



HAL
open science

Three-dimensional finite-difference finite-element frequency-domain wave simulation with multi-level optimized additive Schwarz domain-decomposition preconditioner: A tool for FWI of sparse node datasets

Pierre-Henri Tournier, Pierre Jolivet, Victorita Dolean, Hossein Aghamiry, Stéphane Operto, Sebastian Rizzo

► To cite this version:

Pierre-Henri Tournier, Pierre Jolivet, Victorita Dolean, Hossein Aghamiry, Stéphane Operto, et al.. Three-dimensional finite-difference finite-element frequency-domain wave simulation with multi-level optimized additive Schwarz domain-decomposition preconditioner: A tool for FWI of sparse node datasets. *Geophysics*, 2022, 87 (5), pp.1-84. 10.1190/geo2021-0702.1 . hal-03942570

HAL Id: hal-03942570

<https://hal.science/hal-03942570>

Submitted on 17 Jan 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution 4.0 International License

THREE-DIMENSIONAL FINITE-DIFFERENCE & FINITE-ELEMENT FREQUENCY-DOMAIN WAVE SIMULATION WITH MULTI-LEVEL OPTIMIZED ADDITIVE SCHWARZ DOMAIN-DECOMPOSITION PRECONDITIONER: A TOOL FOR FWI OF SPARSE NODE DATASETS

A PREPRINT

 **P.-H. Tournier**

Sorbonne University, CNRS, France. tournier@ljl1.math.upmc.fr

 **P. Jolivet**

IRIT-CNRS, University of Toulouse, France. Pierre.Jolivet@enseeiht.fr

 **V. Dolean**

Univ. of Strathclyde, United Kingdom/UCA, CNRS, LJAD, France. Victorita.Dolean@univ-cotedazur.fr

 **H. S. Aghamiry**

University Cote d'Azur - CNRS - IRD - OCA, Geoazur, Valbonne, France. aghamiry@geoazur.unice.fr

 **S. Operto**

University Cote d'Azur - CNRS - IRD - OCA, Geoazur, Valbonne, France. operto@geoazur.unice.fr

 **S. Rizzo**

University Cote d'Azur - CNRS - IRD - OCA, Geoazur, Valbonne, France. rizzo@geoazur.unice.fr

October 29, 2021

ABSTRACT

Efficient frequency-domain Full Waveform Inversion (FWI) of long-offset node data can be designed with a few discrete frequencies hence allowing for compact volume of data to be managed. Moreover, attenuation effects can be straightforwardly implemented in the forward problem without computational overhead. However, 3D frequency-domain seismic modeling is challenging since it requires solving a large and sparse linear indefinite system per frequency with multiple right-hand sides. This linear system can be solved by direct or iterative methods. The former are very efficient to process multiple right-hand sides but may suffer from limited scalability for very large problems. Iterative methods equipped with a domain decomposition preconditioner provide a suitable alternative to process large computational domains for sparse node acquisition. The domain decomposition preconditioner relies on an optimized restricted additive Schwarz (ORAS) method, where a Robin or Perfectly-Matched Layer (PML) condition is implemented at the boundaries between the subdomains. The preconditioned system is solved by a Krylov subspace method, while a block low-rank Lower-Upper (LU) decomposition of the local matrices is performed at a preprocessing stage. Multiple sources are processed in group with a pseudo-block method. The accuracy, the computational cost and the scalability of the ORAS solver are assessed against several realistic benchmarks. For the considered benchmarks, a compact wavelength-adaptive 27-point finite-difference stencil on regular

Cartesian grid shows better accuracy and computational efficiency than a P3 finite-element method on h -adaptive tetrahedral mesh, which remains however beneficial to comply with known boundaries such as bathymetry. The scalability of the method, the block processing of multiple RHS, the straightforward implementation of attenuation, which further improves the convergence of the iterative solver, make the method to be a versatile forward engine for large-scale 3D FWI applications from sparse node data sets.

1 INTRODUCTION

Frequency-domain full-waveform inversion (FWI) is suitable for long-offset stationary-recording acquisition, since subsurface models can be reconstructed with a few frequencies and attenuation is easily implemented without computational overhead (Górszczyk et al., 2017; Pratt et al., 1996; Sirgue and Pratt, 2004). Frequency-domain FWI is a data-fitting problem for subsurface parameter estimation constrained by a Helmholtz-type boundary-value equation. This elliptic partial-differential equation is commonly discretized with finite differences or through more elaborate finite elements, which are crafted according to a specific problem (Marfurt, 1984). Its discretization leads to a sparse linear system per frequency whose unknown is the wavefield, the right-hand side (RHS) the seismic source and the matrix coefficients embed the subsurface properties. To solve such a system, we can use either a sparse direct solver (Duff et al., 2017) or an iterative one (Saad, 2003a). While a direct solver is efficient when processing multiple RHSs for problems involving less than ~ 100 million unknowns (Amestoy et al., 2016, 2021; Kostin et al., 2019), the memory overhead generated by the storage of the LU factors and the limited scalability of the LU decomposition make application on large scale problems difficult. The second approach relies on iterative solvers, whose natural scalability and moderate memory demand make them suitable for large-scale problems (see Plessix (2017) for a comprehensive discussion).

However, due to the indefiniteness of the operator, fast convergence of iterative solvers for the discretized Helmholtz equation remains challenging as shown in Ernst and Gander (2012), hence requiring efficient preconditioners. To tackle this issue, many techniques have been developed, which can be divided into three broad categories: incomplete factorizations, e.g., analytic incomplete LU (Gander and Nataf, 2000), multigrid with shifted Laplacian (Erlangga, 2008; Riyanti et al., 2007), and domain decomposition preconditioners including overlapping Schwarz type methods among them (Dolean et al., 2015). In turn, domain decomposition methods can be of different types. We refer the reader to the paper of Gander and Zhang (2019) for an overview of these solvers and the link between them.

Domain decomposition offers a very suitable parallel framework to tackle large scale problems repeatedly, as in the case of inverse problems with multiple RHSs. In this context, an example of the use of a domain decomposition method as a solver for a medical imaging problem based on time-harmonic Maxwell's equations can be found in Tournier et al. (2019) and due to the similarities between Maxwell's and acoustic wave equation (Carcione and Robinson, 2002), a similar type of approach can be applied to Helmholtz equations (Ben Hadj Ali et al., 2011).

