

Large-scale 3D geoelectromagnetic modeling using parallel adaptive high-order finite element method

Journal Article**Author(s):**

Grayver, Alexander; Kolev, Tzanio V.

Publication date:

2015-11

Permanent link:

<https://doi.org/10.3929/ethz-b-000103394>

Rights / license:

[In Copyright - Non-Commercial Use Permitted](#)

Originally published in:

Geophysics 80(6), <https://doi.org/10.1190/GEO2015-0013.1>

Large-scale 3D geoelectromagnetic modeling using parallel adaptive high-order finite element method

Alexander V. Grayver¹ and Tzanio V. Kolev²

ABSTRACT

We have investigated the use of the adaptive high-order finite-element method (FEM) for geoelectromagnetic modeling. Because high-order FEM is challenging from the numerical and computational points of view, most published finite-element studies in geoelectromagnetics use the lowest order formulation. Solution of the resulting large system of linear equations poses the main practical challenge. We have developed a fully parallel and distributed robust and scalable linear solver based on the optimal block-diagonal and auxiliary space preconditioners. The solver was found to be efficient for high finite element orders, unstructured and non-conforming locally refined meshes, a wide range of frequencies, large conductivity contrasts, and number of degrees of freedom (DoFs). Furthermore, the presented linear solver is in essence algebraic; i.e., it acts on the matrix-vector level and thus requires no information about the discretization, boundary conditions, or physical source used, making it readily efficient for a wide range of electromagnetic modeling problems. To get accurate solutions at reduced computational cost, we have also implemented goal-oriented adaptive mesh refinement. The numerical tests indicated that if highly accurate modeling results were required, the high-order FEM in combination with the goal-oriented local mesh refinement required less computational time and DoFs than the lowest order adaptive FEM.

INTRODUCTION

Electromagnetic (EM) methods of geophysics aim at studying subsurface electric conductivity distribution. Because the latter depends on rock composition, fluid content, temperature, and pres-

sure, among others, these methods are widely used in academic and industry environments (Key, 2012; Muñoz, 2014). For a feasibility study or an inversion, EM fields governed by Maxwell's equations need to be calculated for a given 3D electric conductivity distribution. Very often, a large number of such calculations are required (Haber, 2014). Therefore, to accommodate the complexity of the earth and keep the required simulation time feasible, numerical methods capable of delivering accurate solutions and harnessing computational power provided by modern computers are essential.

In this work, we have used the finite-element method (FEM) to calculate EM fields for an arbitrary 3D distribution of electric conductivity. Much research in recent years has gone into developing FEM codes for geoelectromagnetic modeling (Börner, 2010; Farquharson and Miensoopust, 2011; Schwarzbach et al., 2011; Ren et al., 2013; Um et al., 2013; Grayver and Bürg, 2014). They all make use of the so-called Nédélec finite elements, which permit a well-posed representation of EM fields taking into account discontinuities of the normal components (Jin, 2002). In addition to the well-established theory (Monk, 2003), FEM provides elaborated methods for a posteriori error estimation (Ainsworth and Oden, 2000). In combination with adaptive mesh refinement, also known as *h-refinement*, this approach allows us to obtain accurate numerical solutions at decreased computational cost. Of particular interest in geoelectromagnetics is a class of techniques called *goal-oriented error estimators* (Bangerth and Rannacher, 2003). Goal-oriented error estimators steer local mesh refinement in a way that facilitates accurate EM fields at the specified receiver locations using as few degrees of freedom (DoFs) as possible. Recently, Key and Owall (2011) and Ren et al. (2013) demonstrated the efficacy of this approach in the context of geoelectromagnetic modeling.

In this work, we consider a 3D modeling domain split into a set of hexahedra finite elements with a Nédélec basis attached to them. Within each finite element, the solution is typically described by a specific vector polynomial. Similar to the *h-refinement* aimed at increasing the solution accuracy by refining a mesh, one can perform

Manuscript received by the Editor 6 January 2015; revised manuscript received 10 April 2015; published online 26 August 2015; corrected version published online 2 September 2015.

¹Institute of Geophysics, ETH Zürich, Zürich, Switzerland. E-mail: agrayver@erdw.ethz.ch.

²Center for Applied Scientific Computing, Lawrence Livermore National Laboratory, Livermore, California, USA. E-mail: tzanio@llnl.gov.

© 2015 Society of Exploration Geophysicists. All rights reserved.

p -refinement by increasing the polynomial degrees of the finite element. It is well known that for smooth solutions, p -refinement is advantageous and that combining it with h -refinement (the so-called h - p FEM) can lead to exponential convergence rates (Guo and Babuska, 1986; Bürg, 2013). Although h -refinement has been applied in geoelectromagnetic applications (Schwarzbach et al., 2011; Ren et al., 2013), little work has been performed to investigate advantages of higher polynomial degrees for 3D geoelectromagnetic modeling. Most existing FEM modeling codes in this area use the lowest order Nédélec finite elements, often called *edge-based finite elements*. The notable exception is the work of Schwarzbach et al. (2011), who demonstrate that high-order finite elements require fewer DoFs to achieve a certain level of accuracy when compared with the lowest order case. This was shown for the case of a half-space model with the largely smooth solution. High-order Nédélec elements have also been beneficial in EM simulations in other areas, for example, Rieben et al. (2004) and Luo et al. (2011).

The accuracy achieved using high polynomial degrees, however, comes at a cost. First, high-order Nédélec elements are more technical to implement (Zaglmayr, 2006; Bürg, 2012a). To facilitate p -refinement, one also typically needs a hierarchical basis, which is the case we consider in this paper. When using high-order finite elements, the number of DoFs per element gets larger. Some of the DoFs are associated with edges and faces of the element. Hence, they are coupled with DoFs of the elements that share the same edge or face. This extra coupling incurs additional nonzero elements in the system matrix (Šolín et al., 2004). Therefore, higher order polynomials generally decrease sparsity: The stencil size for a method of order p in 3D is $\mathcal{O}(p^3)$ with p being the polynomial degree of the Nédélec element. Furthermore, when building a system matrix or estimating solution error, it is necessary to numerically calculate integrals over elements. This is performed by using quadratures. The order of a quadrature used for integration needs to be proportional to the order of the finite element polynomial. As a result, higher polynomial orders also increase the time spent in building a system matrix, right-side vectors, and estimating solution errors. If implemented naively, a matrix assembly, for example, could require $\mathcal{O}(p^9)$ operations, although with better algorithms the cost can be reduced (Ainsworth et al., 2011; Kirby and Thinh, 2012). However, the largest part of the computational time is typically spent in solving the resulting large and sparse system of linear equations. Construction of robust and scalable solvers is challenging and requires elaborated methods (Grayver and Bürg, 2014; Ren et al., 2014). High polynomial degrees create an additional challenge here. Ainsworth and Coyle (2003) show that in 3D the condition number of a system matrix grows as $\mathcal{O}(p^8)$. Therefore, for a large p , the system matrix not only gets denser but also more ill conditioned, rendering the problem of developing an efficient solver extremely difficult and important.

The above considerations bring up the question: Is using high-order FEM for geoelectromagnetic modeling justified? On one hand, we are interested in having fast modeling codes, and h -refinement together with a fully automatic goal-oriented error estimator already offers a highly efficient tool. On the other hand, we know a priori that the sought solution will exhibit smooth behavior in large parts of a modeling domain (e.g., air or padding cells) due to its diffusive nature (Løseth et al., 2006). In this case, adaptive h -refinement in combination with high polynomial degrees is expected to provide more accurate solutions within a shorter time than h -refine-

ment alone. The purpose of this paper is to address this question. The challenges mentioned earlier are mitigated by using a fully distributed implementation with a robust and scalable solver. To the best of our knowledge, this work is the first study that develops a robust and scalable numerical scheme for 3D geoelectromagnetic modeling using high-order FEM.

It is worth mentioning that the numerical methods and results derived below are essentially independent of the type of physical source. In other words, the presented methods remain valid for the total and secondary field formulations, plane wave or point dipole sources, and the isotropic and anisotropic electric conductivities. These factors do not change the nature of the underlying equation, and as a result, the presented numerical machinery remains valid. Still, to preserve brevity and focus on the topic, we will use magnetotellurics (MT) (Berdichevskii and Dmitriev, 2008; Chave and Jones, 2012) for all numerical experiments.

PROBLEM FORMULATION

Governing equations

We work with the following partial differential equation (PDE) derived from the Maxwell's equations with the time dependence expressed by $e^{i\omega t}$

$$\nabla \times (\mu^{-1} \nabla \times \vec{E}^s) + i\omega\sigma\vec{E}^s = -i\omega(\sigma - \sigma^0)\vec{E}^0 \text{ in } \Omega, \quad (1)$$

where $\Omega \subseteq \mathbb{R}^3$ is a bounded modeling domain, ω is the angular frequency, μ is the magnetic permeability, $\vec{E}^s \in \mathbb{C}^3$ is the complex-valued scattered electric field, and $\sigma: \Omega \rightarrow \mathbb{R}$ denotes the real electric conductivity; i.e., we neglected the displacement currents because they are irrelevant at frequencies of interest. Furthermore, σ^0 is a background model conductivity represented by a horizontally layered medium and $\vec{E}^0 = \vec{E}^0(\omega, \sigma^0, z)$ is an incident electric field calculated analytically.

Equation 1 needs to be augmented with predefined boundary conditions to guarantee uniqueness (Monk, 2003). Conventionally, Dirichlet boundary conditions are used. In this case, tangential components of the scattered field are prescribed to be zero at boundaries; i.e.,

$$\mathbf{n} \times \vec{E}^s = 0 \quad \text{on } \Gamma, \quad (2)$$

where \mathbf{n} is an outward-pointing normal vector on $\Gamma = \partial\Omega$. Alternatively, Neumann boundary conditions can be prescribed, which constrain tangential components of the curl of the scattered electric field as

$$\mathbf{n} \times \frac{1}{\mu} \nabla \times \vec{E}^s = 0 \quad \text{on } \Gamma. \quad (3)$$

From our experience, Neumann boundary conditions often require fewer padding cells because they constrain variations in the field rather than the field itself. In this case, smaller modeling domains can be used without big accuracy deteriorations. Our approach works with both types of boundary conditions, but for simplicity, in the rest of this section, we focus on the Dirichlet case.

The total electric and magnetic fields are calculated as

$$\begin{aligned}\vec{E} &= \vec{E}^0 + \vec{E}^s \\ \vec{H} &= \vec{H}^0 - \frac{1}{i\omega\mu} \nabla \times \vec{E}^s,\end{aligned}\quad (4)$$

where in the latter equation, we made use of Faraday's law.

By splitting $\vec{E}^s := \vec{E}_R^s + i\vec{E}_I^s$ and $\vec{E}^0 := \vec{E}_R^0 + i\vec{E}_I^0$ into real and imaginary parts and inserting into equation 1, we get a system of two coupled real-valued equations:

$$\begin{aligned}\nabla \times (\mu^{-1} \nabla \times \vec{E}_R^s) - \omega\sigma \vec{E}_I^s &= \omega(\sigma - \sigma^0) \vec{E}_I^0 \\ \nabla \times (\mu^{-1} \nabla \times \vec{E}_I^s) + \omega\sigma \vec{E}_R^s &= -\omega(\sigma - \sigma^0) \vec{E}_R^0,\end{aligned}\quad (5)$$

with the corresponding boundary conditions split accordingly.

In the rest of the paper, we work with equations 5 because they permit us to develop an efficient solver as described in the following sections.

Discretization using finite elements

First, we define the space of the 3D vector functions with curl in L^2 as

$$\mathcal{H}(\text{curl}, \Omega) = \{\mathbf{u} \in (L^2(\Omega))^3 \mid \nabla \times \mathbf{u} \in (L^2(\Omega))^3\}.\quad (6)$$

Furthermore, $\mathcal{H}_0(\text{curl}, \Omega)$ denotes the functions $\mathbf{u} \in \mathcal{H}(\text{curl}, \Omega)$, which also satisfy the homogeneous Dirichlet boundary conditions; i.e., $\mathbf{n} \times \mathbf{u} = 0$ on Γ .

