Geophysical Journal International

Geophys. J. Int. (2020) **223,** 840–850
Advance Access publication 2020 July 15
GJI Marine geosciences and applied geophysics

doi: 10.1093/gji/ggaa341

Finite-element EM modelling on hexahedral grids with an FD solver as a pre-conditioner

Nikolay Yavich [1,2] and Michael S. Zhdanov[2,3,4]

[1] *Skolkovo Institute of Science and Technology , CDISE, Moscow* 143026*, Russia. E-mail:* *n.yavich@skoltech.ru*
[2] *Moscow Institute of Physics and Technology Applied Computational Geophysics Lab, Dolgoprudny* 141701*, Russia*
[3] *University of Utah Geology & Geophysics, Salt Lake City, UT* 84112*, USA*
[4] *TechnoImaging LLC, Salt Lake City, UT* 84107*, USA*

Accepted 2020 July 7. Received 2020 June 17; in original form 2019 July 26

SUMMARY

The finite-element (FE) method is one of the most powerful numerical techniques for modelling 3-D electromagnetic fields. At the same time, there still exists the problem of efficient and economical solution of the respective system of FE equations in the frequency domain. In this paper, we concentrate on modelling with adapted hexahedral or logically rectangular grids. These grids are easy to generate, yet they are flexible enough to incorporate real topography and seismic horizons. The goal of this work is to show how a finite-difference (FD) solver can be used as a pre-conditioner for hexahedral FE modelling. Applying the lowest order Nédélec elements, we present a novel pre-conditioned iterative solver for the arising system of linear equations that combines an FD solver and simple smoothing procedure. The particular FD solver that we use relies on the implicit factorization of the horizontally layered earth matrix. We assessed runtime and accuracy of the presented approach on synthetic and real resistivity models (topography of the Black Sea continental slope). We further compared performance of our program versus publicly available Mare2DEM, ModEM and MUMPS programs/libraries. Our examples involve plane-wave and controlled source modelling. The numerical examples demonstrate that the presented approach is fast and robust for models with moderate contrast, supports highly deformed cells, and is quite memory-economical.

Key words: Controlled source electromagnetics (CSEM); Electromagnetic theory; Magnetotellurics; Numerical modelling; Numerical solutions.

1 INTRODUCTION

The role of 3-D electromagnetic (EM) forward modelling can hardly be overestimated nowadays as it is used in EM survey design as well as in EM data interpretation. In the frequency domain, the implementation of a forward modelling routine generally involves the following steps: computational grid generation, conductivity (and possibly permittivity and susceptibility) averaging, system matrix preparation, solution of the system of equations and interpolation of the solution to the receiver locations. In this paper, we address the problem of efficient solution of the finite-element (FE) system of equations as it is the most computationally demanding step.

FE modelling on rectangular, more general hexahedral and tetrahedral grids was discussed in a large number of relatively recent publications (e.g. Silva *et al.* 2012; Cai *et al.* 2014, 2017; Li *et al.* 2016, among many others); however, pre-conditioned iterative solution is rarely considered, though we should note the work of Puzyrev *et al.* (2013), Um *et al.* (2013) and Ren *et al.* (2014).

Some authors studied the use of sparse direct solvers (Kordy *et al.* 2016) appealing to their universality and robustness. These solvers require hundreds of Gb of auxiliary memory and hours of CPU time at the factorization step for typical modelling. With parallelization, this dramatic computational burden is somewhat reduced but remains significant.

To minimize the size of the FE system, grids with hanging nodes or octree grids can be used (Haber & Heldman 2007; Grayver & Kolev 2015). This kind of grids eliminates highly stretched cells outside the survey area. Although attractive in principle, solution of the respective system requires incorporation of a diffusion equation solver, which ultimately raises computational complexity of a single iteration.

Let us limit our consideration to adapted hexahedral grids or logically rectangular grids (see Fig. 1a for 2-D illustration), that is, every cell in such a grid is received by *finite deformation* of the respective cell of a rectangular grid (Fig. 1b). Logically rectangular grids directly accommodate land or seafloor topography or seismic horizons. It can be easily noted that the respective FE matrix shares many properties with the FE or finite-difference (FD) matrix constructed on the rectangular grid.

© The Author(s) 2020. Published by Oxford University Press on behalf of The Royal Astronomical Society.

Downloaded from https://academic.oup.com/gji/article/223/2/840/5871825 by University of Utah Eccles Health Sciences Library user on 04 September 2020

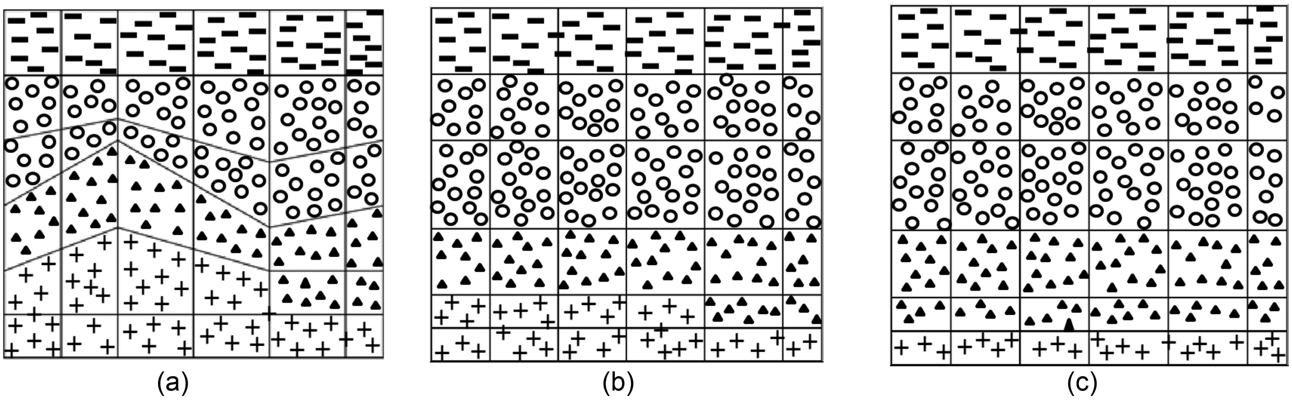


Figure 1. 2-D counterpart of an adapted (deformed) hexahedral grid or logically rectangular grid (a), (undeformed) rectangular grid (b), horizontally layered earth model on the rectangular grid (c). Different filling pattern corresponds to conductivity values.

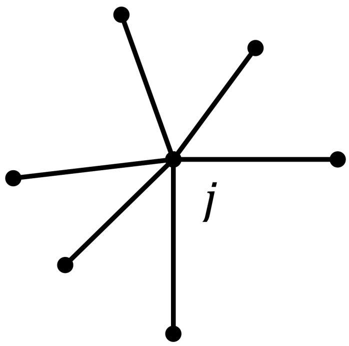


Figure 2. Six edges meeting at an internal node j of the hexahedral grid.

The key point of this paper is to check if an FD solver may be used as a pre-conditioner in FE modelling on logically rectangular grids. As far as the authors are concerned, this type of solution method was not published earlier. The particular FD solver that we will test is based on the implicit and economical factorization of the horizontally layered earth matrix (Fig. 1c, see Appendix A or Yavich *et al.* 2020).

This paper is organized as follows. At first, we formulate the FE system on a hexahedral grid. Next, we introduce FD matrices and discuss their relation to the FE system. After we describe a smoothing procedure, which happens to be necessary to warrant robustness of the FD pre-conditioner for the FE problem. Finally, we present numerical examples that include both synthetic conductivity models, as a model with real marine topography. Our examples involve plane-wave and controlled source modelling.

2 FINITE-ELEMENT SYSTEM OF EQUATIONS

We consider a diffusion of EM field within 3-D heterogeneous conducting media with triaxial-anisotropic electrical conductivity tensor, $\boldsymbol{\sigma}(x, y, z)$:

$$\boldsymbol{\sigma}(x, y, z) = \begin{pmatrix} \sigma_x(x, y, z) & 0 & 0 \\ 0 & \sigma_y(x, y, z) & 0 \\ 0 & 0 & \sigma_z(x, y, z) \end{pmatrix}. \tag{1}$$

The conductivity is assumed to be non-zero in the air. Within forward modelling, we look for the electric field $\mathbf{E}(x, y, z)$ that satisfies the following second-order system of partial differential equations:

$$\text{curl curl } \mathbf{E} - i\omega\mu_0\boldsymbol{\sigma}\ \mathbf{E} = i\omega\mu_0\boldsymbol{J}, \tag{2}$$

where $\boldsymbol{J}$ is the source current density, i is the complex unity, ω is the angular frequency and μ_0 is magnetic permeability of the vacuum (Zhdanov 2002, 2009). These differential equations are solved in a rectangular hexahedral computational domain V. We further preferred secondary field modelling and thus complemented eq. (2) with zero Dirichlet boundary conditions,

$$\mathbf{E} \times \boldsymbol{\nu} = 0, \tag{3}$$

where $\boldsymbol{\nu}$ is the unit vector outward normal to the domain boundary S.

