



Iterative solution methods for 3D controlled-source electromagnetic forward modelling of geophysical exploration scenarios

Michael Weiss¹ · Maya Neytcheva² · Thomas Kalscheuer¹

Received: 30 June 2022 / Accepted: 7 November 2022 / Published online: 6 December 2022
© The Author(s) 2022

Abstract

We develop an efficient and robust iterative framework suitable for solving the linear system of equations resulting from the spectral element discretisation of the curl-curl equation of the total electric field encountered in geophysical controlled-source electromagnetic applications. We use the real-valued equivalent form of the original complex-valued system and solve this arising real-valued two-by-two block system (outer system) using the generalised conjugate residual method preconditioned with a highly efficient block-based PREconditioner for Square Blocks (PRESB). Applying this preconditioner equates to solving two smaller inner symmetric systems which are either solved using a direct solver or iterative methods, namely the generalised conjugate residual or the flexible generalised minimal residual methods preconditioned with the multigrid-based auxiliary-space preconditioner AMS. Our numerical experiments demonstrate the robustness of the outer solver with respect to spatially variable material parameters, for a wide frequency range of five orders of magnitude (0.1–10⁷000 Hz), with respect to the number of degrees of freedom, and for stretched structured and unstructured as well as locally refined meshes. For all the models considered, the outer solver reaches convergence in a small (typically < 20) number of iterations. Further, our numerical tests clearly show that solving the two inner systems iteratively using the indicated preconditioned iterative methods is computationally beneficial in terms of memory requirement and time spent as compared to a direct solver. On top of that, our iterative framework works for large-scale problems where direct solvers applied to the original complex-valued systems succumb due to their excessive memory consumption, thus making the iterative framework better suited for large-scale 3D problems. Comparison to a similar iterative framework based on a block-diagonal and the auxiliary-space preconditioners reveals that the PRESB preconditioner requires slightly fewer iterations to converge yielding a certain gain in time spent to obtain the solution of the two-by-two block system.

Keywords Applied geophysics · Geo-Electromagnetics · Spectral element method · Iterative solution methods · Preconditioning

1 Introduction

Geophysical electromagnetic (EM) methods are crucial tools in mapping the Earth's subsurface in terms of its electric conductivity distribution. These methods are particularly employed for geophysical surveys targeting

hydrocarbon and mineral resources (cf. e.g. [51, 86]), or for investigations concerned with geothermal, hydrogeological or environmental applications (cf. e.g. [65, 82, 87]). Such surveys have recorded large amounts of three-dimensional (3D) EM data in vastly different geological settings. The interpretation of such acquired data sets involves solving an inverse problem to obtain a subsurface model that best fits the recorded data. Although the inverse problem entails considerable computational burden itself, the repeated solution of the 3D EM forward problem within the inversion framework constitutes one of the core factors for the high computational cost of the inversion. This is further aggravated considering that realistic surveys comprise large 3D volumes with many receivers and possibly multiple sources demanding large 3D meshes to accurately represent

✉ Michael Weiss
michael.weiss@geo.uu.se

¹ Department of Earth Sciences, Uppsala University, Uppsala, Sweden

² Department of Information Technology, Uppsala University, Uppsala, Sweden

realistic subsurface structures and topographic variations yielding forward problems of large proportions, often including tens to hundreds of millions degrees of freedom. In light of this, solving the forward problem efficiently and effectively is crucial to limit the computational effort, in particular the time and memory requirements, for the forward problem itself as well as for inversions of realistic industrial 3D data sets.

The 3D EM forward simulation consists of solving the governing partial differential equation (PDE) derived from Maxwell's equations using discretisation. In EM geophysics, a number of different approaches have conventionally been employed in the past few decades to discretise the relevant continuous PDEs. Such methods include the finite difference (FD), finite element (FE) and finite volume (FV) methods. Regardless of the used approach, the corresponding discrete equations are expressed in terms of a linear algebraic system of equations of the form $\mathbf{Ax} = \mathbf{b}$, where \mathbf{A} is a large-scale sparse nonsingular matrix. There exists numerous numerical methods for solving the resulting linear system, which are of two general types - direct and iterative, both with their inherent advantages and drawbacks.

It is well known that obtaining accurate and efficient solutions of the algebraic system of equations arising from Maxwell's equations is a computationally difficult task. This is due to several factors which impact the condition number of the system matrix \mathbf{A} negatively and can make it very ill-conditioned: (1) the influence of the large kernel of the curl operator $\nabla \times$ present in Maxwell's equations [80, 81]; (2) large electric conductivity contrasts, in particular across the air-Earth interface where the contrast amounts to four orders of magnitude or more; (3) the requirement of non-uniform and locally refined meshes; (4) using high-order polynomials p for the space approximation not only yields a more densely populated system matrix but also a more ill conditioned one. In fact, the condition number of the system matrix in 3D increases with $\mathcal{O}(p^8)$ for the finite element method as shown by [1].

Direct methods are based on an exact lower-upper triangular (LU) factorisation of the matrix \mathbf{A} or its Cholesky decomposition LL^T when \mathbf{A} is symmetric and positive definite. Examples of efficient sparse direct solvers that are widely employed in the geophysical EM community include a distributed-memory multifrontal massively parallel sparse direct solver called MUMPS [3, 4] and a shared-memory parallel direct solver known as PARDISO [77]. Direct methods are robust and accurate, even for systems with ill-conditioned matrices. In addition, for the solution of linear systems with multiple right-hand sides as in multi-source surveys in controlled source electromagnetics (CSEM), direct solvers only require a single factorisation of the

system matrix (see e.g. [32, 66, 85]). On the other hand, the memory demands and time requirements grow as $\mathcal{O}(N^2)$ and $\mathcal{O}(N^{4/3})$ in 3D, respectively, where N is the number of unknowns and can limit the size of feasible problems depending on the available computer platform. In regard to (parallel) scalability, direct solvers are considered non-optimal on distributed-memory platforms due to a low computation-to-synchronisation ratio [43, 61]. In practice, this means that, due to the additional overhead caused by communication and synchronisation between the nodes, beyond a certain number of parallel processes direct computations cannot be sped up by adding more computing nodes/cores (see e.g., [32, 68]). Nevertheless, owing to successive advances in direct solution algorithms [5, 78], as well as the increasing availability and power of high-performance computing facilities [60], direct solvers are often used when solving the EM forward problem see (e.g. [29, 32, 35, 68, 71, 72, 81, 85], for details).

Iterative methods, e.g. Krylov subspace iteration methods, include a broad range of solution methods that use successive approximations to acquire progressively more accurate solutions of the considered linear system of equations at each step until reaching convergence. The main advantage of iterative methods is that they require less computational resources than direct methods in terms of time and memory, and are thus better suited to handle large-scale problems. The major computational building blocks of iterative solvers include cheap matrix-vector products and vector operations which can be easily parallelised making them more scalable. However, for ill-conditioned systems, iterative methods, if straightforwardly applied, can exhibit slow convergence or even diverge. Therefore iterative solvers require the usage of an efficient preconditioner. Preconditioners transform the original system of equations into an equivalent one that has a (preconditioned) matrix with a more favourable eigenvalue spectrum. This ought to improve the convergence rate of iterative methods whilst sufficiently offsetting the additional computational burden of building and applying the preconditioner. In this way, preconditioned iterative solvers become viable and cost-effective solution methods, in particular from the perspective of large-scale 3D problems. The task to construct a numerically effective and computationally efficient preconditioner is far from trivial and in many cases the preconditioner might be problem-specific.

Examples of iterative methods and corresponding preconditioners include the Biconjugate gradient (BICG) or the Quasi-minimal residual (QMR) method with different preconditioners such as diagonal or Jacobi scaling [2, 28], the incomplete Cholesky decomposition [83], or the incomplete LU factorisation [89]. Other commonly used iterative solvers are the Generalised minimal residual (GMRES) and the Biconjugate gradient stabilised (BICGStab) methods

which, for instance, have been used by [39] and [49] preconditioned by incomplete LU decomposition. [67] investigate the convergence behaviour of the aforementioned iterative methods preconditioned with a sparse approximation of the inverse of the system matrix based on the minimisation of a suitable Frobenius norm. In addition and in order to solve problems with multiple right-hand sides (RHS), [66] present block GMRES and QMR methods [76] using either an incomplete LU, an approximate inverse or a simplified geometric multigrid preconditioner.

The class of geometric Multigrid (MG) and algebraic Multigrid (AMG) methods has gained popularity, because in many cases they are optimal regarding numerical efficiency, they converge in a number of iterations, that is independent of the number of degrees of freedom, and they also have optimal (or nearly optimal) computational complexity, namely, linear with respect to the number of degrees of freedom. MG methods make use of changing discretisation on a sequence of successively coarser grids. These techniques are based on the observation that although most relaxation methods, e.g. Jacobi and Gauss-Seidel algorithms, may in general converge slowly, they attenuate effectively and very rapidly the oscillatory (high-frequency) modes of the errors of any given mesh. However, standard relaxation methods hardly damp the smooth (low-frequency) components of the error within the few first iterations. Thus, after removing the oscillatory error components, the convergence of these basic iterative methods slows down as the smooth modes are eliminated slowly. However, the smooth modes on a given discretisation mesh naturally become oscillatory modes on a coarser grid, which is the origin of the idea of moving to a coarser mesh discretisation in order to remove the corresponding error mode once the relaxation methods start stalling. The process of moving to coarser grids can be repeated recursively, in this way attenuating effectively and rapidly the oscillatory modes on the different levels, thus reducing the global error quickly. In contrast to MG methods, AMG methods achieve the same effect however, without using any discretisation meshes but creating an analogous algebraic structure based on an aggregation technique. Therefore, AMG methods are more generally applicable than MG methods and are their preferred alternative. Multigrid methods can be used as stand-alone solution methods but more often are used as preconditioners. These methods are considered scalable [31, 94] and can have linear complexity $\mathcal{O}(N)$ in computational and memory load for various boundary value problems [36, 88] making them a viable option for large-scale simulations. Despite this and in comparison to direct and iterative methods, multigrid methods have not been widely used in geoelectromagnetic applications, which might be due to their involved numerical implementation and the fact that

generic multigrid schemes fail for Maxwell's equations due to the large kernel of the curl operator [26, 80, 81]. Since the early 2000s, renewed development in multigrid methods, in particular for the curl-curl operator [46–48, 70], as well as the availability of several software libraries such as *hypre* [38] and *ML* [40], have improved the applicability of MG methods to geophysical EM problems. For example, [6] apply a geometric multigrid preconditioner based on Dendy's black box multigrid solver [33]. Mulder [58] has developed a geometric multigrid method which is either used directly or as a preconditioner and has reported good convergence rates for uniform grid spacing, but inadequate rates for substantially stretched grids. More recently, [52] have implemented an AMG scheme as a preconditioner for Krylov subspace iteration solvers that is shown to provide mesh-independent rate of convergence.

Often MG and AMG methods are used in various preconditioning techniques as a block-solver. An example of this for application with conductors can be found in [30] in the following context. A complex-valued system is rewritten as a real two-by-two block system, preconditioned by a block-diagonal preconditioner. The two linear systems of the diagonal blocks are solved by an iterative method with the auxiliary-space technique [48, 54]. The main idea consists of approximately inverting the curl-curl operator by casting the problem into subspaces in which it can be solved efficiently by AMG methods. This preconditioning framework introduced in [30] has also been adopted to EM modelling problems (cf e.g. [25, 44, 45, 69]).

Two-by-two block systems are encountered in numerical applications in various fields, e.g. discrete Navier-Stokes problems, eddy current problems, linear elasticity, optimisation problems constrained by PDEs etc. As mentioned, these also arise when transforming a given complex-valued system to its real-valued equivalent. As the occurrence of block systems is quite common much research has been carried out and hence efficient preconditioners for block-structured systems can be found in the literature. Here, we restrict our selection to a few robust and efficient techniques for block-structured systems. These include block-diagonal and block-triangular preconditioners [12, 56, 84] and the preconditioned modified Hermitian - skew Hermitian splitting (PMHSS) iteration method [17–19]. Another preconditioning technique presented in various publications (cf. e.g., [9, 10, 15, 27]) is referred to as the PREconditioning for Square Blocks (PRESB).

The objective in this work is to present a robust and efficient iterative solution framework for the algebraic system of equations arising from discretising problems in 3D frequency-domain controlled-source electromagnetic modelling. For the solution of the system, we consider the PRESB preconditioning technique [10] to be used in a Krylov subspace iteration method. An action of the

PRESB preconditioner amounts to solving two additional linear systems, which we solve by an inner iterative method preconditioned by the auxiliary-space preconditioner [48, 54]. We emphasise that PRESB, although resembling the block-diagonal preconditioner presented in some other related work (e.g. [25, 44, 45, 69]), is different. In particular, applying these preconditioner results in slightly different computational procedures although the core of both schemes consists in solving two linear systems. Further and at a later point in this work, we show that the PRESB and block-diagonal preconditioners yield different eigenvalue bounds for the preconditioned block systems. To the best of our knowledge, so far this preconditioner has not been employed in geo-EM related applications. We also stress that we use a spectral element discretisation method, whereas previous works use finite element methods. In contrast to [44, 45] and [69], we do not disregard displacement currents which complicates the system due to the potential negative shift in matrix elements resulting from the displacement current term in some parts of the modelling domain [25].

The remainder of this paper is structured as follows. Section 2 describes the mathematical model, its discretisation and the properties of the arising algebraic system of equations. In Section 2.1, we present the formulation of the continuous problem. We then derive the dimensionless form of the problem at hand in Section 2.1.1 and briefly describe its discretisation using the spectral element method in Section 2.2. Section 3 describes the solution procedure we choose and some possible block preconditioning techniques for two-by-two block system, such as the block-diagonal, the Schur complement-based and the PRESB preconditioners in Sections 3.2, 3.3 and 3.4, respectively. When introducing the PRESB preconditioner, we also present the general block iterative framework. Section 3.4.1 briefly describes the auxiliary-space Maxwell preconditioner. Numerical results for two test problems illustrating the robustness of the solver are presented in Section 4. We compare the performance of the PRESB preconditioner to the diagonal preconditioner employed in [44, 45] and [69]. Some concluding remarks are given in Section 5.

