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 $i \in \{1 : D\}$. Consider that d_i can be satisfactorily approximated by a harmonic function

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

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

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

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

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

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

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

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

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

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

21 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

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

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

36 CRITÉRIOS PARA DEFINIR A PROFUNDIDADE DA CAMADA: DAMPNEY (ESPAÇAMENTO
37 DO GRID) E REIS (ESPAÇAMENTO DAS LINHAS)

1.2 Matrix G

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

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

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

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

42 and

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

43 In this case, the equivalent layer is formed by punctual sources representing monopoles or dipoles (e.g.,
44 Dampney, 1969; Emilia, 1973; Leão and Silva, 1989; Cordell, 1992; Oliveira Jr. et al., 2013; Siqueira et al.,
45 2017; Reis et al., 2020; Takahashi et al., 2020; Soler and Uieda, 2021; Takahashi et al., 2022). Another
46 common approach consists in not defining g_{ij} by using equations 6–8, but other harmonic functions
47 obtained by integrating them over the volume of regular prisms (e.g., Li and Oldenburg, 2010; Barnes and
48 Lumley, 2011; Li et al., 2014; Jirigalatu and Ebbing, 2019). There are also some less common approaches
49 defining the harmonic function g_{ij} (equation 2) as the potential field due to plane faces with constant
50 physical property (Hansen and Miyazaki, 1984), doublets (Silva, 1986) or by computing the double
51 integration of the inverse distance function with respect to z (Guspí and Novara, 2009).

52 A common assumption for most of the equivalent-layer methods is that the harmonic function g_{ij}
53 (equation 2) is independent on the actual physical relationship between the observed potential field and
54 their true sources (e.g., Cordell, 1992; Guspí and Novara, 2009; Li et al., 2014). Hence, g_{ij} can be
55 defined according to the problem. The only condition imposed to this function is that it decays to zero
56 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.
57 However, several methods use a function g_{ij} that preserves the physical relationship between the observed
58 potential field and their true sources. For the case in which the observed potential field is gravity data, g_{ij}
59 is commonly defined as a component of the gravitational field produced at (x_i, y_i, z_i) by a point mass or
60 prism located at (x_j, y_j, z_j), with unit density. On the other hand, g_{ij} is commonly defined as a component

61 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
62 magnetization intensity, when the observed potential field is magnetic data.

63 For all harmonic functions discussed above, the sensitivity matrix $\mathbf{G}$ (equation 3) is always dense. For
64 scattered potential-field data, $\mathbf{G}$ does not have a well-defined structure, regardless of whether the spatial
65 distribution of the equivalent sources is set. Nevertheless, for the particular case in which (i) there is a
66 single equivalent source right below each potential-field datum and (ii) both data and sources rely on
67 planar and regularly spaced grids, Takahashi et al. (2020, 2022) show that $\mathbf{G}$ assumes a block-Toeplitz
68 Toeplitz-block (BTTB) structure. In this case, the product of $\mathbf{G}$ and an arbitrary vector can be efficiently
69 computed via 2D fast Fourier transform as a discrete convolution.

1.3 General formulation

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

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

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

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

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

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

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

80 which is a combination of particular measures of length given by

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

81 and

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

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

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

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

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

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

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

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

89 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
90 subtracting the term $\left(\mathbf{H}^\top\mathbf{G}^\top\mathbf{W}_d\mathbf{G}\,\mathbf{H}\right)\bar{\mathbf{q}}$, we obtain

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

91 where

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

92

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

93

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

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

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

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

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

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

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

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

103 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
104 equation 18.

2 COMPUTATIONAL STRATEGIES

105 COMEÇAR EXPLICANDO AS LIMITAÇÕES DA SOLUÇÃO OBTIDA PELA FORMULAÇÃO
106 GERAL

107   Two important factors affecting the efficiency of a given matrix algorithm are the storage and amount of
108 required arithmetic. Here, we quantify this last factor by counting flops. A flop is a floating point addition,
109 subtraction, multiplication or division (Golub and Van Loan, 2013, p. 12–14).

110   To investigate the efficiency of equivalent-layer methods, we consider how they:

111   (i) set up the linear system (equations 22 and 23);

112 (ii) solve the linear system (equations 22 and 23);

113 (iii) perform potential-field transformations (equation 4).

114 We focus on the overall strategies used by the selected methods.

115 2.1 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 $\mathbf{d}[\mathbf{i}]$, where $\mathbf{i}$ is a list of integer numbers that "pick out" the elements of $\mathbf{d}$ forming the subvector $\mathbf{d}[\mathbf{i}]$. For example, $\mathbf{i} = (1, 6, 4, 6)$ gives the subvector $\mathbf{d}[\mathbf{i}] = [d_1 \ d_6 \ d_4 \ d_6]^\top$. 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,

$$\mathbf{i} = (3:8) \Leftrightarrow \mathbf{d}[\mathbf{i}] = [d_3 \ d_4 \ \ldots \ d_8]^\top$$
$$\mathbf{i} = (:8) \Leftrightarrow \mathbf{d}[\mathbf{i}] = [d_1 \ d_2 \ \ldots \ d_7]^\top \quad ,$$
$$\mathbf{i} = (3:) \Leftrightarrow \mathbf{d}[\mathbf{i}] = [d_3 \ d_4 \ \ldots \ d_D]^\top$$

116 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

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

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 $\mathbf{G}[\mathbf{i}, \mathbf{j}]$. 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:

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

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

118 2.2 Moving window

119   The initial approach to enhance the computational efficiency of the equivalent-layer technique is
120 commonly denoted *moving window* and involves first splitting the observed data d_i, $i \in \{1 : D\}$, into
121 M overlapping subsets (or data windows) formed by D^m data each, $m \in \{1 : M\}$. The data inside the
122 m-th window are usually adjacent to each other and have indices defined by an integer list $\mathbf{i}^m$ having