We assumed that the computational domain is partitioned with a non-overlapping deformed hexahedral grid (see Fig. 1a) for a 2-D illustration, and each cell will be denoted as V_j, $j = 1..m$. We further denote as n the number of internal edges. To find an approximate solution to eqs (2) and (3) on the introduced grid, we follow the conventional FE technique. In this paper, we considered lowest-order Nédélec FE basis functions (Nédélec 1986). For deformed hexahedra, they are completed by the covariant Piola transform (Falk *et al.* 2011), which preserves continuity of the tangential components across interelement faces.

Using these basis functions, $\mathbf{p}_k(x, y, z)$, $k = 1..n$, we expend the unknown electric field, $\mathbf{E}$, as follows:

$$\mathbf{E} \approx \sum_{k=1}^{n} e_k \mathbf{p}_k, \quad \mathbf{e} = (\cdots e_k \cdots)^T \in \mathbb{C}^n. \tag{4}$$

This discretization results in the following system of linear equations:

$$\mathbf{A}\,\mathbf{e} = \mathbf{f}, \tag{5}$$

where $\mathbf{A} = \mathbf{R} - i\omega\mu_0\mathbf{S}$, $\mathbf{R} = (R_{ij})$, $\mathbf{S} = (\mathrm{S}_{ij})$, $\mathbf{f} = (f_i)$, $i = 1..n$, $j = 1..n$,

$$\begin{aligned} R_{ij} &= \int_V \text{curl } \mathbf{p}_i \cdot \text{curl } \mathbf{p}_j \, \mathrm{d}V, \quad \mathrm{S}_{ij} = \int_V \mathbf{p}_i \cdot \boldsymbol{\sigma}\ \mathbf{p}_j \mathrm{d}V, \\ f_i &= i\omega\mu_0 \int_V \mathbf{p}_i \cdot \boldsymbol{J} \mathrm{d}V. \end{aligned} \tag{6}$$

The obtained system is characterized by a large sparse complex symmetric square matrix. The corresponding matrix equation can be solved with both iterative and direct methods.

Downloaded from https://academic.oup.com/gji/article/223/2/840/5871825 by University of Utah Eccles Health Sciences Library user on 04 September 2020

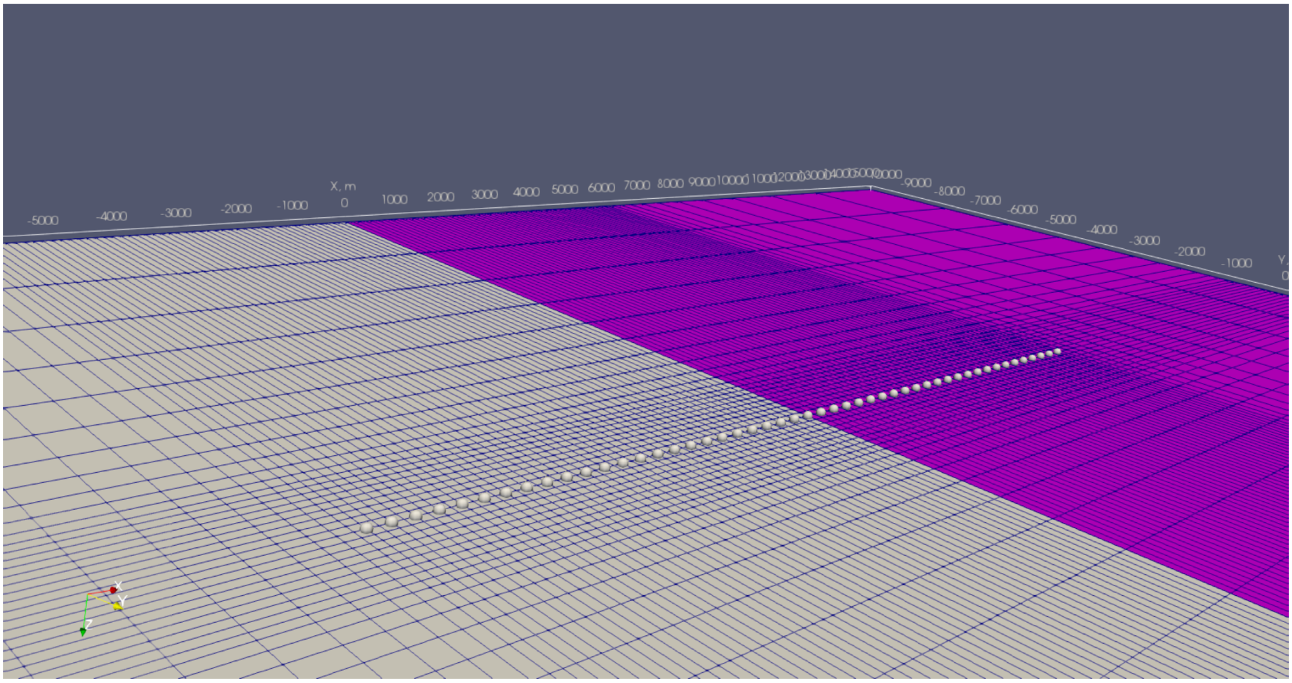


Figure 3. Part of the modelling domain and grid. The grey quarter-space had resistivity 1 Ωm, the magenta quarter-space had resistivity either 10, 100 or 1000 Ωm. Receivers are marked as white balls.

3 PRE-CONDITIONING APPROACHES

3.1 Relation to the FD problem

The essence of pre-conditioning is to find a substitute matrix $\mathbf{B}$ for the system matrix $\mathbf{A}$ such that it is of the same size, has similar spectral properties, but easier to invert than $\mathbf{A}$. After such a pre-conditioner $\mathbf{B}$ was designed, it can be used directly (though this is uncommon),

$$\mathbf{B}^{-1}\mathbf{A}\,\mathbf{e} = \mathbf{B}^{-1}\,\mathbf{f}, \tag{7}$$

or incorporated into the Richardson iterative method,

$$\mathbf{e}_{k+1} = \mathbf{e}_k + \mathbf{B}^{-1}\,(\mathbf{f} - \mathbf{A}\mathbf{e}_k), \tag{8}$$

$k = 0, 1, \ldots$, and $\mathbf{e}_0$ is the initial guess, or more advanced Krylov subspace iterative solvers, for example, BiCGStab or GMRes.

The pre-conditioning approaches discussed below employ a rectangular hexahedral grid obtained by removing deformations (Fig. 1b). On this rectangular grid, the FD discretization is applicable. Introduce the FD matrix,

$$\mathbf{A}_{\mathrm{FD}} = \mathbf{R}_{\mathrm{FD}} - i\omega\mu_0\mathbf{S}_{\mathrm{FD}}, \tag{9}$$

where $\mathbf{R}_{\mathrm{FD}}$ is the FD curl–curl operator and $\mathbf{S}_{\mathrm{FD}}$ is a diagonal matrix of averaged anisotropic conductivities multiplied by the volumes around the respective edges. Note that the deformed and undeformed grids have the same number of edges, thus matrices $\mathbf{A}$ and $\mathbf{A}_{\mathrm{FD}}$ are of the same size. Consequently, we can try to pre-condition our FE system (5) with $\mathbf{A}_{\mathrm{FD}}$ in the following or other forms,

$$\mathbf{e}_{k+1} = \mathbf{e}_k + \mathbf{A}_{\mathrm{FD}}^{-1}\,(\mathbf{f} - \mathbf{A}\mathbf{e}_k). \tag{10}$$

However, inversion of a large FD matrix is a complex task by itself.

To avoid inversion of $\mathbf{A}_{\mathrm{FD}}$**,** we perform another substitution. At this point, any available FD pre-conditioner would be applicable, provided it is robust enough. The particular FD pre-conditioner we used in our implementation relies on a background conductivity model. Introduce $\boldsymbol{\sigma}_b(z)$, a possibly anisotropic conductivity that depends on the vertical coordinate only (Fig. 1c). Following Yavich & Zhdanov (2016), we define matrix corresponding to the FD problem of the background media, $\mathbf{A}_{b\ \mathrm{FD}}$,

$$\mathbf{A}_{b\ \mathrm{FD}} = \mathbf{R}_{\mathrm{FD}} - i\omega\mu_0\mathbf{S}_{b\ \mathrm{FD}}, \tag{11}$$

where $\mathbf{S}_{b\ \mathrm{FD}}$ is a diagonal matrix corresponding to $\boldsymbol{\sigma}_b(z)$.