2 The continuous problem and its discrete formulation

2.1 Formulation of the continuous problem

Starting from Maxwell's equations in frequency domain

$$\begin{aligned}\nabla \times \mathbf{E} &= -i\omega\mu\mathbf{H} \\ \nabla \times \mathbf{H} &= \mathbf{J}_s + \sigma\mathbf{E} + i\omega\varepsilon\mathbf{E},\end{aligned}\quad (1)$$

the governing partial differential equation for the total electric field in CSEM applications is derived by eliminating the magnetic field and reads as follows,

$$\nabla \times \left(\frac{1}{\mu} \nabla \times \mathbf{E} \right) + i\omega\sigma\mathbf{E} - \omega^2\varepsilon\mathbf{E} = -i\omega\mathbf{J}_s \quad \text{in } \Omega, \quad (2a)$$

where Ω is the bounded model domain, \mathbf{E} denotes the total electric field, ω is angular frequency, μ is magnetic permeability, σ is electric conductivity, ε is dielectric permittivity and \mathbf{J}_s is the known imposed source current density. In electromagnetic geophysics, all three material properties μ , σ and ε are assumed to be spatially varying as well as piecewise smooth. However, we note that both the magnetic permeability and dielectric permittivity occupy a relative narrow range. The parameter μ can be safely assumed to lie within $1\mu_0$ to $3\mu_0$, where μ_0 is the magnetic permeability of vacuum $\mu_0 = 4\pi \cdot 10^{-7}$ H/m. The parameter ε ranges from $1\varepsilon_0$ to $81\varepsilon_0$ (relative dielectric permittivity of water) with ε_0 being the dielectric permittivity of free space $\varepsilon_0 = 8.854 \cdot 10^{-12}$ F/m. The remaining physical property σ , on the other hand, is strongly material-dependent and can range from 10^2 S/m for highly conductive massive sulfide ore deposits to 10^{-6} S/m for very resistive metamorphic and igneous rocks. In addition, most applications in EM geophysics set the air conductivity on the order of 10^{-7} to 10^{-10} S/m to ensure that it is sufficiently small but larger than zero.

The range of frequencies employed in CSEM depends on the objective of the survey as well as the system available and its specifications. However, the frequency range influences the size of the computational domain. In particular, to limit computational boundary effects, the domain has to be chosen adequately. Typically, the plane-wave skin depth criterion which is a function of the background conductivity σ_b , permeability μ_b , permittivity ε_b and the employed source frequency f can be used as a guideline to set appropriate domain boundaries. In order to adequately attenuate the electric field at the domain boundaries, the computational domain is extended up to a distance of approximately five plane-wave skin depths from the area of interest. It is further noted that the frequency- and material-dependent wavelength influences the choice of the element size as the grid spacing needs to be sufficiently fine compared to the wavelength in order to ensure accurate numerical computations [57]. Grid spacing is typically chosen as a fraction of the skin depth and based on element order. For first-order approximation, guidelines on grid spacing relevant for finite difference schemes and for finite element modelling in geophysical electromagnetics have been presented in [92] and [91] respectively.

In order to guarantee the uniqueness of the electric field \mathbf{E} , Eq. 2a is subjected to homogeneous Dirichlet boundary conditions on the domain boundary $\partial\Omega$, namely,

$$\mathbf{n} \times \mathbf{E} = 0. \tag{2b}$$

2.1.1 Dimensionless formulation of the curl-curl equation

Given the vastly varying orders of magnitudes of some of the model parameters and the large scale of the space domain, it is preferable to introduce dimensionless quantities for the governing Eq. 2a. The dimensionless forms of the equations serve to properly scale the problem and avoid ill-conditioning due to scaling - the problem becomes independent of measurement units and facilitates better understanding of the nature of the equations without involving physical parameters (see e.g. [22, 24], for more details).

To obtain a dimensionless form we introduce the following new dimensionless variables, denoted by $\tilde{\cdot}$ and the corresponding suitable scaling factors, denoted by subscript f , namely,

$$\begin{aligned} \mathbf{E} &= E_f \tilde{\mathbf{E}}, \quad \mathbf{J}_s = J_{s,f} \tilde{\mathbf{J}}_s, \quad \mathbf{x} = L_f \tilde{\mathbf{x}}, \quad \mu = \mu_f \tilde{\mu}, \\ \omega &= \omega_f \tilde{\omega}, \quad \varepsilon = \varepsilon_f \tilde{\varepsilon}, \quad \sigma = \sigma_f \tilde{\sigma}, \end{aligned}$$

where \mathbf{x} is a three-dimensional space vector. The curl operator $\nabla \times$, being a spatial derivative, transforms in a similar way,

$$\nabla \times = \frac{1}{L_f} \tilde{\nabla} \times.$$

We now replace the dimensional quantities in Eq. 2a with their respective dimensionless equivalents yielding the dimensionless curl-curl equation

$$\begin{aligned} \frac{1}{L_f} \tilde{\nabla} \times \left(\frac{1}{\mu_f \tilde{\mu}} \frac{1}{L_f} \tilde{\nabla} \times E_f \tilde{\mathbf{E}} \right) + i \omega_f \tilde{\omega} \tilde{\sigma}_f \tilde{\sigma} E_f \tilde{\mathbf{E}} \\ - \omega_f^2 \tilde{\omega}^2 \varepsilon_f \tilde{\varepsilon} E_f \tilde{\mathbf{E}} = -i \omega_f \tilde{\omega} J_{s,f} \tilde{\mathbf{J}}_s. \end{aligned} \tag{3}$$

Multiplying Eq. 3 by $\frac{L_f^2 \mu_f}{E_f}$ and collecting the scaling factors for each term in dimensionless coefficients results in

$$\tilde{\nabla} \times \left(\frac{1}{\tilde{\mu}} \tilde{\nabla} \times \tilde{\mathbf{E}} \right) + i \alpha \tilde{\omega} \tilde{\sigma} \tilde{\mathbf{E}} - \beta \tilde{\omega}^2 \tilde{\varepsilon} \tilde{\mathbf{E}} = -i \gamma \tilde{\omega} \tilde{\mathbf{J}}_s, \tag{4}$$

where it can readily be seen that the coefficients α , β and γ are indeed dimensionless and defined as follows

$$\begin{aligned} \alpha &= \omega_f \mu_f \sigma_f L_f^2 \left[s^{-1} \frac{Vs}{Am} m^2 \frac{A}{Vm} \right], \\ \beta &= \omega_f^2 \mu_f \varepsilon_f L_f^2 \left[s^{-2} \frac{Vs}{Am} m^2 \frac{As}{Vm} \right], \\ \gamma &= \frac{\omega_f \mu_f L_f^2 J_{s,f}}{E_f} \left[s^{-1} \frac{Vs}{Am} m^2 \frac{A}{m^2 V} \right]. \end{aligned}$$

The individual scaling factors have to be chosen in a suitable manner. Spatial dimensions are rescaled by L_f

which is taken to be the longest spatial dimension of the model. The latter ensures that the characteristic size of the space discretisation parameter, usually denoted by h , is less than 1. Further, the model conductivities are regularised by choosing σ_f to be equal to the largest conductivity of the conductivity distribution. Similarly, the dielectric permittivities ε and the magnetic permeabilities μ are rescaled with their respective largest value. This means that the largest dimensionless conductivity $\tilde{\sigma}$, permittivity $\tilde{\varepsilon}$ and permeability $\tilde{\mu}$ values are equal to one. The remaining scaling coefficients ω_f , E_f and $J_{s,f}$ are set to one. At this point we also note that, after solving the dimensionless system, to obtain the unscaled electric field we have to rescale the solution vector by $\frac{1}{L_f^2}$.

To simplify the notations, in the sequel we drop $\tilde{\cdot}$ and, unless stated otherwise, we work with the dimensionless parameters and the scaled variables.

2.2 Discretisation approach

In this work, we use a particular discretisation approach, referred to as the *Spectral Element Discretisation*, described in detail in [93]. The main reason for using the spectral element method instead of the more traditional finite element method is to combine and harness advantages of the spectral method in form of high accuracy and favourable convergence properties [42, 62, 73] while retaining the geometrical flexibility of the finite element method (see e.g. [62, 73, 93, 95]). For completeness, we include a short description below.

Following Galerkin’s method, by taking the L^2 -inner product $(\langle f, g \rangle_{L^2} := \int_{\Omega} f \cdot g)$ of the dimensionless curl-curl equation for the electric field Eq. 4 with a vector test function Φ , integrating over Ω and using integration by parts, one obtains the weak form of Eq. 4 (cf. [57]):

Find $\mathbf{E} \in \mathcal{H}_0(\text{curl}, \Omega)$ such that

$$\begin{aligned} \int_{\Omega} \left(\frac{1}{\mu} \nabla \times \mathbf{E} \right) \cdot (\nabla \times \Phi) + i \omega (\alpha \sigma + i \beta \omega \varepsilon) \mathbf{E} \cdot \Phi \, d\Omega \\ = -i \gamma \omega \int_{\Omega} \mathbf{J}_s \cdot \Phi \, d\Omega \end{aligned} \tag{5}$$

for all $\Phi \in \mathcal{H}_0(\text{curl}, \Omega)$, and where $\mathcal{H}_0(\text{curl}, \Omega)$ is defined as

$$\mathcal{H}_0(\text{curl}, \Omega) = \{ \mathbf{E} \in \mathcal{H}(\text{curl}, \Omega) : \mathbf{n} \times \mathbf{E} = 0 \text{ on } \partial\Omega \}. \tag{6}$$

The space $\mathcal{H}(\text{curl}, \Omega)$ denotes the space of functions defined on the bounded domain Ω , which are in the square integrable space $(L^2(\Omega))^3$ and such that the curl is well defined in $(L^2(\Omega))^3$

$$\mathcal{H}(\text{curl}, \Omega) = \{ \mathbf{u} \in (L^2(\Omega))^3 \mid \nabla \times \mathbf{u} \in (L^2(\Omega))^3 \}. \tag{7}$$

The spectral element discretisation of the weak form Eq. 5 of the boundary value problem Eq. 2a starts by

subdividing the domain Ω into a set of non-overlapping hexahedra. In addition, one needs to replace the function space $\mathcal{H}_0(\text{curl}, \Omega)$ by a finite dimensional approximation space consisting of curl-conforming vector functions which are element-wise polynomial functions of arbitrary degree within each element and non-zero for all but a few hexahedra [81]. We note, that due to the specific functional space, the tangential components of the vector basis functions are continuous across the interface of two neighbouring elements, whereas their normal components may be discontinuous across the same element boundary [59]. Letting $\mathcal{H}_{0,h}$ be such an approximate subspace to $\mathcal{H}_0(\text{curl}, \Omega)$ defined on the mesh, one can construct a finite set of basis functions that spans this subspace. This then yields the discrete formulation to find the approximate solution $\mathbf{E}_h \in \mathcal{H}_{0,h}$ such that

$$\begin{aligned} \langle \frac{1}{\mu} \nabla \times \mathbf{E}_h, \nabla \times \Phi \rangle + i\alpha\omega \langle \sigma \mathbf{E}_h, \Phi \rangle \\ - \beta\omega^2 \langle \epsilon \mathbf{E}_h, \Phi \rangle = -i\gamma\omega \langle \mathbf{J}_s, \Phi \rangle \end{aligned} \tag{8}$$

for all $\Phi \in \mathcal{H}_{0,h}$, and where the more compact notation of the inner product is used. Expanding the solution as $\mathbf{E}_h := \sum_{k=1}^{n_{\text{dof}}} E_k \Phi_k$ and inserting it into Eq. 8 yields

$$\begin{aligned} \sum_{k=1}^{n_{\text{dof}}} E_k \langle \frac{1}{\mu} \nabla \times \Phi_k, \nabla \times \Phi_l \rangle + \sum_{k=1}^{n_{\text{dof}}} E_k i\alpha\omega \langle \sigma \Phi_k, \Phi_l \rangle \\ - \sum_{k=1}^{n_{\text{dof}}} E_k \beta\omega^2 \langle \epsilon \Phi_k, \Phi_l \rangle = -i\gamma\omega \langle \mathbf{J}_s, \Phi_l \rangle, \end{aligned} \tag{9}$$

to be fulfilled by any l , or in matrix-vector notation

$$\underbrace{(\mathbf{K} + i\mathbf{M}_\sigma - \mathbf{M}_\epsilon)}_{\mathbf{C}_A} \mathbf{E}_h = \mathbf{b}, \tag{10}$$

where \mathbf{C}_A is the sparse symmetric complex-valued system matrix, \mathbf{E}_h is the solution vector collecting the basis function expansion coefficients E_k which are also referred to as degrees of freedom (DOF) and n_{dof} is the total number of these degrees of freedom. The right-hand side \mathbf{b} contains the source term entries. Further, \mathbf{K} denotes the sparse symmetric positive semi-definite stiffness matrix and \mathbf{M}_σ and \mathbf{M}_ϵ are the sparse symmetric positive definite mass matrices defined by

$$\begin{aligned} \mathbf{K}_{k,l} &= \langle \mu^{-1} \nabla \times \Phi_k, \nabla \times \Phi_l \rangle, \\ \mathbf{M}_{\sigma,k,l} &= \alpha\omega \langle \sigma \Phi_k, \Phi_l \rangle, \\ \mathbf{M}_{\epsilon,k,l} &= \beta\omega^2 \langle \epsilon \Phi_k, \Phi_l \rangle. \end{aligned} \tag{11}$$

We note that the basis functions are real-valued and, hence, so are the matrices. As to the choice of the basis functions for the spectral element discretisation, we refer to [93] for details.

3 Iterative solution and preconditioning approaches

As already noted, the matrix $\mathbf{C}_A = \mathbf{K} + i\mathbf{M}_\sigma - \mathbf{M}_\epsilon$ in (10) is complex symmetric and of very large dimension. Due to the problem parameters, it is also very ill conditioned. Therefore, we need to solve the system with some iterative method and use an efficient preconditioner. We aim at achieving a preconditioned iterative solver, which is robust with respect to the problem parameters $\epsilon, \sigma, \mu, \omega$ and also with respect to the discretisation parameter h .

The system (10) can be solved in two ways. As a first alternative, it can be solved using complex arithmetics, utilising an appropriate Krylov subspace iterative method, such as GMRES [76], BICG [90] and some others. However, the question how to choose a good preconditioner remains open. Standard choices, such as incomplete LU techniques, are known not to be robust. Therefore, we choose the second alternative, i.e. to rewrite the system in real form of twice larger size.