Then, system 2 is multiplied by test functions $\boldsymbol{\psi} \in \mathcal{H}_0(\text{curl}, \Omega)$. Using integration by parts, we obtain the weak formulation that aims at finding $\vec{E}_R^s, \vec{E}_I^s \in \mathcal{H}_0(\text{curl}, \Omega)$ such that

$$\begin{aligned}\langle \mu^{-1} \nabla \times \vec{E}_R^s, \nabla \times \boldsymbol{\psi} \rangle - \omega \langle \sigma \vec{E}_I^s, \boldsymbol{\psi} \rangle &= \omega \langle (\sigma - \sigma^0) \vec{E}_I^0, \boldsymbol{\psi} \rangle \\ \langle \mu^{-1} \nabla \times \vec{E}_I^s, \nabla \times \boldsymbol{\psi} \rangle + \omega \langle \sigma \vec{E}_R^s, \boldsymbol{\psi} \rangle &= -\omega \langle (\sigma - \sigma^0) \vec{E}_R^0, \boldsymbol{\psi} \rangle\end{aligned}\quad (7)$$

for all $\boldsymbol{\psi} \in \mathcal{H}_0(\text{curl}, \Omega)$. Here, the inner product is given by

$$\langle \mathbf{a}, \mathbf{b} \rangle_{L^2(\Omega)} := \int_{\Omega} \mathbf{a} \cdot \mathbf{b}.\quad (8)$$

Let the continuous domain Ω be split into a set \mathbb{T} of nonoverlapping hexahedral cells. To meet the physics of the EM phenomena, we permit normal components of the electric field to be discontinuous, but we require continuous tangential components. It turns out that the functional space $\mathcal{H}_0(\text{curl}, \Omega)$ consists of functions that satisfy these continuity constraints (Monk, 2003). Now, we need to discretize this functional space on our mesh \mathbb{T} . In EM modeling with FEM, Nédélec elements are conventionally adopted because they are a natural finite dimensional subspace of $\mathcal{H}_0(\text{curl}, \Omega)$. Most geoelectromagnetic modeling codes use Nédélec elements of order one (Bossavit, 1998; Monk, 2003) also known as *edge-based ele-*

ments. This finite element has 12 shape functions associated with edges of a hexahedron. For higher polynomial degrees, more shape functions appear also on the faces and in the interior part of the element. The total number N_s of Nédélec shape functions for the d -dimensional hexahedral element of order p can be calculated as (Bürg, 2012a)

$$N_s = (2^{d-1}d + 12(d-2)(p-1) + d(p-1)^{d-1})p.\quad (9)$$

Because we used a real-valued formulation (equation 5), half of the DoFs in an element correspond to the real part and another half to the imaginary part of a field. Table 1 lists the number of DoFs per element for the polynomial degrees that we used in this work. As more DoFs on edges and faces are generated, the coupling between elements increases. Furthermore, to approximate the integral in equation 8, quadratures are used (Šolín et al., 2004). To integrate high-order shape functions with sufficient accuracy, we have to use large quadrature orders. Accordingly, the last column in Table 1 shows the number of points in the constructed quadrature for corresponding polynomial degree.

Let $\mathbf{V}(\mathbb{T})$ be a high-order finite-dimensional Nédélec approximation subspace of $\mathcal{H}_0(\text{curl}, \Omega)$ defined on \mathbb{T} . Then, we seek $\mathbf{E}_R^s, \mathbf{E}_I^s \in \mathbf{V}(\mathbb{T})$ such that

$$\begin{aligned}\langle \mu^{-1} \nabla \times \mathbf{E}_R^s, \nabla \times \boldsymbol{\psi} \rangle - \omega \langle \sigma \mathbf{E}_I^s, \boldsymbol{\psi} \rangle &= \omega \langle (\sigma - \sigma^0) \vec{E}_I^0, \boldsymbol{\psi} \rangle \\ \langle \mu^{-1} \nabla \times \mathbf{E}_I^s, \nabla \times \boldsymbol{\psi} \rangle + \omega \langle \sigma \mathbf{E}_R^s, \boldsymbol{\psi} \rangle &= -\omega \langle (\sigma - \sigma^0) \vec{E}_R^0, \boldsymbol{\psi} \rangle\end{aligned}\quad (10)$$

for all $\boldsymbol{\psi} \in \mathbf{V}(\mathbb{T})$. In this case, the solutions $\mathbf{E}_R^s, \mathbf{E}_I^s$ of equation 10 are expanded using the Nédélec basis:

$$\mathbf{E}_R^s = \sum_{i=1}^n \alpha_i \boldsymbol{\varphi}_i \quad \text{and} \quad \mathbf{E}_I^s = \sum_{i=1}^n \beta_i \boldsymbol{\varphi}_i.\quad (11)$$

Inserting these into formulation 10 with $\boldsymbol{\psi} = \boldsymbol{\varphi}_j$ for any $j = 1, \dots, n$ yields

Table 1. Numbers of real-valued DoFs per Nédélec finite element and corresponding number of points for quadrature used to approximate the integral in equation 8 versus polynomial order p . Note that due to the real-valued formulation 5, there are two sets of DoFs for E_R^s and E_I^s . The total number of DoFs is shown in the table.

Polynomial degree p	Number of DoFs	Quadrature size
1	24	27
2	108	64
3	288	125
4	600	216
5	1080	343
6	1764	512

$$\begin{aligned}
& \sum_{i=1}^n \alpha_i \langle \mu^{-1} \nabla \times \boldsymbol{\varphi}_i, \nabla \times \boldsymbol{\varphi}_j \rangle - \omega \sum_{i=1}^n \beta_i \langle \sigma \boldsymbol{\varphi}_i, \boldsymbol{\varphi}_j \rangle \\
&= \langle \omega(\sigma - \sigma^0) \vec{E}_I^0, \boldsymbol{\varphi}_j \rangle \\
& \sum_{i=1}^n \beta_i \langle \mu^{-1} \nabla \times \boldsymbol{\varphi}_i, \nabla \times \boldsymbol{\varphi}_j \rangle + \omega \sum_{i=1}^n \alpha_i \langle \sigma \boldsymbol{\varphi}_i, \boldsymbol{\varphi}_j \rangle \\
&= -\langle \omega(\sigma - \sigma^0) \vec{E}_R^0, \boldsymbol{\varphi}_j \rangle. \tag{12}
\end{aligned}$$

Rewriting the above equations in a matrix-vector form, we obtain a discrete system of linear equations:

$$\begin{pmatrix} \mathbf{C} & -\mathbf{M} \\ \mathbf{M} & \mathbf{C} \end{pmatrix} \begin{pmatrix} \vec{\mathbf{E}}_R^s \\ \vec{\mathbf{E}}_I^s \end{pmatrix} = \begin{pmatrix} \vec{\mathbf{s}}_R \\ -\vec{\mathbf{s}}_I \end{pmatrix}, \tag{13}$$

where the high-order Nédélec DoFs, $\vec{\mathbf{E}}_R^s = (\alpha_1, \dots, \alpha_n)^T$ and $\vec{\mathbf{E}}_I^s = (\beta_1, \dots, \beta_n)^T$, are the solution vectors of this linear system of equations. The entries of the sparse, symmetric, positive semi-definite (SPSD) matrix $\mathbf{C} \in \mathbb{R}^{n \times n}$ and the sparse, symmetric, positive-definite (SPD) matrix $\mathbf{M} \in \mathbb{R}^{n \times n}$ are given by

$$[\mathbf{C}]_{ij} = \langle \mu^{-1} \nabla \times \boldsymbol{\varphi}_i, \nabla \times \boldsymbol{\varphi}_j \rangle \quad \text{and} \quad [\mathbf{M}]_{ij} = \omega \langle \sigma \boldsymbol{\varphi}_i, \boldsymbol{\varphi}_j \rangle, \tag{14}$$

respectively. The right-side vectors are simply

$$\begin{aligned}
[\vec{\mathbf{s}}_R]_j &= \langle \omega(\sigma - \sigma^0) \vec{E}_I^0, \boldsymbol{\varphi}_j \rangle \quad \text{and} \\
[\vec{\mathbf{s}}_I]_j &= \langle \omega(\sigma - \sigma^0) \vec{E}_R^0, \boldsymbol{\varphi}_j \rangle. \tag{15}
\end{aligned}$$

Multiplying the second row in equation 13 by minus one yields a symmetric system of linear equations:

$$\underbrace{\begin{pmatrix} \mathbf{C} & -\mathbf{M} \\ -\mathbf{M} & \mathbf{C} \end{pmatrix}}_{\mathbf{A}} \underbrace{\begin{pmatrix} \vec{\mathbf{E}}_R^s \\ \vec{\mathbf{E}}_I^s \end{pmatrix}}_{\vec{\mathbf{E}}^s} = \underbrace{\begin{pmatrix} \vec{\mathbf{s}}_R \\ \vec{\mathbf{s}}_I \end{pmatrix}}_{\vec{\mathbf{s}}}. \tag{16}$$

The efficient solution of the above linear system with high-order finite elements on locally refined nonconforming meshes is the major challenge that we address in this paper.

Goal-oriented error estimator

Adaptive mesh refinement is a powerful technique to obtain accurate solutions for boundary-valued problems at a reduced computational cost. This is done by choosing an initial coarse mesh \mathbb{T}_0 and solving equation 5 on a series of subsequently refined meshes. On every subsequent mesh \mathbb{T}_i , a subset of cells in which the solution is most inaccurate is refined. A posteriori error estimator (Bürg, 2012b; Zhong et al., 2012) was used to define those cells. For a solution $\mathbf{F} = \mathbf{F}_R + i\mathbf{F}_I \in \mathbf{V}(\mathbb{T})$ and all cells $K \in \mathbb{T}_i$, error indicators $\eta_K(\mathbf{F})$ are calculated as follows:

$$\eta_K(\mathbf{F})^2 = \eta_{R,K}(\mathbf{F})^2 + \eta_{J,K}(\mathbf{F})^2, \tag{17}$$

where the residual- and jump-based terms are given by

$$\eta_{R,K}(\mathbf{F})^2 = \frac{h_K^2}{p^2} \|\nabla \times (\mu^{-1} \nabla \times \mathbf{F}) + i\omega\sigma\mathbf{F} - s\|_{L^2(K)}^2, \tag{18}$$

and

$$\begin{aligned}
\eta_{J,K}(\mathbf{F})^2 &= \frac{1}{2} \sum_{f=1}^6 \frac{h_f}{p} (\|[\mathbf{n}_f \times (\mu^{-1} \nabla \times \mathbf{F})]\|_{L^2(f)}^2 \\
&\quad + \|[\mathbf{n}_f \cdot (i\omega\sigma\mathbf{F} - s)]\|_{L^2(f)}^2). \tag{19}
\end{aligned}$$

Here, $h_K := \text{diam}(K)$ is the diameter of the hexahedra K and f is an interior face of \mathbb{T}_i and $h_f := \text{diam}(f)$. Square brackets $[\cdot]$ denote the jump of the quantity across the element boundaries with respect to the fixed face normal \mathbf{n}_f .

Having error indicators calculated for all cells, a subset of cells with the most inaccurate solution can be identified and refined. This strategy reduces the global error of a solution (Binev et al., 2004), but more efficient techniques can be derived for geoelectromagnetic methods. We are mostly interested in getting accurate solution at receiver locations. Therefore, even if a part of the domain can significantly affect the global error, its contribution to the accurate solution at a measurement point can be negligible. Therefore, no refinement is likely needed there. Because similar approaches are favorable in many different areas, generic methods have been developed to steer mesh refinement accordingly (Ainsworth and Oden, 2000; Bangerth and Rannacher, 2003). These methods, known as *dual-weighted* or *goal-oriented error estimators*, have been recently applied in geoelectromagnetic modeling (Key and Ovall, 2011; Ren et al., 2013). In addition to the numerical solution of equation 5, a dual problem with fictitious sources at the receiver locations is solved. Let the dual solution be denoted by \mathbf{E}^D (see equation 32 in Ren et al. [2013] for the exact formulation), the weighted error indicator that we used to mark cells for refinement is then expressed as

$$\eta_K^{\text{go}} = \eta_K(\mathbf{E}^s) \eta_K(\mathbf{E}^D), \tag{20}$$

with

$$(\eta^{\text{go}})^2 = \sum_{K \in \mathbb{T}_i} (\eta_K^{\text{go}})^2 \tag{21}$$

being the global goal-oriented error estimator.