123   D^m elements. The number of data D^m forming the data windows are not necessarily equal to each other.
124   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
125   a set of P equivalent sources with scalar physical property p_j, $j \in \{1 : P\}$, and also split them into M
126   overlapping subsets (or source windows) formed by P^m data each, $m \in \{1 : M\}$. The sources inside the
127   m-th window have indices defined by an integer list $\mathbf{j}^m$ having P^m elements. Each source window has a
128   $P^m \times 1$ parameter vector $\mathbf{p}^m$ and is located right below the corresponding m-th data window. Then, each
129   $\mathbf{d}^m \equiv \mathbf{d}[\mathbf{i}^m]$ is approximated by

$$\mathbf{f}^m = \mathbf{G}^m \mathbf{p}^m, \tag{24}$$

130   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
131   2 using only the data and equivalent sources located inside the window m-th. The main idea of the moving-
132   window approach is using the $\tilde{\mathbf{p}}^m$ estimated for each window to obtain (i) an estimate $\tilde{\mathbf{p}}$ of the parameter
133   vector for the entire equivalent layer or (ii) a given potential-field transformation $\mathbf{t}$ (equation 4). The main
134   advantages of this approach is that (i) the estimated parameter vector $\tilde{\mathbf{p}}$ or transformed potential field are
135   not obtained by solving the full, but smaller linear systems and (ii) the full matrix $\mathbf{G}$ (equation 3) is never
136   stored.

137   Leão and Silva (1989) presented a pioneer work using the moving-window approach. Their method
138   requires a regularly-spaced grid of observed data on a horizontal plane z_0. The data windows are defined by
139   square local grids of $\sqrt{D'} \times \sqrt{D'}$ adjacent points, all of them having the same number of points D'. The
140   equivalent sources in the m-th data window are located below the observation plane, at a constant vertical
141   distance Δz_0. They are arranged on a regular grid of $\sqrt{P'} \times \sqrt{P'}$ adjacent points following the same
142   grid pattern of the observed data. The local grid of sources for all data windows have the same number
143   of elements P'. Besides, they are vertically aligned, but expands the limits of their corresponding data
144   windows, so that $D' < P'$. Because of this spatial configuration of observed data and equivalent sources,
145   we have that $\mathbf{G}^m = \mathbf{G}'$ (equation 24) for all data windows (i.e., $\forall\, m \in \{1 : M\}$), where $\mathbf{G}'$ is a $D' \times P'$
146   constant matrix.

147   By omitting the normalization strategy used by Leão and Silva (1989), their method consists in directly
148   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
149   as follows:

$$t_c^m = \left(\mathbf{G}'\mathbf{a}'\right)^\top \left[\mathbf{G}'\left(\mathbf{G}'\right)^\top + \mu\,\mathbf{I}_{D'}\right]^{-1} \mathbf{d}^m, \quad m \in \{1 : M\}, \tag{25}$$

150   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
151   5 by using all equivalent sources in the m-th subset and only the coordinate of the central point in the
152   m-th data window. Due to the presumed spatial configuration of the observed data and equivalent sources,
153   $\mathbf{a}'$ is the same for all data windows. Note that equation 25 combines the potential-field transformation
154   (equation 4) with the solution of the undetermined problem (equation 23) for the particular case in which
155   $\mathbf{H} = \mathbf{W}_q = \mathbf{I}_{P'}$ (equations 9 and 13), $\mathbf{W}_d = \mathbf{I}_{D'}$ (equation 12), $\bar{p} = \mathbf{0}$ (equation 14), where $\mathbf{I}_{P'}$ and $\mathbf{I}_{D'}$
156   are identity matrices of order P' and D', respectively, and $\mathbf{0}$ is a vector of zeros.

157   The method proposed by Leão and Silva (1989) can be outlined by the Algorithm 1. Note that Leão and
158   Silva (1989) directly compute the transformed potential t_c^m at the central point of each data window without
159   explicitly computing and storing an estimated for $\mathbf{p}^m$ (equation 24). It means that their method allows
160   computing a single potential-field transformation. A different transformation or the same one evaluated at
161   different points require running their moving-data window method again.

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 $(\mathbf{G}'\mathbf{a}')^\top \left[\mathbf{G}' (\mathbf{G}')^\top + \mu \, \mathbf{I}_{D'} \right]^{-1}$;
7 $m = 1$;
8 **while** $m < M$ **do**
9   $\quad$ Compute t_c^m (equation 25) ;
10   $\quad$ $m \leftarrow m + 1$;
11 **end**

---

162 Soler and Uieda (2021) generalized the method proposed by Leão and Silva (1989) for irregularly spaced
163 data on an undulating surface. A direct consequence of this generalization is that a different submatrix
164 $\mathbf{G}^m \equiv \mathbf{G}[\mathbf{i}^m, \mathbf{j}^m]$ (equation 24) must be computed for each window. Differently from Leão and Silva
165 (1989), Soler and Uieda (2021) store the computed $\tilde{\mathbf{p}}^m$ for all windows and subsequently use them to obtain
166 a desired potential-field transformation (equation 4) as the superposed effect of all windows. The estimated
167 $\tilde{\mathbf{p}}^m$ for all windows are combined to form a single $P \times 1$ vector $\tilde{\mathbf{p}}$, which is an estimate for original
168 parameter vector $\mathbf{p}$ (equation 3). For each data window, Soler and Uieda (2021) solve an overdetermined
169 problem (equation 22) for $\tilde{\mathbf{p}}^m$ by using $\mathbf{H} = \mathbf{W}_q = \mathbf{I}_{Pm}$ (equations 9 and 13), $\mathbf{W}_d^m$ (equation 12) equal to
170 a diagonal matrix of weights for the data inside the m-th window and $\bar{p} = \mathbf{0}$ (equation 14), so that

$$\left[(\mathbf{G}^m)^\top \mathbf{W}_d^m \mathbf{G}^m + \mu \, \mathbf{I}_{P'} \right] \tilde{\mathbf{p}}^m = (\mathbf{G}^m)^\top \mathbf{W}_d^m \mathbf{d}^m \,. \tag{26}$$