As is well known, any $n \times n$ complex system of the form $(A + iB)(\mathbf{x} + i\mathbf{y}) = (\mathbf{u} + i\mathbf{w})$ can be transformed into a $2n \times 2n$ real system in several equivalent ways. A straightforward computation shows some of the equivalent forms of the arising two-by-two block real systems,

$$\begin{aligned} \begin{bmatrix} A & -B \\ B & A \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ \mathbf{y} \end{bmatrix} = \begin{bmatrix} \mathbf{u} \\ \mathbf{w} \end{bmatrix}, \quad \begin{bmatrix} B & A \\ A & -B \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ \mathbf{y} \end{bmatrix} = \begin{bmatrix} \mathbf{w} \\ \mathbf{u} \end{bmatrix}, \\ \begin{bmatrix} A & B \\ -B & A \end{bmatrix} \begin{bmatrix} \mathbf{y} \\ \mathbf{x} \end{bmatrix} = \begin{bmatrix} \mathbf{w} \\ \mathbf{u} \end{bmatrix}. \end{aligned} \tag{12}$$

Which form to use may depend on the properties of the matrices A and B . In our case the block \mathbf{M}_σ is symmetric and positive definite and we recast Eq. 10 into a real-equivalent form by splitting the electric field into real and imaginary parts $\mathbf{E}_h = \mathbf{E}_{\mathbf{R}h} + i\mathbf{E}_{\mathbf{I}h}$, thus obtaining a two-by-two block system as in the middle form of Eq. 12. We then slightly alter the system by introducing the substitution $\widehat{\mathbf{E}}_{\mathbf{I}h} = -\mathbf{E}_{\mathbf{I}h}$ and obtain the following discrete system of linear equations

$$\mathbf{A} \begin{bmatrix} \mathbf{E}_{\mathbf{R}h} \\ -\mathbf{E}_{\mathbf{I}h} \end{bmatrix} = \underbrace{\begin{bmatrix} \mathbf{M}_\sigma & -(\mathbf{K} - \mathbf{M}_\epsilon) \\ \mathbf{K} - \mathbf{M}_\epsilon & \mathbf{M}_\sigma \end{bmatrix}}_{\mathbf{A}} \begin{bmatrix} \mathbf{E}_{\mathbf{R}h} \\ -\mathbf{E}_{\mathbf{I}h} \end{bmatrix} = \begin{bmatrix} \mathbf{b}_{\mathbf{I}} \\ \mathbf{b}_{\mathbf{R}} \end{bmatrix}, \tag{13}$$

with the same definitions for the matrices as given in Eq. 11.

In the sequel, we work with the system described in Eq. 13 unless otherwise stated, and equations and expressions involving matrices written in upper case bold letters denote problem-specific cases, whereas expressions with matrices written in calligraphic letters are valid in general.

3.1 Preconditioners for indefinite matrices of two-by-two block form

The matrix in equation (13) belongs to the class of indefinite matrices of two-by-two block form,

$$\mathcal{A} = \begin{bmatrix} A & B^T \\ C & -D \end{bmatrix}, \tag{14}$$

where, in general $A \in \mathbb{R}^{n \times n}$, $B \in \mathbb{R}^{m \times n}$, $C \in \mathbb{R}^{m \times n}$ and $D \in \mathbb{R}^{m \times m}$, $n \geq m$. As pointed out in [23], under suitable partitioning, any linear system can be written in the two-by-two block-structured form (14) or in some equivalent form, such as $\begin{bmatrix} A & B^T \\ -C & D \end{bmatrix}$ or $\begin{bmatrix} A & -B^T \\ C & D \end{bmatrix}$. For comprehensive studies of the properties of these matrices we refer to [23] and [7] and the references therein.

The experience how to precondition matrices of the form (14) is very rich. For the case when $C = B$ we mention briefly some classical approaches, based on the exact block-LU factorisation of the matrix, namely,

$$\begin{aligned} \begin{bmatrix} A & B^T \\ B & D \end{bmatrix} &= \begin{bmatrix} A & 0 \\ B & S \end{bmatrix} \begin{bmatrix} I_n & A^{-1}B^T \\ 0 & I \end{bmatrix} \\ &= \begin{bmatrix} I_n & 0 \\ BA^{-1} & I_m \end{bmatrix} \begin{bmatrix} A & 0 \\ 0 & S \end{bmatrix} \begin{bmatrix} I_n & A^{-1}B^T \\ 0 & I_m \end{bmatrix}. \end{aligned}$$

Here I_n and I_m denote identity matrices of size n and m , respectively. In, for instance, [23] and [13], the matrices $\mathcal{P}_T = \begin{bmatrix} \tilde{A} & 0 \\ B & \tilde{S} \end{bmatrix}$ and $\mathcal{P}_D = \begin{bmatrix} \tilde{A} & 0 \\ 0 & \tilde{S} \end{bmatrix}$ are shown to be very good preconditioners to the matrix \mathcal{A} in equation (14), provided that we can construct \tilde{A} and \tilde{S} to be very good approximations of A and S and also accurately solve systems with those when applying the preconditioner.

Matrices of the type (14) with square blocks occur not only when solving complex linear systems in real arithmetic, but also in various important applications, one example being the discrete first order necessary conditions in optimal control problems, constrained by partial differential equations. Below, we discuss preconditioners that utilise the same dimension of the diagonal and off-diagonal blocks.

3.2 Block-diagonal preconditioning

In a series of works (cf. [30, 44, 45, 55, 69, 96, 97]), based on some special norm techniques, another efficient preconditioner of block-diagonal form for matrices with square blocks has been derived and used. This preconditioner reads as follow for the general block systems of form $\begin{bmatrix} A & -B \\ B & A \end{bmatrix}$ and $\begin{bmatrix} A & B \\ B & -A \end{bmatrix}$,

$$\mathcal{P}_{bd1} = \begin{bmatrix} A + B & 0 \\ 0 & A + B \end{bmatrix}. \tag{15}$$

If A and the symmetric part of B are symmetric and positive semi-definite matrices, the intersection of their nullspaces is empty and $A + B$ is invertible. Furthermore, it has been shown that the spectrum of $\mathcal{P}_{bd1}^{-1}\mathcal{A}$ is contained in the intervals $[-1 - \frac{1}{2}] \cup [\frac{1}{2}, 1]$ (see [79]). As a consequence, the condition number of the preconditioned system satisfies $\kappa((\mathcal{P}_{bd1}^{-1}\mathcal{A})^2) \leq 2$ (cf. [10, 30, 96]). The above theoretical estimates show that this preconditioner is robust with respect to the discretisation and material parameters.

Applied to system Eq. 13, the preconditioning matrix is as follows

$$\mathbf{P}_{bd1} = \begin{bmatrix} \mathbf{M}_\sigma + (\mathbf{K} - \mathbf{M}_\varepsilon) & 0 \\ 0 & \mathbf{M}_\sigma + (\mathbf{K} - \mathbf{M}_\varepsilon) \end{bmatrix}. \tag{16}$$

For comparison purposes, in Section 4 we present numerical illustrations of the performance of \mathbf{P}_{bd1} for the target problem.

3.3 Schur complement-based preconditioning

In [63, 64] for systems as in equation (13), efficient preconditioners have been derived, based on a particular high quality approximation ($\hat{\mathbf{S}}$) of the arising exact (negative) Schur complement $\mathbf{S} = \mathbf{M}_\sigma + (\mathbf{M}_\sigma + (\mathbf{K} - \mathbf{M}_\varepsilon))\mathbf{M}_\sigma^{-1}(\mathbf{M}_\sigma + (\mathbf{K} - \mathbf{M}_\varepsilon))$, namely, $\hat{\mathbf{S}} = (\mathbf{M}_\sigma + (\mathbf{K} - \mathbf{M}_\varepsilon))\mathbf{M}_\sigma^{-1}(\mathbf{M}_\sigma + (\mathbf{K} - \mathbf{M}_\varepsilon))$. The corresponding spectral bound for the Schur complement is shown to be

$$\lambda(\hat{\mathbf{S}}^{-1}\mathbf{S}) \in \left[\frac{1}{2}, 1 \right].$$

Thus, the block-diagonal preconditioner reads as follows (cf. [64]),

$$\mathbf{P}_{bd2} = \begin{bmatrix} \mathbf{M}_\sigma & 0 \\ 0 & \hat{\mathbf{S}} \end{bmatrix}.$$

We see that the computation cost to solve a system with \mathbf{P}_{bd2} includes two solutions with $\mathbf{M}_\sigma + (\mathbf{K} - \mathbf{M}_\varepsilon)$ and one solution with \mathbf{M}_σ , which is diagonal in the spectral element approximation for orthogonal hexahedral elements.

Alternatively, one can use a block lower-triangular preconditioner with the same Schur complement approximation,

$$\mathbf{P}_{bt2} = \begin{bmatrix} \mathbf{M}_\sigma & 0 \\ \mathbf{K} - \mathbf{M}_\varepsilon & \hat{\mathbf{S}} \end{bmatrix}.$$

The cost to use \mathbf{P}_{bt2} , compared to that when using \mathbf{P}_{bd1} is higher and requires one more matrix-vector multiplication with $\mathbf{K} - \mathbf{M}_\varepsilon$.

As noted in [10], this preconditioner has been shown to require slightly larger computational time compared to the PRESB preconditioner, outlined in Section 3.4 and therefore has not been tested in this study.

3.4 The PRESB preconditioner

Next, we consider the following preconditioner, referred to as 'PRESB', which stands for 'PREconditioning for Square Blocks' (cf. e.g., [10, 14, 15]). PRESB is derived for two-by-two block matrices (with square blocks) of the general form

$$\mathcal{A} = \begin{bmatrix} A & -b B_2 \\ a B_1 & A \end{bmatrix},$$

with the assumption that A is symmetric positive definite and that a and b are non-zero scalars having the same sign. The theory covers also more general cases, for instance, when A is semi-definite, however these fall out of the scope of this paper. The preconditioner is of the form

$$\mathcal{P} = \begin{bmatrix} A & -b B_2 \\ a B_1 & A + \sqrt{ab}(B_1 + B_2) \end{bmatrix} \tag{17}$$

and it is shown (e.g., [10, 14, 15]) that all the eigenvalues of the preconditioned matrix $\mathcal{P}^{-1}\mathcal{A}$ are real and are contained in the interval $[\frac{1}{2}, 1]$, independently of the discretisation parameter h , the parameters a and b and any other problem parameters that may be included in the matrix blocks. For completeness, we include the derivation of this result in Theorem 3.1 for $a = b = 1$ and more general types of matrices.

Theorem 3.1 (See [8]) *Let the real $n \times n$ matrices A and B be symmetric and positive semi-definite, such that $\mathcal{N}(A) \cap \mathcal{N}(B) = \emptyset$ and $A + B$ is nonsingular, and where $\mathcal{N}(A)$ and $\mathcal{N}(B)$ denote the nullspaces of A and B . Then the eigenvalues of the generalised eigenvalue problem*

$$\begin{bmatrix} A & -B \\ B & A \end{bmatrix} \begin{bmatrix} v \\ w \end{bmatrix} = \lambda \begin{bmatrix} A & -B \\ B & A + 2B \end{bmatrix} \begin{bmatrix} v \\ w \end{bmatrix} \tag{18}$$

are all real and contained in the interval $[\frac{1}{2}, 1]$.

Proof Rewrite the problem (18) equivalently as

$$\begin{bmatrix} 0 & 0 \\ 0 & 2B \end{bmatrix} \begin{bmatrix} v \\ w \end{bmatrix} = (1 - \lambda) \begin{bmatrix} A & -B \\ B & A + 2B \end{bmatrix} \begin{bmatrix} v \\ w \end{bmatrix} \tag{19}$$

We see that $\lambda = 1$ for any v and $w \in \mathcal{N}(B)$. Let now $w \notin \mathcal{N}(B)$, thus, $\lambda \neq 1$. Denoting $\alpha = \frac{1}{1-\lambda}$, we rewrite problem (19) once again as

$$\begin{bmatrix} 0 & 0 \\ 0 & 2\alpha B \end{bmatrix} \begin{bmatrix} v \\ w \end{bmatrix} = \begin{bmatrix} A & -B \\ B & A + 2B \end{bmatrix} \begin{bmatrix} v \\ w \end{bmatrix}. \tag{20}$$

Relation (20) is equivalent to

$$Av = Bw, \quad Bv + (A + 2B)w = 2\alpha Bw. \tag{21}$$

From equation (21)(left) we obtain that $A(v + w) = (A + B)w$ and from equation (21)(right) $(A + B)(v + w) = 2\alpha Bw$. Thus,

$$v + w = 2\alpha(A + B)^{-1}Bw = 2\alpha(I - (A + B)^{-1}A)w.$$

Further, $A(v + w) = (A + B)w = 2\alpha A(I - (A + B)^{-1}A)w$ and finally

$$w = 2\alpha(A + B)^{-1}A(I - (A + B)^{-1}A)w.$$

Denote $G = (A + B)^{-1}A$ and let its eigenvalues be μ . Since A and B are symmetric positive semi-definite and $A + B$ is nonsingular, then all μ are real, nonnegative and less than 1. To estimate λ we use the following observation (with I being the identity matrix of appropriate dimension),

$$\begin{aligned} 2G(I - G) &= 2 \left(\frac{1}{2}I - \left(\frac{1}{2}I - G \right) \right) \left(\frac{1}{2}I + \left(\frac{1}{2}I - G \right) \right) \\ &= \left(\frac{1}{2}I - 2 \left(\frac{1}{2}I - G \right)^2 \right). \end{aligned}$$

From the latter expression we conclude that $\lambda = \frac{1}{2}(1 + (1 - 2\mu)^2)$. Using the bounds for μ , we obtain the following parameter-independent bounds for λ ,

$$\frac{1}{2} \leq \frac{1}{2}(1 + (1 - 2\mu)^2) = \lambda \leq 1. \tag{22}$$

It can easily be verified that inserting 0 and 1 for μ yields the upper bound for λ , whereas setting $\mu = \frac{1}{2}$ gives the lower bound for λ . \square

One additional advantage of the PRESB preconditioner is that it possesses the following block factorisation,

$$\begin{bmatrix} A & -B^T \\ B & A + B + B^T \end{bmatrix} = \begin{bmatrix} I & -I \\ 0 & I \end{bmatrix} \begin{bmatrix} A + B & 0 \\ 0 & I \end{bmatrix} \begin{bmatrix} I & 0 \\ B & I \end{bmatrix} \begin{bmatrix} I & 0 \\ 0 & A + B^T \end{bmatrix} \begin{bmatrix} I & I \\ 0 & I \end{bmatrix} \tag{23}$$

For the problem, considered in this paper, we have that $B_2 = B_1^T$ and $a = b = 1$. The preconditioner to \mathbf{A} reads then

$$\mathbf{P} = \begin{bmatrix} \mathbf{M}_\sigma & -(\mathbf{K} - \mathbf{M}_\varepsilon) \\ \mathbf{K} - \mathbf{M}_\varepsilon & \mathbf{M}_\sigma + 2(\mathbf{K} - \mathbf{M}_\varepsilon) \end{bmatrix}. \tag{24}$$

From equation (23), we directly see the computational cost of applying the PRESB preconditioner, namely, solutions of linear systems with $A + B$ and $A + B^T$, one multiplication with the matrix B and three vector additions. For the matrix \mathbf{P} the off-diagonal blocks are symmetric, and to solve a system of the form

$$\mathbf{P} \begin{bmatrix} \mathbf{w}_1 \\ \mathbf{w}_2 \end{bmatrix} = \begin{bmatrix} \mathbf{f}_1 \\ \mathbf{f}_2 \end{bmatrix}, \tag{25}$$

we solve twice with $\mathbf{H} = \mathbf{M}_\sigma + (\mathbf{K} - \mathbf{M}_\varepsilon)$, multiply with \mathbf{M}_σ and form three vector updates [10–12]. The computational procedure is shown in Algorithm 1.