In this paper, we present a new solver for 3D frequency-domain wave simulation as a forward engine for FWI, which consists in combining the Krylov subspace generalized minimal residual (GMRES) solver (Saad, 2003a) with an optimized restricted additive Schwarz (ORAS) domain-decomposition preconditioner (Bonazzoli et al., 2019; Graham et al., 2017). We will present both one-level and two-level methods and discuss their domain of application. Two-level methods are obtained from one-level methods by adding a coarse space correction in order to achieve scalability with respect to the number of subdomains. However, a specificity of wave propagation problems, in particular those arising in exploration geophysics, is the sensitivity of the method to the physical parameters such as the range of wave numbers and the level of heterogeneity of the medium (Plessix, 2017). To manage this issue, several coarse space correction strategies have been investigated for Helmholtz problems including those of interest here in Bootland et al. (2021a,b) and the coarse space based on a geometrical grid (aka the grid coarse space) was considered to be the best strategy for solving the applications of interest especially in the case of parsimonious discretization with the lowest number of points per wavelength possible. This choice was pushed further and assessed in more detail on a few test cases in Dolean et al. (2020a,b). On the other side, spectral coarse spaces like those based upon Dirichlet-to-Neumann (DtN) operator (Bootland and Dolean, 2019; Conen et al., 2014) remain potentially promising alternatives for inverse problems involving multiple RHSs where the parallel precomputation of the coarse space components could bring a clear advantage and improve considerably the overall computation time. Compared to the well-known preconditioner based upon shifted Laplacian (Erlangga, 2008), let's add that the optimized Schwarz preconditioners we introduce here are less sensitive to the shift (added attenuation) and can be used without it.

In this study, we assess two discretization schemes of the Helmholtz equation. The first relies on continuous Lagrange finite elements of order 3 (P3) on unstructured tetrahedral meshes to comply with complex geometries and adapt the size of the elements to the local wavelength (h -adaptivity). Our rationale behind the choice of the polynomial order

is to choose the coarsest discretization providing the desired accuracy while being able to capture heterogeneities during FWI whose size is of the order of half the wavelength, namely the theoretical resolution of FWI (Virieux and Operto, 2009). The second discretization scheme relies on the recently proposed compact wavelength-adaptive 27-point finite-difference method on uniform Cartesian grid (Aghamiry et al., 2021). The governing idea of this method compared to the classical 27-point stencil (Operto et al., 2007) is to make its accuracy uniform with respect to the local wavelength through a local adaptation of the stencil coefficients. Finite difference methods are the most popular ones in exploration geophysics due to their simplicity and their ability to perform well on structured meshes with the lowest number of degrees of freedom for a prescribed accuracy. On the other side, finite elements may be favored when their versatility for dealing with unstructured geometries and non-trivial topographies is needed. The different paradigms governing these two methods prompt us to provide a re-assessment of their pros and cons in the framework of Helmholtz problems.

This paper is organized as follows. In the first section, we review the ORAS preconditioner and its two-level variant with the grid coarse space. We present their formulation and review several key algorithmic aspects. Then, we briefly review the two discretization schemes and their dispersion properties. The third section presents numerical results. We first define the optimal tuning of the ORAS solver that achieves the best compromise between accuracy and computational efficiency. Then, we assess the two discretization schemes against four large-scale benchmarks. The solutions are validated against analytical solutions for the first two benchmarks and the convergent Born series (CBS) method (Osnabrugge et al., 2016) for the last two benchmarks, which involve complex media. We conclude the numerical section with a weak and strong scalability analysis. In the final discussion section, we discuss the pros and cons of the proposed method against alternative approaches based on sparse direct solver and explicit time marching methods with a numerical example.

2 Method

We consider the Helmholtz equation, which is the simplest mathematical model of frequency-domain acoustic wave propagation:

$$(\Delta + k^2(\mathbf{x})) u(\mathbf{x}, \omega) = f(\mathbf{x}, \omega), \text{ in a subsurface domain } \Omega, \quad (2.1)$$

where u is the monochromatic pressure wavefield, f the monochromatic source, $k(\mathbf{x}, \omega) = \omega/c(\mathbf{x})$ is the wavenumber with ω denoting the angular frequency, $c(\mathbf{x})$ the wavespeed (which is complex valued in viscous media) and $\mathbf{x} = (x, y, z) \in \Omega$. Let us denote its boundary by $\partial\Omega = \Gamma_D \cup \Gamma_R$ where Dirichlet (free-surface) conditions are imposed on Γ_D ($u = 0$) and a Robin condition on Γ_R ($\frac{\partial u}{\partial \mathbf{n}} + iku = 0$). The Robin condition is a standard first order approximation to the far field Sommerfeld radiation condition enabling the description of the wave behavior in a bounded domain, while the physical domain is not bounded. Alternatively, perfectly-matched layer boundary conditions can be applied on Γ_R (Bérenger, 1994; Chew and Weedon, 1994). The resulting linear system, while being complex symmetric, is not Hermitian and this will condition the choice of iterative method to solve it. The method that is reviewed below can be extended to more general form of the wave equation including heterogeneous density (e.g., Operto et al., 2007), anisotropic effects (e.g., Operto et al., 2014), and elastic effects (Gosselin-Cliche and Giroux, 2014; Li et al., 2020, 2015), with however a potential impact of the physics on the convergence behavior of the method (Plessix, 2017). In the sequel of this study, the discretized form of equation 2.1 is written in matrix form as

$$A\mathbf{u} = \mathbf{f}. \quad (2.2)$$

where \mathbf{u} is the vector of degrees of freedom.

2.1 Optimized Schwarz preconditioners

We review the optimized Schwarz preconditioners that we use to solve efficiently the linear system (2.2). A well-known iterative solver for this type of indefinite linear systems is the Krylov subspace Generalized Minimal RESidual Method (GMRES) (Saad, 2003a). However, the Helmholtz operator requires efficient preconditioning, which can be done by domain decomposition (Dolean et al., 2015). Note that we will employ right preconditioning to solve (2.2): we will solve

$$AM^{-1}\mathbf{y} = \mathbf{f} \quad \text{where } \mathbf{u} = M^{-1}\mathbf{y} \quad (2.3)$$

with preconditioner M^{-1} . Compared to the left-preconditioned variant, right-preconditioned GMRES is usually considered more stable, and the residual of the preconditioned system is independent of the preconditioner.