171 The overall steps of their method are defined by the Algorithm 2. Note that Algorithm 2 starts with a
172 residuals vector $\mathbf{r}$ that is iteratively updated. At each iteration, the potential field predicted a source window
173 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   $\quad$ Set the matrix $\mathbf{W}_d^m$;
9   $\quad$ Compute the matrix $\mathbf{G}^m$;
10   $\quad$ Compute $\tilde{\mathbf{p}}^m$ (equation 26) ;
11   $\quad$ $\tilde{\mathbf{p}}[\mathbf{j}^m] \leftarrow \tilde{\mathbf{p}}[\mathbf{j}^m] + \tilde{\mathbf{p}}^m$;
12   $\quad$ $\mathbf{r} \leftarrow \mathbf{r} - \mathbf{G}[:, \mathbf{j}^m] \tilde{\mathbf{p}}^m$;
13   $\quad$ $m \leftarrow m + 1$;
14 **end**

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

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

2.4 Row update

Mendonça and Silva (1994) proposes an algebraic reconstruction technique (ART) (e.g., van der Sluis and van der Vorst, 1987, p. 58) to estimate a parameter vector $\tilde{\mathbf{p}}$ for a regular grid of P equivalent sources on a horizontal plane z_0. Such methods iterate on the linear system rows to estimate corrections for the parameter vector, which may substantially save computer time and memory required to compute and store

199 the full linear system matrix along the iterations. The convergence of such *row-update methods* depends on
200 the linear system condition. The main advantage of such methods is not computing and storing the full
201 linear system matrix, but iteratively using its rows. The particular ART method proposed by Mendonça and
202 Silva (1994) considers that

$$\mathbf{d} = \begin{bmatrix} \mathbf{d}_e \\ \mathbf{d}_r \end{bmatrix}, \quad \mathbf{G} = \begin{bmatrix} \mathbf{G}_e \\ \mathbf{R}_r \end{bmatrix}, \tag{27}$$

203 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,
204 respectively. Mendonça and Silva (1994) designate $\mathbf{d}_e$ and $\mathbf{d}_r$ as, respectively, *equivalent* and *redundant*
205 data. With the exception of a normalization strategy, Mendonça and Silva (1994) calculate a $P \times 1$ estimated
206 parameter vector $\tilde{\mathbf{p}}$ by solving an underdetermined problem (equation 23) involving only the equivalent
207 data $\mathbf{d}_e$ (equation 27) for the particular case in which $\mathbf{H} = \mathbf{W}_p = \mathbf{I}_P$ (equations 9 and 13), $\mathbf{W}_d = \mathbf{I}_{D_e}$
208 (equation 12) and $\bar{p} = \mathbf{0}$ (equation 14), which results in

$$\begin{aligned} \left(\mathbf{F} + \mu\, \mathbf{I}_{D_e} \right) \mathbf{u} &= \mathbf{d}_e \\ \tilde{\mathbf{p}} &= \mathbf{G}_e^\top \mathbf{u} \end{aligned}, \tag{28}$$

209 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
210 parameter vector $\tilde{\mathbf{p}}$ obtained from equation 28 leads to a $D_r \times 1$ residuals vector

$$\mathbf{r} = \mathbf{d}_r - \mathbf{G}_r \tilde{\mathbf{p}} \tag{29}$$

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

212   The overall method of Mendonça and Silva (1994) is defined by Algorithm 4. It is important noting
213 that the number D_e of equivalent data in $\mathbf{d}_e$ increases by one per iteration, which means that the order
214 of the linear system in equation 28 also increases by one at each iteration. Those authors also propose
215 a computational strategy based on Cholesky factorization (e.g., Golub and Van Loan, 2013, p. 163) for
216 efficiently updating $\left(\mathbf{F} + \mu\, \mathbf{I}_{D_e} \right)$ at a given iteration (line 16 in Algorithm 4) by computing only its new
217 elements with respect to those computed in the previous iteration.

2.5 Reparameterization

219   Another approach for improving the computational performance of equivalent-layer technique consists
220 in setting a $P \times Q$ reparameterization matrix $\mathbf{H}$ (equation 9) with $Q << P$. This strategy has been used
221 in applied geophysics for decades (e.g., Skilling and Bryan, 1984; Kennett et al., 1988; Oldenburg et al.,
222 1993; Barbosa et al., 1997) and is known as *subspace method*. The main idea relies in reducing the linear
223 system dimension from the original P-space to a lower-dimensional subspace (the Q-space). An estimate
224 $\tilde{\mathbf{q}}$ for the reparameterized parameter vector $\mathbf{q}$ is obtained in the Q-space and subsequently used to obtain
225 an estimate $\tilde{\mathbf{p}}$ for the parameter vector $\mathbf{p}$ (equation 3) in the P-space by using equation 9. Hence, the key
226 aspect of this *reparameterization approach* is solving an appreciably smaller linear inverse problem for $\tilde{\mathbf{q}}$
227 than that for the original parameter vector $\tilde{\mathbf{p}}$ (equation 3).

228   Oliveira Jr. et al. (2013) have used this approach to describe the physical property distribution on the
229 equivalent layer in terms of piecewise bivariate polynomials. Specifically, their method consists in splitting
230 a regular grid of equivalent sources into source windows inside which the physical-property distribution
231 is described by bivariate polynomial functions. The key aspect of their method relies on the fact that the
232 total number of coefficients required to define the bivariate polynomials is considerably smaller than the

---

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 $r_{\texttt{max}}$ in $\mathbf{r}$;
5 Define the index $i_{\texttt{max}}$ of $r_{\texttt{max}}$;
6 Define the list of indices $\mathbf{i}_r$ of the remaining data in $\mathbf{r}$;
7 Define $\mathbf{d}_e = \mathbf{d}[i_{\texttt{max}}]$;
8 Compute $(\mathbf{F} + \mu \, \mathbf{I}_{D_e})$ and $\mathbf{G}_e$;
9 Compute $\tilde{\mathbf{p}}$ (equation 28) ;
10 Compute $\mathbf{r} = \mathbf{d}[\mathbf{i}_r] - \mathbf{G}[\mathbf{i}_r, :] \, \tilde{\mathbf{p}}$;
11 Define the maximum absolute value $r_{\texttt{max}}$ in $\mathbf{r}$;
12 **while** $(r_{\texttt{max}} > \epsilon)$ **do**
13   Define the index $i_{\texttt{max}}$ of $r_{\texttt{max}}$;
14   Define the list of indices $\mathbf{i}_r$ of the remaining elements in $\mathbf{r}$;
15     $\mathbf{d}_e \leftarrow \begin{bmatrix} \mathbf{d}_e \\ \mathbf{d}[i_{\texttt{max}}] \end{bmatrix}$;
16   Update $(\mathbf{F} + \mu \, \mathbf{I}_{D_e})$ and $\mathbf{G}_e$;
17   Update $\tilde{\mathbf{p}}$ (equation 28) ;
18   Update $\mathbf{r} = \mathbf{d}[\mathbf{i}_r] - \mathbf{G}[\mathbf{i}_r, :] \, \tilde{\mathbf{p}}$;
19   Define the maximum absolute value $r_{\texttt{max}}$ in $\mathbf{r}$;
20 **end**