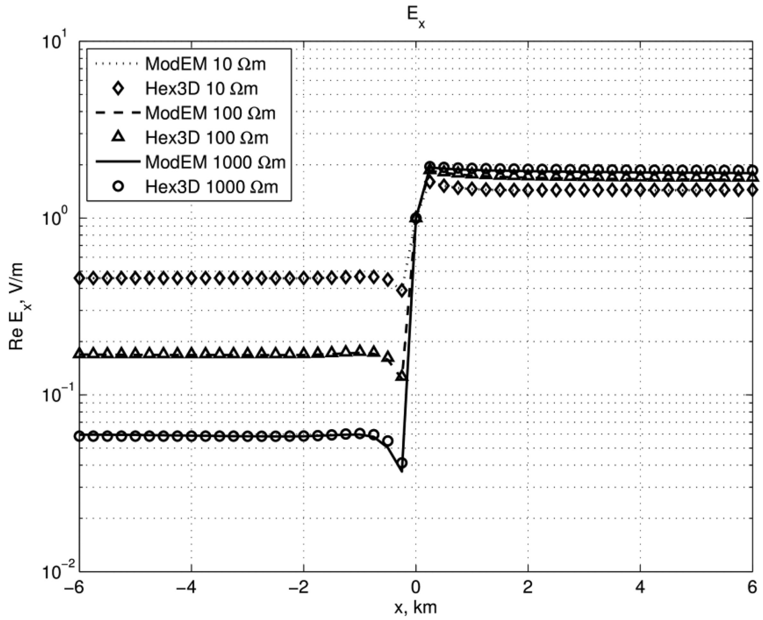
We should note that matrix $\mathbf{A}_{b\ \mathrm{FD}}$ is of the same size as $\mathbf{A}$ and multiplication of the inverse matrix $\mathbf{A}_{b\ \mathrm{FD}}^{-1}$ with a given vector can be rapidly (in at most $O(n^{\frac{4}{3}})$ arithmetical operations) and economically performed, see Appendix A. Therefore, it is practical to apply $\mathbf{A}_{b\ \mathrm{FD}}^{-1}$ as a pre-conditioner to eq. (5) as described in eq. (8) or using BiCGStab. We will refer $\mathbf{A}_{b\ \mathrm{FD}}$ as FD GF pre-conditioner since $\mathbf{A}_{b\ \mathrm{FD}}^{-1}$ implements FD Green's functions.

Pre-conditioning of an FE problem with an FD matrix is known to work efficiently for diffusion and acoustic problems (Heikokla *et al.* 1999). Interestingly, this does not work for EM problems (see Numerical Examples). We regard this to the rich null spaces of matrices $\mathbf{R}$ and $\mathbf{R}_{\mathrm{FD}}$. It can be easily noted that whenever at least one hexahedron is deformed (rather than just stretched or squeezed), the null spaces of these matrices are different. Consequently, the respective error components are not reduced within the pre-conditioned iterative solver (the importance of the null space within iterative solution of Maxwell equations is discussed for example in Hiptmair 1998).

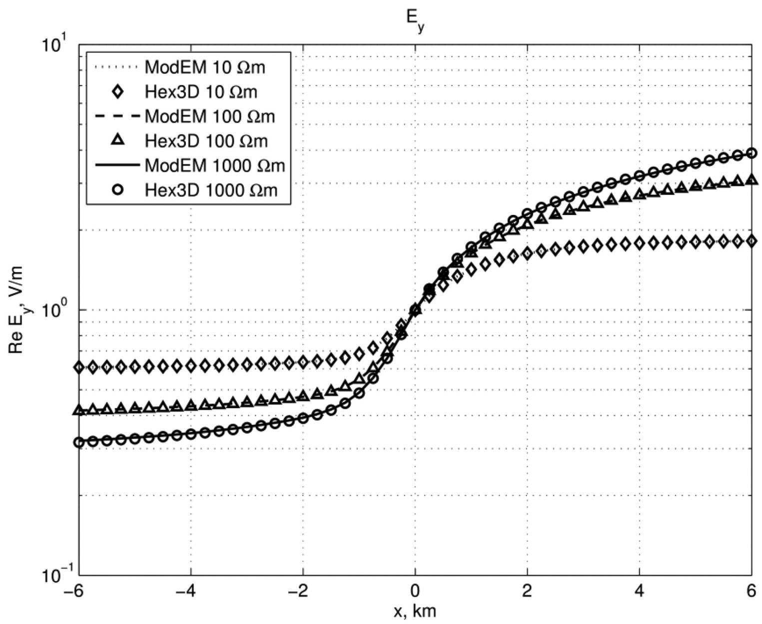
3.2 Smoothing procedure

Since the appearance of multigrid methods for EM problems (Arnold *et al.* 2000), several procedures to reduce null-space error components of the curl–curl operator are known. They enforce the charge conservation law either globally (in the whole computational domain) or locally (in points, lines, or plains). The latter

Downloaded from https://academic.oup.com/gji/article/223/2/840/5871825 by University of Utah Eccles Health Sciences Library user on 04 September 2020



(a)



(b)

Figure 4. Electric field responses due to plane wave at 1 Hz for two-quarter-space model. Responses were computed using hexahedral FE program (Hex3D) and ModEM. The resistivity of the west quarter-space was 1 Ωm, the resistivity of east quarter-space was either 10, 100 or 1000 Ωm. (a)E_x component of the response due to x-polarized plane wave; b) E_y component of the response due to y-polarized plane wave.

Table 1. Iteration count and CPU time of hexahedral FE program (Hex3D) and ModEM. Iterative solver tolerance was 1e−7. The resistivity of the west quarter-space was 1 Ωm; the resistivity of east quarter-space was either 10, 100 or 1000 Ωm.

East quarter-space resistivity (Ωm)	Hex3D Iteration count/CPU time (min) per polarization	ModEM CPU time (min) per polarization
10	58/2.86	3.86
100	211/10.46	3.92
1000	433/21.71	3.92

Table 2. Iteration count and CPU time of hexahedral FE program (Hex3D) for a sequence of grids. Iterative solver tolerance was 1e−7. The resistivity of the west quarter-space was 1 Ωm; the resistivity of east quarter-space was 100 Ωm.

Modelling grid	Discrete problem size	Hex3D Iteration count/CPU time (min) per polarization
136 × 56 × 75	1 669 835	211/10.46
68 × 28 × 37	200 565	197/0.80
34 × 14 × 18	23 090	142/0.03

Downloaded from https://academic.oup.com/gji/article/223/2/840/5871825 by University of Utah Eccles Health Sciences Library user on 04 September 2020

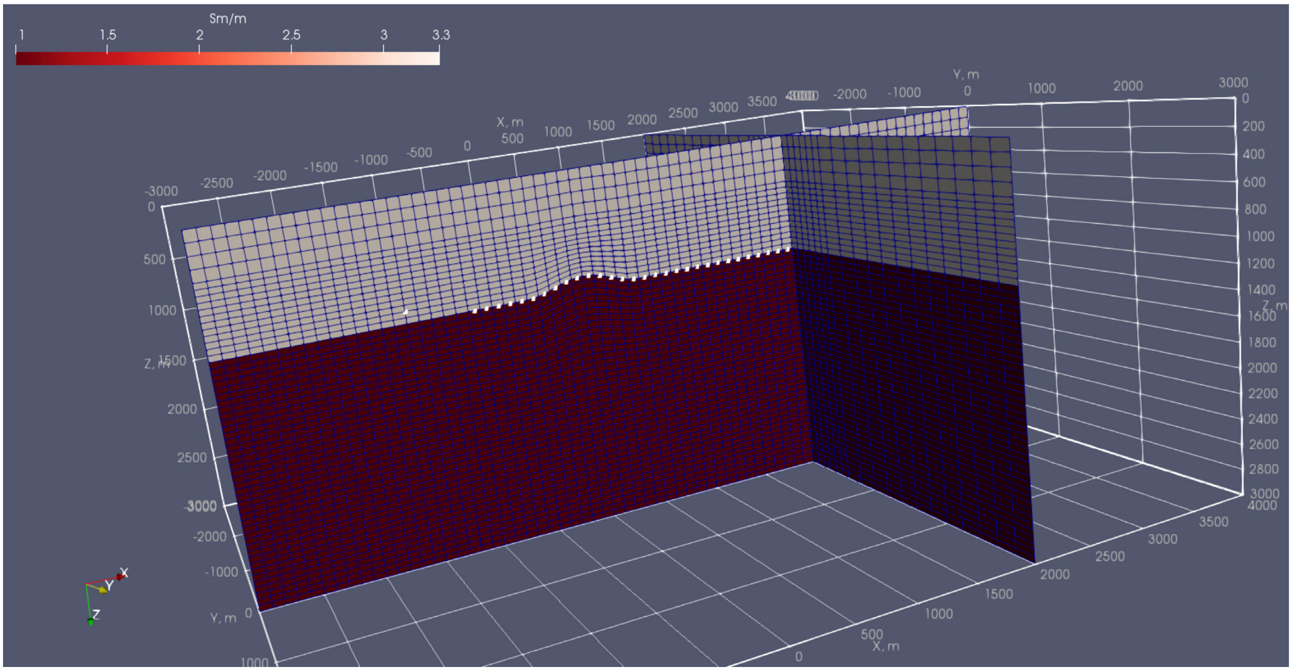


Figure 5. Model with simple seafloor bathymetry: sea-water layer (grey), sediments (red), source (leftmost white disk) and receivers (the other white disks). The modelling grid was adapted to the seafloor (blue lines).