Our approach is based on the optimized restricted additive Schwarz (ORAS) method. Let us first introduce the one-level variant. In order to use the domain decomposition method, we first partition Ω into overlapping subdomains $\{\Omega_j\}_{j=1}^N$. Note that graph partitioners such as Metis (Karypis, 2013) and Scotch (Pellegrini, 2018) provide a non-overlapping partition from which the overlapping counterpart can be inferred by adding more layers in a recursive

manner when subdomains with larger overlap are required. Now that we have a domain decomposition, we can define the restriction to a given subdomain Ω_j via a Boolean matrix $R_j \in \mathbb{R}^{n_j \times n}$ where n_j is the number of degrees of freedom in Ω_j . Since our subdomains overlap, we also make use of a partition of unity having matrix form $D_j \in \mathbb{R}^{n_j \times n_j}$, which is diagonal and satisfies $\sum_{j=1}^N R_j^T D_j R_j = I$. Note that R_j^T acts as an extension by zero outside of Ω_j . We can now define the one-level optimized restricted additive Schwarz (ORAS) preconditioner as

$$M_{\text{ORAS}}^{-1} = \sum_{j=1}^N R_j^T D_j B_j^{-1} R_j,$$

where B_j is the local matrix on Ω_j and Robin or impedance boundary conditions are assumed on $\partial\Omega_j \setminus \partial\Omega$ (as mentioned previously, perfectly-matched layer boundary conditions can be used instead in the case of regular partitioning). Each contribution in the sum of the above equation can be computed locally in parallel. The ORAS method is a one-level method as it relies on local subdomain solutions and local transmission of data. This is reflected in the scalability of such a method, as the number of iterations usually grows as we increase the number of subdomains N . Therefore, in order to achieve robustness with respect to N , a coarse space needs to be included, leading to a two-level method.

In a two-level method, the typical characteristics of the method (like the diameter H of the elements in the coarse grid or the number of eigenvectors in the coarse space) can be tuned in order to achieve the desired robustness, i.e., the weak dependence on the wave number k or the heterogeneities at the same time as the scalability. The typical two-level preconditioner is mainly characterized by the thin and long matrix Z and can be written in a generic algebraic way as follows

$$M_{2,\text{ORAS}}^{-1} = M_{\text{ORAS}}^{-1} P + \underbrace{Z E^{-1} Z^\dagger}_Q. \quad (2.4)$$

Here M_{ORAS}^{-1} is the one-level preconditioner, Z is a rectangular matrix with full column rank, $E = Z^\dagger A Z$ is the so-called coarse grid matrix, $Q = Z E^{-1} Z^\dagger$ is the so-called coarse grid correction matrix, and $P = I - A Q$ is a projection matrix. The most natural coarse space is the one based on a coarser mesh or grid, the so-called grid coarse space. In the case of finite elements for example, let us consider \mathcal{T}^H , a coarse simplicial mesh of Ω with mesh diameter H , and $V^H \subset V$, the corresponding finite element space. Let Z the interpolation matrix between the coarse and the fine finite element spaces. Then, in this case, $E = Z^\dagger A Z$ is really the stiffness matrix of the problem discretized on the coarse mesh. Note that the inclusion of the coarse space can be done in several ways, the one indicated in (2.4) is based on an adapted deflation technique by making use of the projector P outside the coarse space for an increased efficiency.

As mentioned previously, the two discretization schemes considered in this paper are wavelength-adaptive 27-point finite differences and P3 finite elements. In the case of finite differences, the solution is defined by its values on the nodes of the Cartesian grid with a resolution of four points per wavelength ; this unfortunately means that a coarser grid discretization would lead to a poor representation of the solution, invalidating the benefits of a second level of preconditioning based on a coarse grid. Thus, the one-level ORAS preconditioner will be used for the solution of linear systems stemming from finite-difference discretization.

However, in the case of P3 finite elements, the much larger set of degrees of freedom allows for a relevant physical representation of the solution even on a coarser mesh with a resolution of two points per wavelength. Thus, contrary to the finite difference case, the second-level based on a coarse mesh discretization for finite elements still maintains a pertinent physical meaning and has a significant positive impact on convergence. Hence, the two-level variant will be used for finite element discretizations.

In the next section, we give more details on the practical aspects of the parallel implementation of the one- and two-level ORAS preconditioners.

2.2 Algorithmic aspects and parallel implementation

One-level method The first step is to decompose the computational domain into overlapping subdomains. For finite-difference discretizations, the regular grid is partitioned into overlapping cuboids. For finite-element discretizations on unstructured meshes, we rely on automatic graph partitioners such as Metis and Scotch, which produce a non-overlapping decomposition of the set of mesh elements while minimizing interfaces between subdomains and conserving good load-balancing. The overlapping decomposition is then obtained by simply adding successive layers of elements to reach the desired size of overlap. The choice of the width of the overlap region is discussed in more details below.

The implementation of the domain decomposition method used here relies on Message Passing Interface (MPI) parallelism. Each subdomain of the decomposition is assigned to one MPI process (the number of subdomains N is then equal to the number of MPI processes).

The parallel implementation operates only on distributed quantities; for instance, a global vector \mathbf{v} is represented as a distributed vector, i.e., a collection of local vectors $(\mathbf{v}_j)_{1 \leq j \leq N}$ where the values on the unknowns shared between multiple subdomains are the same:

$$\mathbf{v}_j = R_j \mathbf{v}, \quad j = 1, \dots, N.$$

The MPI process in charge of subdomain j then holds the restriction $R_j \mathbf{v}$ to subdomain j . Distributed operators can be defined in a similar manner; for example, the global operator A can be written as

$$A = \left(\sum_{i=1}^N R_i^T D_i R_i \right) A \quad (2.5)$$

$$= \sum_{i=1}^N R_i^T D_i A_i R_i, \quad (2.6)$$

where $A_i = R_i A R_i^T$ is the local restriction of A to subdomain i and the last equality holds under the hypothesis that D_i vanishes at the subdomain interface ; we then have $D_i R_i A = D_i A_i R_i$.

Thus, the distributed result $(\mathbf{u}_j)_{1 \leq j \leq N}$ of the parallel matrix-vector product with A applied to vector \mathbf{v} distributed as $(\mathbf{v}_i)_{1 \leq i \leq N}$ is

$$\mathbf{u}_j = R_j A \mathbf{v} \quad (2.7)$$

$$= R_j \sum_{i=1}^N R_i^T D_i A_i R_i \mathbf{v} \quad (2.8)$$

$$= R_j \sum_{i=1}^N R_i^T D_i A_i \mathbf{v}_i \quad (2.9)$$

$$= D_j A_j \mathbf{v}_j + \sum_{i \in \mathcal{O}(j)} R_j R_i^T D_i A_i \mathbf{v}_i, \quad (2.10)$$

where $\mathcal{O}(j)$ is the set of neighbors of subdomain j . Exchange operators $R_j R_i^T$ correspond to neighbor-to-neighbor MPI communications, transferring the values of the local input vector on the shared overlap region of subdomain i to corresponding neighboring subdomains j . Thus, applying operator A in parallel amounts to performing local weighted matrix-vector products $D_i A_i \mathbf{v}_i$ on each MPI process, exchanging the result in the overlapping region with neighboring subdomains, and summing contributions.