---

233   original number of equivalent sources. Hence, they formulate a linear inverse problem for estimating the
234   polynomial coefficients and use them later to compute the physical property distribution on the equivalent
235   layer.

236   The method proposed by Oliveira Jr. et al. (2013) consists in solving an overdetermined problem (equation
237   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
238   that

$$\left(\mathbf{H}^\top \mathbf{G}^\top \mathbf{G} \, \mathbf{H} + \mu \, \mathbf{W}_q\right) \tilde{\mathbf{q}} = \mathbf{H}^\top \mathbf{G}^\top \, \mathbf{d} \,, \tag{30}$$

239   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
240   regularization (e.g., Aster et al., 2019, p. 103). Note that, in this case, the prior information is defined in the
241   P-space for the original parameter vector $\mathbf{p}$ and then transformed to the Q-space. Another characteristic of
242   their method is that it is valid for processing irregularly-spaced data on an undulating surface.

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

249   Before Oliveira Jr. et al. (2013) and Mendonça (2020), Barnes and Lumley (2011) also proposed a
250   computationally efficient method for equivalent-layer technique based on reparameterization. A key
251   difference, however, is that Barnes and Lumley (2011) did not set a $P \times Q$ reparameterization matrix $\mathbf{H}$
252   (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 solving the overdetermined problem for $\tilde{\mathbf{q}}$ (equation 30) via conjugate-gradient method (e.g., Golub and Van Loan, 2013, sec. 11.3).

2.6 Wavelet compression

Previously to Barnes and Lumley (2011), the idea of transforming the dense matrix $\mathbf{G}$ (equation 3) into a sparse one has already been used in the context of equivalent-layer technique. Li and Oldenburg (2010) proposed a method that applies the wavelet transform to the original dense matrix $\mathbf{G}$ and sets to zero the small coefficients that are below a given threshold, which results in an approximating sparse representation of $\mathbf{G}$ in the wavelet domain. Their method requires

$$\mathbf{d}_w \approx \mathbf{G}_w \, \mathbf{p}_w \, , \tag{31}$$

$$\begin{aligned} \mathbf{d}_w &= \mathcal{W} \, \mathbf{d} \\ \mathbf{p}_w &= \mathcal{W}^\top \mathbf{p} \\ \mathbf{G}_w &= \mathcal{W} \, \mathbf{G} \, \mathcal{W} \end{aligned} \tag{32}$$

$$\left(\mathbf{G}_s^\top \mathbf{G}_s + \mu \, \mathbf{W}_p \right) \tag{33}$$

PAREI AQUI

2.7 Iterative methods using the original $\mathbf{G}$

Ideia: descrever o conjugate gradient e depois as diferenças dos outros métodos iterativos - esta ideia é ruim porque vai ser necessario reescrever os sistemas com matrizes CCT e CTC. Isso necessita definir as matrizes Wq e Wd como WqTWq ou WqWqT, a depender do problema.

Xia and Sprowl (1991)

Xia et al. (1993)

Siqueira et al. (2017)

Jirigalatu and Ebbing (2019)

2.8 Discrete convolution

Takahashi et al. (2020)

Takahashi et al. (2022)

3 TEXTO ANTIGO

276 3.0.1 The wavelet compression and lower-dimensional subspace

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

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

$$\tilde{\mathbf{A}} = \mathbf{W_2} \, \mathbf{A} \, \mathbf{W_2}^\top , \tag{34}$$

284 where $\tilde{\mathbf{A}}$ is the expanded original sensitivity matrix in the wavelet bases with many elements zero or close
285 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
286 is obtained by retaining only the large elements of the $\tilde{\mathbf{A}}$. Thus, the elements of $\tilde{\mathbf{A}}$ whose amplitudes
287 fall below a relative threshold are discarded. In **?**, the original sensitivity matrix $\mathbf{A}$ is high compressed
288 resulting in a sparce matrix $\tilde{\mathbf{A}}_\mathbf{s}$ with a few percent of nonzero elements and the the inverse problem is
289 solved in the wavelet domain by using $\tilde{\mathbf{A}}_\mathbf{s}$ and a incomplete conjugate gradient least squares, without an
290 explicit regularization parameter and a limited number of iterations. The solution is obtained by solving the
291 following linear system

$$\tilde{\mathbf{A}}_\mathbf{L}^\top \, \tilde{\mathbf{A}}_\mathbf{L} \, \tilde{\mathbf{p}}_\mathbf{L}^* = \tilde{\mathbf{A}}_\mathbf{L}^\top \, \tilde{\mathbf{d}}^o , \tag{35}$$

292 where $\tilde{\mathbf{p}}_\mathbf{L}^*$ is obtained by solving the linear system given by equation 35,

$$\tilde{\mathbf{A}}_\mathbf{L} = \tilde{\mathbf{A}}_\mathbf{s} \, \tilde{\mathbf{L}}^{-1}, \tag{36a}$$

$$\tilde{\mathbf{p}}_\mathbf{L} = \tilde{\mathbf{L}}\tilde{\mathbf{p}}, \tag{36b}$$

$$\tilde{\mathbf{d}}^o = \mathbf{W_2} \, \mathbf{d}^o, \tag{36c}$$

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

$$\tilde{\mathbf{p}} = \tilde{\mathbf{L}}^{-1} \, \tilde{\mathbf{p}}_\mathbf{L}^* , \tag{37}$$