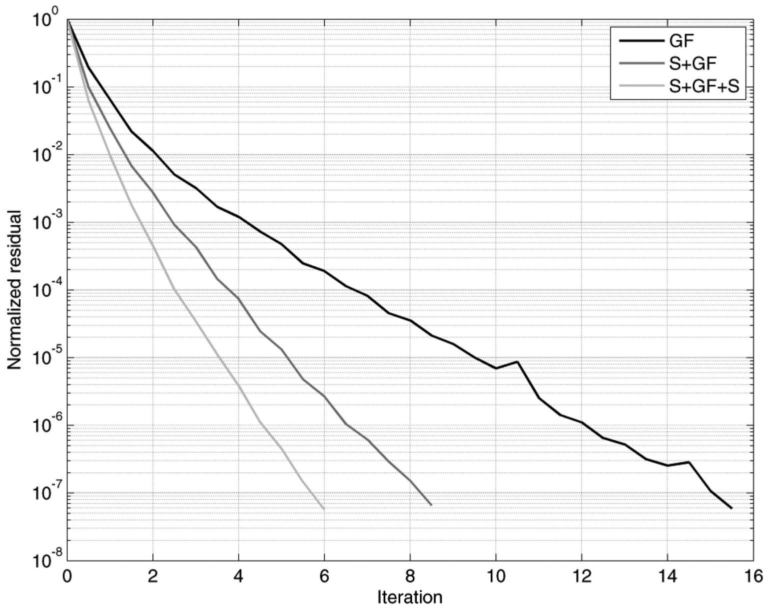


Figure 6. BiCGStab convergence history pre-conditioned with either $\mathbf{A}_{b\ \mathrm{FD}}$ matrix only (GF), or combined with pre-smoothing (S + GF), or pre- and post-smoothing (S + GF + S) for the model with simple seafloor bathymetry

Table 3. Performance of the BiCGStab solver pre-conditioned with either $\mathbf{A}_{b\ \mathrm{FD}}$ matrix only (GF), or combined with pre-smoothing (S + GF), or pre- and post-smoothing (S + GF + S) for the model with simple seafloor bathymetry.

Pre-conditioned BiCGStab solver	Iteration count	CPU time (min)	CPU time (min) per iteration
S + GF + S	6	0.16	0.026
S + GF	9	0.17	0.019
GF	16	0.22	0.014

procedures are referred to as *smoothers* within multigrid methods because they smooth out spatial error components.

The smoothing procedure that has the smallest arithmetical complexity performs point smoothing in the block Jacobi fashion. Let p be the number of internal grid nodes. At any internal node of a logically rectangular hexahedral grid, six edges meet (Fig. 2). Consider a node j. Let $\mathbf{B}_j$ be an $n \times n$ matrix that coincides with $\mathbf{A}$ at those columns and rows which correspond to the six edges, while coinciding with the identity matrix at the rest of the entries.

Downloaded from https://academic.oup.com/gji/article/223/2/840/5871825 by University of Utah Eccles Health Sciences Library user on 04 September 2020

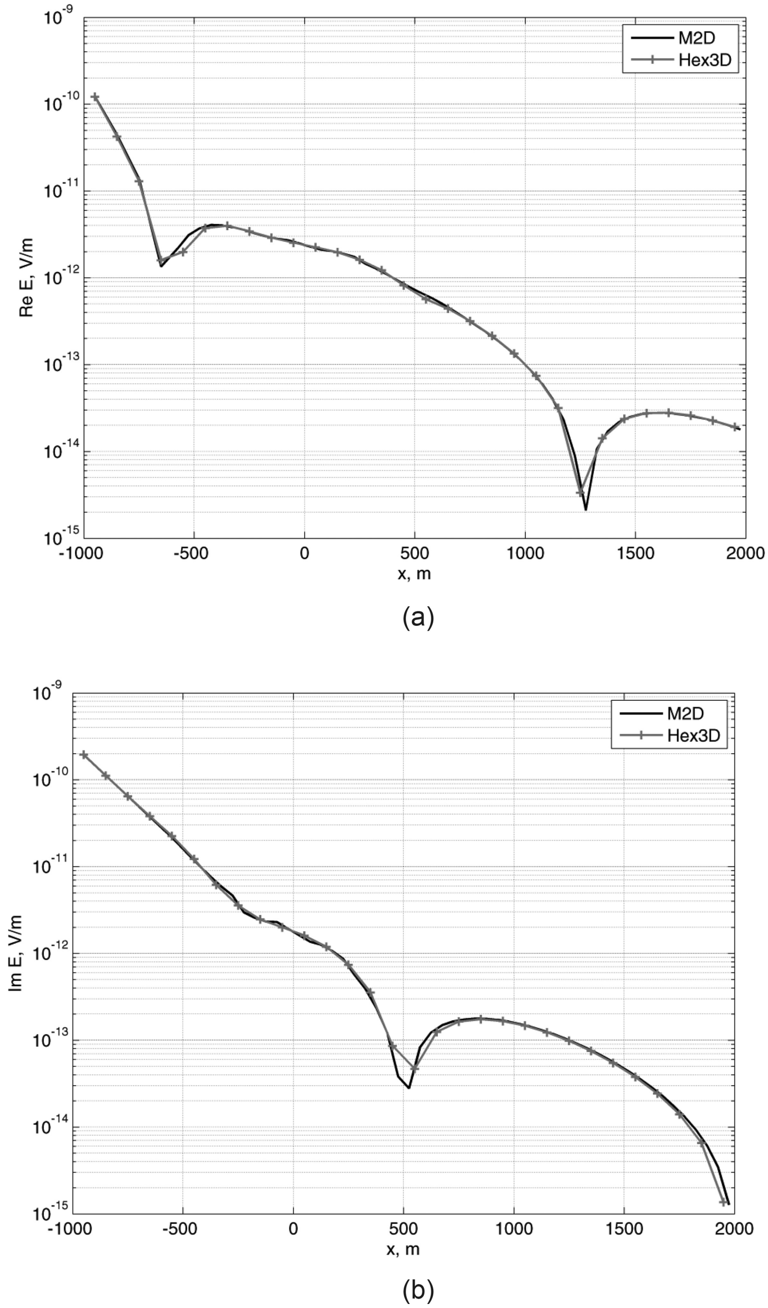


Figure 7. Real and imaginary parts of the inline electric field components tangential to the seafloor. Hexahedral FE modelling (Hex3D) compared with 2-D FE modelling (M2D) of mare2DEM for the model with simple seafloor bathymetry

Note that the matrix $\mathbf{B}_j$ is easy to invert and solution of a system with matrix $\mathbf{B}_j$ enforces the charge conservation law in the point j. Now, the block Jacobi smoother can be written as follows:

$$\mathbf{e}_{k+1} = \mathbf{e}_k + \eta \sum_{j=1}^{p} \mathbf{B}_j^{-1} \, (\mathbf{f} - \mathbf{A}\,\mathbf{e}_k). \tag{12}$$

This procedure has linear arithmetical complexity and naturally parallelizable. The scalar parameter η should be positive and less than 1. In our experiments, we picked $\eta = 0.5$ as this choice takes in account the fact that each edge is involved at most twice in the stencils, Fig. 2. (Yavich & Scholl 2012).

Now we have to combine the pre-conditioning step (eqs 10 and 11) with the smoothing step (eq. 12). The next three-step algorithm resulted in a robust solver,

$$\begin{aligned} \mathbf{e}_{k+1/3} &= \mathbf{e}_k + \eta \sum_{j=1}^{p} \mathbf{B}_j^{-1} \, (\mathbf{f} - \mathbf{A}\,\mathbf{e}_k), \\ \mathbf{e}_{k+2/3} &= \mathbf{e}_{k+1/3} + \mathbf{A}_{b\,FD}^{-1} \left(\mathbf{f} - \mathbf{A}\,\mathbf{e}_{k+1/3}\right), \\ \mathbf{e}_{k+1} &= \mathbf{e}_{k+2/3} + \eta \sum_{j=1}^{p} \mathbf{B}_j^{-1} \left(\mathbf{f} - \mathbf{A}\,\mathbf{e}_{k+2/3}\right). \end{aligned} \tag{13}$$

Practically, we do not implement eq. (13) as is, rather these three steps were implemented as a pre-conditioner to the BiCGStab iterative solver. Following the multigrid terminology, the first step we will be referred to as pre-smoothing, the last step we will be referred to as post-smoothing.