In the same manner, the distributed result $(\mathbf{u}_j)_{1 \leq j \leq N}$ of the application of the one-level ORAS preconditioner M_{ORAS}^{-1} to a distributed vector $(\mathbf{v}_i)_{1 \leq i \leq N}$ is

$$\mathbf{u}_j = R_j M_{\text{ORAS}}^{-1} \mathbf{v} \quad (2.11)$$

$$= R_j \sum_{i=1}^N R_i^T D_i B_i^{-1} R_i \mathbf{v} \quad (2.12)$$

$$= R_j \sum_{i=1}^N R_i^T D_i B_i^{-1} \mathbf{v}_i \quad (2.13)$$

$$= D_j B_j^{-1} \mathbf{v}_j + \sum_{i \in \mathcal{O}(j)} R_j R_i^T D_i B_i^{-1} \mathbf{v}_i, \quad (2.14)$$

which amounts to performing local weighted solves $D_i B_i^{-1} \mathbf{v}_i$ on each MPI process, exchanging the result in the overlapping region with neighboring subdomains, and summing contributions. The idea is then to precompute local factorizations of matrices B_i in each subdomain using a direct solver such as the Multifontal Massively Parallel Solver (MUMPS) (MUMPS team, 2021). This can be viewed as the *setup* cost of the preconditioner. Then, during the iterative solution of the linear system, local solves $B_i^{-1} \mathbf{v}_i$ occurring in the application of the preconditioner can be performed by forward-backward substitution using the local factorization of B_i in each subdomain.

All operations involved in the iterative solution of the linear system can be distributed the same way, and we have all the ingredients to solve the problem with parallel distributed preconditioned GMRES.

Note on the size of overlap For wave propagation problems, the idea of the Optimized Schwarz method is to construct B_i as a restriction of the global operator A on subdomain i with an approximation of transparent boundary conditions at the subdomain interface. In the case of finite-differences discretizations, a good approximation of transparent boundary conditions can be implemented using PMLs on the interface boundaries of the cuboid-shaped subdomains. In this case, the idea is to add a PML in each direction in the overlap region, with the width of the overlap sufficiently large for the PML to be a good enough approximation of transparent boundary conditions. In practice, we add three grid points in each direction (with a corresponding overlap width of six grid elements) for most of the experiments in this paper as it is found to be the best compromise (note that increasing the overlap width increases the computational cost), although we observed that the optimal amount of overlap can increase with frequency.

Note on processing multiple right-hand sides Frequency-domain FWI requires forward simulations with multiple sources, which means solving linear systems with multiple right-hand sides (RHSs). Instead of opting for the naive approach which consists in solving the linear systems successively for each RHS, we can use various ways to accelerate computations when processing multiple RHSs, from pseudo-block Krylov methods to block Krylov method with recycling such as the Block Generalized Conjugate Residual Method with Inner Orthogonalization with inexact breakdown and deflated restarting (BGCRO-DR) (Giraud et al., 2021). The simplest ones are pseudo-block methods, where operations for each RHS are simply fused together in order to achieve higher arithmetic intensity and decrease the number of global synchronizations. In particular, for our one-level ORAS method, this means that we can also leverage the multi-RHS processing capabilities of the direct solver for local problems when computing the application of the ORAS preconditioner M_{ORAS}^{-1} to a block of input vectors, using the factorization of B_i to perform forward elimination and backward substitution simultaneously on each input vector in order to compute the action of B_i^{-1} on the whole block.

Two-level method As mentioned previously, the coarse space we consider here is based on a coarse grid discretization of the PDE. Since a finite-difference grid with less than four points per wavelength cannot carry a meaningful representation of the solution, we only use the two-level method in the finite element case, where the larger set of degrees of freedom of the P3 discretization allows for a relevant representation of the solution even on a coarser mesh resolution of two points per wavelength.

For the two-level method, two meshes of the same domain must be considered simultaneously. Here we will consider nested meshes, so that the fine mesh is generated as a uniform refinement in which edges of all tetrahedra are divided uniformly in s . In this work we choose $s = 2$, which means that the coarse mesh is constructed with a resolution level of two points per wavelength. The required resolution of four points per wavelength will then be achieved on the fine mesh obtained by splitting.

The number of unknowns in the coarse problem is then $s^d = 8$ times smaller than the fine problem. However, for large frequencies the coarse problem still reaches tens of millions of unknowns. Thus, instead of relying on an exact factorization of the coarse matrix E , we solve it iteratively using a preconditioned GMRES method. The preconditioner for the coarse problem is again a one-level ORAS preconditioner ; this can be seen as a two-level recursive DD method. This outer-inner strategy requires the use of flexible methods such as FGMRES (Saad, 1993). Moreover, we observe that in practice an accurate solution of the coarse problem is not required: the best compromise is obtained by choosing a loose relative GMRES tolerance of 10^{-1} for the coarse problem.

Finally, in order to minimize communications between levels, a natural choice is to use the same subdomain partitioning for the coarse and fine levels. This means that each MPI process will be in charge of two (the fine and the coarse) discretizations of the same spatial subdomain, with fine subdomains being merely uniformly refined subdomains of the coarse grid. Thus, by setting the width of the overlap to 1 for the coarse mesh partitioning, the overlap at the fine level will then have width $s = 2$.

2.3 Discretization of the Helmholtz problem

We now review the two discretization methods that we consider in this study. The first relies on finite elements with Lagrange polynomials while the second one relies on finite differences.

2.3.1 Finite-element frequency-domain (FEFD) discretization

The first discretization scheme relies on the finite element method (Zienkiewicz et al., 2005), with piecewise Lagrange polynomials on a simplicial mesh \mathcal{T}^h of Ω , which has a characteristic element diameter h . The weak form of the

problem is to find $u \in V$ such that $a(u, v) = F(v)$, $\forall v \in V_0$ where

$$a(u, v) = \int_{\Omega} (\nabla u \cdot \nabla \bar{v} - k^2 u \bar{v}) \, dx + \int_{\Gamma_R} iku \bar{v} \, ds, \quad F(v) = \int_{\Omega} f \bar{v} \, dx, \quad (2.15)$$

and $V_0 = \{u \in H^1(\Omega) : u = 0 \text{ on } \Gamma_D\}$. Denoting the associated trial space $V^h \subset H^1(\Omega)$ and test space $V_0^h \subset V_0$, the discrete problem is to find $u_h \in V^h$ such that $a(u_h, v_h) = F(v_h)$, $\forall v_h \in V_0^h$. Let $\{\phi_j\}_{j=1}^n$ be the nodal basis for V_0 . Then we can rewrite the discrete weak formulation as a (complex) linear system

$$A\mathbf{u} = \mathbf{f}, \quad (2.16)$$

where the coefficient matrix $A \in \mathbb{C}^{n \times n}$ and right-hand side vector $\mathbf{f} \in \mathbb{C}^n$ are given by $A_{i,j} = a(\phi_j, \phi_i)$ and $f_i = F(\phi_i)$. We then seek the solution $\mathbf{u} \in \mathbb{C}^n$ of the (in general) complex symmetric indefinite system (2.16) to give $u_h(x) = \sum_{j=1}^n u_j \phi_j(x)$.