For automatic adaptive mesh refinement, the fixed fraction strategy (Dörfler, 1996) was used and the smallest subset $\mathcal{T} \subseteq \mathbb{T}_i$ of cells that satisfies

$$\sum_{K \in \mathcal{T}} (\eta_K^{\text{go}})^2 \geq (\theta \eta^{\text{go}})^2 \tag{22}$$

was selected for a predefined $0 < \theta \leq 1$. The selected cells are split in all three dimensions, and the procedure is repeated until η is small or the initial error has been sufficiently reduced.

Nonconforming meshes

It is practically impossible to obtain a conforming hexahedral 3D mesh during local refinement. A mesh is called *conforming* if all cells share the whole edge or face. To avoid this limitation, we al-

lowed nonconforming meshes (Šolín et al., 2004). Figure 1 shows a simple nonconforming hexahedral mesh. In this case, additional constraints need to be imposed for a solution to conform with the continuity requirements of the $\mathcal{H}(\text{curl}, \Omega)$ functional space. Specifically, all fine DoFs on fine-coarse interfaces are linearly interpolated from the coarse DoFs on the interfaces. Note that the constrained DoFs are eliminated, so linear system 16 is ultimately posed only on the unconstrained conforming unknowns. A detailed description of the arbitrary-order FEM on nonconforming meshes can be found in Bangerth and Kayser-Herold (2009).

EFFICIENT SOLVER

The resulting system of linear equation 16 is quite difficult to solve numerically. Such factors as the low frequencies required in geoelectromagnetic methods, high-dimensional null space of the curl operator, large conductivity contrasts (e.g., across the air-ground interface), and nonuniform and locally refined meshes all make the system matrix very ill conditioned. Furthermore, the condition number of the system matrix grows as $\mathcal{O}(p^8)$ making high polynomial degrees more challenging. Taking into account the fact that solution of this system typically comprises most of the modeling time, development of a robust and scalable solver plays a vital role. In this context, the robustness of a solver means the ability to preserve fast convergence for a wide range of model parameters (e.g., frequency and material contrasts) and number of unknowns, whereas (strong) scalability means that when run in parallel, the required computational time can be reduced proportionally with the number of processes used.

Developing a robust and scalable solver for EM modeling is a non-trivial task even for the lowest order Nédélec elements. One could apply direct solvers that are very robust, but they do not scale well (Grayver and Bürg, 2014). Therefore, for large problems, the iterative Krylov solvers are the only viable alternative. The performance of these methods largely depends on a preconditioner used. Some of the most general, robust, and user-friendly preconditioners in current practice belong to the family of algebraic multigrid (AMG) methods, which couple a simple relaxation scheme with a hierarchy of algebraically constructed, coarse-grid problems. Parallel implementations of AMG methods have been under intensive research and development in the past few decades, and several scalable software libraries are currently available (Henson and Yang, 2002; Gee et al., 2006) with some support for definite Maxwell problems (Hu et al., 2006; Kolev and Vassilevski, 2009). In geoelectromagnetic applications, multigrid and domain-decomposition preconditioners have been shown to be extremely efficient in handling large-scale problems (Grayver and Bürg, 2014; Ren et al., 2014). These works, however, consider only the lowest order Nédélec elements and do not discuss complications related to nonconforming meshes. Generally, much less work has been done in developing efficient preconditioners for high-order Nédélec elements (Zaglmayr, 2006). In this work, we have extended the approach from Grayver and Bürg (2014) to the arbitrary order Nédélec elements and nonconforming hexahedral meshes. This approach is based on the auxiliary-space Maxwell solver (AMS) (Kolev and Vassilevski, 2009) from Hypr (2014), which is discussed in the following subsection.

To solve the indefinite symmetric system (equation 16), we have applied flexible generalized minimal residual (FGMRES) Krylov subspace method (Saad, 2003). To precondition this system, we used a real, SPD block-diagonal matrix:

$$\mathbf{P}_A = \begin{pmatrix} \mathbf{B} & \mathbf{0} \\ \mathbf{0} & \mathbf{B} \end{pmatrix}, \quad (23)$$

where $\mathbf{B} = \mathbf{C} + \mathbf{M}$. In Appendix A, we prove that the condition number of $\mathbf{P}_A^{-1}\mathbf{A}$ is $\leq \sqrt{2}$, independent of the finite element order, matrix size, frequency, and conductivity contrasts; see lemma 4.1 in Chen et al. (2010). Because of that, FGMRES requires a small number of iterations to converge to the solution of equation 16.

Although optimal, in the sense that applying, e.g., MINRES to equation 16 with a preconditioner \mathbf{P}_A will require a bounded number of iterations, preconditioner 23 is not yet practically useful until we find a way to apply its inverse to a vector quickly. A product of \mathbf{P}_A^{-1} with a vector requires solving two linear systems:

$$\mathbf{B}\mathbf{x} = \mathbf{b}. \quad (24)$$

Note that \mathbf{B} is a discretization of a second-order definite Maxwell problem; i.e., we have reduced the inversion of the operator 1 with a mass matrix coefficient $i\omega\sigma$ to inverting the same operator with the real coefficient $\omega\sigma$. Because FGMRES requires one matrix-vector product per iteration, N_{iter} iterations will therefore incur the solution of $2N_{\text{iter}}$ problems 24. We have calculated the approximate solution of this system efficiently using the conjugate gradients (CG) method with the auxiliary space preconditioner AMS, as described below. Note that the inner system is solved by using a predefined tolerance value. This causes preconditioner 23 to change at every iteration. Hence, we need to use Krylov methods that permit this, and FGMRES is one of them.

Auxiliary space preconditioner

In this section, we provide a general overview of the theory and implementation of AMS in the low-order conforming case, and its extensions to the high-order nonconforming case. Defining the AMS components in the latter case is nontrivial, and we consider it to be one of the main contributions of this paper.

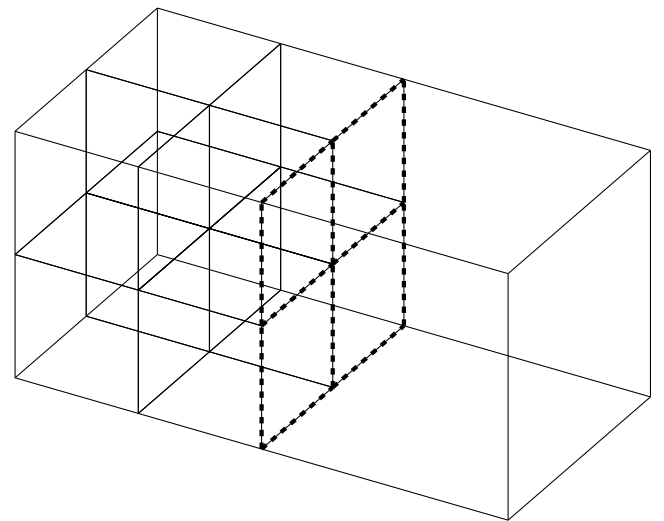


Figure 1. Nonconforming hexahedral mesh. The DoFs associated with the edges and faces marked with the dashed line need to be constrained to fulfill the continuity requirements of the $\mathcal{H}(\text{curl})$ functional space that is used.

Consider first the case of low-order elements on a conforming mesh \mathbb{T} of size h . Let $S(\mathbb{T})$ be the space of linear continuous nodal finite elements on the same mesh as $\mathbf{V}(\mathbb{T})$, and let $\mathbf{S}(\mathbb{T}) = S(\mathbb{T})^d$ be its vector counterpart. Denote with G_h and $\mathbf{\Pi}_h$ the matrix representations of the mapping $\varphi \in S(\mathbb{T}) \mapsto \nabla\varphi \in \mathbf{V}(\mathbb{T})$ and the nodal Nédélec interpolation from $\mathbf{S}(\mathbb{T})$ to $\mathbf{V}(\mathbb{T})$. The matrix G_h has as many rows as the number of edges in the mesh, with each row having two nonzero entries: $+1$ and -1 in the columns corresponding to the edge vertices. Furthermore, $\mathbf{\Pi}_h$ can be computed based only on G_h and the coordinates of the vertices of the mesh. For example, the x -component of $\mathbf{\Pi}_h$ is

$$\Pi_h^x = \frac{1}{2} \mathbf{D}_{G_h x} |G_h|, \quad (25)$$

where \mathbf{x} is the vector of x -coordinates of the mesh vertices, $\mathbf{D}_{G_h x}$ is a diagonal matrix with the diagonal given by the vector $G_h \mathbf{x}$, and $|\cdot|$ denotes componentwise absolute value of a matrix.

In these settings, the auxiliary space AMG preconditioner for \mathbf{B} is a subspace correction method using the subspaces $\mathbf{V}(\mathbb{T})$, $G_h S(\mathbb{T})$, and $\mathbf{\Pi}_h \mathbf{S}(\mathbb{T})$. Its additive form reads (Xu, 1996)

$$\Lambda_h^{-1} + G_h \mathcal{B}_h^{-1} G_h^T + \mathbf{\Pi}_h \mathcal{B}_h^{-1} \mathbf{\Pi}_h^T, \quad (26)$$

where Λ_h is a smoother for B , whereas \mathcal{B}_h and \mathcal{B}_h are efficient preconditioners for the variational auxiliary-space matrices $G_h^T \mathbf{B} G_h$ and $\mathbf{\Pi}_h^T \mathbf{B} \mathbf{\Pi}_h$, respectively. Because these matrices come from nodal elliptic forms, they can be handled efficiently by the classical AMG, which is especially efficient on unstructured problems with large coefficient jumps. The AMS implementation in *hypre* is a parallel version of this algorithm, using the AMG solver BoomerAMG (Henson and Yang, 2002) in the auxiliary subspaces.

The theoretical foundation for equation 26 is the HX decomposition (Hiptmair and Xu, 2007) of finite-element functions $\mathbf{u}_h \in \mathbf{V}(\mathbb{T})$ in the form

$$\mathbf{u}_h = \mathbf{v}_h + \mathbf{\Pi}_h \mathbf{z}_h + \nabla\varphi_h, \quad (27)$$

with $\mathbf{v}_h \in \mathbf{V}(\mathbb{T})$, $\mathbf{z}_h \in \mathbf{S}(\mathbb{T})$, and $\varphi_h \in S(\mathbb{T})$ satisfying the stability estimates

$$h^{-1} \|\mathbf{v}_h\|_0 + \|\mathbf{z}_h\|_1 \leq C \|\nabla \times \mathbf{u}_h\|_0 \quad \text{and} \quad \|\nabla\varphi_h\|_0 \leq C \|\mathbf{u}_h\|_0. \quad (28)$$

These estimates are critical for efficient preconditioning because they imply that the nonnodal component \mathbf{v}_h has a small relative norm, and therefore it can be handled by simple pointwise smoothing.

In the high-order conforming case, we can still define the spaces $S(\mathbb{T})$ and $\mathbf{S}(\mathbb{T})$ using nodal elements of the same order as $\mathbf{V}(\mathbb{T})$, and the HX decomposition (equation 27) continues to hold. However, the structure of the G_h matrix is more complicated in this case, and although it is still independent of the mesh geometry, its evaluation is more technical. Furthermore, matrix $\mathbf{\Pi}_h$ cannot be determined without additional information about the high-order bases in $\mathbf{V}(\mathbb{T})$ and $\mathbf{S}(\mathbb{T})$, and in practice it has to be computed by evaluating the Nédélec DoFs of the nodal vector basis functions on each element, instead of equation 25. Specifically, let

$$\mathbf{F} = \sum_{i=1}^n \alpha_i(F) \boldsymbol{\varphi}_i \in \mathbf{V}(\mathbb{T}) \quad (29)$$

be the Nédélec interpolant of a given (sufficiently smooth) vector field F . In the low-order case, $\alpha_i(F)$ are the integrals of the tangential components of F over the edges of \mathbb{T} , whereas in the high-order case, they depend on the choice of the high-order basis (but they typically involve integral moments over the edges, faces, and the element volumes). Let $\{\varphi_k\}$ be the basis functions in $S(\mathbb{T})$, then

$$(G_h)_{ik} = \alpha_i(\nabla\varphi_k) \quad \text{and} \quad (\Pi_h^x)_{ik} = \alpha_i((\varphi_k, 0, 0)). \quad (30)$$

These computations can be parallelized and automated in general finite-element libraries (Bangerth et al., 2011; MFEM, 2014).