Downloaded from https://academic.oup.com/gji/article/223/2/840/5871825 by University of Utah Eccles Health Sciences Library user on 04 September 2020

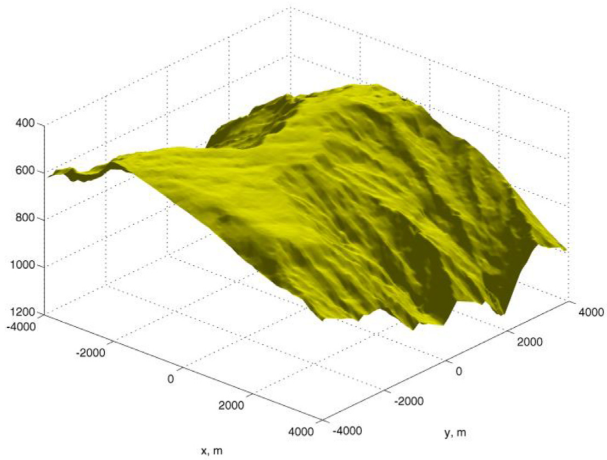


Figure 8. Seafloor bathymetry.

To wrap up this section, let us summarize the properties of the designed pre-conditioner. Arithmetical complexity of a single application of eq. (13) is $O(n^{\frac{4}{3}})$. Initialization is momentary since it involves manipulation with 1-D discrete problems only, needed to prepare for multiplication of $\mathbf{A}_{b\ \mathrm{FD}}^{-1}$ by a vector. Inversion of 6 × 6 block arising in $\mathbf{B}_j^{-1}$ is performed on-the-fly to minimize storage expenses.

4 NUMERICAL EXAMPLES

In this section, we present the results of 3-D numerical modelling experiments with the pre-conditioning scheme (13) introduced earlier. The scheme was incorporated into the BiCGStab iterative solver. The finite-element system of equations (5) was implemented in Matlab (program of Cai *et al.* 2014). The finite-difference pre-conditioner introduced in eq. (11) (Yavich & Zhdanov 2016) and smoother (12) were implemented in C/C++ and linked to the Matlab code. Computations were performed on a Linux cluster node. All the presented results involve sequential computations.

4.1 Modelling on rectangular grids

We start off our experiments with a simple model (Fig. 3) consisting of two quarter-spaces of different resistivity. The west ($x < 0$) has resistivity 1 Ωm, while the resistivity of the east quarter-space ($x > 0$) was different in different tests, 10, 100 and 1000 Ωm, respectively. A plane-wave electric field response at 1 Hz was recorded along a 12 km profile, perpendicular to the contact interface. The resistivity of the air was 10^6 Ωm.

In this case, we used an FE grid that matched the FD grid. The modelling domain occupied the volume [−85; 85] × [−80; 80] × [−53; 71] km^3 and cells in the central part had the size 125 × 125 × 50 m^3. Part of the grid is shown in Fig. 3. The grid had 136 × 56 × 75 cells, making the discrete problem size 1 669 835 unknowns.

We compared accuracy and runtime of our program versus those of ModEM 2019, Egbert & Kelbert (2012) and Kelbert *et al.* (2014). ModEM has implemented Fortran 95 and its solver part combines QMR iterations with ILU pre-conditioner and static divergence corrections. ModEM supports transfer function computation only, thus minor edits were implemented to obtain the electric fields due to a particular polarization.

Both of the programs used the same modelling grid and accuracy of the iterative solvers was set to 1e−7. Fig. 4 illustrates the computed responses by the two programs for the three different resistivities of the east quarter-space. The E_x component of the response due to x-polarized plane wave is shown in Fig. 4(a), while E_y component of the response due to y-polarized plane wave is shown in Fig. 4(b). The electric fields were normalized so that they would pass through the unity at $x = 0$.

We observe a fairly good match of the responses. Though, some discrepancies are notable in the 1000 Ωm case. We regard this to different incorporation of the quarter-space: our program uses secondary field modelling, while ModEM uses primary field modelling.

Smoothing was not applied in this test since hexahedrons were not deformed. Table 1 presents iteration count and CPU time of our hexahedral FE program and ModEM. Our program was faster in case of moderate resistivity contrast, while in other cases, the number of iterations tends to grow. For this model, the pre-conditioning matrix $\mathbf{A}_{b\ \mathrm{FD}}$ corresponds to a uniform half-space model. Consequently, this approach loses robustness. On the other hand, ModEM performs fairly invariant to resistivity increase. We should note that our program is experimental and involves Matlab code, nevertheless, in some cases, its performance is competitive with the industry-standard program.

We also investigated the impact of the discrete problem size in Table 2. The originally generated grid 136 × 56 × 75 was coarsened either two or four times in each direction by removing the respective grid lines. The table reports iteration count and CPU time required for modelling with these grids. We see that the grid size has a small impact on the iteration count, implying that the resented approach is applicable for large-scale problems.

4.2 Simple bathymetry modelling

In this section, we illustrate performance of the designed algorithm applied for marine CSEM modelling in the presence of 2-D simple seafloor topography. We considered a model formed by a 1000 m deep sea-water layer and of 0.3 Ωm and homogenous seafloor of 1 Ωm. The seafloor contains an elevation of 110 m uniform in the y-direction (Fig. 5). The top of the elevation is located along $x = 0$ and it spans approximately from $x = -500$ m to $x = 500$ m.

We modelled a response of a x-directed dipole source centred in [−1500; 0; 900] m, that is, 100 m above the seafloor and 1.5 km to the west of the elevation. The source was emitting at 1 Hz. The seafloor receivers were inline to the source, along $y = 0$ and tangential to the seafloor.

For this set-up, a hexahedral grid was generated. The modelling grid covered the volume [−3; 4] × [−3; 3] × [−5; 3] km^3, had the smallest cell size 100 × 100 × 50 m^3 and was adapted to seafloor topography (Fig. 5). The grid had dimensions 70 × 36 × 81, making the size of the discrete problem 590 335.

Fig. 6 shows BiCGStab convergence history pre-conditioned with either $\mathbf{A}_{b\ \mathrm{FD}}$ matrix (11) only, or combined with pre-smoothing, or pre- and post-smoothing (13). See also Table 3. In all of the three cases, the iterative solver converged to the target residual of 1e−7 quite fast: 16, 9 and 6 iterations respectively. Though, we should note that the use of smoothing increased convergence speed considerably. Also, note that smoothing steps increased computational complexity of a single iteration (Table 3). Nevertheless, the shortest CPU time was received when two smoothing steps were performed per iteration.

Downloaded from https://academic.oup.com/gji/article/223/2/840/5871825 by University of Utah Eccles Health Sciences Library user on 04 September 2020

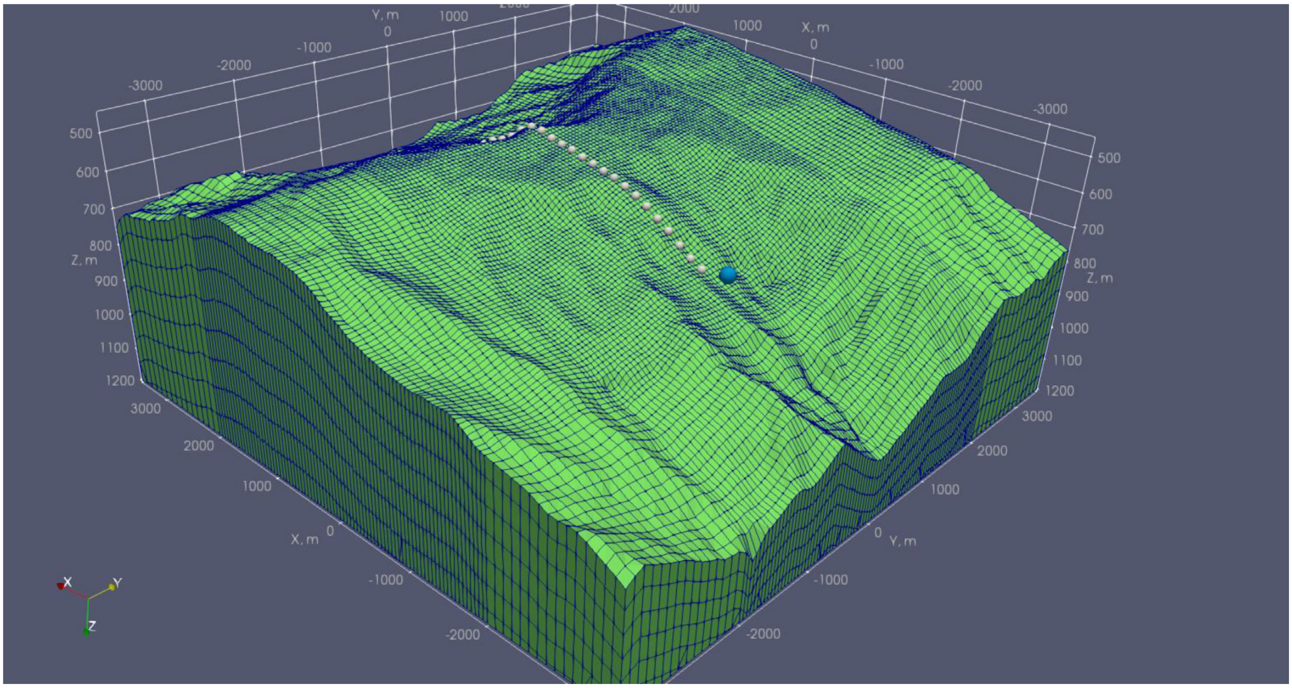


Figure 9. Hexahedral grid adapted to seafloor bathymetry. The source is marked as a blue ball, receivers are marked as white balls.