The oscillatory nature of solutions to the Helmholtz equation requires a sufficiently fine mesh in order to get a good approximation to the problem. Moreover, when the wave number k is increasing, for the same level of accuracy the number of grid points must increase faster than k increases, in order to avoid pollution effect (Babuska and Sauter, 1997). This growth depends on the discretization and becomes less important when the precision of the method increases in the case of finite element methods. For example, for piecewise linear (P1) finite element approximation on simplicial elements of diameter h , we require that h behaves as $\mathcal{O}(k^{-3/2})$ whereas for piecewise cubic (P3) finite elements the criteria relaxes and h should decrease as $\mathcal{O}(k^{-7/6})$. To find the best compromise between accuracy and computational efficiency, a common practice in finite-element simulation is to consider a fixed number of points G per wavelength $\lambda = 2\pi k^{-1}$ (i.e., $G \approx \lambda/h$), or simply taking h decreasing as $\mathcal{O}(k^{-1})$. In practice, this is implemented by using h -adaptive unstructured meshing. In this study, we wish to use constant G with $G = 4$ to be consistent with the theoretical half-wavelength resolution of FWI (Virieux and Operto, 2009). Indeed, the fill-in of the matrix increases as the polynomial degree increases. Therefore, a careful accuracy analysis of the FEFD method should be performed to find the smallest polynomial order providing the desired accuracy for $G = 4$ as discussed in the *Dispersion analysis* section.

2.3.2 Finite-difference frequency-domain (FDFD) discretization

The second discretization scheme relies on the 27-point finite-difference stencil on regular Cartesian grid (constant grid interval h) in which compact 2^{nd} -order accurate stencils minimize the numerical bandwidth and maximizes the sparsity of A while reaching a high-order accuracy by mixing consistent mass and stiffness matrices on different (rotated) coordinate systems (Aghamiry et al., 2021; Brossier et al., 2010; Chen et al., 2012; Operto et al., 2014, 2007). These sparsity and compactness properties are useful to minimize the matrix fill-in induced by a sparse direct solver, which is used to solve the local problems in each subdomain of the preconditioner. The stiffness and consistent mass matrices are weighted by coefficients that are computed by least-squares minimization of the numerical dispersion and anisotropy. Classical implementation minimizes the dispersion jointly for a finite number of G , which means that the same coefficients are used in each row of the matrix A (Brossier et al., 2010; Chen et al., 2012; Operto et al., 2014, 2007). Alternatively, G can be first tabulated such that it covers the range of values found in the earth and the corresponding table of weighting coefficients are estimated once and for all by minimizing the phase-velocity numerical dispersion for each value of G treated separately (the weights become functions of G). Then, for a simulation at a given frequency f , each row of the matrix A (which is tied to a given grid point) is built by picking in the table the weights corresponding to the local $G = \lambda/h$ hence leading to the λ -adaptive 27-point stencil (Aghamiry et al., 2021). We will use the latter in this study since Aghamiry et al. (2021) showed that the λ -adaptivity increases the accuracy of the 27-point stencil quite significantly in heterogeneous media, in particular in presence of sharp contrast. Since the grid is regular, we will set h such that $h = \lambda_{min}/4$ according to the FWI resolution criterion.

In summary, the discretization rules used in FEFD and FDFD methods rely on two different paradigms. The discretization of the finite-element method is performed such that G is roughly constant by adapting the size of the elements to the local wavelengths (h -adaptivity). By doing so, the accuracy of the numerical scheme is supposed to be similar all over the mesh. The discretization of the finite-difference method relies on a constant h and hence G varies from one grid point to the next. However, we locally adapt the weights of the stencil such that its accuracy is as uniform as possible as a function of G .

2.3.3 Dispersion analysis of FEFD and FDFD methods

The results of the numerical dispersion analysis of the finite-element and finite-difference methods as developed by (Ainsworth and Wajid, 2010) and Aghamiry et al. (2021), respectively, are shown in Fig. 1. The figure shows the

normalized numerical phase velocity against $1/G$ for plane waves in homogeneous media. The dispersion curve for the finite-element method suggests that the accuracy of P2 elements is insufficient for $G = 4$, while a good accuracy is expected for P3 elements over a wide range of G (Fig. 1a). The dispersion curves of the λ -adaptive FDFD stencil suggests that its accuracy in homogeneous media is similar to that of the P3-FE stencil (Fig. 1b).

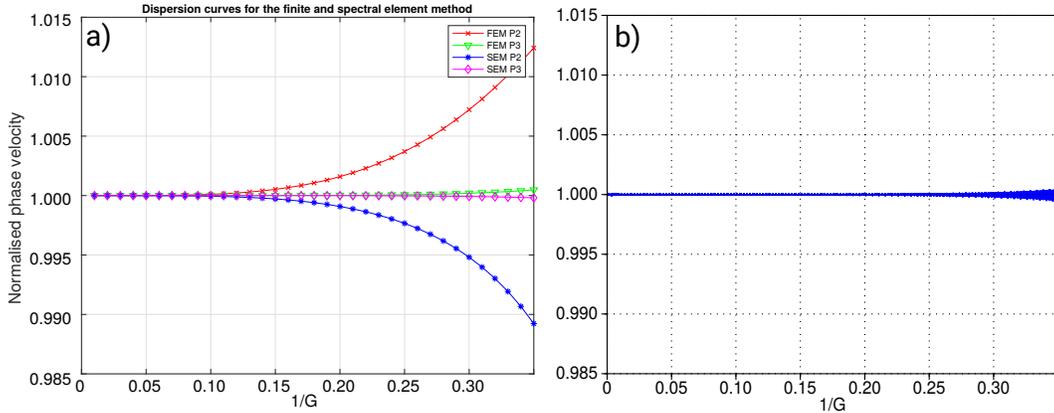


Figure 1: Dispersion curves for (a) P2 and P3 spectral and classical finite elements, (b) the 27 FD scheme with adaptive coefficients. The normalized phase velocity is the ratio between the numerical phase velocity and the wavespeed.