The matrix \mathbf{H} is of the size of the original complex matrix \mathbf{C}_A and can be also large. Therefore, in Steps 1 and 3 in Algorithm 1 we should also use an (inner) preconditioned iterative method. Thus, the overall solution procedure becomes of the so-called 'inner-outer' solution type. Due to the usage of an inner iterative solver the

-
- 1 Solve $\mathbf{H}\mathbf{g} = \mathbf{f}_1 + \mathbf{f}_2$
 - 2 Compute $\mathbf{M}_\sigma \mathbf{g}$ and $\mathbf{f}_1 - \mathbf{M}_\sigma \mathbf{g}$
 - 3 Solve $\mathbf{H}\mathbf{h} = \mathbf{f}_1 - \mathbf{M}_\sigma \mathbf{g}$
 - 4 Compute $\mathbf{w}_1 = \mathbf{g} + \mathbf{h}$ and $\mathbf{w}_2 = -\mathbf{h}$
-

Algorithm 1 Solving linear system with preconditioner \mathbf{P} .

PRESB preconditioner is of variable type and this imposes the additional requirement that the outer iterative Krylov subspace solver should be capable of handling variable preconditioning. Iterative methods that can be used as outer methods with variable inner preconditioning are the Flexible GMRES (FGMRES, [75]), the Generalized Conjugate Residual (GCR) method [34] and the Generalized Conjugate Gradient method [16]. In this study we use CGR as an outer solver and either the preconditioned GCR or FGMRES methods for the inner solver with a suitable Algebraic Multigrid (AMG) preconditioning.

For clarity, the implementation of the iterative framework is summarised in Algorithm 2. In particular, the outer iterative solution procedure is given and solving the two inner systems in Algorithm 1 using an (inner) preconditioned iterative method refers to the inner solution method. Note that the outer loop is restarted every $m + 1$ iterations and that the current iterate \mathbf{x}_{m+1} is taken as the new starting guess \mathbf{x}_0 . It follows that at most m direction vectors \mathbf{p} and m matrix-vector products with \mathbf{p} need to be stored.

As can be seen, each outer iteration step involves solving a linear system with preconditioner \mathbf{P} as described in Algorithm 1. Note that the right-hand side vector \mathbf{f} used to solve system Eq. 25 corresponds to the normalised residual vector \mathbf{r} of the current iteration of the outer loop.

Input: $\mathbf{A}, \mathbf{b}, \mathbf{H}, \mathbf{M}_\sigma$, initial guess \mathbf{x}_0 , tol
Output: \mathbf{E}_h

- 1 Set $\mathbf{r}_0 = \mathbf{b} - \mathbf{A}\mathbf{x}_0$
- 2 **for** $i = 0, \dots, m$ **do**
- 3 Solve $\mathbf{P}\mathbf{p}_i = \mathbf{r}_i$ using Algorithm 1
- 4 $\mathbf{q}_i = \mathbf{A}\mathbf{p}_i$
- 5 $\mathbf{q}_i = \mathbf{q}_i - \sum_{j=0}^{i-1} \mathbf{q}_j \frac{(\mathbf{q}_i, \mathbf{q}_j)}{(\mathbf{q}_j, \mathbf{q}_j)}$
- 6 $\mathbf{p}_i = \mathbf{p}_i - \sum_{j=0}^{i-1} \mathbf{p}_j \frac{(\mathbf{q}_i, \mathbf{q}_j)}{(\mathbf{q}_j, \mathbf{q}_j)}$
- 7 $\alpha_i = \frac{(\mathbf{r}_i, \mathbf{q}_i)}{(\mathbf{q}_i, \mathbf{q}_i)}$
- 8 $\mathbf{x}_{i+1} = \mathbf{x}_i + \alpha_i \mathbf{p}_i$
- 9 $\mathbf{r}_{i+1} = \mathbf{r}_i - \alpha_i \mathbf{q}_i$
- 10 **if** $\frac{\|\mathbf{r}_{i+1}\|_2}{\|\mathbf{r}_0\|_2} < \text{tol}$ **then** $\mathbf{E}_h = \mathbf{x}_{i+1}$ & Stop
- 11 **end**

Algorithm 2 PRESB-preconditioned CGR method.

3.4.1 Auxiliary-space Maxwell preconditioner

From Algorithm 1 and the form of the matrix blocks, it is seen that we have to solve systems with $\mathbf{H} = \mathbf{M}_\sigma + (\mathbf{K} - \mathbf{M}_\epsilon)$. The matrix \mathbf{H} is of type of a discrete form of the following variational problem arising from Maxwell’s equations

$$\text{Find } \mathbf{u} \in \mathcal{H}_{0,h} \quad (c \nabla \times \mathbf{u}, \nabla \times \mathbf{v}) + (d \mathbf{u}, \mathbf{v}) = (\mathbf{f}, \mathbf{v})$$

for all $\mathbf{v} \in \mathcal{H}_{0,h}$ (26)

where c and d are scalars describing the magnetic and electric properties of the media, and where $\mathcal{H}_{0,h}$ is a subspace to \mathcal{H}_0 (see Section 2.2 for the definition).

We choose to solve linear systems of equations involving \mathbf{H} by another (inner) preconditioned iterative solver. As a preconditioner we use a suitable AMG method. Efficient multigrid-based solvers for this problem are already available, described in detail in [48] and [53, 54], and implemented in the library hypre [37, 38] as the auxiliary-space Maxwell solver AMS [45, 54]. The basic premise of this approach is based on the Hitpmair-Xu (HX) decomposition [48]. Without going into details, the underlying idea of their method lies in the understanding that the space $\mathcal{H}_0(\text{curl}, \Omega)$ possesses a stable decomposition (see Section 3 in [48] for a detailed discussion on decomposition of spaces) of form

$$\mathcal{H}_0(\text{curl}, \Omega) = \left(H_0^1(\Omega) \right)^3 + \text{grad} H_0^1(\Omega) \tag{27}$$

which implies that solving $H_0^1(\Omega)$ -elliptic variational problems can be the basis for a preconditioner for $\mathcal{H}_0(\text{curl}, \Omega)$ -elliptic problems [48]. In essence, the auxiliary-space preconditioning technique comes down to approximately inverting the discrete curl-curl operator by transferring the problem into subspaces. Solving the problem in these (nodal) auxiliary spaces is much simpler than in the initial space as one can exploit the power of the AMG methods. As the auxiliary-space preconditioner can be viewed as a black box, we refrain from going into details and refer to [44, 48, 53, 54] and [45] for details on the underlying theory and its implementation.

For completeness, we briefly specify the required user inputs for the auxiliary-space preconditioner. Besides the system matrix and right-hand side vector described by Eq. 26, AMS requires the commonly called discrete gradient matrix which describes the edges of the mesh in terms of its vertices [53, 54]. It follows that the number of rows of this matrix corresponds to the number of degrees of freedom (edges) and the number of columns to that of the number of vertices in the mesh. In case of first order curl conforming basis functions, each row possesses two nonzero entries in the corresponding columns of the vertices composing the edge. In particular, the entries will be $+1$ and -1 , where the

sign is based on the orientation of the edge. In addition, the auxiliary-space preconditioner needs the coordinates of the vertices in the mesh to construct the solver. Alternatively, the user may also choose to input the representations of the constant vector fields in the edge element basis [53, 54].

Note that the implementation of the auxiliary-space preconditioner AMS supports arbitrary order Nedelec elements as well as non-conforming quadrilateral/hexahedral meshes (i.e. meshes with hanging nodes).

We mention that the AMS implementation in `hypre` [37, 38] holds for the positive semi-definite case [54] while convergence in the indefinite case might be slow or not reached at all. This could potentially pose some difficulties as, depending on the frequency as well as the air space conductivity considered, the system we solve might change type and become indefinite. For example, if the conductivity of air is set to 10^{-8} S/m and the corresponding dielectric permittivity of air is $8.854 \cdot 10^{-12}$ F/m, then the system becomes indefinite in the air space of the domain at frequencies > 180 Hz due to the negative shift by \mathbf{M}_ϵ and the larger kernel of \mathbf{K} [25]. Thus, for large frequencies the system matrix becomes highly indefinite.

In the numerical experiments, we use AMS as implemented in `hypre` [53, 54] as the preconditioner for the iterative method applied to solve the inner systems (Steps 1 and 3 in Algorithm 1).

4 Numerical experiments

We illustrate the performance and the robustness of the solution procedure and the PRESB preconditioner for the EM equations on the following two test problems.

Problem 1 *The first problem we consider is a simple 1D model of a layered Earth as depicted in Fig. 1(a) and consists of a conductive layer of 0.01 S/m extending from a depth of 500 to 1000 m embedded in a 10^{-4} S/m half-space. The air conductivity is set to 10^{-8} S/m. The source is a 200 m long grounded cable directed along the x-axis stretching from $(-100, 0, 0)$ to $(100, 0, 0)$ m and is thus centred around the origin which has coordinates $(0, 0, 0)$ m. The source moment is 100 Am. The source wire itself is divided into four segments of 50 m length each. The computational domain is discretised using nonuniform stretched rectangular elements. The smallest cell sizes are located around the source and steadily grow in size towards the domain boundaries. The relevant information of this model setup is summarised in Table 1.*

Problem 2 *This problem is a 3D model that includes a conductive feature representative of, for example, an ore body buried within a resistive background and is covered*

by a thin conductive layer as shown in Fig. 1(b). The conductive body which is inclined and of dimensions $1000 \times 662 \times 3000$ m³ has an electric conductivity of 1 S/m. The surrounding host rock and the top layer have conductivities of 10^{-4} and 0.01 S/m, respectively. The body is located at 2000 m depth and extends downwards to 5000 m depth. The top and the bottom of the structure stretch from -3000 to -4000 m and -2000 to -3000 m in x-direction respectively. In y-direction, the body extends from -331 to 331 m. The source is a grounded cable with its endpoints at coordinates $(-75, 0, 0)$ and $(59, 0, 0)$ m. The source moment is 100 Am. The computational domain is meshed with unstructured hexahedral elements (see Fig. 2) using the open-source meshing software `gmsh` [41]. Other relevant information pertaining to this setup is given in Table 1.

The numerical procedures described in Sections 2 and 3 are implemented in a parallel fashion using the Message Passing Interface (MPI) and employ the open-source libraries `PETSc` [20, 21] and `hypre` [37, 38].

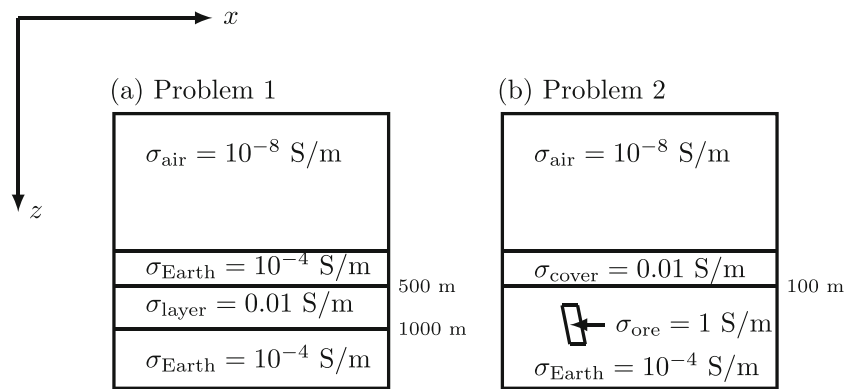
All experiments are performed on AMD Ryzen Threadripper 2950X 16-core processor with a clock frequency of 3.5 GHz and with 128 GB RAM. The operating system is Ubuntu 20.04. All numerical simulations are run with two MPI processes in order to show that the developed iterative framework and all libraries used within it can run in parallel fashion.

As described in Section 3, when the system is solved iteratively, the outer iteration method for the two-by-two block system is the GCR method. The inner two systems arising from applying the preconditioner are solved using either GCR or FGMRES provided by `PETSc`. In both cases, AMS is used as a preconditioner. For comparison we provide experiments, where the inner systems are solved by the sparse direct solver MUMPS, also available through `PETSc`.

We remark that we also tested `hypre`'s ILU called *pilu* as a potential preconditioner for the inner systems. However, we do not include any results of these simulations as all of them failed to reach convergence for Problems 1 and 2. We thus note that ILU decomposition does not constitute a suitable inner preconditioner for the system to be solved in the proposed iterative framework.

For all presented tests, the electric conductivity of the air is set to 10^{-8} S/m. The magnetic permeability and dielectric permittivity are assumed to be the corresponding constants of free space for all but a few indicated exceptions. Unless stated otherwise, by default the simulations are run using the smallest problem size (see Table 1) and are stopped once the relative residual of the outer solver GCR falls below the threshold of 10^{-12} . Regarding the stopping tolerance for the inner GCR solver preconditioned with AMS, experiments show that it can be chosen to be

Fig. 1 Schematic profile of Problems 1 and 2. None of the drafts is true to scale. The illustrating sketches include the conductivities of the different geophysical units



much larger without affecting the iteration count of the outer solver significantly. Table 2 indicates that the stopping tolerance for the inner solver can be set to 10^{-3} while still allowing for a good compromise between the number of outer and inner iterations to keep the solver robust and the computational time down. Further, we note that the given numbers of degrees of freedom (DOF) correspond to the real-valued system, which is twice that of the complex-valued system.

