2017) solves the linear system in *it* iterations. The main
653 cost of this method (equations 49,50, 51 and 52)is the matrix-vector multiplication to asses the predicted
654 data ($2N^2$) and three simply element by element vector sum, subtraction and division ($3N$ total)

$$f_{siqueira} = it(3N + 2N^2) \tag{69}$$

5.1.7 Convolutional equivalent layer for gravity data (?)

656 This methods replaces the matrix-vector multiplication of the iterative fast-equivalent technique (Siqueira
657 et al., 2017) by three steps, involving a Fourier transform, an inverse Fourier transform, and a Hadamard
658 product of matrices (equation 58). Considering that the first column of our BCCB matrix has $4N$ elements,
659 the flops count of this method is

$$f_{convgrav} = \kappa 4N\log_2(4N) + it(27N + \kappa 8N\log_2(4N)) \tag{70}$$

660 In the resultant count we considered a *radix-2* algorithm for the fast Fourier transform and its inverse,
661 which has a κ equals to 5 and requires $\kappa 4N\log_2(4N)$ flops each. The Hadarmard product of two matrices
662 of $4N$ elements with complex numbers takes $24N$ flops. Note that equation 70 is different from the one
663 presented in **?** because we also added the flops necessary to calculate the eigenvalues in this form. It does
664 not differentiate much in order of magnitude because the iterative part is the most costful.

665 5.1.8 Convolutional equivalent layer for magnetic data (**?**)

666   The convolutional equivalent layer for magnetic data uses the same flops count of the main operations as
667 in the gravimetric case (equation 58), the difference is the use of the conjugate gradient algorithm to solve
668 the inverse problem. It requires a Hadamard product outside of the iterative loop and the matrix-vector and
669 vector-vector multiplications inside the loop as seem in equation 67.

$$f_{convmag} = \kappa 16N \log_2(4N) + 24N + it(\kappa 16N \log_2(4N) + 60N) \tag{71}$$

670 5.1.9 Deconvolutional method

671   The deconvolution method does not require an iterative algorithm, rather it solves the estimative of the
672 physical properties in a single step using the $4N$ eigenvalues of the BCCB matrix as in the convolutional
673 method. From equation 59 it is possible to deduce this method requires two fast Fourier transform
674 $(\kappa 4N \log_2(4N))$, one for the eigenvalues and another for the data transformation, a element by element
675 division $(24N)$ and finally, a fast inverse Fourier transform for the final estimative $(\kappa 4N \log_2(4N))$.

$$f_{deconv} = \kappa 12N \log_2(4N) + 24N \tag{72}$$

676   Using the deconvolutional method with a Wiener stabilization adds two multiplications of complex
677 elements of the conjugates eigenvalues $(24N$ each$)$ and the sum of $4N$ elements with the stabilization
678 parameter μ as shown in equation 60

$$f_{deconvwiener} = \kappa 12N \log_2(4N) + 76N \tag{73}$$

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

ACKNOWLEDGMENTS

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. (2019). *Parameter Estimation and Inverse Problems* (Elsevier), 3 edn.

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

Blakely, R. J. (1996). *Potential Theory in Gravity and Magnetic Applications* (Cambridge University press)

Bott, M. H. P. (1960). The use of Rapid Digital Computing Methods for Direct Gravity Interpretation of Sedimentary Basins. *Geophysical Journal International* 3, 63–67. doi:10.1111/j.1365-246X.1960.tb00065.x

Chan, R. H.-F. and Jin, X.-Q. (2007). *An introduction to iterative Toeplitz solvers*, vol. 5 (SIAM)

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

708 Dampney, C. N. G. (1969). The equivalent source technique. *GEOPHYSICS* 34, 39–53. doi:10.1190/1.
709   1439996

710 Emilia, D. A. (1973). Equivalent sources used as an analytic base for processing total magnetic field
711   profiles. *GEOPHYSICS* 38, 339–348. doi:10.1190/1.1440344

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

714 Gonzalez, S. P., Barbosa, V. C. F., and Oliveira Jr., V. C. (2022). Analyzing the ambiguity of the remanent-
715   magnetization direction separated into induced and remanent magnetic sources. *Journal of Geophysical*
716   *Research: Solid Earth* 127, 1–24. doi:10.1029/2022JB024151

717 Guspí, F., Introcaso, A., and Introcaso, B. (2004). Gravity-enhanced representation of measured geoid
718   undulations using equivalent sources. *Geophysical Journal International* 159, 1–8. doi:10.1111/j.
719   1365-246X.2004.02364.x