3 Numerical results

3.1 Benchmarks and experimental design

3.1.1 Benchmarks

We validate the FEFD/FDFD ORAS solver against four benchmarks (Table 1). We limit ourselves to isotropic acoustic media with a constant density equal to 1. However, the conclusions of these tests should apply to visco acoustic media with heterogeneous density. All of the λ -adaptive FDFD simulations are performed with a discretization rule of four grid point per minimum wavelength, while the FEFD simulations are performed on h -adaptive tetrahedral mesh with four grid point per local wavelength, unless stated otherwise. The unstructured tetrahedral meshes are generated with the Mmg re-meshing software (Dapogny et al., 2014).

Homogeneous and linear velocity models

The first two benchmarks involve an infinite homogeneous medium and an infinite laterally-homogeneous medium where the velocity linearly increases in the y direction ($c(x, y, z) = c_0 + \alpha \times y$), referred to as gradient model. The size of both models is $10 \text{ km} \times 20 \text{ km} \times 10 \text{ km}$ and the frequency is 7.5 Hz. The wavespeed in the homogeneous medium is 1500 m/s. The grid interval in the Cartesian grid is 50 m, leading to $G=4$ for both FEFD and FDFD simulations (Table 1). In the gradient model, the wavespeed increases from 1500 m/s at $y=0 \text{ km}$ to 8500 m/s at $y=20 \text{ km}$ hence covering the wavespeeds encountered in the earth's crust and upper mantle (the velocity gradient is 0.35 s^{-1}). The grid interval for the FDFD simulation is 50 m. Accordingly, G increases from 4 to ~ 23 , while an h -adaptive mesh is used to perform FEFD simulation (Table 1).

The last two benchmarks involve the complex SEG/EAGE overthrust and GO_3D_OBS geomodels.

SEG/EAGE overthrust model

The overthrust model of dimensions $20 \text{ km} \times 20 \text{ km} \times 4.65 \text{ km}$ is representative of the exploration geophysics scale, where G typically ranges between 4 and 12 in a regular Cartesian grid. It represents a complex thrust sedimentary succession on top of a structurally decoupled extensional and rift basement block (Aminzadeh et al., 1997) (Figure 2a). A complex weathered zone also characterizes it in the near-surface with sharp lateral velocity variations and several sand channels. The velocities range between 2179 m/s at the surface and 6000 m/s at the basement. The frequency is 10 Hz. For the FDFD simulation, we re-mesh the original model with a grid interval of

Table 1: Main specifications of the four benchmark models. c_m, c_M : Minimum and maximum wavespeeds. f : Frequency. $\lambda_{min}, \lambda_{max}$: Minimum and maximum wavelength. G_{min}, G_{max} : Smallest and highest G . N_λ : Maximum number of propagated wavelengths.

Models	$c_m(m/s)$	$c_M(m/s)$	$f(Hz)$	$\lambda_m(m)$	$\lambda_M(m)$	G_m	G_M	N_λ
Homogeneous	1500	1500	7.5	375	375	4	4	40
Linear	1500	8500	7.5	375	1133	4	22.7	29
Overthrust	2179	6000	10	218	600	4.4	12	50
GO_3D_OBS	1500	8639.1	3.75	400	2303.8	4	23	255

50 m and 8 grid points in the PMLs along each face of the 3D grid of dimensions $417 \times 417 \times 109$ (Table 1). The unstructured tetrahedral mesh used for the FEFD simulation is illustrated in Figure 3(b-c).

GO_3D_OBS crustal geomodel

The 3D GO_3D_OBS crustal geomodel has been designed to assess seismic imaging techniques for deep crustal exploration (Górszczyk and Operto, 2021). It covers a continental margin at a regional scale, where the wider range of wavespeeds found in the crust and upper mantle makes G to range between 4 and 25 in a regular Cartesian grid. It embeds the main structural factors that characterize subduction zones and has been inspired by the FWI case study performed in the eastern Nankai trough by Górszczyk et al. (2017). We select a target of the model of dimensions $20 \text{ km} \times 102 \text{ km} \times 28.4 \text{ km}$ (Figure 2b). The simulations are performed for a frequency of 3.75 Hz except for the strong scaling analysis presented at the end of the section where a frequency as high as 10 Hz is processed. For the FDFD simulation, we discretize this target with a grid interval of 100 m with eight grid points in the PMLs, leading to a finite-difference grid of dimensions $217 \times 1037 \times 300$, hence 67.5 million degrees of freedom. For the FEFD simulation, we build an unstructured mesh that complies with the bathymetry. Below the bathymetry, the size of the elements is set according to the local wavelength such that $G \approx 4$ (Figure 3).

3.1.2 Experimental setup

For the first two benchmarks, the accuracy of the FEFD and FDFD solutions are assessed against analytical solutions. The reader is referred to Kuvshinov and Mulder (2006) for the analytical solution of the Helmholtz equation in a velocity gradient model.

For the last two benchmarks, the FEFD and FDFD solutions are assessed against a reference solution computed with the highly-accurate Convergent Born Series (CBS) method (Osna-brugge et al., 2016). Compared to finite-difference or finite-element methods that rely on discretization of differential operators, the CBS method belongs to another class of methods referred to as volume integral methods based upon the Green’s function theorem. More specifically, the solution of the Helmholtz equation in heterogeneous media is inferred from Fourier-domain analytic expression of the Green’s functions in homogeneous media by solving the scattered-field wave equation with material contrasts of arbitrary scattering strength starting from the zero-order term of the Born series and iteratively recovering the higher-order scattering terms. Therefore, we expect to achieve an accuracy of the CBS solutions at the level of machine precision while Osna-brugge et al. (2016) conclude that the CBS method is nine orders more accurate than the pseudo-spectral time-domain method. For all the extracted wavefields using CBS, the stopping criterion of iteration is set according to a backward error of 10^{-12} .

In addition to visual comparison, we assess the accuracy of the FEFD and FDFD stencils by computing the ℓ_1 norm of the difference between the CBS and FEFD/FDFD wavefields (field Err in Table 5) according to the formula

$$\text{Err} = \frac{|\mathbf{WR}(\mathbf{u}_{cbs} - \mathbf{u}_{fe/fd})|}{|\mathbf{WR}(\mathbf{u}_{cbs})|} + \frac{|\mathbf{WI}(\mathbf{u}_{cbs} - \mathbf{u}_{fe/fd})|}{|\mathbf{WI}(\mathbf{u}_{cbs})|}, \quad (3.1)$$

where \mathbf{u}_{cbs} and $\mathbf{u}_{fe/fd}$ denotes the CBS and FEFD/FDFD wavefields, respectively, \mathcal{R} and \mathcal{I} denote the real and imaginary parts of a complex number, respectively, and \mathbf{W} is a diagonal matrix applying a linear gain with distance from the source to the wavefield for amplitude balancing. We also mute the contribution of the nearest points to the source to avoid introducing some bias generated by the different implementation of a point source in the FEFD and FDFD schemes.