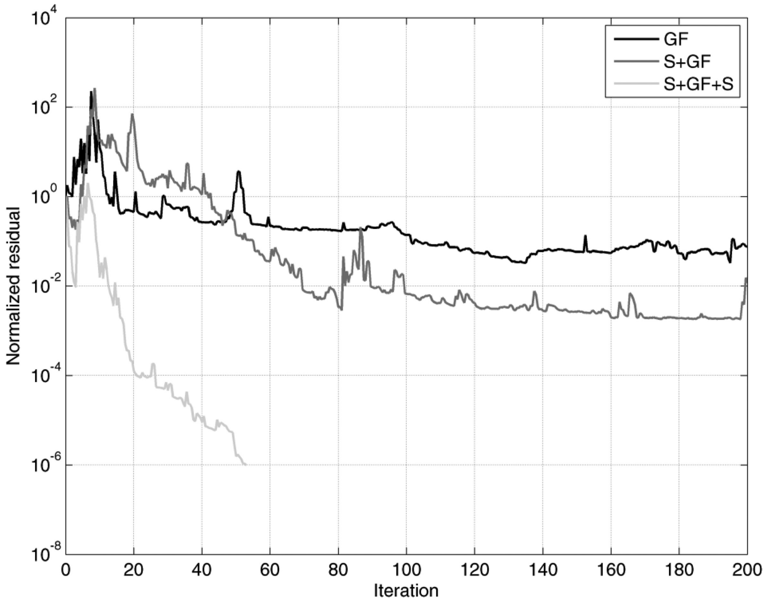


Figure 10. BiCGStab convergence pre-conditioned with FD GF completed with either no smoothing (GF), pre-smoothing (S + GF) or pre- and post-smoothing (S + GF + S).

Table 4. Performance of the pre-conditioned iterative and sparse direct solvers.

Solver	CPU time	Peak memory usage
BiCGStab/S + GF + S	5.3 min	1 Gb
MUMPS	factorize 8.0 hr solve 0.7 min	56 Gb

To conclude this experiment, we illustrate and compare the computed response with that of publicly available program 2.5-D finite-element code Mare2DEM (Key 2016). We should note that the model used in this experiment is 2-D with y being the strike direction, consequently such a comparison is fair. Fig. 7 shows real and imaginary parts of the electric field components. We observe a good match, though some inaccuracies are notable near sign reversals.

Downloaded from https://academic.oup.com/gji/article/223/2/840/5871825 by University of Utah Eccles Health Sciences Library user on 04 September 2020

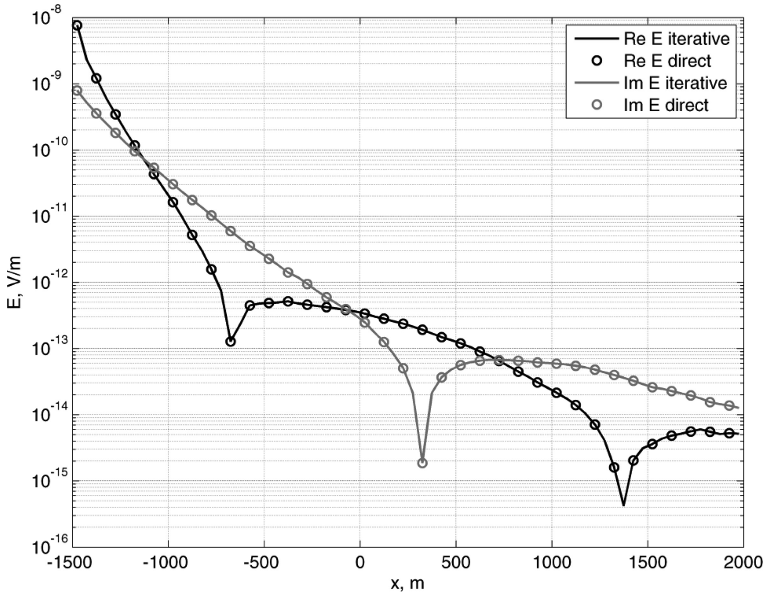


Figure 11. Responses computed with the pre-conditioned iterative and sparse direct solvers. Real and imaginary parts of the inline electric field component tangential to the seafloor are shown.

4.3 Real bathymetry modelling

We modelled a response of a horizontal electric dipole of 1 Hz towed near the seafloor. We assumed a homogeneous 1 Ωm subsurface, while seafloor bathymetry represents an 8×8 km^2 area of the Black Sea continental slope (Daudina *et al.* 2014; Yavich *et al.* 2019), Fig. 8. In this area, depth varies from 450 to 1150 m. Sea-water conductivity was 0.3 Ωm.

For this set-up, we generated a non-uniform $115 \times 100 \times 55$ grid (Fig. 9) with overall 1.851 million discrete electric field unknowns. The grid was adapted to the seafloor bathymetry.

Fig. 10 illustrates convergence of the BiCGStab iterative solver, pre-conditioned with $A_{b\ \mathrm{FD}}$ completed with either no smoothing, pre-smoothing or pre- and post-smoothing. In the last case, the pre-conditioned iterative solver converged in 53 iterations in 5.3 min to the residual norm of 1e−6. In other cases, no convergence was observed. We conclude that for realistic models, pre- and post-smoothing are necessary to gain robustness. Note that both smoothing and the FD GF pre-conditioner are naturally parallelizable (Yavich & Scholl 2012; Yavich *et al.* 2017).

Finally, we linked to the commonly used MUMPS sparse direct solver (Amestoy *et al.* 2001) to assess their performance versus the discussed approach. In this study, we have looked at sequential performance leaving investigation of scalability aside. MUMPS 5.1.1 was used and complex symmetricity of the matrix was employed during factorization.

Table 4 shows CPU time and memory usage of the pre-conditioned iterative and MUMPS sparse direct solvers. It took 8 hr and 56 Gb of memory for the direct solver to factorize the matrix. In contrast, the designed pre-conditioned iterative solver required only near 1 Gb of memory. We conclude that in this example, the iterative solver was roughly 90 times faster and 50 times more memory-economical.

Both of the approaches can gain some speedup in a parallel environment, but the computational burden of the sparse direct factorization still will be significant.

Finally, Fig. 11 illustrates responses modelled with the two solvers. The responses are essentially identical. This demonstrates the fact that pre-conditioning eqs (10) and (12) corresponds to the equivalent transformation of the original linear system (5).

5 CONCLUSIONS

We designed and tested a pre-conditioning scheme for the hexahedral FE system of equations resulting from the discretization on a logically rectangular grid. The approach combines the FD GF pre-conditioner and pre- and post-smoothing procedures. Accuracy and runtime were compared versus commonly used in the EM community programs ModEM and Mare2DEM. The numerical examples presented above demonstrate that the presented approach is fast and robust for models with moderate contrast. As far as the authors are concerned, this type of pre-conditioning has not been published earlier. An attempt to further leverage the presented approach with the contraction-operator transformation, which can potentially further accelerate FE forward modelling, will be performed. Another way to gain robustness would be to substitute point smoothing with line smoothing (Mulder 2006).

Arbitrary unstructured tetrahedral and hexahedral grids or grids with hanging nodes presumably cannot be used with the presented approach. Our consideration was limited to lowest order Nédélec basis functions for logically rectangular grids, while the developed pre-conditioner can be directly reused for higher-order basis functions. We will also try to reuse this approach for time-domain modelling, where accurate incorporation of land topography is of paramount importance.

ACKNOWLEDGEMENTS

The authors acknowledge the Skoltech, MIPT, University of Utah's Consortium for Electromagnetic Modeling and Inversion (CEMI) and TechnoImaging for help with this research and permission to publish. The authors appreciate fruitful discussions with their

Downloaded from https://academic.oup.com/gji/article/223/2/840/5871825 by University of Utah Eccles Health Sciences Library user on 04 September 2020

colleagues, Drs Hongzhu Cai, Mikhail Malovichko and Pavel Pushkarev. We also thank Drs G. Egbert and A. Kelbert for the opportunity to use ModEM software in this study. We would like to appreciate the Skoltech CDISE's high-performance computing cluster, Zhores (Zacharov *et al.* 2019), for providing the computing resources that have contributed to the results reported herein.

This research was supported by the Russian Science Foundation, project No. 18-71-10071.