We now consider the general settings of high-order elements on a nonconforming mesh. In this case, each finite element space has two versions: a conforming one, e.g., $V_c(\mathbb{T})$, in which the *hanging* DoFs are constrained by the conforming (*real*) DoFs, and a nonconforming one, e.g., $V_{nc}(\mathbb{T})$, in which the nonconforming DoFs (*hanging* and *real*) are unconstrained. The matrix representation \mathbf{P}_V of the embedding operator $\mathbf{E} \in V_c(\mathbb{T}) \mapsto \mathbf{E} \in V_{nc}(\mathbb{T})$ is precisely the conforming interpolation; see the section on nonconforming meshes. One can similarly define the matrices \mathbf{P}_S and \mathbf{P}_S representing the conforming interpolation between the nodal spaces $S_c(\mathbb{T})$ and $S_c(\mathbb{T})$ and $S_{nc}(\mathbb{T})$ and $S_{nc}(\mathbb{T})$, respectively. We also introduce the matrix representations, \mathbf{R}_V , \mathbf{R}_S , and \mathbf{R}_S of the restriction operators from the nonconforming to the conforming space, which ignore the nonconforming components of their input. Note that with these definitions, $\mathbf{R}_V \mathbf{P}_V$, $\mathbf{R}_S \mathbf{P}_S$, and $\mathbf{R}_S \mathbf{P}_S$ are just the identity matrices on the conforming DoFs.

As mentioned in the section “Nonconforming meshes,” the definite Maxwell matrix \mathbf{B} in equation 24 is defined on the space $V_c(\mathbb{T})$, i.e., $\mathbf{B} = \mathbf{B}_c = \mathbf{P}_V^T \mathbf{B}_{nc} \mathbf{P}_V$, where \mathbf{B}_{nc} is the $V_{nc}(\mathbb{T})$ version of \mathbf{B} , assembled without interpolatory DoFs constraints. Therefore, to apply AMS to \mathbf{B} , we need to construct a “conforming discrete gradient” matrix $G_{c,h}$ that corresponds to the gradient mapping from $S_c(\mathbb{T})$ to $V_c(\mathbb{T})$. To obtain an appropriate definition for $G_{c,h}$, we consider the diagram

$$\begin{array}{ccc} S_c(\mathbb{T}) & \xrightarrow{G_{c,h}} & V_c(\mathbb{T}) \\ \mathbf{P}_S \downarrow \uparrow \mathbf{R}_S & & \mathbf{P}_V \downarrow \uparrow \mathbf{R}_V \\ S_{nc}(\mathbb{T}) & \xrightarrow{G_{nc,h}} & V_{nc}(\mathbb{T}) \end{array} \quad (31)$$

where $G_{nc,h}$ is the matrix representation of the gradient mapping $\varphi_{nc} \in S_{nc}(\mathbb{T}) \mapsto \nabla\varphi_{nc} \in V_{nc}(\mathbb{T})$, which can be computed element by element, as before, without imposing any constraints on the hanging DoFs. We now require that this diagram commutes from $S_c(\mathbb{T})$ to $V_{nc}(\mathbb{T})$; i.e.,

$$\mathbf{P}_V G_{c,h} = G_{nc,h} \mathbf{P}_S. \quad (32)$$

This means that $G_{c,h}$ and $G_{nc,h}$ are compatible when computing the gradient of a conforming nodal function. Because $\mathbf{R}_V \mathbf{P}_V = I$, this implies the conforming discrete gradient definition

$$G_{c,h} = \mathbf{R}_V G_{nc,h} \mathbf{P}_S, \quad (33)$$

which is what we pass to AMS. Note that by commutativity, the variational auxiliary space matrix for $G_{c,h}$ is the constrained version of the variational auxiliary space matrix for $G_{nc,h}$:

$$G_{c,h}^T \mathbf{B}_c G_{c,h} = G_{c,h}^T \mathbf{P}_V^T \mathbf{B}_{nc} \mathbf{P}_V G_{c,h} = \mathbf{P}_S^T (G_{nc,h}^T \mathbf{B}_{nc} G_{nc,h}) \mathbf{P}_S. \quad (34)$$

Similar considerations imply that the conforming Nédélec interpolation matrix should be defined as

$$\mathbf{\Pi}_{c,h} = \mathbf{R}_V \mathbf{\Pi}_{nc,h} \mathbf{P}_S. \quad (35)$$

In the high-order case, $\mathbf{\Pi}_{nc,h}$ and consequently $\mathbf{\Pi}_{c,h}$ can be computed elementwise by evaluating the Nédélec DoFs of the nodal vector basis functions as discussed earlier. We note, however, that for low-order elements on quadrilateral/hexahedral meshes, matrix $\mathbf{\Pi}_{c,h}$ can still be computed based only on $\mathbf{G}_{c,h}$ and the conforming vertex coordinates as in equation 25. Indeed, the entries of \mathbf{R}_V and \mathbf{P}_S are positive in this case, so

$$|G_{c,h}| = \mathbf{R}_V |G_{nc,h}| \mathbf{P}_S. \quad (36)$$

Furthermore, the conformity of the mesh implies $G_{nc,h} \mathbf{x}_{nc} = G_{nc,h} \mathbf{P}_S \mathbf{x}_c = \mathbf{P}_V G_{c,h} \mathbf{x}_c$, so

$$\mathbf{R}_V \mathbf{D}_{G_{nc,h} \mathbf{x}_{nc}} = \mathbf{D}_{G_{c,h} \mathbf{x}_c} \mathbf{R}_V. \quad (37)$$

Therefore,

$$\begin{aligned} \mathbf{\Pi}_{c,h}^x &= \mathbf{R}_V \mathbf{\Pi}_{nc,h}^x \mathbf{P}_S = \frac{1}{2} \mathbf{R}_V \mathbf{D}_{G_{nc,h} \mathbf{x}_{nc}} |G_{nc,h}| \mathbf{P}_S \\ &= \frac{1}{2} \mathbf{D}_{G_{c,h} \mathbf{x}_c} |G_{c,h}|, \end{aligned} \quad (38)$$

which is in agreement with equation 25.

NUMERICAL EXPERIMENTS

The numerical scheme described in the previous section has been implemented in a fully distributed fashion using message passing interface (MPI). Several third-party libraries have been used for this purpose: deal.II for FEM machinery (Bangerth et al., 2011), p4est for distributed mesh handling (Burstedde et al., 2011), and PETSc for distributed linear algebra (Balay et al., 2014). Finally, the open-source implementation of AMS preconditioner from the *hypr* library was used (Kolev and Vassilevski, 2009; Baker et al., 2012).

In this section, we run a series of numerical tests with high-order FEM using MT modeling as an example. Calculating the impedance tensor in MT requires solving system 16 for two orthogonal source polarizations. Because we used the secondary field formulation and homogeneous boundary conditions, the system matrix remains the same for the polarizations and for the dual problems. This eliminates expensive recalculation of the system matrix entries in equation 16. Denoting the horizontal components of the electric and magnetic fields for the two source polarizations by $E_{x1}, E_{y1}, H_{x1}, H_{y1}$ and $E_{x2}, E_{y2}, H_{x2}, H_{y2}$, the MT impedance tensor \mathbf{Z} reads (Berdichevskii and Dmitriev, 2008)

$$\begin{pmatrix} E_{x1} & E_{x2} \\ E_{y1} & E_{y2} \end{pmatrix} = \begin{pmatrix} Z_{xx} & Z_{xy} \\ Z_{yx} & Z_{yy} \end{pmatrix} \begin{pmatrix} H_{x1} & H_{x2} \\ H_{y1} & H_{y2} \end{pmatrix}. \quad (39)$$

In practice, one typically plots apparent resistivity and phase calculated from an impedance tensor component in the following way:

$$\rho_{ij}^{\text{app}} = \frac{1}{\omega \mu} |Z_{ij}|^2 \quad \text{and} \quad \phi_{ij} = \arg(Z_{ij}). \quad (40)$$

The results presented in this section are divided into two parts as follows: first, we perform a series of tests to study robustness and scalability of the implemented numerical scheme for different polynomial orders of the Nédélec element, followed by a convergence study aimed at investigating which polynomial order and refinement strategy perform best.

Robustness and scalability of the solver

To study robustness of the solver, we calculated solutions for a series of models. For all tests shown, the electric conductivity of the air was set to 10^{-9} S/m. The outer FGMRES solver was stopped once the normalized relative residual attained the value of $\varepsilon = 10^{-8}$; i.e.,

$$\frac{\|\mathbf{A}\bar{\mathbf{E}}^s - \bar{\mathbf{s}}\|_2}{\|\bar{\mathbf{s}}\|_2} < \varepsilon. \quad (41)$$

A much higher relative residual of 10^{-2} was used for the inner CG solver. Recall that equation 24 is solved whenever a product of the inverse preconditioner 23 with a vector is needed. Experiments showed that solving this system with a low tolerance does not improve convergence of the outer solver significantly. Therefore, the value 10^{-2} was found sufficient to preserve robustness and keep the computational time low. The conductivity model we used for this test is shown in Figure 2. This model poses several challenges: First, a large conductivity contrast across the air-ground interface. Second, we used a stretched grid, which may deteriorate convergence of some preconditioners (Mulder, 2006). Initially, the model was

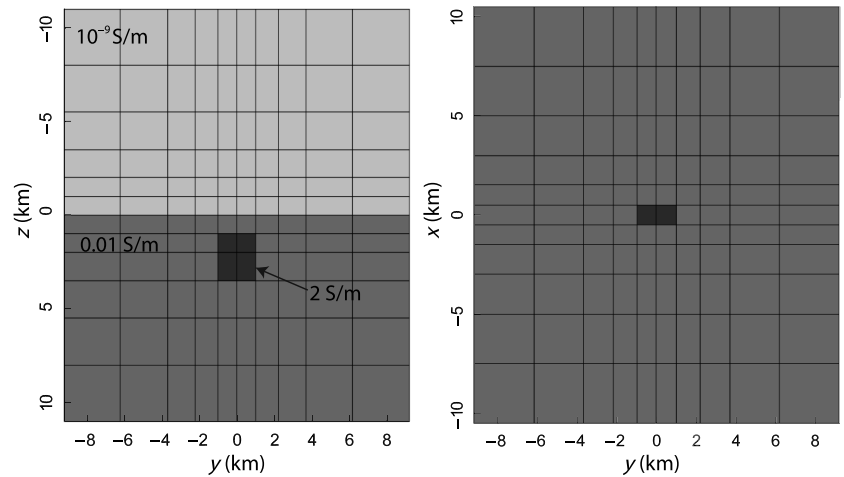


Figure 2. The conductivity model and the initial mesh used to study the robustness and scalability of the solver. The vertical and horizontal sections at $x = 0$ and $z = 0$ are shown. The corresponding conductivity values are given in the text.

split into 1320 cells. In each subsequent cycle, the model is uniformly refined in one or more spatial dimensions. This approximately doubles the number of DoFs. Note that we show the number of *real-valued* DoFs, which is twice the number of corresponding *complex-valued* DoFs. The numbers of outer and the average numbers of inner iterations for some of the models and $p = 1$ are given in Table 2. The degree $p = 1$ corresponds to the conventionally used lowest order Nédélec elements. We see that for frequencies ranging from 0.01 to 1000 Hz and a wide range of prob-

lem sizes, the solver performs in a robust way. The number of outer iterations stays virtually constant, and the inner iteration counts slightly increase with the problem size. Tables 3 and 4 present the results for higher polynomial degrees of $p = 2$ and $p = 3$, respectively. The solver performed remarkably well for these problems. The number of outer iterations does not show any dependence on the degree p , whereas the inner solver required on average more iterations for higher polynomial degrees. This is anticipated because the condition number of matrix \mathbf{B} grows as $\mathcal{O}(p^8)$. However, the

Table 2. Numerical results for the model shown in Figure 2 and variable frequencies for Nédélec elements of degree $p = 1$. The number of outer FGMRES and the average numbers of inner CG iterations are denoted by N_{iter} and $\bar{N}_{\text{iter}}^{\text{CG}}$, respectively.