The focus of this work is on the convergence behaviour of the described iterative framework. Nonetheless in order to demonstrate that the iterative algorithm has the required modelling accuracy compared to other established methods, a numerical comparison of the electromagnetic responses to finite element solutions for the 3D model (Problem 2) is shown in Fig. 5 in Appendix A. Details are disclosed in Appendix A.

The conductivity distributions of Problems 1 and 2 shown in Fig. 1 involve several difficulties that might negatively affect the convergence of the used iterative methods, such as large conductivity contrasts, in particular at the air-Earth interface and to a lesser extent within the Earth at interfaces between different entities, non-uniformly stretched cells, unstructured elements and locally refined meshes. Hence, on the basis of these problems, we ascertain the robustness of the solvers.

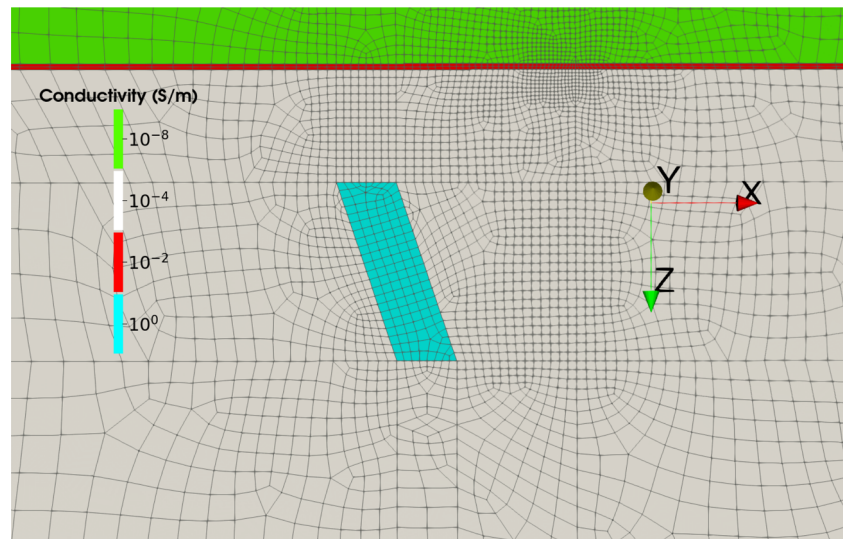
As predicted by the convergence properties of the PRESB preconditioner, we observe that across the tested frequency range of 0.1 to 10'000 Hz the iteration counts of the outer solver remain virtually stable (see Tables 3 and 4 as well

as Fig. 3). It is worth noting that solving the two inner systems (Steps 1 and 3 in Algorithm 1) iteratively leads to only a slight increase of the outer iteration count compared to the corresponding count where the inner systems are solved by a direct method and thus exactly (compare outer iteration counts in Tables 3 and 4). Another observation is that solving the two inner systems iteratively requires considerably less memory than when employing a direct solver for the same two inner systems. It is also evident that using preconditioned iterative solution techniques for solving the two inner systems instead of a direct solver reduces the computational time spent to reach convergence with the outer iterative algorithm. This gain in time only manifests if the preconditioned iterative algorithm applied to the two inner systems is efficient which in fact means the applied solution technique needs to converge to the desired tolerance in few iterations. In Tables 3 and 4, we observe that the inner preconditioned iterative method performs effectively for all but the highest (10'000 Hz) tested frequencies. In this case, the solution time increases drastically (approximately by a factor of five compared to the next lower frequency of 8000 Hz) while the outer iteration count remains unaffected which points to a break down of the efficiency of the inner preconditioned iterative solver. This is indeed the case as visualised in Fig. 4, which shows the convergence of the inner preconditioned iterative solver for the two inner systems at two stages where the residual of the outer algorithm dips below 10^{-4} and 10^{-8} , respectively. It can be seen that for a frequency of 10'000 Hz the inner solver needs significantly more iterations to reach

Table 1 Model information for Problems 1 and 2

Model	Problem 1 - Layered Earth	Problem 2 - 3D model
Domain size [km ³]	30 × 30 × 30	30 × 36 × 30
Conductivities [S/m]	$\sigma_{\text{air}} = 10^{-8}$, $\sigma_{\text{Earth}} = 10^{-4}$, $\sigma_{\text{layer}} = 10^{-2}$	$\sigma_{\text{air}} = 10^{-8}$, $\sigma_{\text{Earth}} = 10^{-4}$, $\sigma_{\text{cover}} = 0.01$, $\sigma_{\text{ore}} = 1$
Approximation order	1st	1st
# elements	54 × 54 × 54	332'580
# degrees of freedom	980'100	2'033'986

Fig. 2 Problem 2: a selected extract of the xz -cross section of the discretisation mesh along plane $y = 0$ m



convergence than for the other frequencies, which explains the observed increase in overall solution time. Thus, to achieve good overall performance we need a good solver for the inner systems. As a side note, clearly, the deterioration of the performance of the inner iterative solver can be detected and, if memory consumption allows to do so, one might switch to solving the inner systems directly.

As alluded to in Section 3.4.1, the inner system \mathbf{H} becomes indefinite for frequencies larger than 180 Hz and, as already noted, this is due to the negative shift from \mathbf{M}_e and the large nullspace of the stiffness matrix \mathbf{K} . This affects negatively the performance of the preconditioner for the inner solver as AMS is not devised for indefinite systems (cf. [54]). This effect can clearly be observed for the highest frequency of 10'000 Hz in Fig. 4.

In this study, we use AMS-preconditioned GCR as an inner solver and we illustrate that the AMS preconditioner is not efficient enough as soon as the inner systems become

strongly indefinite. However, we see that, as theory predicts, the outer convergence rate does not get affected because of that. Thus, a viable option for higher frequencies is to use another inner solver, that is more suitable for such inner systems.

The experiments convincingly show that the convergence of the outer solver is independent of the problem size—the number of iterations of the outer solver remains nearly constant and the solution time increases linearly with increasing number of degrees of freedom, see Table 5. For additional insight in the performance of the PRESB preconditioner, in Table 6 we compare the solution times and peak memory requirements, when the inner systems are solved both iteratively and by a direct method and those when solving the original complex-valued system in Eq. 10. As expected, the iterative solver has significantly lower memory demands and is faster than when direct methods are used. For simulations in which either the complex system or

Table 2 Problem 1: Influence on the outer iteration count N_{it}^{outer} , average inner iterations \bar{N}_{it}^{inner} and solution time when relaxing the stopping tolerance for the inner solver

		inner stopping tolerance				
		10^{-6}	10^{-5}	10^{-4}	10^{-3}	10^{-2}
f=1 Hz	N_{it}^{outer}	10	10	10	10	12
	\bar{N}_{it}^{inner}	22	18	15	11	8
	time [s]	106.0	88.7	73.4	56.6	47.0
f=100 Hz	N_{it}^{outer}	19	19	20	22	25
	\bar{N}_{it}^{inner}	20	16	12	9	6
	time [s]	186.0	152.5	119.0	93.5	71.5
f=10'000 Hz	N_{it}^{outer}	18	17	18	18	19
	\bar{N}_{it}^{inner}	124	101	74	55	34
	time [s]	1064.5	821.7	636.6	470.1	309.7

Table 3 Problem 1: Comparison of outer iteration count (N_{it}^{outer}), solution time (time [s]) and memory requirements (peak memory usage for simulation; mem[GB]) when using iterative (GCR and FGMRES) and direct (DMUMPS) methods as inner solvers

Inner solver	GCR			FGMRES			DMUMPS		
	N_{it}^{outer}	time [s]	mem [GB]	N_{it}^{outer}	time [s]	mem [GB]	N_{it}^{outer}	time [s]	mem [GB]
0.1	7	40.0	4.3	7	39.3	4.3	6	121.8	14.1
1	10	56.7	4.4	10	55.8	4.3	9	147.3	14.1
10	16	79.3	4.4	16	78.3	4.4	15	145.3	14.0
100	22	93.5	4.5	22	92.7	4.3	18	157.2	14.1
1000	19	70.2	4.4	19	69.3	4.3	17	166.7	14.0
5000	18	62.8	4.4	18	62.1	4.3	17	153.0	14.0
8000	18	96.8	4.4	18	95.6	4.4	18	157.8	14.2
10`000	18	460.7	4.4	18	443.4	4.4	18	168.7	14.1

the inner systems are solved directly one can easily spot the increase in memory consumption as well as in solution time. The execution time given in the tables include the overall time for the iterative solver to converge, including the time needed to construct the AMS preconditioner. Whenever direct methods are used, the analysis and factorisation times are included. For complex systems we report the time for analysis, factorisation and solution time. Memory requirements in particular may become the limiting factor for large problems. This is observed for the largest problem size where the direct solvers demand approximately 151 and 198 GB of memory respectively, which exceeds the available memory of the computer platform used to perform these tests. In contrast, using an iterative method for the inner systems uses only around 30 GB of memory.

In order to test the robustness of the solver with respect to magnetic permeability μ , we assign different values of relative magnetic permeability μ_r to the ore body in Problem 2. In particular, μ_r is set to 2, 5 and 10. Per definition, the relative magnetic permeability is defined as $\mu_r = \frac{\mu}{\mu_0}$. We point out that the magnetic permeability for different rock types

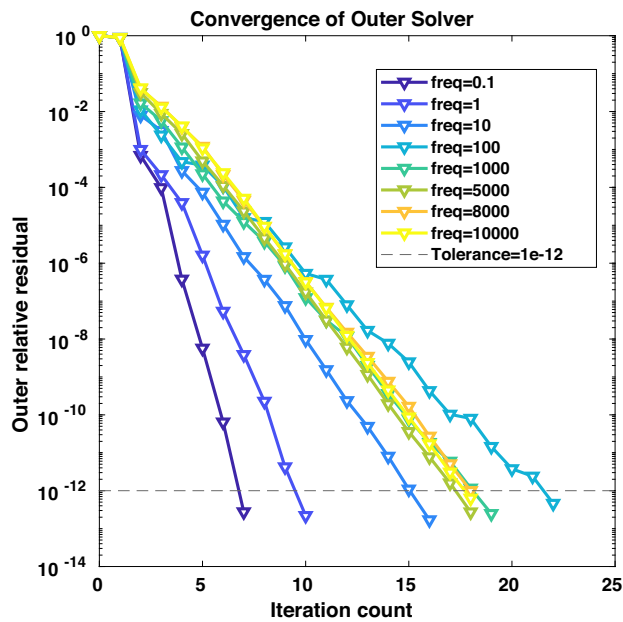
falls within a very narrow range between $1\mu_0$ and $3\mu_0$. Further, a relative magnetic permeability value of 10 can safely be considered to cover the vast majority of possibilities encountered in geo-EM applications. The test results are presented in Table 7 and show that the influence of spatially variable magnetic permeability distributions is minimal thus confirming that the solver is robust also with respect to material parameter μ .

We next consider the robustness of the solver with regard to variable dielectric permittivity ϵ . To do so, we assume that the surrounding host rock around the buried body and the cover layer on top of the host rock have relative dielectric permittivities values of 5 and 20, respectively. In a geophysical sense, this could be representative of a weakly fracture granite covered by some kind of glacial deposits. Further, the relative magnetic permeability of the ore body is taken to be either 1 or 10. The results for this set of parameter value combinations are given in Table 8, and, in comparison to Table 7, these imply that the solver remains robust with regard to variable dielectric permittivity distributions.

Table 4 Problem 2: Comparison of outer iteration count (N_{it}^{outer}), solution time (time [s]) and memory requirements (peak memory usage for simulation; mem[GB]) when using iterative (GCR and FGMRES) and direct (DMUMPS) methods as inner solvers

Inner solver	GCR			FGMRES			DMUMPS		
	N_{it}^{outer}	time [s]	mem [GB]	N_{it}^{outer}	time [s]	mem [GB]	N_{it}^{outer}	time [s]	mem [GB]
0.1	9	594.5	10.6	9	586.3	10.6	7	1118.7	70.6
1	12	385.3	10.6	12	380.0	10.6	12	1158.9	70.8
10	16	284.8	10.6	16	286.4	10.6	15	1182.4	70.8
100	23	278.6	10.6	23	278.3	10.6	20	1225.0	70.9
1000	20	214.1	10.6	20	210.5	10.6	18	1211.1	70.8
5000	18	174.3	10.6	18	172.5	10.6	17	1196.5	70.8
8000	18	324.3	10.6	18	322.0	10.6	17	1197.5	70.8
10`000	19	1775.1	10.6	18	1725.7	10.6	18	1207.0	70.8

(a) Inner solver: GCR preconditioned with AMS



(b) Inner solver: DMUMPS

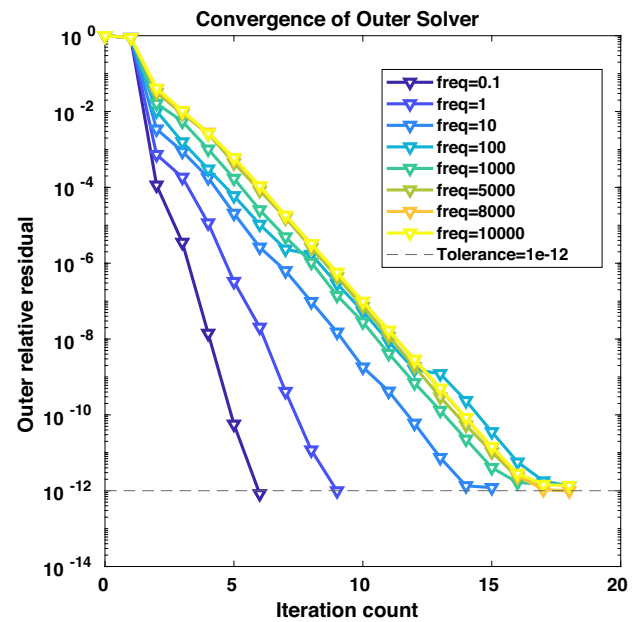


Fig. 3 Problem 1: Convergence curves for the outer solver for a range of frequencies. The two inner systems are solved iteratively by AMS-preconditioned GCR or by the direct solver MUMPS

Taken all together, our examples demonstrate that the solver is robust with respect to all material properties σ , μ , and ε relevant in geophysical settings and frequency ω as well as with respect to the discretisation parameter. Moreover, the outer solver converges within relatively few iterations. The whole iterative framework is computationally inexpensive in terms of memory requirement when compared to direct solvers. On top of that, the procedure also saves a considerable amount of computational time given that the inner systems can be solved efficiently (i.e. in a few iterations).