REFERENCES

Amestoy, P.R., Duff, I.S., Koster, J. & L'Excellent, J.-Y., 2001. A fully asynchronous multifrontal solver using distributed dynamic scheduling, *SIAM J. Matrix Anal. Appl.,* **23,** 15–41.

Arnold, D.N., Falk, R.S. & Winther, R., 2000. Multigrid in H(div) and H(curl), *Numer. Math.,* **85,** 197–217.

Cai, H., Xiong, B., Han, M. & Zhdanov, M., 2014. 3D controlled-source electromagnetic modeling in anisotropic medium using edge-based finite element method, *Comput. Geosci.,* **73,** 164–176.

Cai, H., Hu, X., Li, J., Endo, M. & Xiong, B., 2017. Parallelized 3D CSEM modeling using edge-based finite element with total field formulation and unstructured mesh, *Comput. Geosci.,* **99,** 125–134.

Da Silva, N.V., Morgan, J.V., MacGregor, L. & Warner, M., 2012, A finite element multifrontal method for 3D CSEM modeling in the frequency domain, *Geophysics,* **77**(2), E101–E115.

Daudina, D., Malovichko, M., Myasoedov, N., Nikishin, A., Poludetkina, E., Tikhotskiy, S. & Tokarev, M., 2014. Joint inversion of multi-type geophysical and geochemical data for hydrocarbon systems exploration at sea shelf, in *6th EAGE Saint Petersburg International Conference and Exhibition,* https://doi.org/10.3997/2214-4609.20140152.

Egbert, G.D. & Kelbert, A., 2012. Computational recipes for electromagnetic inverse problems, *Geophys. J. Int.,* **189**(1), 251–267.

Falk, R.S., Gatto, P. & Monk, P., 2011. Hexahedral H(div) and H(curl) finite elements, *ESAIM: M2AN,* **45**(1), 115–143.

Grayver, A.V. & Kolev, Tz.V., 2015, Large-scale 3D geoelectromagnetic modeling using parallel adaptive high-order finite element method, *Geophysics,* **80**(6), E277–E291.

Haber, E. & Heldmann, S., 2007. An octree multigrid method for quasi-static Maxwell's equations with highly discontinuous coefficients, *J. Comput. Phys.,* **223**(2), 783–796.

Heikkola, E., Kuznetsov, Y.A. & Lipnikov, K.N., 1999, Fictitious domain methods for the numerical solution of three-dimensional acoustic scattering problems, *J. Comput. Acoust.,* **7**(3), 161–183.

Hiptmair, R., 1998. Multigrid method for Maxwell's equations, *SIAM J. Numer. Anal*., **36**(1), 204–225.

Kelbert, A., Meqbel, N., Egbert, G.D. & Tandon, K., 2014. ModEM: a modular system for inversion of electromagnetic geophysical data, *Comput. Geosci.,* **66,** 40–53.

Key, K., 2016. MARE2DEM: a 2-D inversion code for controlled-source electromagnetic and magnetotelluric data, *Geophys. J. Int.,* **207**(1), 571–588.

Kordy, M., Wannamaker, P., Maris, V., Cherkaev, E. & Hill, G., 2016. 3-D magnetotelluric inversion including topography using deformed hexahedral edge finite elements and direct solvers parallelized on SMP computers – Part I: forward problem and parameter Jacobians, *Geophys. J. Int.,* **204,** 74–93.

Li, J., Farquharson, C.G. & Hu, X., 2016. 3D vector finite-element electromagnetic forward modeling for large loop sources using a total-field algorithm and unstructured tetrahedral grids, *Geophysics,* **82**(1), E1–E16.

Mulder, W.A., 2006, A multigrid solver for 3D electromagnetic diffusion, *Geophys. Prospect.,* **54,** 633–649.

Nédélec, J.-C., 1986. A new family of mixed finite elements in R^3, *Numer. Math.,* **50**(1), 57–81.

Puzyrev, V., Koldan, J., de la Puente, J., Houzeaux, G., Vázquez, M. & Cela, J.M., 2013, A parallel finite-element method for three-dimensional controlled-source electromagnetic forward modelling, *Geophys. J. Int.,* **193**(2), 678–693.

Ren, Zh., Kalscheuer, Th., Greenhalgh, St. & Maurer, H., 2014, A finite-element-based domain-decomposition approach for plane wave 3D electromagnetic modeling, *Geophysics,* **79**(6), E255–E268.

Um, E.S., Commer, M. & Newman, G.A., 2013. Efficient pre-conditioned iterative solution strategies for the electromagnetic diffusion in the Earth: finite-element frequency-domain approach, *Geophys. J. Int.,* **193**(3), 1460–1473.

Yavich, N. & Scholl, C., 2012. Advances in multigrid solution of 3D forward MCSEM problem, in *5th EAGE St. Petersburg International Conference and Exhibition on Geosciences,* EAGE. https://doi.org/10.3997/2214-4609.20143665

Yavich, N. & Zhdanov, M., 2016. Contraction pre-conditioner in finite-difference electromagnetic modelling, *Geophys. J. Int.,* **206,** 1718–1729.

Yavich, N., Malovichko, M., Khokhlov, N. & Zhdanov, M., 2017. Advanced method of FD electromagnetic modeling based on contraction operator, *79th EAGE Conference and Exhibition,* https://doi.org/10.3997/2214-4609.201701354.

Yavich, N., Malovichko, M. & Zhdanov, M., 2019, Towards efficient finite-element EM modeling on hexahedral grids, *81st EAGE Conference and Exhibition,* https://doi.org/10.3997/2214-4609.201901487.

Yavich, N., Malovichko, M. & Shlykov, A., 2020. Parallel simulation of audio- and radio-magnetotelluric data, *Minerals,* **10**(1), 42.

Zacharov, I. *et al.* 2019. "Zhores"—Petaflops supercomputer for data-driven modeling, machine learning and artificial intelligence installed in Skolkovo Institute of Science and Technology, *Open Eng.,* **9,** 512–520.

Zaslavsky, M., Druskin, V., Davydycheva, S., Knizhnerman, L., Abubakar, A. & Habashy, T., 2011, Hybrid finite-difference integral equation solver for 3D frequency domain anisotropic electromagnetic problems, *Geophysics,* **76**(2), F123–F137.

Zhdanov, M., 2002. *Geophysical Inverse Theory and Regularization Problems*. Elsevier.

Zhdanov, M., 2009, *Geophysical Electromagnetic Theory and Methods*. Elsevier.

APPENDIX A: FAST INVERSION OF $\mathbf{A}_{b\ \mathrm{FD}}$

In this appendix, we describe an approach for direct factorization of matrix $\mathbf{A}_{b\ \mathrm{FD}}$, an FD matrix corresponding to Maxwell equations in a possibly anisotropic medium of conductivity $\boldsymbol{\sigma}_b(z)$. Eq. (2) in this case read as follows in explicit form:

$$\begin{aligned}
-\frac{\partial^2 E_x}{\partial y^2} - \frac{\partial^2 E_x}{\partial z^2} + \frac{\partial^2 E_y}{\partial x \partial y} + \frac{\partial^2 E_z}{\partial x \partial z} - i\omega\mu_0\sigma_{bx}(z)E_x = i\omega\mu_0 J_x,\\
-\frac{\partial^2 E_y}{\partial x^2} - \frac{\partial^2 E_y}{\partial z^2} + \frac{\partial^2 E_x}{\partial x \partial y} + \frac{\partial^2 E_z}{\partial y \partial z} - i\omega\mu_0\sigma_{by}(z)E_y = i\omega\mu_0 J_y,\\
-\frac{\partial^2 E_z}{\partial x^2} - \frac{\partial^2 E_z}{\partial y^2} + \frac{\partial^2 E_x}{\partial x \partial z} + \frac{\partial^2 E_y}{\partial y \partial z} - i\omega\mu_0\sigma_{bz}(z)E_z = i\omega\mu_0 J_z.
\end{aligned} \tag{A1}$$

We assume that the non-uniform FD grid is formed by $N_x \times N_y \times N_z$ cells. In complexity and size estimates, we assume the numbers of grid cells in all directions are of the same order. We will present an algorithm to efficiently solve

$$\mathbf{A}_{b\ \mathrm{FD}} v = g \tag{A2}$$

for v while $g \in \mathbb{C}^n$ is given. For magnetic field modelling on Lebedev grids, a similar algorithm was presented in Zaslavsky *et al.* (2011). The below-described approach is analogous to using double Fourier transform for solution of differential equation system (A1).