The source is a point source located on a grid point where the temporal source signature is a delta function: $s(\mathbf{x}, \omega) = \delta(\mathbf{x} - \mathbf{x}_s)$, where δ denotes the delta function and \mathbf{x}_s denotes the source coordinates. We implement PMLs along each face of the grid.

We perform the simulation on the Occigen supercomputer of CINES (<https://www.cines.fr>). The computer nodes contain two Haswell E5-2690V3@2.6 GHz processors with 128 Giga bytes of shared memory and 12 cores per processor. The high-speed network is Infiniband FDR 56 Gbit/s.

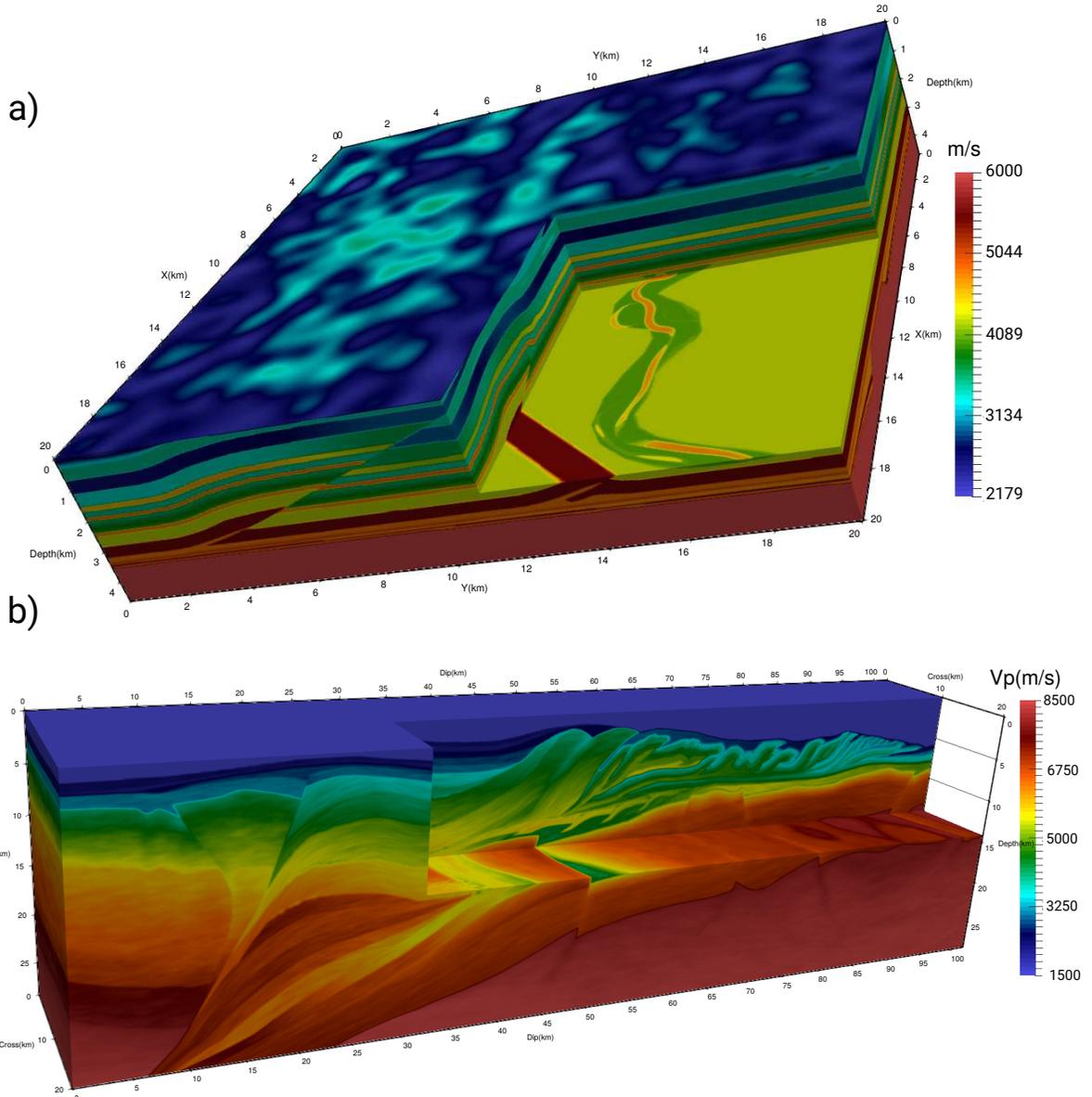


Figure 2: The two realistic benchmarks used to assess the FEFD/FDFD ORAS-GMRES solver. (a) The 3D SEG/EAGE overthrust model (Aminzadeh et al., 1997), representative of the exploration-geophysics scale. (b) Target of the regional GO_3D_OBS model representing the crust of a subduction zone (Górszczyk and Operto, 2021).

3.2 Tuning of the ORAS solver

Before showing the results of the ORAS solver for the four benchmarks when the subsurface model is discretized with the FEFD and FDFD methods, we tune key ingredients of the ORAS solver such that the best compromise between accuracy and computational efficiency is reached. These ingredients include the stopping criterion of iterations, arithmetic precision (single versus double precision), the solver used to factorize local matrices and the orthogonalization scheme of the Krylov basis. We perform this tuning with the 3D GO_3D_OBS benchmark and the FDFD scheme.