MPI processes	Number of DoFs	Frequency (Hz)							
		0.01		0.1		10		1000	
		N_{iter}	$\bar{N}_{\text{iter}}^{\text{CG}}$	N_{iter}	$\bar{N}_{\text{iter}}^{\text{CG}}$	N_{iter}	$\bar{N}_{\text{iter}}^{\text{CG}}$	N_{iter}	$\bar{N}_{\text{iter}}^{\text{CG}}$
2	9434	8	2	11	2	18	1	16	1
4	69,284	9	3	12	3	19	1	18	1
8	530,312	9	4	13	3	20	2	18	1
32	4,148,240	9	5	13	4	20	2	18	1
128	32,812,064	9	7	13	6	20	3	20	2
224	130,729,664	10	7	13	6	20	3	20	2

Table 3. Same as Table 2, but for Nédélec elements of degree $p = 2$.

MPI processes	Number of DoFs	Frequency (Hz)							
		0.01		0.1		10		1000	
		N_{iter}	$\bar{N}_{\text{iter}}^{\text{CG}}$	N_{iter}	$\bar{N}_{\text{iter}}^{\text{CG}}$	N_{iter}	$\bar{N}_{\text{iter}}^{\text{CG}}$	N_{iter}	$\bar{N}_{\text{iter}}^{\text{CG}}$
2	69,284	9	5	12	4	18	2	17	4
8	530,312	9	6	13	5	18	2	18	3
16	2,088,368	9	9	12	9	20	4	18	4
48	8,265,408	9	8	13	8	20	4	18	4
64	16,462,688	10	10	13	10	20	5	18	4

Table 4. Same as Table 2, but for Nédélec elements of degree $p = 3$.

MPI processes	Number of DoFs	Frequency (Hz)							
		0.01		0.1		10		1000	
		N_{iter}	$\bar{N}_{\text{iter}}^{\text{CG}}$	N_{iter}	$\bar{N}_{\text{iter}}^{\text{CG}}$	N_{iter}	$\bar{N}_{\text{iter}}^{\text{CG}}$	N_{iter}	$\bar{N}_{\text{iter}}^{\text{CG}}$
4	227,070	9	6	12	5	18	3	17	3
16	1,763,244	9	7	12	7	19	3	18	3
48	6,979,464	9	11	12	11	20	5	18	4
128	13,895,064	9	7	13	7	20	4	18	4

growth in iterations is rather moderate and overall the solver remains robust. Of particular importance for large-scale problems is the fact that the number of MPI processes used to solve a problem does not affect convergence of the solver for any polynomial degree.

To test the robustness of the solver on nonconforming unstructured and locally refined meshes, we constructed a model with undulated topography. Specifically, we used the topography of the Kronotsky volcano located at the Kamchatka peninsula. In this example, the electric field was calculated at the frequency of 1 Hz using Nédélec elements of degree $p = 2$. The initial mesh was adaptively refined using goal-oriented error estimator with $\theta = 0.65$. Figure 3 displays the initial and refined meshes with positions of the receivers. Table 5 summarizes the number of outer and inner iterations that were required to solve the resulting systems at different refinement steps. Clearly, the solver remains robust and the number of iterations is virtually constant despite the unstructured mesh and local refinement that we used.

Then, we investigated the strong scalability of the solver. To this end, a uniformly refined version of the model shown in Figure 2 with 8,265,408 DoFs was taken and the solution at the frequency of 0.1 Hz was calculated using $p = 1$ and $p = 2$. The model consisted of 1,351,680 and 168,960 cells for $p = 1$ and $p = 2$, respectively. Figure 4 illustrates computational time versus the number of MPI processes used to calculate the solution. In addition to the time required by the solver, we have also plotted the time spent on the system matrix and the right-side vector assembly. These two operations comprise most of the total required CPU time. Quadratic elements ($p = 2$) have roughly four times more DoFs per elements than lowest order $p = 1$ elements. This increases the coupling between elements and results in more nonzero entries in the system matrix. Therefore, given the same number of DoFs, higher polynomial degrees incur more computational time because of lower sparsity. Evidence for this is seen in Figure 4a and 4b with quadratic elements being roughly three times more expensive. At the same time, better scalability is observed for $p = 2$ because of the higher computation-to-communication ratio. This can be seen when comparing time required by the solver for $p = 1, 2$ in Figure 4a and 4b. For the lowest polynomial degree, the time does not decrease proportionally to the number of MPI processes used, indicating suboptimal scalability.

Convergence study

Having confirmed that the presented solver is robust and scalable, we have not yet answered the question of whether higher polynomial degrees are advantageous for geoelectromagnetic modeling. In this section, we study the convergence of several different discretization strategies. We considered four scenarios:

- 1) *Uniform h-refinement* — At each subsequent step, a mesh is globally refined in all three dimensions.
- 2) *Uniform p-refinement* — At each subsequent step, the polynomial degree is globally increased by one.
- 3) *Adaptive h-refinement* — The mesh is locally refined using a goal-oriented error estimator and fixed fraction strategy described in the section on error estimation. The lowest order edge-based elements were used; i.e., $p = 1$.
- 4) *Adaptive h-refinement and uniform p-refinement* — The same as the previous, but with higher polynomial degrees $p > 1$.

For this test, we have chosen the COMMEMI 3D-2 model (Figure 5). This model contains two extensive 3D anomalies that cause the solution to have large gradients and be discontinuous in the vicinity of the conductivity contrasts. It is important to use a model that has nonsmooth solution with discontinuities because this behavior occurs in many practical scenarios. Because no analytic solution exists for this model, the solution error was estimated using equation 17. This error estimator was shown to represent the solution error well (Chen et al., 2010; Bürg, 2012b; Zhong et al., 2012). The goal-oriented error estimator was run for 64 receivers uniformly spaced along the x -direction from -40 to 40 km (see Figure 5) and $\theta = 0.75$. In what follows, the error for x polarization is shown. Among the two polarizations, this one is more challenging to resolve accurately due to the geometry and position of the anomaly.

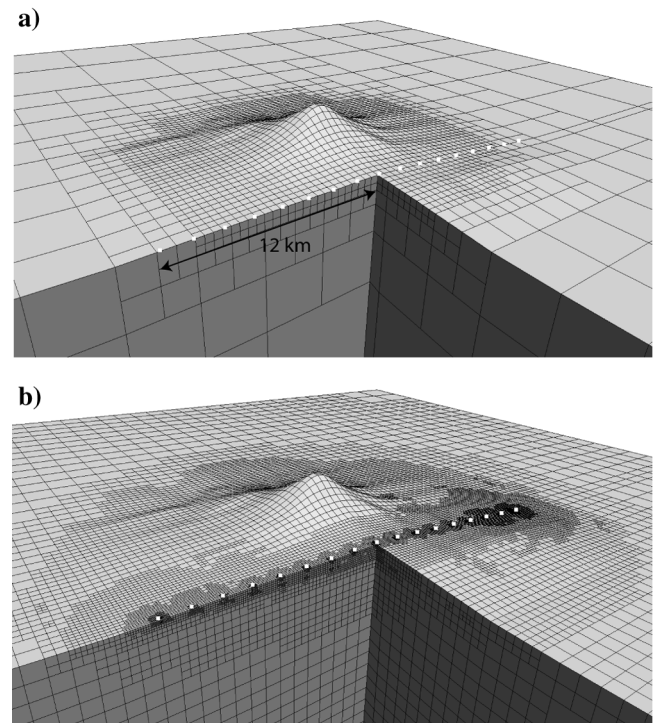


Figure 3. (a) Initial and (b) adaptively refined meshes of the Kronotsky volcano model with 11,752 and 169,966 cells, respectively. Receivers are depicted with white rectangles. The air layer on the top is not shown.

Table 5. Numerical results for the model shown in Figure 3 and 18 refinement steps. The number of outer FGMRES and the average numbers of inner CG iterations are denoted by N_{iter} and $\bar{N}_{\text{iter}}^{\text{CG}}$, respectively.

Refinement step	Number of DoFs	N_{iter}	$\bar{N}_{\text{iter}}^{\text{CG}}$
0	647,544	22	3
6	786,436	22	3
12	2,044,156	23	3
18	9,291,488	24	4

ous objects with respect to the incident field. Neumann boundary conditions were adopted for this test.

Figure 6a plots convergence curves for the aforementioned refinement strategies. Adaptive h -refinement was performed for polynomial degrees up to three using goal-oriented error estimator. It is evident from the figure that uniform h -refinement is the worst strategy. Although the error decreases steadily, the convergence rate is small: We expect at most first-order convergence in the energy norm (Monk, 2003), thus requiring too many DoFs to reduce the solution error sufficiently. Despite that the solution is largely nonsmooth in some areas around anomalous objects, much better convergence is attained by using uniform p -refinement. Note that each marker on the magenta line essentially corresponds to a solution error obtained for increasing polynomial degrees p . Once we resort to adaptive h -refinement, convergence is further improved. The dramatic improvement in the convergence rate is achieved by going from Nédélec elements of degree $p = 1$ to the elements of degree $p = 2$. Using quadratic elements, we need roughly 10 times fewer DoFs to achieve the same error. This ties well with the theoretical predictions. Increasing the polynomial degree further and using $p = 3$ improves the convergence rate, but for the considered problem the benefit is rather moderate, which can probably be explained by the lack of smoothness in the solution.

The convergence study in which the estimated solution error is plotted against the number of DoFs does not reveal one important practical aspect. Namely, how long does it take to attain a certain error level? In the end, we are interested in having a fast numerical method. To demonstrate this, we thus also plotted errors versus CPU time in Figure 6b. Before interpreting these results, a few com-

ments on methodology are needed. All tests were run using 48 MPI processes, and the shown runtimes include time spent in the following operations: building the system matrix and right-side vectors, solving three systems (two for orthogonal source polarizations and a dual problem), and estimating errors. For adaptive h - and p -refinements, the cumulative time to achieve the corresponding error is shown. In other words, to achieve a certain level, we need to go through several refinement cycles. Therefore, runtimes spent on all previous refinement cycles need to be included. This is in contrast to the uniform h - and p -refinements in which independent times for each of the refinement cycles are shown. As is evident from Figure 6b, the results take a different shape when seen in terms of runtime. It turns out that uniform h - and p -refinement strategies require the same time to achieve a specified solution error, even though the latter results in a system with significantly fewer unknowns. This smaller system, however, is much denser and ill conditioned, thereby making higher polynomial degrees more resource demanding. At the initial stages, the lowest order Nédélec elements appear to be the fastest. However, due to the better convergence rate, quadratic elements soon outperform, whereas the efficiency gain provided by the cubic elements is not substantial. Therefore, at least for the model considered, the optimal strategy from the computational point of view is to use adaptive h -refinement with the Nédélec elements of degree $p = 2$. Of note is that our results stay in a good agreement with similar studies from other areas (Wang et al., 2009).

Figures 7–9 illustrate the estimated solution errors, i.e., $\eta_K(\mathbf{E}^s)^2$, $K \in \mathbb{T}$ for the initial coarse mesh and adaptively refined meshes obtained after 14 refinement steps, and different polynomial de-

Figure 4. The time required to assemble the system matrix and the right-side vector and to solve the resulting system with 8,265,408 DoFs for polynomial degrees of (a) $p = 1$ and (b) $p = 2$ versus the number of MPI processes used.