4.1 Convergence behaviour for loop sources

In this section, we test the convergence behaviour of our algorithm for Problem 1 when using a horizontal or vertical loop source corresponding to vertical or horizontal magnetic dipole, respectively, instead of line sources. The loop sources are square loops with a side length of 10 m, and the centres of the horizontal and vertical loops are located at coordinates $(0, 0, 0)$ and $(0, 0, 5)$, respectively. Each side is subdivided into four wire segments. We remark that the simulations for the loop sources are performed using the intermediate problem size with $3'641'400$ degrees of freedom.

Tables 9 and 10 show the relevant convergence parameters, namely the outer iteration count and the time taken to reach convergence when solving all the systems in the

algorithm iteratively. For comparison, we also run the simulations where the inner systems are solved using a direct solver. As already pointed out earlier, solving the inner systems by iterative methods is more cost-effective, both in terms of time as well as memory required to carry out the simulation. However, we observe that the more complicated nature of the loop source that enters the problem through the right-hand side increases the overall simulation times when compared to those of a line source (see Table 5). This is due to an increase in iterations required to reach convergence for the inner systems as demonstrated by the average number of inner iterations per system given in Tables 9 and 10. We recall that the increase in time can be compensated for by relaxing the inner stopping criteria which results in fewer inner iterations at the expense of more outer iterations (see Table 2). In the case of loop sources, this might become a viable option to keep the simulation times low. Even though the time gain of solving the inner systems by a preconditioned iterative method is not as large as for the same simulations with line sources, the overall memory consumption still presents a major advantage over using direct solvers as the method of choice to solve the inner systems.

4.2 Comparisons of the performance of \mathbf{P}_{bd_1} and \mathbf{P}

We next present a comparison of the PRESB preconditioner \mathbf{P} with the block diagonal preconditioner \mathbf{P}_{bd_1} , used in related works (e.g. [30, 44, 45, 69]). For the sake of clarity,

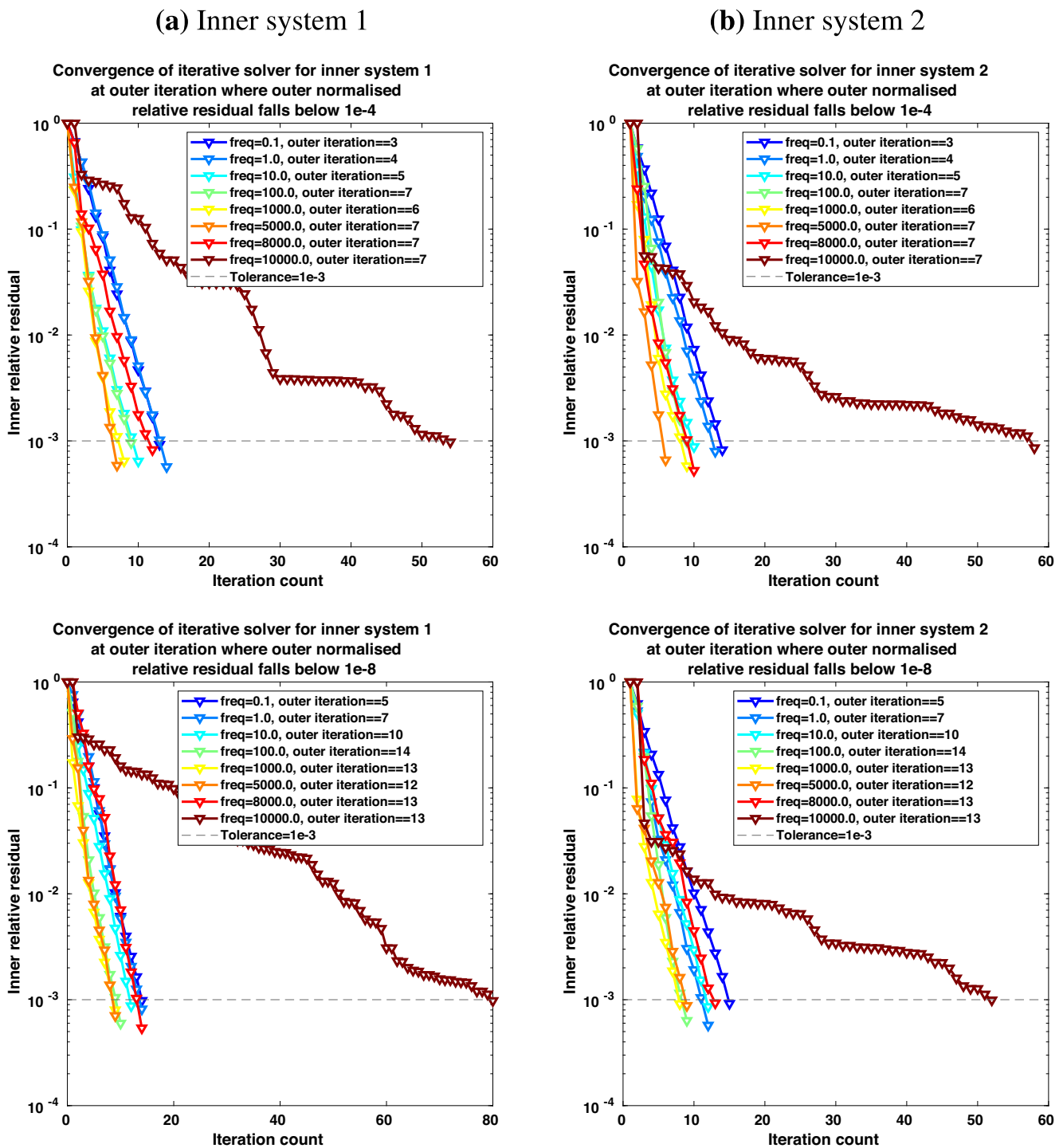


Fig. 4 Problem 1: Convergence curves of AMS-preconditioned GCR for inner systems in Algorithm 1 across a broad range of frequencies for outer iteration where stopping criteria of 10^{-4} and 10^{-8} in outer normalised relative residual are reached

we briefly outline the general setup of the solver. Recall the two-by-two system to be solved

$$\begin{bmatrix} \mathbf{M}_\sigma & -(\mathbf{K} - \mathbf{M}_\varepsilon) \\ \mathbf{K} - \mathbf{M}_\varepsilon & \mathbf{M}_\sigma \end{bmatrix} \begin{bmatrix} \mathbf{E}_R \\ -\mathbf{E}_I \end{bmatrix} = \begin{bmatrix} \mathbf{f}_I \\ \mathbf{f}_R \end{bmatrix}, \tag{28}$$

preconditioned with a block diagonal matrix of the form

$$\mathbf{P}_{bd1} = \begin{bmatrix} \mathbf{M}_\sigma + (\mathbf{K} - \mathbf{M}_\varepsilon) & 0 \\ 0 & \mathbf{M}_\sigma + (\mathbf{K} - \mathbf{M}_\varepsilon) \end{bmatrix}. \tag{29}$$

We note that none of the aforementioned works (namely [30, 44, 45, 69]) include displacement currents, that is \mathbf{M}_ε is present neither in their block systems nor in the used

Table 5 Problem 1: Comparison of outer iteration counts (N_{it}^{outer}) and solving times (time [s]) for various frequencies and problem sizes

#DOF	frequency [Hz]							
	0.1		10		1000		8000	
	N_{it}^{outer}	time[s]	N_{it}^{outer}	time[s]	N_{it}^{outer}	time[s]	N_{it}^{outer}	time[s]
980*100	7	42.2	16	79.3	19	70.2	18	96.8
3*641*400	8	152.9	15	286.4	18	272.6	19	310.2
6*879*600	8	343.4	16	646.5	18	521.8	18	790.5

Table 6 Problem 1: Comparison of solving times (time [s]) and peak memory consumption (mem[GB]) for a frequency of 100 Hz and various problem sizes between iterative outer solves employing either an iterative (preconditioned GCR) or a direct (DMUMPS) inner solver and direct solves of the original complex-valued system in Eq. 10

#DOF	Iterative Method: GCR				Direct Solver: ZMUMPS	
	Preconditioned GCR		DMUMPS		-	
	time[s]	mem[GB]	time[s]	mem[GB]	time[s]	mem[GB]
980*100	93.5	4.4	157.2	14.1	166.8	9.0
3*641*400	368.4	16.4	1625.0	74.8	1874.1	55.5
6*879*600	661.0	30.1	-	out of memory	-	out of memory

Table 7 Problem 2: Comparison of outer iteration counts (N_{it}^{outer}) and solution times (time [s]) for various frequencies and different magnetic permeability of the ore body

frequency [Hz]	relative magnetic permeability μ_r of ore body							
	1		2		5		10	
	N_{it}^{outer}	time[s]	N_{it}^{outer}	time[s]	N_{it}^{outer}	time[s]	N_{it}^{outer}	time[s]
0.1	9	594.5	9	586.0	10	656.7	11	691.8
10	16	284.8	16	282.3	16	180.1	16	279.2
100	23	278.6	23	275.4	23	273.7	24	284.1
8000	18	324.3	18	345.0	18	344.7	18	331.4

Table 8 Problem 2: Comparison of iteration counts (N_{it}^{outer}) and solution times (time [s]) for various frequencies; simulations run with a variable dielectric permittivity distribution and two different magnetic permeabilities for the ore body

relative dielectric permittivity of air, cover, host rock and ore body	$\epsilon_r^{air} = 1, \epsilon_r^{cover} = 20, \epsilon_r^{Earth} = 5, \epsilon_r^{ore\ body} = 1$							
relative magnetic permeability of ore body	$\mu_r = 1$				$\mu_r = 10$			
frequency [Hz]	N_{it}^{outer}	time[s]	N_{it}^{outer}	time[s]	N_{it}^{outer}	time[s]	N_{it}^{outer}	time[s]
0.1	9	590.4	11	691.6				
10	16	282.3	16	279.8				
100	23	278.5	24	285.1				
8000	18	341.0	18	322.4				

Table 9 Problem 1: Comparison of iteration counts (N_{it}^{outer}), average inner iteration count per system (\bar{N}_{it}^{inner}), solution times (time [s]) and peak memory usage (mem [GB]) for various frequencies; simulations

Inner solver	GCR				DMUMPS		
	N_{it}^{outer}	\bar{N}_{it}^{inner}	time [s]	mem [GB]	N_{it}^{outer}	time [s]	mem [GB]
freq [Hz]							
0.1	9	24	351.9	18.3	4	1435.8	75.5
10	13	26	532.4	18.3	10	1510.0	75.7
1000	15	21	499.1	18.3	12	1533.8	75.8
8000	16	22	539.5	18.3	13	1547.9	75.8

are run using the intermediate problem size with 3'641'400 degrees of freedom and using a horizontal loop source corresponding to a vertical magnetic dipole source

block preconditioners. In addition, the outer solver used in these publications is either a GMRES or FGMRES, and the two inner systems are solved using a AMS-preconditioned CG. Hence, the choices of the outer and inner solvers differ, however GCR and GMRES are shown to be mathematically equivalent, (cf. e.g. [50]). Thus, we can make a fair comparison by testing the performance of the block diagonal preconditioner for our problems. Table 11 shows the performance of both block preconditioners in terms of outer iteration count (N_{it}^{outer}), time spent inside the algorithm (time [s]) and peak memory consumption (mem[GB]) for each simulation across a frequency range spanning five orders of magnitude. It is easily discernible that both preconditioners exhibit similar behaviour: (1) an increase of outer iterations for higher frequencies; (2) highest iteration count for the frequency of 100 Hz; (3) a steep rise in time spent to reach the solution for frequency 10'000.

The comparison clearly shows that the PRESB preconditioner is more efficient than the block-diagonal preconditioner, even though, compared to the application cost of the block-diagonal preconditioner, applying it requires one additional matrix-vector multiplication and four vector update operations. Note, in particular, that the outer

iteration counts when using the PRESB preconditioner are equal to or lower than when applying the block-diagonal one. The same behaviour also holds true (except for frequency 100 Hz) when reducing the outer stopping tolerance to 10^{-8} for simulations in which the block-diagonal preconditioner is used. As the outer iteration count is smaller for runs using PRESB, the time spent to achieve convergence is lower.

All in all, given the presented evidence, the PRESB preconditioner is to be preferred. As the implementation of both preconditioners is fairly similar, changing to the improved preconditioner is beneficial and thus encouraged.

5 Conclusions and outlook

In this work, we present an efficient and robust iterative framework for solving the linear algebraic system arising from the spectral element discretisation of the curl-curl equation of the total electric field describing the behaviour of the electromagnetic field for 3D CSEM problems in frequency domain. We transform the original complex-valued system into an equivalent real-valued form. Then,

Table 10 Problem 1: Comparison of iteration counts (N_{it}^{outer}), average inner iteration count per system (\bar{N}_{it}^{inner}), solution times (time [s]) and peak memory usage (mem [GB]) for various frequencies; simulations

Inner solver	GCR				DMUMPS		
	N_{it}^{outer}	\bar{N}_{it}^{inner}	time [s]	mem [GB]	N_{it}^{outer}	time [s]	mem [GB]
freq [Hz]							
0.1	9	27	380.8	18.3	5	1445.8	75.5
10	13	27	553.0	18.3	10	1512.0	75.7
1000	16	22	551.3	18.3	12	1536.0	75.7
8000	17	23	617.4	18.3	13	1547.1	75.8

are run using the intermediate problem size with 3'641'400 degrees of freedom and using a vertical loop source corresponding to a horizontal magnetic dipole source

Table 11 Problem 1: Comparison of outer iteration count (N_{it}^{outer}), time taken to obtain the solution (time [s]) and memory requirements (maximum memory usage for simulation; mem[GB]) when applying different block preconditioners

freq [Hz]	PRESB (tol= 10^{-12})			Block Diagonal (tol= 10^{-12})			Block Diagonal (tol= 10^{-8})		
	N_{it}^{outer}	time [s]	mem [GB]	N_{it}^{outer}	time [s]	mem [GB]	N_{it}^{outer}	time [s]	mem [GB]
0.1	7	40.0	4.3	11	65.7	4.4	7	38.6	4.3
1	10	56.7	4.4	16	90.5	4.4	11	58.6	4.4
10	16	79.3	4.4	27	143.3	4.5	17	86.5	4.4
100	22	93.5	4.5	38	167.2	4.6	20	81.9	4.4
1000	19	70.2	4.4	34	130.7	4.6	20	72.6	4.4
5000	18	62.8	4.4	32	122.9	4.5	21	67.4	4.4
8000	18	96.8	4.4	33	191.8	4.5	21	111.0	4.4
10000	18	460.7	4.4	33	993.6	4.5	21	609.6	4.5

we solve the resulting real-valued two-by-two block system by the GCR method preconditioned with a highly efficient block-based preconditioner PRESB. Our numerical experiments confirm the theoretical estimates that the PRESB-preconditioned GCR solver is robust with respect to spatially variable material parameters σ , μ and ε and with respect to the number of degrees of freedom by showing experiments across five orders of magnitude with regard to problem size and for stretched structured, unstructured as well as locally refined meshes. For the models considered in this work, the iterative algorithm reaches convergence within a small (typically < 20) number of outer iterations.