296 and

$$\mathbf{p} = \mathbf{W_2} \, \tilde{\mathbf{p}} . \tag{38}$$

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

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

305 **?** overcame the solution of intractable large-scale equivalent-layer problem by using the subspace method
306 (e.g., Skilling and Bryan, 1984; Kennett et al., 1988; Oldenburg et al., 1993; Barbosa et al., 1997). The
307 subspace method reduces the dimension of the linear system of equations to be solved. Given a higher-
308 dimensional space (e.g., M-dimensional model space, $\mathbb{R}^M$), there exists many lower-dimensional subspaces
309 (e.g., Q-dimensional subspace) of $\mathbb{R}^M$. The linear inverse problem related to the equivalent-layer technique
310 consists in finding an M-dimension parameter vector $\mathbf{p} \in \mathbb{R}^M$ which adequately fits the potential-field
311 data. The subspace method looks for a parameter vector who lies in a Q-dimensional subspace of $\mathbb{R}^M$
312 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
313 parameter vector in the subspace method can be written as

$$\mathbf{p} = \mathbf{V}\,\boldsymbol{\alpha}\,, \tag{39}$$

314 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 $\mathbb{R}^M$.
315 In equation 39, the parameter vector $\mathbf{p}$ is defined as a linear combination in the space spanned by Q basis
316 vectors $\mathbf{v}_i = 1, ..., Q$ and $\boldsymbol{\alpha}$ is a Q-dimensional unknown vector to be determined. The main advantage of
317 the subspace method is that the linear system of M equations in M unknowns to be originally solved is
318 reduced to a new linear system of Q equations in Q unknowns which requires much less computational
319 effort since $Q << M$, i.e.:

$$\mathbf{V}^\top \mathbf{A}^\top \mathbf{A} \mathbf{V} \boldsymbol{\alpha}^* = \mathbf{V}^\top \mathbf{d}^o\,. \tag{40}$$

320 To avoid the storage of matrices $\mathbf{A}$ and $\mathbf{V}$, **?** evaluates an element of the matrix $\mathbf{AV}$ by calculating the dot
321 product between the row of matrix $\mathbf{A}$ and the column of the matrix $\mathbf{B}$. After estimating $\boldsymbol{\alpha}^*$ (equation 40)
322 belonging to a Q-dimensional subspace of $\mathbb{R}^M$, the distribution over the equivalent layer $\mathbf{p}$ in the $\mathbb{R}^M$ is
323 obtained by applying equation 39. The choice of the Q basis vectors $\mathbf{v}_i = 1, ..., Q$ (equation 39) in the
324 subspace method is not strict. **?**, for example, chose the eigenvectors yielded by applying the singular value
325 decomposition of the matrix containing the gridded data set. The number of eigenvectors used to form
326 basis vectors will depend on the singular values.

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

3.0.2 The quadtree discretization

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

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

341 By using the quadtree discretization, **?** were able to jointly process multiple components of airborne
342 gravity-gradient data using a single layer of equivalent sources. To our knowledge, **?** are the pioneers on
343 processing full-tensor gravity-gradient data jointly. In addition to computational feasibility, **?**'s (**?**) method

344 reduces low-frequency noise and can also remove the drift in time-domain from the survey data. Those
345 authors stressed that the $G_{zz}-$component calculated through the single estimated equivalent-layer model
346 projected on a grid at a constant elevation by inverting full gravity-gradient data has the low-frequency error
347 reduced by a factor of 2.4 as compared to the inversion of an individual component of the gravity-gradient
348 data.

3.0.3   The reparametrization of the equivalent layer

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

$$\mathbf{p}^k = \mathbf{B}^k \, \mathbf{c}^k \,, \qquad k \,=\, 1 \dots Q \,, \tag{41}$$

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

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

$$\left(\mathbf{B}^\top \mathbf{A}^\top \mathbf{A} \mathbf{B} \,+\, \mu \mathbf{I}\right) \mathbf{c}^* \,=\, \mathbf{B}^\top \mathbf{A}^\top \mathbf{d}^o \,, \tag{42}$$

360 where μ is a regularizing parameter, $\mathbf{c}^*$ is an estimated H-dimensional vector containing all coefficients
361 describing all polynomial functions within all equivalent-source windows which compose the entire
362 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
363 such that the main-diagonal blocks are $\mathbf{B}^k$ matrices (equation 41) and all off-diagonal blocks are zero
364 matrices. For ease of the explanation of equation 42, we keep only the zeroth-order Tikhonov regularization
365 and omitting the first-order Tikhonov regularization (**?**) which was also used by Oliveira Jr. et al. (2013).

366   The main advantage of the PEL is solve H-dimensional system of equations (equation 42), where H
367 totalizes the number of polynomial coefficients composing all equivalent-source windows, requiring a
368 lower computational effort since $H <<< N$. To avoid the storage of matrices $\mathbf{A}$ and $\mathbf{B}$, Oliveira Jr. et al.
369 (2013) evaluate an element of the matrix $\mathbf{AB}$ by calculating the dot product between the row of matrix $\mathbf{A}$
370 and the column of the matrix $\mathbf{B}$. After estimating all polynomial coefficients of all windows, the estimated
371 coefficients ($\mathbf{c}^*$ in equation 42) are transformed into a single physical-property distribution encompassing
372 the entire equivalent layer.