720 Guspí, F. and Novara, I. (2009). Reduction to the pole and transformations of scattered magnetic data using
721   Newtonian equivalent sources. *GEOPHYSICS* 74, L67–L73. doi:10.1190/1.3170690

722 Hansen, R. O. and Miyazaki, Y. (1984). Continuation of potential fields between arbitrary surfaces.
723   *GEOPHYSICS* 49, 787–795. doi:10.1190/1.1441707

724 Horn, R. A. and Johnson, C. R. (1991). *Topics in Matrix Analysis* (Cambridge University Press), 1 edn.

725 Jain, A. K. (1989). *Fundamentals of Digital Image Processing* (Pearson), 1 edn.

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

728 Kellogg, O. D. (1967). *Foundations of Potential Theory* (Springer-Verlag), reprint from the first edition of
729   1929 edn.

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

732 Leão, J. W. D. and Silva, J. B. C. (1989). Discrete linear transformations of potential field data. *Geophysics*
733   54, 497–507. doi:10.1190/1.1442676

734 Li, Y., Nabighian, M., and Oldenburg, D. W. (2014). Using an equivalent source with positivity for low-
735   latitude reduction to the pole without striation. *GEOPHYSICS* 79, J81–J90. doi:10.1190/geo2014-0134.
736   1

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

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

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

743 Menke, W. (2018). *Geophysical data analysis: Discrete inverse theory* (Elsevier), 4 edn.

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

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

748 Reis, A. L. A., Oliveira Jr., V. C., and Barbosa, V. C. F. (2020). Generalized positivity constraint on
749   magnetic equivalent layers. *Geophysics* 85, 1–45. doi:10.1190/geo2019-0706.1

750 Roy, A. (1962). Ambiguity in geophysical interpretation. *GEOPHYSICS* 27, 90–99. doi:10.1190/1.
751   1438985

752 Silva, J. B. C. (1986). Reduction to the pole as an inverse problem and its application to low-latitude
753   anomalies. *GEOPHYSICS* 51, 369–382. doi:10.1190/1.1442096
754 Siqueira, F., Oliveira Jr., V. C., and Barbosa, V. C. F. (2017). Fast iterative equivalent-layer technique for
755   gravity data processing: A method grounded on excess mass constraint. *GEOPHYSICS* 82, G57–G69.
756   doi:10.1190/GEO2016-0332.1
757 Skilling, J. and Bryan, R. (1984). Maximum entropy image reconstruction-general algorithm. *Monthly*
758   *Notices of the Royal Astronomical Society, Vol. 211, NO. 1, P. 111, 1984* 211, 111
759 Soler, S. R. and Uieda, L. (2021). Gradient-boosted equivalent sources. *Geophysical Journal International*
760   227, 1768–1783. doi:10.1093/gji/ggab297
761 Takahashi, D., Oliveira Jr., V. C., and Barbosa, V. C. (2022). Convolutional equivalent layer for magnetic
762   data processing. *Geophysics* 87, 1–59
763 Takahashi, D., Oliveira Jr., V. C., and Barbosa, V. C. F. (2020). Convolutional equivalent layer for gravity
764   data processing. *GEOPHYSICS* 85, G129–G141. doi:10.1190/geo2019-0826.1
765 van der Sluis, A. and van der Vorst, H. A. (1987). Numerical solution of large, sparse linear algebraic
766   systems arising from tomographic problems. In *Seismic tomography with applications in global*
767   *seismology and exploration geophysics*, ed. G. Nolet (D. Reidel Publishing Company), chap. 3. 49–83
768 Van Loan, C. F. (1992). *Computational Frameworks for the fast Fourier transform*. Frontiers in Applied
769   Mathematics (SIAM)
770 Xia, J. and Sprowl, D. R. (1991). Correction of topographic distortion in gravity data. *Geophysics* 56,
771   537–541
772 Xia, J., Sprowl, D. R., and Adkins-Heljeson, D. (1993). Correction of topographic distortions in potential-
773   field data; a fast and accurate approach. *Geophysics* 58, 515–523. doi:10.1190/1.1443434
774 Zhao, G., Chen, B., Chen, L., Liu, J., and Ren, Z. (2018). High-accuracy 3D Fourier forward modeling
775   of gravity field based on the Gauss-FFT technique. *Journal of Applied Geophysics* 150, 294–303.
776   doi:10.1016/j.jappgeo.2018.01.002
777 Zidarov, D. (1965). Solution of some inverse problems of applied geophysics. *Geophysical Prospecting*
778   13, 240–246. doi:10.1111/j.1365-2478.1965.tb01932.x