The efficiency of the iterative framework depends on the efficiency of the solution of the two inner systems of equations, being a part of PRESB. In this study, these two inner systems are either solved using a direct method or an iterative GCR/FGMRES method with the auxiliary-space preconditioner AMS. Our numerical tests clearly show that it is computationally beneficial to employ the suggested preconditioned iterative solver to the two inner systems instead of a direct solver in terms of overall memory requirements and time spent to obtain the solution for the two-by-two block system. Further and even though the iterative framework works with a system twice the size of the original complex-valued system, the iterative method works on large-scale problems where direct solvers succumb due to their excessive memory consumption, even on smaller less performance oriented computers and platforms.

From the above, it becomes clear that the efficiency of the overall algorithm stands and falls with the performance of the auxiliary-space preconditioner which is devised for semi-definite $\mathcal{H}_0(\text{curl}, \Omega)$ problems. We note that the system matrix of the two inner systems does not necessarily adhere to semi-definiteness. In fact, the system matrix becomes indefinite for frequencies larger than 180 Hz. Our experiments show that the performance of AMS loses its

advantages in terms of time over the direct solver for a frequency of 10'000 Hz. Thus, to achieve the full efficiency of PRESB across all possible frequency ranges, we need to use either frequency-dependent inner solvers or an advanced variant of AMS that could ensure better efficiency in the indefinite case.

Comparison to a similar iterative framework preconditioned with a block-diagonal preconditioner reveals that our approach converges in slightly fewer iterations resulting in certain execution time gains. It is noted that previous works show strong scalability for the iterative solver as well as for AMS. Further, AMS provides support for high-order curl-conforming element discretisations. Our future research is thus focused on extending our framework for high-order basis functions and to upscale our code in order to exploit the performance offered by modern distributed-memory platforms.

Appendix A: Accuracy for the 3D model compared to finite element solutions

In the following, we present results from numerical simulations obtained by our iterative framework and by a finite element code developed by [74], that uses a direct solver. In particular, we deploy twelve receivers across the target body of the 3D model (Problem 2) in cross-line configuration, meaning the receiver line is perpendicular to the transmission direction. The receiver line is offset by 3.5 km in x -direction. The receivers range from -3 to -0.5 km and from 0.5 to 3 km in y -direction at intervals of 500 m. Figure 5 shows the electric and magnetic fields excited by line source transmitting at 10 Hz for both the iterative framework and the finite element code denoted by SEM and FEM respectively. The responses for both numerical schemes are in good agreement with each other. Further, Table 12 displays the mean relative deviations of the electric

Responses of cross-line receivers at 3.5 km offset for $f=10$ Hz

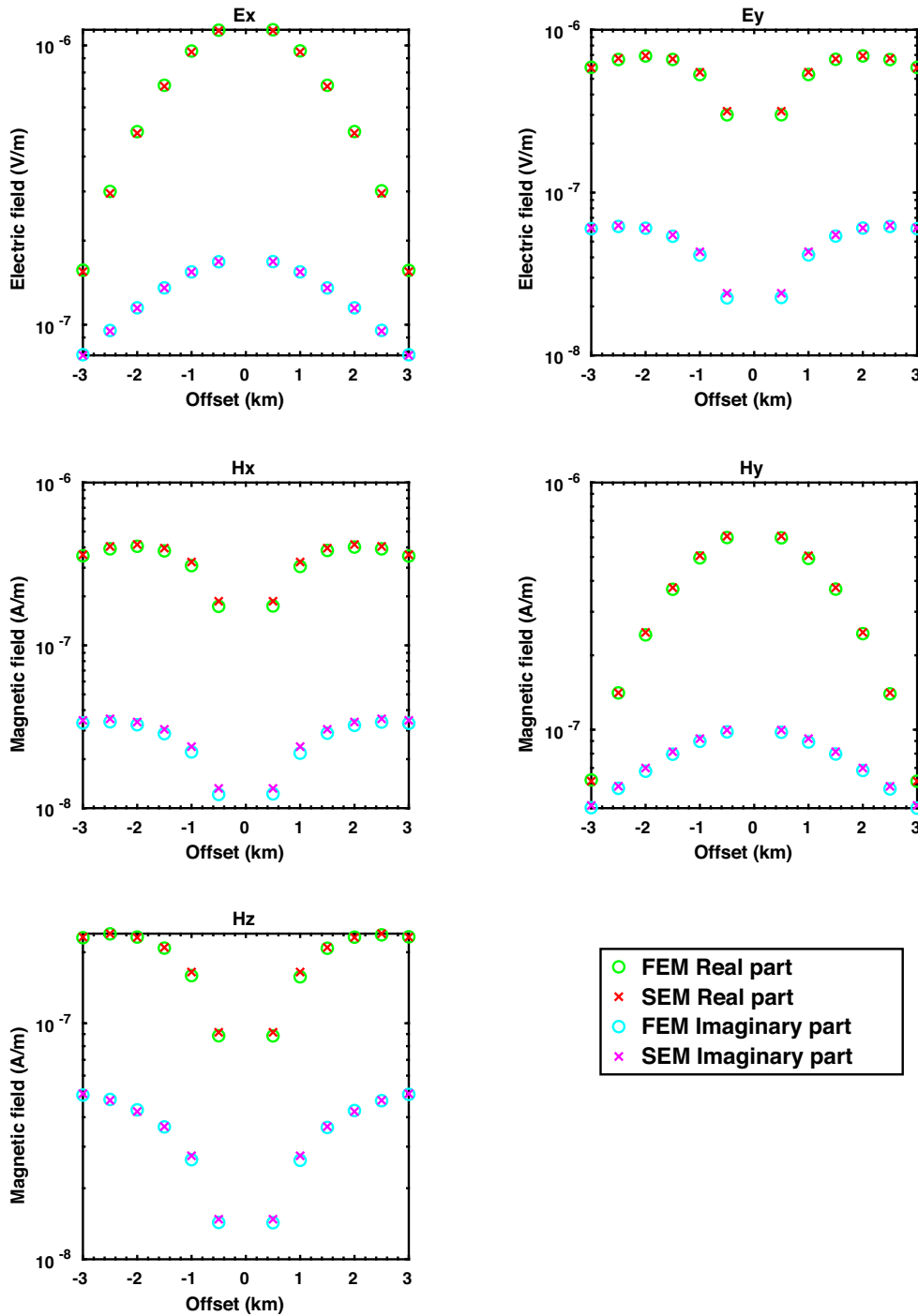


Fig. 5 Problem 2: Comparison the electric and magnetic field components split in real and imaginary part obtained with our iterative solver (SEM) and a finite element code [74, FEM,]

Table 12 Problem 2: Average relative deviations of all the field components between the spectral element and finite element results for the 3D model. Note that the finite element solutions are taken as the reference

Field component	Average relative deviations in %			
	f=10 Hz		f=100 Hz	
	Real	Imaginary	Real	Imaginary
E_x	1.10	0.26	2.40	1.60
E_y	1.90	2.60	1.90	2.10
H_x	4.10	6.00	3.10	4.80
H_y	1.30	2.40	5.60	2.50
H_z	1.60	1.80	3.00	1.50

and magnetic field components across all receivers thus confirming the observed agreement. Note, that the relative deviations are given for frequencies of 10 and 100 Hz.

Acknowledgements We thank Paula Rulff for providing the results for the 3D model. This work was partly funded by Uppsala's Center for Interdisciplinary Mathematics (CIM). The computations were enabled by resources in project SNIC 2021/22-883 provided by the Swedish National Infrastructure for Computing (SNIC) at UPPMAX, partially funded by the Swedish Research Council through grant agreement no. 2018-05973.

Funding Open access funding provided by Uppsala University.

Data Availability The data underlying this article can be shared upon reasonable request. Since the modelling code is to be embedded in a larger software package, it cannot be made freely available.

Declarations

Conflict of Interests The authors have no competing interests to declare that are relevant to the content of this article.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

- Ainsworth, M., Coyle, J.: Conditioning of hierarchic p-version Nédélec elements on meshes of curvilinear quadrilaterals and hexahedra. *SIAM J. Numer. Anal.* **41**(2), 731–750 (2003)
- Alumbaugh, D.L., Newman, G.A., Prevost, L., Shadid, J.N.: Three-dimensional wideband electromagnetic modeling on massively parallel computers. *Radio Sci.* **31**(1), 1–23 (1996)
- Amestoy, P., Buttari, A., L'Excellent, J.-Y., Mary, T.: Performance and scalability of the block low-rank multifrontal factorization on multicore architectures. *ACM Transactions on Mathematical Software* **45**, 2:1–2:26 (2019)
- Amestoy, P., Duff, I.S., Koster, J., L'Excellent, J.-Y.: A fully asynchronous multifrontal solver using distributed dynamic scheduling. *SIAM J. Matrix Anal. Appl.* **23**(1), 15–41 (2001)
- Amestoy, P.R., Guermouche, A., L'Excellent, J.-Y., Pralet, S.: Hybrid scheduling for the parallel solution of linear systems. *Parallel Comput.* **32**(2), 136–156 (2006)
- Aruliah, D.A., Ascher, U.M.: Multigrid preconditioning for Krylov methods for time-harmonic Maxwell's equations in three dimensions. *SIAM J. Sci. Comput.* **24**(2), 702–718 (2002)
- Axelsson, O.: Unified analysis of preconditioning methods for saddle point matrices. *Numerical Linear Algebra with Applications* **22**(2), 233–253 (2015)
- Axelsson, O.: Optimality properties of a square block matrix preconditioner with applications. *Computers & Mathematics with Applications* **80**(2), 286–294 (2020)
- Axelsson, O., Boyanova, P., Kronbichler, M., Neytcheva, M., Wu, X.: Numerical and computational efficiency of solvers for two-phase problems. *Computers & Mathematics with Applications* **65**(3), 301–314 (2013)
- Axelsson, O., Farouq, S., Neytcheva, M.: Comparison of preconditioned Krylov subspace iteration methods for PDE-constrained optimization problems. *Numerical Algorithms* **73**(3), 631–663 (2016)
- Axelsson, O., Farouq, S., Neytcheva, M.: A preconditioner for optimal control problems, constrained by Stokes equation with a time-harmonic control. *J. Comput. Appl. Math.* **310**, 5–18 (2017)
- Axelsson, O., Lukáš, D.: Preconditioning methods for eddy-current optimally controlled time-harmonic electromagnetic problems. *J. Numer. Math.* **27**(1), 1–21 (2019)
- Axelsson, O., Neytcheva, M.: Eigenvalue estimates for preconditioned saddle point matrices. *Numerical Linear Algebra with Applications* **13**(4), 339–360 (2006)
- Axelsson, O., Neytcheva, M.: Preconditioners for two-by-two block matrices with square blocks. Department of Information Technology, Uppsala Universitet (2018)
- Axelsson, O., Neytcheva, M., Ahmad, B.: A comparison of iterative methods to solve complex valued linear algebraic systems. *Numerical Algorithms* **66**(4), 811–841 (2014)
- Axelsson, O., Vassilevski, P.S.: A black box generalized conjugate gradient solver with inner iterations and variable-step preconditioning. *SIAM J. Matrix Anal. Appl.* **12**(4), 625–644 (1991)
- Bai, Z.-Z.: Rotated block triangular preconditioning based on PMHSS. *Science China Mathematics* **56**(12), 2523–2538 (2013)
- Bai, Z.-Z., Benzi, M., Chen, F.: On preconditioned MHSS iteration methods for complex symmetric linear systems. *Numerical Algorithms* **56**(2), 297–317 (2011)
- Bai, Z.-Z., Benzi, M., Chen, F., Wang, Z.-Q.: Preconditioned MHSS iteration methods for a class of block two-by-two linear systems with applications to distributed control problems. *IMA J. Numer. Anal.* **33**(1), 343–369 (2013)
- Balay, S., Abhyankar, S., Adams, M.F., Benson, S., Brown, J., Brune, P., Buschelman, K., Constantinescu, E., Dalcin, L., Dener, A., Eijkhout, V., Gropp, W.D., Hapla, V., Isaac, T., Jolivet, P., Karpeev, D., Kaushik, D., Knepley, M.G., Kong, F., Kruger, S., May, D.A., McInnes, L.C., Mills, R.T., Mitchell, L., Munson, T., Roman, J.E., Rupp, K., Sanan, P., Sarich, J., Smith, B.F., Zampini, S., Zhang, H., Zhang, H., Zhang, J.: PETSc/TAO users manual