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

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

3.0.4 The iterative scheme without solving a linear system

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

Following this class of methods of iterative equivalent-layer technique that does not solve linear systems, Siqueira et al. (2017) developed a fast iterative equivalent-layer technique for processing gravity data in which the sensitivity matrix $\mathbf{A}$ (equation 3) is replaced by a diagonal matrix $N \times N$, i.e.:

$$\tilde{\tilde{\mathbf{A}}} = 2\,\pi\,\gamma\,\mathbf{\Delta S^{-1}}\,,\tag{43}$$

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:

$$\tilde{\tilde{\mathbf{A}}}^{\top}\,\tilde{\tilde{\mathbf{A}}}\mathbf{\Delta}\,\hat{\mathbf{p}}^k \;=\; \tilde{\tilde{\mathbf{A}}}^{\top}\,\mathbf{r}^k\,,\tag{44}$$

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.,

$$r_i^k = d_i^o \,-\, d_i^k\,.\tag{45}$$

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

Because $\tilde{\tilde{\mathbf{A}}}$, in equation 44, is a diagonal matrix (equation 43), 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

$$\Delta \hat{p}^k_{\,i} = \frac{\Delta s_i\,r_i^k}{2\,\pi\,\gamma}\,.\tag{46}$$

The mass distribution over the equivalent layer is updated by:

$$\hat{p}_i^{k+1} = \hat{p}_i^k \,+\, \Delta \hat{p}^k_{\,i}\,.\tag{47}$$

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

$$p_i^o = \frac{\Delta s_i \, d_i^o}{2 \, \pi \, \gamma} \, . \tag{48}$$

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

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

$$\mathbf{p}_{k+1} = \mathbf{p}_k + \omega \left[\mathbf{A_1}^\top (\mathbf{d_1} - \mathbf{M A_1 p}_k) + \mathbf{A_2}^\top (\mathbf{d_2} - \mathbf{M A_2 p}_k) \right] \, , \tag{49}$$

417 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
418 corresponding gravity gradient kernels. Jirigalatu and Ebbing (2019) applied their method for processing
419 two horizontal curvature components of Falcon airborne gravity gradient.

3.0.5 The convolutional equivalent layer with BTTB matrices

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

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

436 By using the original idea of the convolutional equivalent layer proposed by **?** for gravimetric data
437 processing, **?** developed the convolutional equivalent layer for magnetic data processing. By assuming
438 a regularly spaced grid of magnetic data at a constant height and a planar equivalent layer of dipoles, **?**
439 proved that the sensitivity matrix linked with this layer possess a BTTB structure in the specific scenario
440 where each dipole is exactly beneath each observed magnetic data point. **?** used a conjugate gradient
441 least-squares (CGLS) algorithm which does not require an inverse matrix or matrix-matrix multiplication.
442 Rather, it only requires matrix-vector multiplications per iteration, which can be effectively computed using
443 the 2D FFT as a discrete convolution. The matrix-vector product only uses the elements that constitute the

444 first column of the associated BTTB matrix, resulting in computational time and memory savings. **?** (**?**)
445 showed the robustness of the convolutional equivalent layer in processing magnetic survey that violates the
446 requirement of regular grids in the horizontal directions and flat observation surfaces.

447   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.
448 To solve it efficiently, these authors involked the auxiliary linear system

$$\mathbf{w} = \mathbf{C}\mathbf{v} \,, \tag{50}$$

449 where $\mathbf{w}$ and $\mathbf{v}$ are, respectively, vectors of data and parameters completed by zeros and $\mathbf{C}$ is a BCCB
450 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.
451 The first column of $\mathbf{C}$ is obtained by rearranging the first column of the sensitivity matrix $\mathbf{A}$ (equation 3).
452 Because a BCCB matrix is diagonalized by the 2D unitary discrete Fourier transform (DFT), $\mathbf{C}$ can be
453 written as

$$\mathbf{C} = \left(\mathbf{F}_{2Q} \otimes \mathbf{F}_{2P}\right)^* \mathbf{\Lambda} \left(\mathbf{F}_{2Q} \otimes \mathbf{F}_{2P}\right) \,, \tag{51}$$

454 where the symbol "$\otimes$" denotes the Kronecker product (**?**), $\mathbf{F}_{2Q}$ and $\mathbf{F}_{2P}$ are the $2Q \times 2Q$ and $2P \times 2P$
455 unitary DFT matrices (**?**, p. 31), respectively, the superscritpt "$*$" denotes the complex conjugate and $\mathbf{\Lambda}$ is
456 a $4QP \times 4QP$ diagonal matrix containing the eigenvalues of $\mathbf{C}$. Due to the diagonalization of the matrix
457 $\mathbf{C}$, the auxiliary system (equation 50) can be rewritten by using equation 51 and premultiplying both sides
458 of the result by $\left(\mathbf{F}_{2Q} \otimes \mathbf{F}_{2P}\right)$, i.e.,

$$\mathbf{\Lambda} \left(\mathbf{F}_{2Q} \otimes \mathbf{F}_{2P}\right) \mathbf{v} = \left(\mathbf{F}_{2Q} \otimes \mathbf{F}_{2P}\right) \mathbf{w} \,. \tag{52}$$

459 By applying the vec-operator (**?**) to both sides of equation 52, by premultiplying both sides of the result by
460 $\mathbf{F}_{2Q}^*$ and then postmultiplying both sides of the result by $\mathbf{F}_{2P}^*$

$$\mathbf{F}_{2Q}^* \left[\mathbf{L} \circ \left(\mathbf{F}_{2Q} \, \mathbf{V} \, \mathbf{F}_{2P}\right)\right] \mathbf{F}_{2P}^* = \mathbf{W} \,, \tag{53}$$

461 where "$\circ$" denotes the Hadamard product (**?**, p. 298) and $\mathbf{L}$, $\mathbf{V}$ and $\mathbf{W}$ are $2Q \times 2P$ matrices obtained
462 by rearranging, along their rows, the elements forming the diagonal of matrix $\mathbf{\Lambda}$, vector $\mathbf{v}$ and vector $\mathbf{w}$,
463 respectively. The left side of equation 53 contains the 2D Inverse Discrete Fourier Transform (IDFT) of the
464 term in brackets, which in turn represents the Hadamard product of matrix $\mathbf{L}$ and the 2D DFT of matrix $\mathbf{V}$.
465 Matrix $\mathbf{L}$ contains the eigenvalues of $\mathbf{\Lambda}$ (equation 51) and can be efficiently computed by using only the
466 first column of the BCCB matrix $\mathbf{C}$ (equation 50).

467   Actually, in **?** (**?**, **?**) a fast 2D discrete circular convolution (Van Loan, 1992) is used to process very
468 large gravity and magnetic datasets efficiently. The convolutional equivalent layer was applied to perform
469 upward continuation of large magnetic datasets. Compared to the classical Fourier approach, **?**'s (**?**) method
470 produces smaller border effects without using any padding scheme.