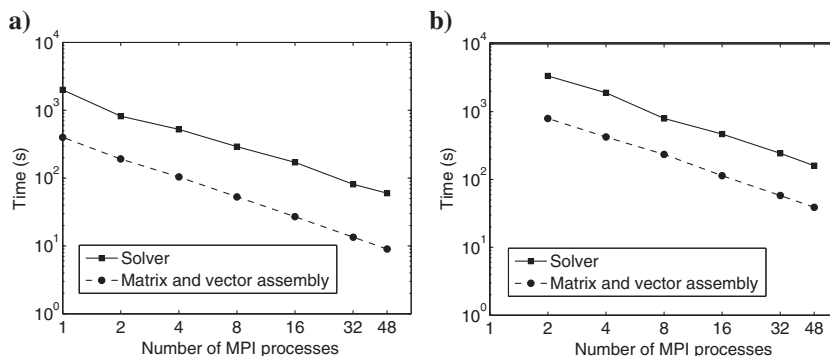
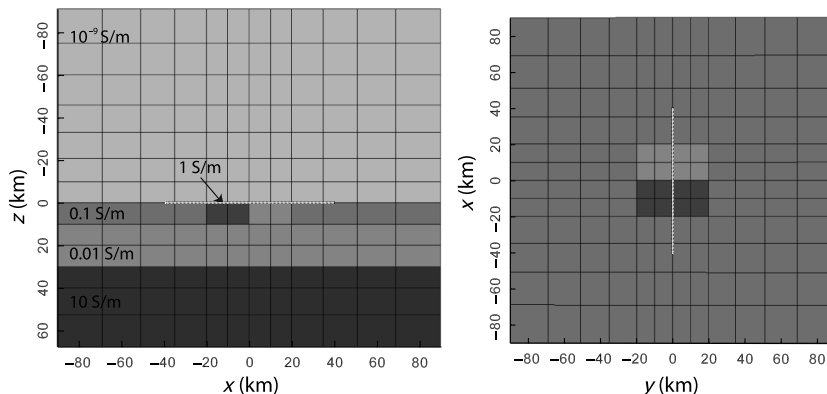


Figure 5. The initial discretization of the COMMEMI 3D-2 model (Zhdanov et al., 1997) used for the convergence test. The vertical and horizontal sections at $y = 0$ and $z = 0$ are shown. In total, 64 receiver stations are marked with white crosses. The corresponding conductivity values are given in the text.



greens. All figures have the same color scale. As anticipated, the initial errors are the largest in the vicinity of conductivity contrasts. At the final stages, the errors were reduced substantially and exhibit more uniform behavior. When comparing results obtained for different polynomial degrees, several observations can be made: First, for higher polynomial degrees, fewer cells are sufficient to achieve the same or often better accuracy. Second, in regions where the solution is smooth, e.g., in the air, using higher polynomial degrees proves particularly efficient. Even though the mesh remains coarse there, the solution errors decrease when using higher p . In contrast,

in the vicinity of conductivity contrasts, solution is nonsmooth and h -refinement is the best choice in this case.

Finally, in Figure 10, we show the apparent resistivity and phase at receivers for the initial and refined meshes and different polynomial degrees. Remarkably, even for the initial coarse mesh, the solution obtained with polynomial degree $p = 3$ already gives quite accurate results. The results for the adaptively refined meshes are almost identical, indicating the efficiency of our error estimator, although much fewer DoFs were required for higher p to attain the same accuracy.

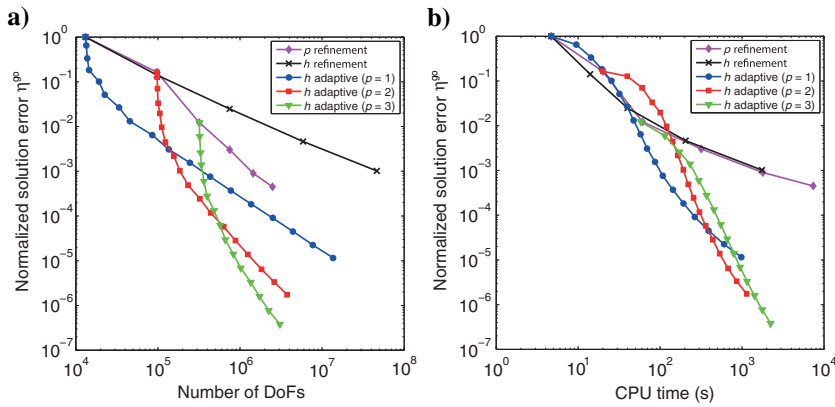


Figure 6. Normalized estimated global solution errors η^{s0} (see equation 21) versus the (a) number of DoFs and (b) runtimes for different refinement strategies. Each marker corresponds to a different forward problem.

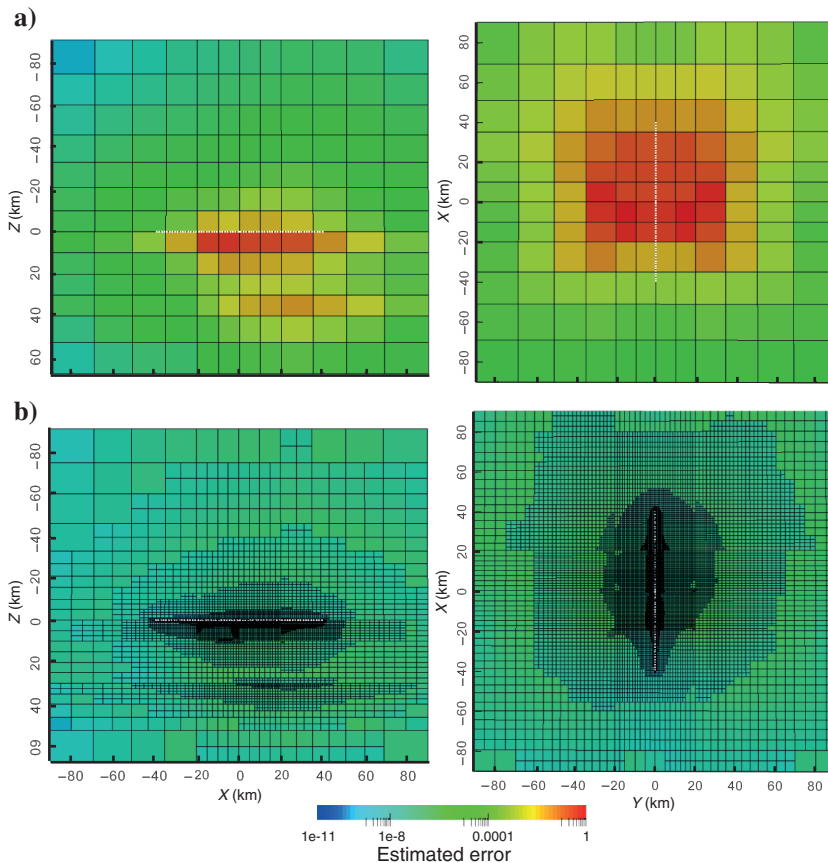


Figure 7. Estimated solution error $\eta_K(E^s)^2$ on the (a) initial mesh and (b) mesh obtained after 14 steps of adaptive mesh refinement for the Nédélec element of polynomial degree $p = 1$. The positions of 64 receivers are shown with white rectangles.

Figure 8. Same as Figure 7, but for Nédélec elements of polynomial degree $p = 2$.

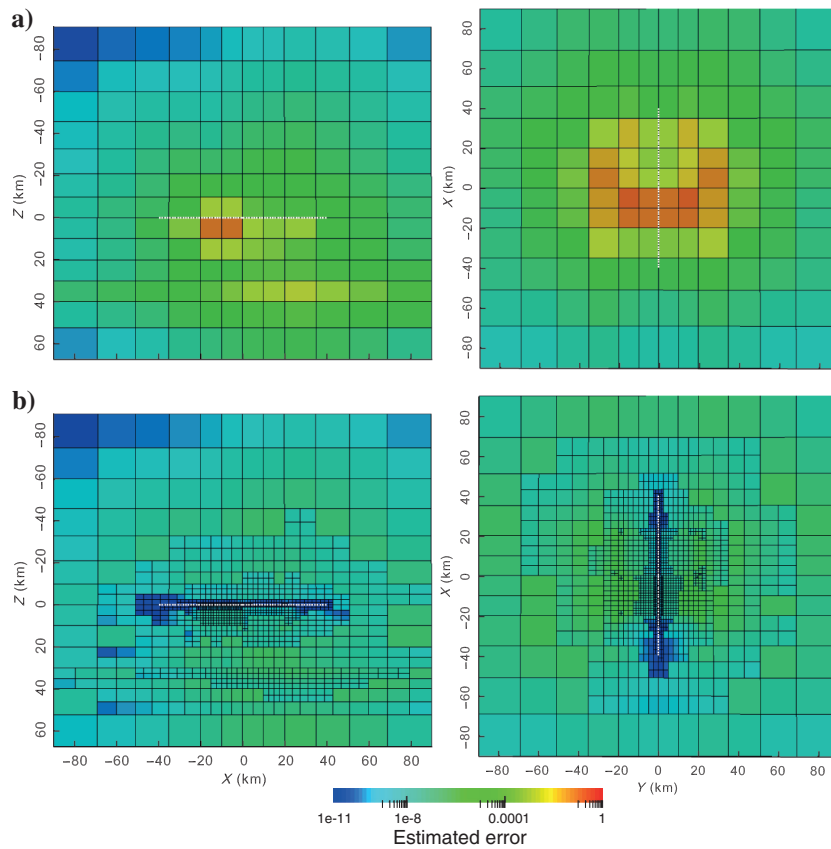
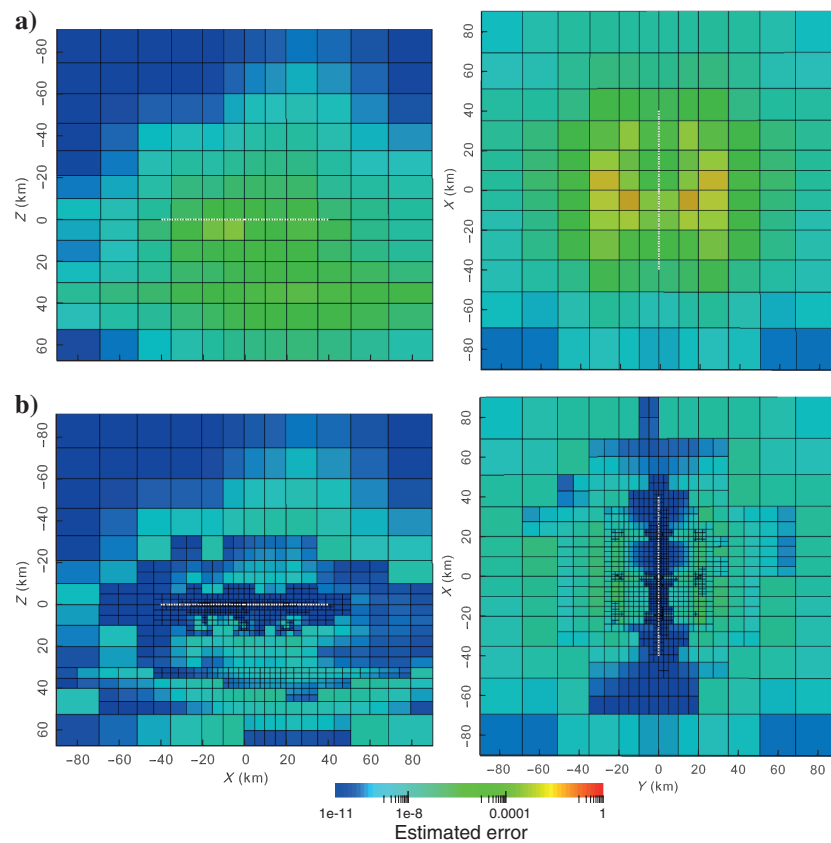


Figure 9. Same as Figure 7, but for Nédélec elements of polynomial degree $p = 3$.