Matrix $\mathbf{A}_{b\ \mathrm{FD}}$ inherits the structure of eq. (A1). It can be presented in the following way:

$$\mathbf{A}_{b\ \mathrm{FD}} = \begin{pmatrix} A_{xx} & A_{xy} & A_{xz} \\ A_{yx} & A_{yy} & A_{yz} \\ A_{zx} & A_{zy} & A_{zz} \end{pmatrix}. \tag{A3}$$

Downloaded from https://academic.oup.com/gji/article/223/2/840/5871825 by University of Utah Eccles Health Sciences Library user on 04 September 2020

We will use a special representation of matrix $\mathbf{A}_{b\ \mathrm{FD}}$. To introduce it, we need some auxiliary notations. Note that discretization of eq. (A1) involves backward, forward and central FE operators. The Kronecker product (Appendix B) will let us write components of $\mathbf{A}_{b\ \mathrm{FD}}$ explicitly,

$$
\begin{aligned}
A_{xx} &= I_d^z \otimes A_d^y \otimes I_n^x + A_d^z \otimes I_d^y \otimes I_n^x - i\omega\mu_0\Sigma^x \otimes I_d^y \otimes I_n^x,\\
A_{xy} &= I_d^z \otimes F_{dn}^y \otimes F_{nd}^x,\\
A_{xz} &= I_{dn}^z \otimes I_d^y \otimes F_{nd}^x,\\
&\ldots\\
A_{zz} &= I_n^z \otimes A_d^y \otimes I_d^x + I_n^z \otimes I_d^y \otimes A_d^x - i\omega\mu_0\Sigma^Z \otimes I_d^y \otimes I_n^x.
\end{aligned}
\tag{A4}
$$

Here, I_d^x, I_d^y, etc., are identity matrices; Σ^x, Σ^y and Σ^z are diagonal matrices corresponding to $\sigma_{bx}(z)$, $\sigma_{by}(z)$ and $\sigma_{bz}(z)$, respectively; F_{nd}^x and F_{dn}^y are backword and forward difference operators resulting from discretization of $\frac{\partial^2}{\partial x\partial y}$; matrices A_d^x, etc., correspond to discretization of $\frac{\partial^2}{\partial x^2}$. An important feature of all the matrices involved in the right-hand side of eq. (A4) is that they are either diagonal, bidiagonal or tridiagonal. Also they are relatively small in size, $O(N_x)$, since they correspond to 1-D ordinary differential operators. However, the blocks A_{xx}, etc., are large and have a large band.

We will now simplify the structure of $\mathbf{A}_{b\ \mathrm{FD}}$ and invert it using spectral and singular value decompositions of earlier introduced matrices A_d^x, F_{nd}^x, etc. This is achieved in three steps.

Step 1–Symmetrization. Matrix $\mathbf{A}_{b\ \mathrm{FD}}$ is not symmetric generally. We can symmetrize it by using a diagonal matrix,

$$
\mathbf{D} = \operatorname{diag}\left(D_d^z \otimes D_d^y \otimes D_n^x,\ \ D_d^z \otimes D_n^y \otimes D_d^x,\ \ D_n^z \otimes D_d^y \otimes D_d^x\right). \tag{A5}
$$

Here, D_d^x, etc., are diagonal matrices of grid steps in the respective directions. Now we can put,

$$
\tilde{\mathbf{A}} = \mathbf{D}^{\frac{1}{2}}\ \mathbf{A}_{b\ \mathrm{FD}}\ \mathbf{D}^{-\frac{1}{2}}. \tag{A6}
$$

Now solving (A2) is equivalent to solving

$$
\tilde{\mathbf{A}}\,\tilde{v} = \tilde{g}\,, \tag{A7}
$$

with

$$
\tilde{v} = \mathbf{D}^{\frac{1}{2}}\ v,\quad \tilde{g} = \mathbf{D}^{\frac{1}{2}}\ g. \tag{A8}
$$

After this scaling, matrix $\tilde{A}$ is always symmetric.

Step 2–Diagonalization. Matrices involved in $\tilde{\mathbf{A}}$ can now be diagonalized using eigenvalue decomposition. At this step we diagonalize those matrices that correspond to FD differentiation in the horizontal direction. The respective eigenvector basis is denoted as W_d^x, W_d^y, etc. Define,

$$
\mathbf{W} = \operatorname{diag}\left(I_d^z \otimes W_d^y \otimes W_n^x,\ \ I_d^z \otimes W_n^y \otimes W_d^x,\ \ I_n^z \otimes W_d^y \otimes W_d^x\right). \tag{A9}
$$

and put

$$
\mathbf{T} = \mathbf{W}^T\ \tilde{\mathbf{A}}\,\mathbf{W}. \tag{A10}
$$

Matrix $\mathbf{T}$ benefits from the eigenvector basis and has the following form:

$$
\mathbf{T} = \begin{pmatrix} T_{xx} & T_{xy} & T_{xz}\\ T_{yx} & T_{yy} & T_{yz}\\ T_{zx} & T_{zy} & T_{zz}\end{pmatrix}, \tag{A11}
$$

$$
\begin{aligned}
T_{xx} &= I_d^z \otimes \Lambda_d^y \otimes I_n^x + A_d^z \otimes I_d^y \otimes I_n^x - i\omega\mu_0\Sigma^x \otimes I_d^y \otimes I_n^x,\\
T_{xy} &= -I_d^z \otimes (\Lambda_{nd}^y)^{\frac{1}{2}T} \otimes (\Lambda_{nd}^x)^{\frac{1}{2}},\\
T_{xz} &= -F_{dn}^z \otimes I_d^y \otimes (\Lambda_{nd}^x)^{\frac{1}{2}},\\
&\ldots\\
T_{zz} &= I_n^z \otimes \Lambda_d^y \otimes I_d^z + I_n^z \otimes I_d^y \otimes \Lambda_d^x - i\omega\mu_0\Sigma^z \otimes I_d^y \otimes I_n^x.
\end{aligned}
\tag{A12}
$$

Switching from $\tilde{\mathbf{A}}$ to $\mathbf{T}$ in (A7) gives us,

$$
\mathbf{T}\,u = f,\quad f = \mathbf{W}^T\,\tilde{g},\quad u = \mathbf{W}^T\,\tilde{v}. \tag{A13}
$$

An important feature of the above matrix is that all FD derivatives with respect to x and y are gone. All of the nine blocks of $\mathbf{T}$ are either diagonal, bidiagonal or tridiagonal. Consequently, we can factorize it efficiently (see the next step).

Step 3–Factorization. In order to solve eq. (A13), it remains to discuss an approach to factorize $\mathbf{T}$. As it was noted above, matrix $\mathbf{T}$ involves only FD derivatives with respect to z. Further note that T_{zz} block is diagonal. Consequently, we can eliminate the respective subset of unknowns while keeping sparsity of the remaining blocks of the matrix. Under an appropriate renumbering, the obtained matrix will be *block diagonal* with $O(N_x N_y)$ blocks and each block being a symmetric seven-diagonal matrix of size $O(N_z)$. Thus, we can factorize every block using LDL^T-algorithm in linear time. This is performed on-the-fly.

Solution of eq. (A2) can be summarized as follows: apply diagonal scaling $\mathbf{D}$ (A8), then perform conversion to the eigenvalue basis (A13), after this, find the discrete harmonics, then perform conversion from the eigenvalue basis, and finally remove diagonal scaling.

All the operations performed in the algorithm above are linear with respect to the problem size n except conversions from/to the eigenvalue basis which require $O(N_z N_x N_y(N_x + N_y))$ or $O(n^{\frac{4}{3}})$ operations. Thus, the latter complexity dominates in the solution of eq. (A2). At initialization, eigenvalue decomposition of four tridiagonal matrices is performed. The overall complexity of this step is $O\,(n_x^2) = O(n^{\frac{2}{3}})$.

APPENDIX B: KRONECKER PRODUCT AND ITS PROPERTIES

Given an $m \times n$ matrix $\mathbf{A} = \{a_{ij}\}$ and some matrix $\mathbf{B}$, their Kronecker product matrix $\mathbf{A} \otimes \mathbf{B}$ is a block matrix defined as

$$
\mathbf{A} \otimes \mathbf{B} = \begin{pmatrix} a_{11}B & \cdots & a_{1n}B\\ & \cdots & \\ a_{m1}B & \cdots & a_{mn}B\end{pmatrix}. \tag{B1}
$$

We used the following properties of this product within this note:

$$
(\mathbf{A} \otimes \mathbf{B})^T = \mathbf{A}^T \otimes \mathbf{B}^T, \tag{B2}
$$

$$
(\mathbf{A} \otimes \mathbf{B})^{-1} = \mathbf{A}^{-1} \otimes \mathbf{B}^{-1}, \tag{B3}
$$

$$
(\mathbf{A} \otimes \mathbf{B})\ (\mathbf{C} \otimes \mathbf{D}) = (\mathbf{AC}) \otimes (\mathbf{BD}). \tag{B4}
$$

The identity (B3) holds under the assumption that $\mathbf{A}$ and $\mathbf{B}$ are invertible, while (B4) holds under the assumption that the respective matrix products are well defined.

Downloaded from https://academic.oup.com/gji/article/223/2/840/5871825 by University of Utah Eccles Health Sciences Library user on 04 September 2020