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

478 ### 3.0.6 The deconvolutional equivalent layer with BTTB matrices

479 To avoid the iterations of the conjugate gradient method in **?**, we can employ the deconvolution process.
480 Equation 53 shows that estimate the matrix $\mathbf{V}$, containing the elements of parameter vector $\mathbf{p}$, is a
481 inverse problem that could be solved by deconvolution. From equation 53, the matrix $\mathbf{V}$ can be obtain by
482 deconvolution, i.e.

$$\mathbf{V} = \mathbf{F}_{2Q}^* \left[\frac{(\mathbf{F}_{2Q} \mathbf{W} \mathbf{F}_{2P})}{\mathbf{L}} \right] \mathbf{F}_{2P}^* . \tag{54}$$

483 Equation 54 shows that the parameter vector (in matrix $\mathbf{V}$) can be theoretically obtain by dividing each
484 potential-field observations (in matrix $\mathbf{W}$) by each eigenvalues (in matrix $\mathbf{L}$). Hence, the parameter vector
485 is constructed by element-by-element division of data by eigenvalues.

486 However, the deconvolution often is extremely unstable. This means that a small change in data can lead
487 to an enormous change in the estimated parameter. Hence, equation 54 requires regularization to be useful.
488 We usede wiener deconvolution to obtain a stable solution, i.e.,

$$\mathbf{V} = \mathbf{F}_{2Q}^* \left[(\mathbf{F}_{2Q} \mathbf{W} \mathbf{F}_{2P}) \frac{\mathbf{L}^*}{(\mathbf{L} \mathbf{L}^* + \mu)} \right] \mathbf{F}_{2P}^* , \tag{55}$$

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

491 ## 3.1 Solution stability

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

494 Let us assume noise-free potential-field data $\mathbf{d}$, we estimate a physical-property distribution $\mathbf{p}$ (estimated
495 solution) within the equivalent layer. Then, the noise-free data $\mathbf{d}$ are contaminated with additive D different
496 sequences of pseudorandom Gaussian noise, creating different noise-corrupted potential-field data $\mathbf{d}_\ell^o$,
497 $\ell = 1, ..., D$. From each $\mathbf{d}_\ell^o$, we estimate a physical-property distribution $\hat{\mathbf{p}}_\ell$ within the equivalent layer.

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

$$\delta p_\ell = \frac{\| \hat{\mathbf{p}}_\ell - \mathbf{p} \|_2}{\| \mathbf{p} \|_2}, \quad \ell = 1, ..., D, \tag{56}$$

500 and

$$\delta d_\ell = \frac{\| \mathbf{d}_\ell^o - \mathbf{d} \|_2}{\| \mathbf{d} \|_2}, \quad \ell = 1, ..., D. \tag{57}$$

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

$$\delta p_\ell \le \kappa \, \delta d_\ell, \quad \ell = 1, ..., D, \tag{58}$$

502 where κ is the constant of proportionality between the model perturbation δp_ℓ (equation 56) and the data
503 perturbation δd_ℓ (equation 57). The constant κ acts as the condition number of an invertible matrix in a
504 given inversion, and thus measures the instability of the solution. The larger (smaller) the value of κ the
505 more unstable (stable) is the estimated solution.

506   Equation 58 shows a linear relationship between the model perturbation and the data perturbation. By
507   plotting δp_ℓ (equation 56) against δd_ℓ (equation 57) produced by a set of D estimated solution obtained by
508   applying a given equivalent-layer method, we obtain a straight line behaviour described by equation 58.
509   By applying a linear regression, we obtain a fitted straight line whose estimated slope (κ in equation 58)
510   quantifies the solution stability.

511   Here, the analysis of solution stability is numerically conducted by applying the classical equivalent-
512   layer technique with zeroth-order Tikhonov regularization, the convolutional method for gravimetric and
513   magnetic data, the deconvolutional method (equation 54) and the deconvolutional method with different
514   values for the Wiener stabilization (equation 55).

4 NUMERICAL SIMULATIONS

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

4.1 Floating-point operations calculation

522 To measure the computational effort of the different algorithms to solve the equivalent layer linear system,
523 a non-hardware dependent method can be useful because allow us to do direct comparison between them.
524 Counting the floating-point operations (*flops*), i.e., additions, subtractions, multiplications and divisions
525 is a good way to quantify the amount of work of a given algorithm (Golub and Van Loan, 2013). For
526 example, the number of *flops* necessary to multiply two vectors $\mathbb{R}^N$ is $2N$. A common matrix-vector
527 multiplication with dimension $\mathbb{R}^{N \times N}$ and $\mathbb{R}^N$, respectively, is $2N^2$ and a multiplication of two matrices
528 $\mathbb{R}^{N \times N}$ is $2N^3$. Figure **??** shows the total flops count for the different methods presented in this review
529 with a crescent number of data, ranging from $10,000$ to $1,000,000$ for the gravity equivalent layer and
530 figure **??** for magnetic data.

4.1.1 Normal equations using Cholesky decomposition

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

$$f_{classical} = \frac{5}{6}N^3 + 4N^2 \tag{59}$$

4.1.2 Window method (Leão and Silva, 1989)

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

$$f_{window} = N\frac{5}{6}M_w N_w^2 + 4N_w M_w \tag{60}$$

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

543 The polynomial equivalent layer uses a simliar approach od moving windows from Leão and Silva (1989).
544 For this operations calculation (equation 42) we used a first degree polynomial (two variables) and each
545 window contains $N_s = 1,000$ observed data and $M_s = 1,000$ equivalent sources. Following the steps
546 given in (Oliveira Jr. et al., 2013) the total *flops* becomes