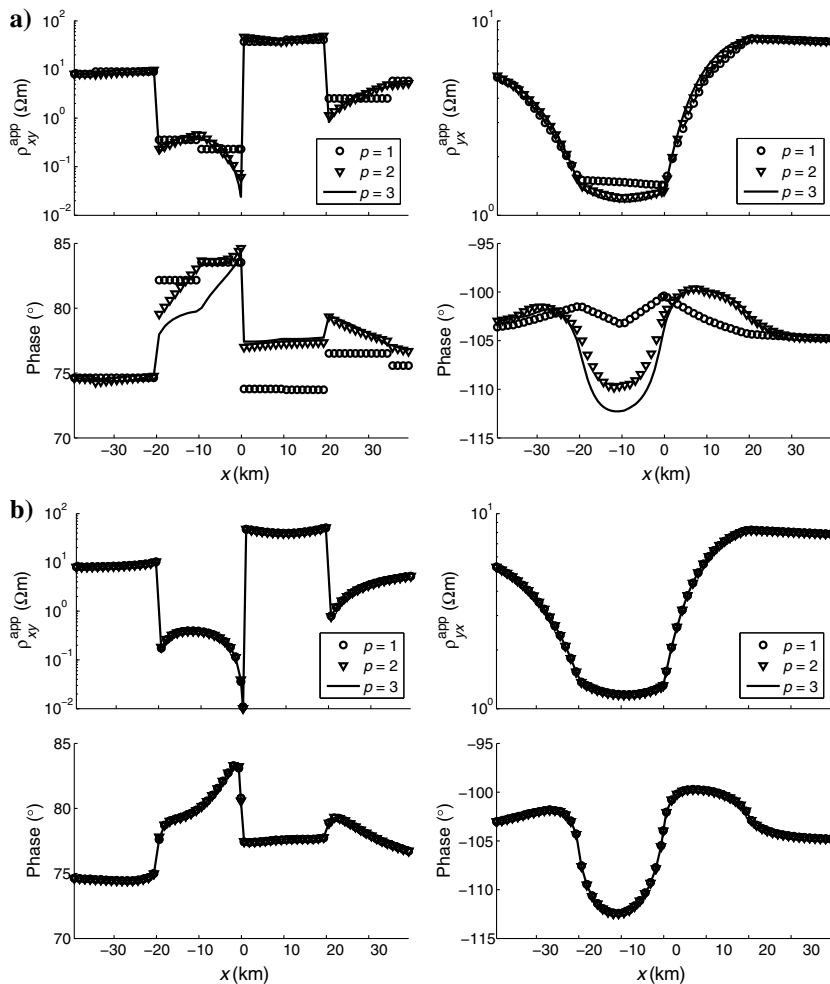


Figure 10. Apparent resistivity and phase for off-diagonal components of Z at the receivers (see Figure 5) and different polynomial degrees of the Nédélec finite element for the (a) initial coarse mesh and (b) adaptively refined meshes.

CONCLUSIONS

We have presented a numerical scheme for geoelectromagnetic modeling using high-order FEM. To date, Nédélec elements of the lowest degree have been mostly used to model time-harmonic EM phenomena, although the FEM permits using arbitrary polynomial degrees to represent solution in each element. Although being favorable from a theoretical point of view, Nédélec elements of higher polynomial degrees are very challenging from a numerical perspective. They are inherently more expensive to deal with, and they affect the condition number of the resulting system of linear equations. To overcome these challenges, we have developed a fully distributed modeling code with a robust and scalable solver. We have presented the solver in very generic settings that make it valid for a large variety of problems and formulations. The solver was shown to be robust for different polynomial degrees of the finite element, wide range of frequencies, and various problem sizes. In addition, this solver scales well, rendering our method suitable for large-scale problems with 10^8 unknowns. Of high practical importance is the fact that because the solver heavily relies on multigrid methods, new developments in this active area will readily allow a reduced required computational time.

For a further efficiency gain, we have implemented a fully automatic adaptive mesh refinement approach. Accurate EM responses at the receiver positions were obtained by using highly efficient

goal-oriented error estimator, which steers the mesh refinement in such a way that an accurate solution at the receivers is obtained using as few DoFs as possible. By performing a series of convergence tests on a model with a solution that has discontinuities, we have found that using adaptive mesh refinement in combination with Nédélec elements of higher polynomial degrees provides the highest accuracy with respect to the number of DoFs. Still, because higher polynomial degrees are more resource demanding, they may be less attractive from a computational point of view. For the cases studied here, quadratic elements together with adaptive local mesh refinement delivered the optimal combination. Because many models of practical interest share the properties with the model we have used, we expect this conclusion to hold for a wider range of relevant problems.

ACKNOWLEDGMENTS

A. V. Grayver is appreciative of A. Kuvshinov for providing insightful feedback and support for this work. The authors acknowledge the constructive comments by editor X. Garcia, Z. Ren, and two anonymous reviewers. This work was performed under the auspices of the U.S. Department of Energy by the Lawrence Livermore National Laboratory under contract nos. DE-AC52-07NA27344 and LLNL-JRNL-665742.

APPENDIX A

CONDITION NUMBER OF THE BLOCK-DIAGONAL PRECONDITIONER

In this appendix, we evaluate the condition number for the block-diagonal preconditioner 23 of the 2×2 real reformulation 16. This result holds for any matrix $\mathbf{C} + i\mathbf{M}$, in which the *stiffness* part \mathbf{C} is real and SPSD with a nontrivial kernel, whereas the *mass* part \mathbf{M} is real and SPD. The 2×2 real system corresponding to $\mathbf{C} + i\mathbf{M}$ has a matrix

$$\mathcal{A} = \begin{pmatrix} \mathbf{C} & -\mathbf{M} \\ -\mathbf{M} & -\mathbf{C} \end{pmatrix}. \quad (\text{A-1})$$

We define

$$\mathcal{P} = \begin{pmatrix} \mathbf{C} + \mathbf{M} & 0 \\ 0 & \mathbf{C} + \mathbf{M} \end{pmatrix}. \quad (\text{A-2})$$

Because $\mathbf{C} + \mathbf{M}$ is SPD, matrix \mathcal{P} is also SPD, and it is in particular invertible. Matrix \mathcal{A} , however, is symmetric but indefinite, so to control its negative eigenvalues, we introduce the “squared” preconditioned operator $\mathcal{S} = (\mathcal{A}\mathcal{P}^{-1})^2$, which is SPSD in the \mathcal{P}^{-1} inner product:

$$\begin{aligned} (\mathcal{S}z_1, z_2)_{\mathcal{P}^{-1}} &= (\mathcal{P}^{-1}\mathcal{A}\mathcal{P}^{-1}\mathcal{A}\mathcal{P}^{-1}z_1, z_2) \\ &= (\mathcal{P}^{-1}z_1, \mathcal{A}\mathcal{P}^{-1}\mathcal{A}\mathcal{P}^{-1}z_2) \\ &= (z_1, \mathcal{S}z_2)_{\mathcal{P}^{-1}}, \end{aligned} \quad (\text{A-3})$$

and $(\mathcal{S}z, z)_{\mathcal{P}^{-1}} = \|\mathcal{A}\mathcal{P}^{-1}z\|_{\mathcal{P}^{-1}}^2 \geq 0$. Now, \mathcal{P} will be a good preconditioner for \mathcal{A} if the condition number $\kappa(\mathcal{S})$ is bounded, i.e., if there are positive constants c and C , independent of the problem size, such that

$$|(\mathcal{A}z, z)| \leq C(\mathcal{P}z, z) \quad (\text{A-4})$$

and

$$c(\mathcal{P}z, z) \leq (\mathcal{A}\mathcal{P}^{-1}\mathcal{A}z, z) \quad (\text{A-5})$$

for all vectors $z = (x \ y)^T$. The first inequality is a bound on the spectral radius of $\mathcal{A}\mathcal{P}^{-1}$, in which the spectral radius is given by the largest (in magnitude) eigenvalue. The second inequality estimates the eigenvalue of \mathcal{S} that is closest to the origin.

Starting from $(\mathcal{A}z, z) = (\mathbf{C}x, x) - (\mathbf{C}y, y) - 2(\mathbf{M}x, y)$ and applying the triangle, Schwarz, and $2ab \leq a^2 + b^2$ inequalities we get

$$\begin{aligned} |(\mathcal{A}z, z)| &\leq (\mathbf{C}x, x) + (\mathbf{C}y, y) + 2\|x\|_{\mathbf{M}}\|y\|_{\mathbf{M}} \\ &\leq (\mathbf{C}x, x) + (\mathbf{C}y, y) + \|x\|_{\mathbf{M}}^2 + \|y\|_{\mathbf{M}}^2, \end{aligned} \quad (\text{A-6})$$

which is the first estimate above with $C = 1$. This estimate is sharp for any vectors $x = \pm y$ in the kernel of \mathbf{C} .

For the second inequality, note that by writing $\mathbf{C} = (\mathbf{C} + \mathbf{M}) - \mathbf{M}$, we get

$$\mathbf{C}(\mathbf{C} + \mathbf{M})^{-1}\mathbf{M} = \mathbf{M} - \mathbf{M}(\mathbf{C} + \mathbf{M})^{-1}\mathbf{M} = \mathbf{M}(\mathbf{C} + \mathbf{M})^{-1}\mathbf{C}. \quad (\text{A-7})$$

Therefore, it can be verified that

$$\mathcal{A}\mathcal{P}^{-1}\mathcal{A} = \begin{pmatrix} U & 0 \\ 0 & U \end{pmatrix} \quad \text{and} \quad \mathcal{P} = \begin{pmatrix} U + V & 0 \\ 0 & U + V \end{pmatrix}, \quad (\text{A-8})$$

where $U = \mathbf{C}(\mathbf{C} + \mathbf{M})^{-1}\mathbf{C} + \mathbf{M}(\mathbf{C} + \mathbf{M})^{-1}\mathbf{M}$ and $V = 2\mathbf{C}(\mathbf{C} + \mathbf{M})^{-1}\mathbf{M}$. This means that the second estimate above, $c(\mathcal{P}z, z) \leq (\mathcal{A}\mathcal{P}^{-1}\mathcal{A}z, z)$, is equivalent with the bound $c((U + V)x, x) \leq (Ux, x)$.

Using the symmetry of \mathbf{C} and \mathbf{M} and the Schwarz inequality in the $(\mathbf{C} + \mathbf{M})^{-1}$ inner product we obtain

$$\begin{aligned} (Vx, x) &= 2((\mathbf{C} + \mathbf{M})^{-1}\mathbf{M}x, \mathbf{C}x) \\ &\leq \|\mathbf{M}x\|_{(\mathbf{C} + \mathbf{M})^{-1}}^2 + \|\mathbf{C}x\|_{(\mathbf{C} + \mathbf{M})^{-1}}^2 = (Ux, x). \end{aligned} \quad (\text{A-9})$$

Therefore, we get the second estimate with $c = 1/2$. This estimate is sharp for any x and y that are nonzero eigenvectors of $\mathbf{M}^{-1}\mathbf{C}$.