- (Tech. Re. No. ANL-21/39 - Revision 3.17). Argonne National Laboratory (2022)
21. Balay, S., Gropp, W.D., McInnes, L.C., Smith, B.F.: Efficient management of parallelism in object oriented numerical software libraries. In: Arge, E., Bruaset, A.M., Langtangen, H.P. (eds.) *Modern Software Tools in Scientific Computing*, pp. 163–202. Birkhäuser Press (1997)
 22. Barenblatt, G.I., Barenblatt, G.I., Isaakovich, B.G.: *Scaling, Self-Similarity, and Intermediate Asymptotics: Dimensional Analysis and Intermediate Asymptotics* (No. 14). Cambridge University Press, Cambridge (1996)
 23. Benzi, M., Golub, G.H., Liesen, J.: Numerical solution of saddle point problems. *Acta Numer.* **14**, 1–137 (2005)
 24. Bhaskar, R., Nigam, A.: Qualitative physics using dimensional analysis. *Artif. Intell.* **45**(1-2), 73–111 (1990)
 25. Bin Zubair Syed, H., Farquharson, C., MacLachlan, S.: Block preconditioning techniques for geophysical electromagnetics. *SIAM J. Sci. Comput.* **42**(3), B696–B721 (2020)
 26. Börner, R.-U.: Numerical modelling in geo-electromagnetics: Advances and challenges. *Surv. Geophys.* **31**(2), 225–245 (2010)
 27. Boyanova, P., Do-Quang, M., Neytcheva, M.: Efficient preconditioners for large scale binary Cahn-Hilliard models. *Computational Methods in Applied Mathematics* **12**(1), 1–22 (2012)
 28. Cai, H., Xiong, B., Han, M., Zhdanov, M.: 3D controlled-source electromagnetic modeling in anisotropic medium using edge-based finite element method. *Comput. Geosci.* **73**, 164–176 (2014)
 29. Castillo-Reyes, O., de la Puente, J., Cela, J.M.: PETGEM: A parallel code for 3D CSEM forward modeling using edge finite elements. *Comput. Geosci.* **119**, 123–136 (2018)
 30. Chen, J., Chen, Z., Cui, T., Zhang, L.-B.: An adaptive finite element method for the eddy current model with circuit/field couplings. *SIAM J. Sci. Comput.* **32**(2), 1020–1042 (2010)
 31. Chow, E., Falgout, R.D., Hu, J.J., Tuminaro, R.S., Yang, U.M.: A survey of parallelization techniques for multigrid solvers. *Parallel Processing for Scientific Computing* 179–201 (2006)
 32. Da Silva, N.V., Morgan, J.V., MacGregor, L., Warner, M.: A finite element multifrontal method for 3D CSEM modeling in the frequency domain. *Geophysics* **77**(2), E101–E115 (2012)
 33. Dendy, J.: Black box multigrid. *J. Comput. Phys.* **48**(3), 366–386 (1982)
 34. Eisenstat, S.C., Elman, H.C., Schultz, M.H.: Variational iterative methods for nonsymmetric systems of linear equations. *SIAM J. Numer. Anal.* **20**(2), 345–357 (1983)
 35. Elías, M.W., Zyserman, F.I., Rosas-Carbajal, M., Manassero, M.C.: Three-dimensional modelling of controlled source electromagnetic surveys using non-conforming finite element methods. *Geophys. J. Int.* **229**(2), 1133–1151 (2022)
 36. Falgout, R.D.: An introduction to algebraic multigrid (Tech. Rep.). Lawrence Livermore National Lab., Livermore, CA (United States) (2006)
 37. Falgout, R.D., Jones, J.E., Yang, U.M.: The design and implementation of hypre, a library of parallel high performance preconditioners. In: *Numerical Solution of Partial Differential Equations on Parallel Computers*, pp. 267–294. Springer (2006)
 38. Falgout, R.D., Yang, U.M.: hypre: A library of high performance preconditioners. In: *International Conference on Computational Science*, pp. 632–641 (2002)
 39. Farquharson, C.G., Miensopust, M.P.: Three-dimensional finite-element modelling of magnetotelluric data with a divergence correction. *J. Appl. Geophys.* **75**(4), 699–710 (2011)
 40. Gee, M., Siefert, C., Hu, J., Tuminaro, R., Sala, M.G.: *ML 5.0 smoothed aggregation user's guide* (Tech. Rep. No. SAND2006-2649). Sandia National Laboratories (2006)
 41. Geuzaine, C., Remacle, J.-F.: Gmsh: A 3-D finite element mesh generator with built-in pre-and post-processing facilities. *Int. J. Numer. Meth. Eng.* **79**(11), 1309–1331 (2009)
 42. Gottlieb, D., Orszag, S.A.: *Numerical analysis of spectral methods: Theory and Applications*. SIAM (1977)
 43. Gould, N.I., Scott, J.A., Hu, Y.: A numerical evaluation of sparse direct solvers for the solution of large sparse symmetric linear systems of equations. *ACM Transactions on Mathematical Software (TOMS)* **33**(2), 10–es (2007)
 44. Grayver, A.V., Bürg, M.: Robust and scalable 3-D geoelectromagnetic modelling approach using the finite element method. *Geophys. J. Int.* **198**(1), 110–125 (2014)
 45. Grayver, A.V., Kolev, T.V.: Large-scale 3D geoelectromagnetic modeling using parallel adaptive high-order finite element method. *Geophysics* **80**(6), E277–E291 (2015)
 46. Greif, C., Schötzau, D.: Preconditioners for the discretized time-harmonic Maxwell equations in mixed form. *Numerical Linear Algebra with Applications* **14**(4), 281–297 (2007)
 47. Haber, E., Heldmann, S.: An octree multigrid method for quasi-static Maxwell's equations with highly discontinuous coefficients. *J. Comput. Phys.* **223**(2), 783–796 (2007)
 48. Hiptmair, R., Xu, J.: Nodal auxiliary space preconditioning in H (curl) and H (div) spaces. *SIAM J. Numer. Anal.* **45**(6), 2483–2509 (2007)
 49. Jahandari, H., Farquharson, C.G.: Finite-volume modelling of geophysical electromagnetic data on unstructured grids using potentials. *Geophys. J. Int.* **202**(3), 1859–1876 (2015)
 50. Jiránek, P., Rozložník, M., Gutknecht, M.H.: How to make simpler GMRES and GCR more stable. *SIAM J. Matrix Anal. Appl.* **30**(4), 1483–1499 (2009)
 51. Kalscheuer, T., Juhojuntti, N., Vaittinen, K.: Two-dimensional magnetotelluric modelling of ore deposits: Improvements in model constraints by inclusion of borehole measurements. *Surv. Geophys.* **39**(3), 467–507 (2018)
 52. Koldan, J., Puzyrev, V., de la Puente, J., Houzeaux, G., Cela, J.M.: Algebraic multigrid preconditioning within parallel finite-element solvers for 3-D electromagnetic modelling problems in geophysics. *Geophys. J. Int.* **197**(3), 1442–1458 (2014)
 53. Kolev, T.V., Vassilevski, P.S.: Parallel H1-based auxiliary space AMG solver for H (curl) problems (Tech. Rep.). Lawrence Livermore National Lab., Livermore, CA (United States) (2006)
 54. Kolev, T.V., Vassilevski, P.S.: Parallel auxiliary space AMG for H (curl) problems. *J. Comput. Math.* 604–623 (2009)
 55. Kollmann, M.: Efficient iterative solvers for saddle point systems arising in PDE-constrained optimization problems with inequality constraints. Johannes Kepler University Linz (2013)
 56. Kolmbauer, M., Langer, U.: A robust preconditioned Min-Res solver for distributed time-periodic eddy current optimal control problems. *SIAM J. Sci. Comput.* **34**(6), B785–B809 (2012)
 57. Monk, P., et al.: *Finite Element Methods for Maxwell's Equations*. Oxford University Press, Oxford (2003)
 58. Mulder, W.A.: A multigrid solver for 3D electromagnetic diffusion. *Geophys. Prospect.* **54**(5), 633–649 (2006)
 59. Nédélec, J.-C.: Mixed finite elements in \mathbb{R}^3 . *Numer. Math.* **35**(3), 315–341 (1980)
 60. Newman, G.A.: A review of high-performance computational strategies for modeling and imaging of electromagnetic induction data. *Surv. Geophys.* **35**(1), 85–100 (2014)
 61. Pardo, D., Paszynski, M., Collier, N., Alvarez, J., Dalcin, L., Calo, V.M.: A survey on direct solvers for Galerkin methods. *SeMA Journal* **57**(1), 107–134 (2012)
 62. Patera, A.T.: A spectral element method for fluid dynamics: laminar flow in a channel expansion. *J. Comput. Phys.* **54**(3), 468–488 (1984)
 63. Pearson, J.W., Wathen, A.J.: A new approximation of the Schur complement in preconditioners for PDE-constrained optimization. *Numerical Linear Algebra with Applications* **19**(5), 816–829 (2012)

64. Pearson, J.W., Wathen, A.J.: Fast iterative solvers for convection-diffusion control problems. *Electron. Trans. Numer. Anal.* **40**, 294–310 (2013)
65. Pedersen, L.B., Bastani, M., Dynesius, L.: Groundwater exploration using combined controlled-source and radiomagnetotelluric techniques. *Geophysics* **70**(1), G8–G15 (2005)
66. Puzyrev, V., Cela, J.M.: A review of block Krylov subspace methods for multisource electromagnetic modelling. *Geophys. J. Int.* **202**(2), 1241–1252 (2015)
67. Puzyrev, V., Koldan, J., de la Puente, J., Houzeaux, G., Vázquez, M., Cela, J.M.: A parallel finite-element method for three-dimensional controlled-source electromagnetic forward modelling. *Geophys. J. Int.* **193**(2), 678–693 (2013)
68. Puzyrev, V., Koric, S., Wilkin, S.: Evaluation of parallel direct sparse linear solvers in electromagnetic geophysical problems. *Comput. Geosci.* **89**, 79–87 (2016)
69. Qiu, C., Yin, C., Liu, Y., Ren, X., Chen, H., Yan, T.: Solution of large-scale 3D controlled-source electromagnetic modeling problem using efficient iterative solvers. *Geophysics* **86**(4), E283–E296 (2021)
70. Reitzinger, S., Schöberl, J.: An algebraic multigrid method for finite element discretizations with edge elements. *Numerical Linear Algebra with Applications* **9**(3), 223–238 (2002)
71. Ren, Z., Kalscheuer, T., Greenhalgh, S., Maurer, H.: A goal-oriented adaptive finite-element approach for plane wave 3-D electromagnetic modelling. *Geophys. J. Int.* **194**(2), 700–718 (2013)
72. Rochlitz, R., Skibbe, N., Günther, T.: custEM: Customizable finite-element simulation of complex controlled-source electromagnetic data. *Geophysics* **84**(2), F17–F33 (2019)
73. Rønquist, E.M., Patera, A.T.: A Legendre spectral element method for the Stefan problem. *Int. J. Numer. Meth. Eng.* **24**(12), 2273–2299 (1987)
74. Rulff, P., Buntin, L.M., Kalscheuer, T.: Efficient goal-oriented mesh refinement in 3-D finite-element modelling adapted for controlled source electromagnetic surveys. *Geophys. J. Int.* **227**(3), 1624–1645 (2021)
75. Saad, Y.: A flexible inner-outer preconditioned GMRES algorithm. *SIAM J. Sci. Comput.* **14**(2), 461–469 (1993)
76. Saad, Y.: *Iterative Methods for Sparse Linear Systems*. SIAM (2003)
77. Schenk, O., Gärtner, K.: Solving unsymmetric sparse systems of linear equations with PARDISO. *Futur. Gener. Comput. Syst.* **20**(3), 475–487 (2004)
78. Schenk, O., Gärtner, K.: On fast factorization pivoting methods for sparse symmetric indefinite systems. *Electron. Trans. Numer. Anal.* **23**(1), 158–179 (2006)
79. Schöberl, J., Zulehner, W.: Symmetric indefinite preconditioners for saddle point problems with applications to PDE-constrained optimization problems. *SIAM J. Matrix Anal. Appl.* **29**(3), 752–773 (2007)
80. Schwarzbach, C.: Stability of finite element solutions to Maxwell's equations in frequency domain. Unpublished doctoral dissertation, TU Bergakademie Freiberg (2009)
81. Schwarzbach, C., Börner, R.-U., Spitzer, K.: Three-dimensional adaptive higher order finite element simulation for geoelectromagnetics—a marine CSEM example. *Geophys. J. Int.* **187**(1), 63–74 (2011)
82. Sheard, S.N., Ritchie, T.J., Christopherson, K.R., Brand, E.: Mining, environmental, petroleum, and engineering industry applications of electromagnetic techniques in geophysics. *Surv. Geophys.* **26**(5), 653–669 (2005)
83. Smith, J.T.: Conservative modeling of 3-D electromagnetic fields, Part II: Biconjugate gradient solution and an accelerator. *Geophysics* **61**(5), 1319–1324 (1996)
84. Stoll, M., Wathen, A.: Preconditioning for partial differential equation constrained optimization with control constraints. *Numerical Linear Algebra with Applications* **19**(1), 53–71 (2012)
85. Streich, R.: 3D finite-difference frequency-domain modeling of controlled-source electromagnetic data: Direct solution and optimization for high accuracy. *Geophysics* **74**(5), F95–F105 (2009)
86. Streich, R.: Controlled-source electromagnetic approaches for hydrocarbon exploration and monitoring on land. *Surv. Geophys.* **37**(1), 47–80 (2016)
87. Streich, R., Becken, M., Ritter, O.: Imaging of CO₂ storage sites, geothermal reservoirs, and gas shales using controlled-source magnetotellurics: Modeling studies. *Geochemistry* **70**, 63–75 (2010)
88. Trottenberg, U., Oosterlee, C.W., Schuller, A.: *Multigrid*. Elsevier, New York (2000)
89. Um, E.S., Commer, M., Newman, G.A.: 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 (2013)
90. Van der Vorst, H.A.: Bi-CGSTAB: A fast and smoothly converging variant of Bi-CG for the solution of nonsymmetric linear systems. *SIAM J. Sci. Stat. Comput.* **13**(2), 631–644 (1992)
91. Wannamaker, P.E., Stodt, J.A., Rijo, L.: A stable finite element solution for two-dimensional magnetotelluric modelling. *Geophys. J. Int.* **88**(1), 277–296 (1987)
92. Weaver, J.T.: *Mathematical Methods for Geo-Electromagnetic Induction*, vol. 7. Research Studies Press, Baldock (1994)
93. Weiss, M., Kalscheuer, T., Ren, Z.: Spectral element method for 3-D controlled-source electromagnetic forward modelling using unstructured hexahedral meshes. *Geophys. J. Int.* **232**(2), 1427–1454 (2022)
94. Yang, U.M.: Parallel algebraic multigrid methods—high performance preconditioners. In: *Numerical Solution of Partial Differential Equations on Parallel Computers*, pp. 209–236. Springer (2006)
95. Yin, C., Liu, L., Liu, Y., Zhang, B., Qiu, C., Huang, X.: 3D frequency-domain airborne EM forward modelling using spectral element method with Gauss–Lobatto–Chebyshev polynomials. *Explor. Geophys.* **50**(5), 461–471 (2019)
96. Zulehner, W.: Nonstandard norms and robust estimates for saddle point problems. *SIAM J. Matrix Anal. Appl.* **32**(2), 536–560 (2011)
97. Zulehner, W.: Efficient solvers for saddle point problems with applications to PDE-constrained optimization. In: *Advanced Finite Element Methods and Applications*, pp. 197–216. Springer (2013)

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.