$$f_{pel} = \frac{1}{3}H^3 + 2H^2 + 2NM_sH + H^2N + 2HN + 2NP \tag{61}$$

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

4.1.4   Conjugate gradient least square (CGLS)

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

$$f_{cgls} = 2N^2 + it(4N^2 + 12N) \tag{62}$$

4.1.5   Wavelet compression method with CGLS (?)

554 For the wavelet method (equation 35) we have calculated a coompression rate of 98% ($C_r = 0.02$)
555 for the threshold as the authors used in **?** and the wavelet transformation requiring $\log_2(N)$ *flops* each
556 (equations 34 and 36c), with its inverse also using the same number of operations (equation 38). Combined
557 with the conjugate gradient least square necessary steps and iterations, the number of *flops* are

$$f_{wavelet} = 2NC_r + 4N\log_2(N) + it(4N\log_2(N) + 4NC_r + 12C_r) \tag{63}$$

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

559 The fast equivalent layer from Siqueira et al. (2017) solves the linear system in *it* iterations. The main
560 cost of this method (equations 44, 45, 46 and 47)is the matrix-vector multiplication to asses the predicted
561 data ($2N^2$) and three simply element by element vector sum, subtraction and division ($3N$ total)

$$f_{siqueira} = it(3N + 2N^2) \tag{64}$$

4.1.7   Convolutional equivalent layer for gravity data (?)

563 This methods replaces the matrix-vector multiplication of the iterative fast-equivalent technique (Siqueira
564 et al., 2017) by three steps, involving a Fourier transform, an inverse Fourier transform, and a Hadamard
565 product of matrices (equation 53). Considering that the first column of our BCCB matrix has $4N$ elements,
566 the flops count of this method is

$$f_{convgrav} = \kappa 4N\log_2(4N) + it(27N + \kappa 8N\log_2(4N)) \tag{65}$$

567 In the resultant count we considered a *radix-2* algorithm for the fast Fourier transform and its inverse,
568 which has a κ equals to 5 and requires $\kappa 4N\log_2(4N)$ flops each. The Hadarmard product of two matrices
569 of $4N$ elements with complex numbers takes $24N$ flops. Note that equation 65 is different from the one
570 presented in **?** because we also added the flops necessary to calculate the eigenvalues in this form. It does
571 not differentiate much in order of magnitude because the iterative part is the most costful.

4.1.8 Convolutional equivalent layer for magnetic data (**?**)

The convolutional equivalent layer for magnetic data uses the same flops count of the main operations as in the gravimetric case (equation 53), the difference is the use of the conjugate gradient algorithm to solve the inverse problem. It requires a Hadamard product outside of the iterative loop and the matrix-vector and vector-vector multiplications inside the loop as seem in equation 62.

$$f_{convmag} = \kappa 16N \log_2(4N) + 24N + it(\kappa 16N \log_2(4N) + 60N) \tag{66}$$

4.1.9 Deconvolutional method

The deconvolution method does not require an iterative algorithm, rather it solves the estimative of the physical properties in a single step using the $4N$ eigenvalues of the BCCB matrix as in the convolutional method. From equation 54 it is possible to deduce this method requires two fast Fourier transform ($\kappa 4N \log_2(4N)$), one for the eigenvalues and another for the data transformation, a element by element division ($24N$) and finally, a fast inverse Fourier transform for the final estimative ($\kappa 4N \log_2(4N)$).

$$f_{deconv} = \kappa 12N \log_2(4N) + 24N \tag{67}$$

Using the deconvolutional method with a Wiener stabilization adds two multiplications of complex elements of the conjugates eigenvalues ($24N$ each) and the sum of $4N$ elements with the stabilization parameter μ as shown in equation 55

$$f_{deconvwiener} = \kappa 12N \log_2(4N) + 76N \tag{68}$$

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

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

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

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

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

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

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

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

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

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

626 Leão, J. W. D. and Silva, J. B. C. (1989). Discrete linear transformations of potential field data. *Geophysics*
627   54, 497–507. doi:10.1190/1.1442676

628 Li, Y., Nabighian, M., and Oldenburg, D. W. (2014). Using an equivalent source with positivity for low-
629   latitude reduction to the pole without striation. *GEOPHYSICS* 79, J81–J90. doi:10.1190/geo2014-0134.
630   1

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

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

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

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

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

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

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

644 Roy, A. (1962). Ambiguity in geophysical interpretation. *GEOPHYSICS* 27, 90–99. doi:10.1190/1.
645   1438985

646 Silva, J. B. C. (1986). Reduction to the pole as an inverse problem and its application to low-latitude
647   anomalies. *GEOPHYSICS* 51, 369–382. doi:10.1190/1.1442096

648 Siqueira, F., Oliveira Jr., V. C., and Barbosa, V. C. F. (2017). Fast iterative equivalent-layer technique for
649   gravity data processing: A method grounded on excess mass constraint. *GEOPHYSICS* 82, G57–G69.
650   doi:10.1190/GEO2016-0332.1

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

653 Soler, S. R. and Uieda, L. (2021). Gradient-boosted equivalent sources. *Geophysical Journal International*
654   227, 1768–1783. doi:10.1093/gji/ggab297

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

657 Takahashi, D., Oliveira Jr., V. C., and Barbosa, V. C. F. (2020). Convolutional equivalent layer for gravity
658   data processing. *GEOPHYSICS* 85, G129–G141. doi:10.1190/geo2019-0826.1

659 van der Sluis, A. and van der Vorst, H. A. (1987). Numerical solution of large, sparse linear algebraic
660   systems arising from tomographic problems. In *Seismic tomography with applications in global*
661   *seismology and exploration geophysics*, ed. G. Nolet (D. Reidel Publishing Company), chap. 3. 49–83
662 Van Loan, C. F. (1992). *Computational Frameworks for the fast Fourier transform*. Frontiers in Applied
663   Mathematics (SIAM)
664 Xia, J. and Sprowl, D. R. (1991). Correction of topographic distortion in gravity data. *Geophysics* 56,
665   537–541
666 Xia, J., Sprowl, D. R., and Adkins-Heljeson, D. (1993). Correction of topographic distortions in potential-
667   field data; a fast and accurate approach. *Geophysics* 58, 515–523. doi:10.1190/1.1443434
668 Zidarov, D. (1965). Solution of some inverse problems of applied geophysics. *Geophysical Prospecting*
669   13, 240–246. doi:10.1111/j.1365-2478.1965.tb01932.x