Combining the two estimates and using the fact that they are sharp, we get that the block-diagonal preconditioner leads to the following (optimal) condition number:

$$\kappa(\mathcal{P}^{-1}\mathcal{A}) = \kappa(\mathcal{A}\mathcal{P}^{-1}) = \sqrt{\kappa(\mathcal{S})} = \frac{C}{\sqrt{c}} = \sqrt{2}. \quad (\text{A-10})$$

REFERENCES

- Ainsworth, M., G. Andriamaro, and O. Davydov, 2011, Bernstein-Bezier finite elements of arbitrary order and optimal assembly procedures: *SIAM Journal on Scientific Computing*, **33**, 3087–3109, doi: [10.1137/11082539X](https://doi.org/10.1137/11082539X).
- Ainsworth, M., and J. Coyle, 2003, Conditioning of hierarchic p -version Nédélec elements on meshes of curvilinear quadrilaterals and hexahedra: *SIAM Journal on Numerical Analysis*, **41**, 731–750, doi: [10.1137/S003614290239590X](https://doi.org/10.1137/S003614290239590X).
- Ainsworth, M., and J. T. Oden, 2000, *A posteriori error estimation in finite element analysis*: John Wiley & Sons.
- Baker, A., R. Falgout, T. Kolev, and U. Yang, 2012, Scaling hypre's multigrid solvers to 100,000 cores, in M. W. Berry, K. A. Gallivan, E. Gallopoulos, A. Grama, B. Philippe, Y. Saad, and F. Saied, eds., *High performance scientific computing: Algorithms and applications*: Springer, 261–279.
- Balay, S., S. Abhyankar, M. F. Adams, J. Brown, P. Brune, K. Buschelman, V. Eijkhout, W. D. Gropp, D. Kaushik, M. G. Knepley, L. C. McInnes, K. Rupp, B. F. Smith, and H. Zhang, 2014, *PETSc users manual: Technical Report ANL-95/11 — Revision 3.5*, Argonne National Laboratory.
- Bangerth, W., C. Burstedde, T. Heister, and M. Kronbichler, 2011, Algorithms and data structures for massively parallel generic adaptive finite element codes: *ACM Transactions on Mathematical Software (TOMS)*, **38**, 14.
- Bangerth, W., and O. Kayser-Herold, 2009, Data structures and requirements for hp finite element software: *ACM Transactions on Mathematical Software*, **36**, 4, doi: [10.1145/1486525.1486529](https://doi.org/10.1145/1486525.1486529).
- Bangerth, W., and R. Rannacher, 2003, *Adaptive finite element methods for differential equations*: Springer.
- Berdichevskii, M. M. N., and V. I. Dmitriev, 2008, *Models and methods of magnetotellurics*: Springer.
- Binev, P., W. Dahmen, and R. DeVore, 2004, Adaptive finite element methods with convergence rates: *Numerische Mathematik*, **97**, 219–268, doi: [10.1007/s00211-003-0492-7](https://doi.org/10.1007/s00211-003-0492-7).
- Börner, R.-U., 2010, Numerical modelling in geo-electromagnetics: Advances and challenges: *Surveys in Geophysics*, **31**, 225–245, doi: [10.1007/s10712-009-9087-x](https://doi.org/10.1007/s10712-009-9087-x).
- Bossavit, A., 1998, *Computational electromagnetism*: Academic Press.
- Bürg, M., 2012a, *A fully automatic hp -adaptive refinement strategy*: Ph.D. thesis, Institut für Angewandte und Numerische Mathematik.

- Bürg, M., 2012b, A residual-based a posteriori error estimator for the hp -finite element method for Maxwell's equations: *Applied Numerical Mathematics*, **62**, 922–940, doi: [10.1016/j.apnum.2012.02.007](https://doi.org/10.1016/j.apnum.2012.02.007).
- Bürg, M., 2013, Convergence of an automatic hp -adaptive finite element strategy for Maxwell's equations: *Applied Numerical Mathematics*, **72**, 188–204, doi: [10.1016/j.apnum.2013.04.008](https://doi.org/10.1016/j.apnum.2013.04.008).
- Burstedde, C., L. C. Wilcox, and O. Ghattas, 2011, `p4est`: Scalable algorithms for parallel adaptive mesh refinement on forests of octrees: *SIAM Journal on Scientific Computing*, **33**, 1103–1133, doi: [10.1137/100791634](https://doi.org/10.1137/100791634).
- Chave, A. D., and A. G. Jones, 2012, *The magnetotelluric method: Theory and practice*: Cambridge University Press.
- Chen, J., Z. Chen, T. Cui, and L.-B. Zhang, 2010, An adaptive finite element method for the eddy current model with circuit/field couplings: *SIAM Journal on Scientific Computing*, **32**, 1020–1042, doi: [10.1137/080713112](https://doi.org/10.1137/080713112).
- Dörfler, W., 1996, A convergent adaptive algorithm for Poisson's equation: *SIAM Journal on Numerical Analysis*, **33**, 1106–1124, doi: [10.1137/0733054](https://doi.org/10.1137/0733054).
- Farquharson, C. G., and M. P. Miensoop, 2011, Three-dimensional finite-element modelling of magnetotelluric data with a divergence correction: *Journal of Applied Geophysics*, **75**, 699–710, doi: [10.1016/j.jappgeo.2011.09.025](https://doi.org/10.1016/j.jappgeo.2011.09.025).
- Gee, M., C. Siefert, J. Hu, R. Tuminaro, and M. Sala, 2006, *ML 5.0 smoothed aggregation user's guide*: Technical Report SAN D2006-2649, Sandia National Laboratories.
- Grayver, A. V., and M. Bürg, 2014, Robust and scalable 3-D geo-electromagnetic modelling approach using the finite element method: *Geophysical Journal International*, **198**, 110–125, doi: [10.1093/gji/ggu119](https://doi.org/10.1093/gji/ggu119).
- Guo, B., and I. Babuska, 1986, The h - p version of the finite element method: *Computational Mechanics*, **1**, 21–41, doi: [10.1007/BF00298636](https://doi.org/10.1007/BF00298636).
- Haber, E., 2014, *Computational methods in geophysical electromagnetics*, Society for Industrial and Applied Mathematics, Mathematics in Industry.
- Henson, V. E., and U. M. Yang, 2002, BoomerAMG: A parallel algebraic multigrid solver and preconditioner: *Applied Numerical Mathematics*, **41**, 155–177, doi: [10.1016/S0168-9274\(01\)00115-5](https://doi.org/10.1016/S0168-9274(01)00115-5).
- Hiptmair, R., and J. Xu, 2007, Nodal auxiliary space preconditioning in H (curl) and H (div) spaces: *SIAM Journal on Numerical Analysis*, **45**, 2483–2509, doi: [10.1137/060660588](https://doi.org/10.1137/060660588).
- Hu, J., R. Tuminaro, P. Bochev, C. Garasi, and A. Robinson, 2006, Toward an h -independent algebraic multigrid method for Maxwell's equations: *SIAM Journal on Scientific Computing*, **27**, 1669–1688, doi: [10.1137/040608118](https://doi.org/10.1137/040608118).
- Hypre, 2014, High performance preconditioners, <http://www.llnl.gov/CASC/hypre>, accessed 30 July 2015.
- Jin, J., 2002, *The finite element method in electromagnetics*: Wiley.
- Key, K., 2012, Marine electromagnetic studies of seafloor resources and tectonics: *Surveys in Geophysics*, **33**, 135–167, doi: [10.1007/s10712-011-9139-x](https://doi.org/10.1007/s10712-011-9139-x).
- Key, K., and J. Owall, 2011, A parallel goal-oriented adaptive finite element method for 2.5-D electromagnetic modelling: *Geophysical Journal International*, **186**, 137–154, doi: [10.1111/j.1365-246X.2011.05025.x](https://doi.org/10.1111/j.1365-246X.2011.05025.x).
- Kirby, R. C., and K. T. Thinh, 2012, Fast simplicial quadrature-based finite element operators using Bernstein polynomials: *Numerische Mathematik*, **121**, 261–279, doi: [10.1007/s00211-011-0431-y](https://doi.org/10.1007/s00211-011-0431-y).
- Kolev, T. V., and P. Vassilevski, 2009, Parallel auxiliary space AMG for H (curl) problems: *Journal of Computational Mathematics*, **27**, 604–623, doi: [10.4208/jcm.2009.27.5.013](https://doi.org/10.4208/jcm.2009.27.5.013).
- Løseth, L. O., H. M. Pedersen, B. Ursin, L. Amundsen, and S. Ellingsrud, 2006, Low-frequency electromagnetic fields in applied geophysics: Waves or diffusion?: *Geophysics*, **71**, no. 4, W29–W40, doi: [10.1190/1.2208275](https://doi.org/10.1190/1.2208275).
- Luo, X.-J., M. S. Shephard, L.-Q. Lee, L. Ge, and C. Ng, 2011, Moving curved mesh adaptation for higher-order finite element simulations: *Engineering with Computers*, **27**, 41–50, doi: [10.1007/s00366-010-0179-5](https://doi.org/10.1007/s00366-010-0179-5).
- MFEM, 2014, Modular parallel finite-element methods, <http://mfem.org>, accessed 30 July 2015.
- Monk, P., 2003, *Finite-element methods for Maxwell's equations*: Oxford University Press.
- Mulder, W., 2006, A multigrid solver for 3D electromagnetic diffusion: *Geophysical Prospecting*, **54**, 633–649, doi: [10.1111/j.1365-2478.2006.00558.x](https://doi.org/10.1111/j.1365-2478.2006.00558.x).
- Muñoz, G., 2014, Exploring for geothermal resources with electromagnetic methods: *Surveys in Geophysics*, **35**, 101–122, doi: [10.1007/s10712-013-9236-0](https://doi.org/10.1007/s10712-013-9236-0).
- Ren, Z., T. Kalscheuer, S. Greenhalgh, and H. Maurer, 2013, A goal-oriented adaptive finite-element approach for plane wave 3-d electromagnetic modelling: *Geophysical Journal International*, **194**, 700–718, doi: [10.1093/gji/ggt154](https://doi.org/10.1093/gji/ggt154).
- Ren, Z., T. Kalscheuer, S. Greenhalgh, and H. Maurer, 2014, A finite-element-based domain-decomposition approach for plane wave 3D electromagnetic modeling: *Geophysics*, **79**, no. 6, E255–E268, doi: [10.1190/geo2013-0376.1](https://doi.org/10.1190/geo2013-0376.1).
- Rieben, R., D. White, and G. R., 2004, High-order symplectic integration methods for finite element solutions to time dependent Maxwell equations: *IEEE Transactions on Antennas and Propagation*, **52**, 2190–2195, doi: [10.1109/TAP.2004.832356](https://doi.org/10.1109/TAP.2004.832356).
- Saad, Y., 2003, *Iterative methods for sparse linear systems* 2nd ed.: Society for Industrial and Applied Mathematics.
- Schwarzbach, C., R.-U. Börner, and K. Spitzer, 2011, Three-dimensional adaptive higher order finite element simulation for geoelectromagnetics — A marine CSEM example: *Geophysical Journal International*, **187**, 63–74, doi: [10.1111/j.1365-246X.2011.05127.x](https://doi.org/10.1111/j.1365-246X.2011.05127.x).
- Šolín, P., K. Segeth, and I. Doležal, 2004, *Higher-order finite-element methods*: Chapman & Hall/CRC.
- Um, E. S., M. Commer, and G. A. Newman, 2013, Efficient preconditioned iterative solution strategies for the electromagnetic diffusion in the Earth: Finite-element frequency-domain approach: *Geophysical Journal International*, **193**, 1460–1473, doi: [10.1093/gji/ggt071](https://doi.org/10.1093/gji/ggt071).
- Wang, Y., W. Bangerth, and J. Ragusa, 2009, Three-dimensional h -adaptivity for the multigroup neutron diffusion equations: *Progress in Nuclear Energy*, **51**, 543–555, doi: [10.1016/j.pnucene.2008.11.005](https://doi.org/10.1016/j.pnucene.2008.11.005).
- Xu, J., 1996, The auxiliary space method and optimal preconditioning techniques for unstructured grids: *Computing*, **56**, 215–235, doi: [10.1007/BF02238513](https://doi.org/10.1007/BF02238513).
- Zaglmayr, S., 2006, *High order finite element methods for electromagnetic field computation*: Ph.D. thesis, Johannes Kepler University.
- Zhdanov, M., I. M. Varentsov, J. Weaver, N. Golubev, and V. Krylov, 1997, Methods for modelling electromagnetic fields results from COMMEMI — The international project on the comparison of modelling methods for electromagnetic induction: *Journal of Applied Geophysics*, **37**, 133–271, doi: [10.1016/S0926-9851\(97\)00013-X](https://doi.org/10.1016/S0926-9851(97)00013-X).
- Zhong, L., L. Chen, S. Shu, G. Wittum, and J. Xu, 2012, Convergence and optimality of adaptive edge finite element methods for time-harmonic Maxwell equations: *Mathematics of Computation*, **81**, 623–642, doi: [10.1090/S0025-5718-2011-02544-5](https://doi.org/10.1090/S0025-5718-2011-02544-5).