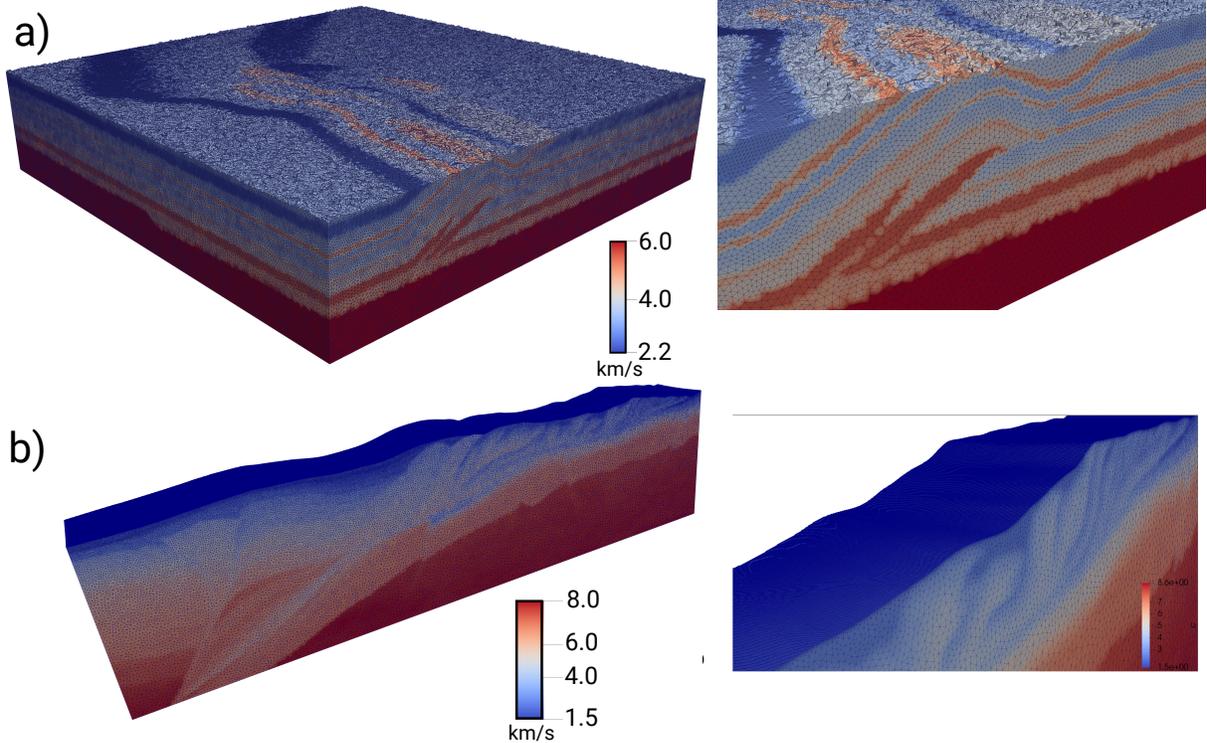


Figure 3: Tetrahedral meshing of (a) the 3D SEG/EAGE overthrust and (b) GO_3D_OBS models. The right panels show a zoom.

3.2.1 Stopping criterion of GMRES iterations

The simulations are performed in single precision on 660 cores. As stopping criterion of iteration, we use the relative backward error defined by

$$\varepsilon_{gmres} = \frac{\|Au - f\|_2^2}{\|f\|_2^2}. \quad (3.2)$$

We test $\varepsilon_{gmres} = 10^{-2}$, 10^{-3} , 10^{-4} and 10^{-5} (Table 2 and Figures 4 and 5). The number of iterations required to satisfy this criterion is 13, 32, 45 and 61, respectively. The errors of the FDFD simulations performed with $\varepsilon_{gmres} = 10^{-4}$ and $\varepsilon_{gmres} = 10^{-5}$ are similar and hence the former value is used in the remaining of the paper. Using $\varepsilon_{gmres} = 10^{-4}$ instead of $\varepsilon_{gmres} = 10^{-5}$ reduces the number of iterations from 61 to 45 and the computational time by 27%.

Table 2: Cost of a simulation in the GO_3D_OBS model as a function of the GMRES backward error criterion. *#it*: number of iterations. $T_s(s)$: Time elapsed in GMRES to compute the solution. *Err*: Error between the CBS and FDFD wavefields as defined by equation 3.1.

Stopping criterion of GMRES iteration			
ε_{gmres}	<i>#it</i>	$T_s(s)$	Err
10^{-2}	13	3.5	0.8663
10^{-3}	32	8.2	0.2567
10^{-4}	45	11.5	0.2096
10^{-5}	61	15.9	0.2093

3.2.2 Arithmetic precision

The results shown in the section *FEFD versus FDFD discretization: accuracy and computational cost* are performed in single precision. We check here that a similar accuracy is reached when we perform simulation in single and

double precisions. Compared to the previous section, the simulations are performed on 1344 cores because the double precision simulation runs out of memory on 660 cores. Single precision allows for the solution time to be decreased by a factor two compared to double precision simulations (fourth column of Table 3) while the error between the FDFD and the CBS simulations is the same, hence validating the single precision option (last column of Table 3).

Table 3: Footprint of arithmetic precision on computational cost and accuracy. #it: Number of iterations. $T_f(s)$: Elapsed time for local LU factorizations. $T_s(s)$: Elapsed time for GMRES. *Err*: Error between the CBS and FDFD wavefields as defined by equation 3.1.

Arithmetic precision				
Precision	#it	$T_f(s)$	$T_s(s)$	Err
Double	53	7.9	14.0	0.2097
Single	53	4.4	7.1	0.2096

3.2.3 Solution of the local problems and multiple right-hand sides

In this section, we assess two direct solvers and orthogonalization schemes of the Krylov basis. We test the Intel MKL PARDISO solver (Alappat et al., 2020; Böllhofer et al., 2019, 2020) and the MUMPS solver (Amestoy et al., 2018, 2019b; MUMPS team, 2021) to factorize the local matrices tied to each subdomain. When using MUMPS, we test both the full-rank (FR) and the block low-rank (BLR) version of the solver, referred to as MUMPS_{FR} and MUMPS_{BLR}, respectively (Amestoy et al., 2016, 2019a). In the latter case, we use as compression threshold $\varepsilon_{BLR} = 10^{-3}$. We also compare the performances when a classical Gram-Schmidt (CGS) and a modified Gram-Schmidt (MGS) algorithms are used for the orthogonalization of the Krylov basis (Saad, 2003b) and we perform the computations in single and double precisions. Moreover, we perform the simulation with one RHS and 130 RHSs, the latter mimicking a sparse seabed node acquisition. For multi-RHS simulation, we test a pseudo-block Krylov method, a block Krylov method and block Krylov with recycling by processing the 130 RHSs in group of 20 (Jolivet and Tournier, 2016; Parks et al., 2006). The results are outlined in Table 4. The smaller computational time is reached with MUMPS_{BLR} although the BLR approximation generates some iteration overheads. The CGS orthogonalization provides substantial computational saving compared to the MGS counterpart in case of multi-RHS simulation. These simulations further confirm that single precision arithmetic decreases by a factor almost two the cost of the simulation compared to double precision counterpart. Concerning multi-RHS simulation, the best performance is obtained with the pseudo-block Krylov method with a speedup of $7.2 \times 130/256.8 = 3.64$ compared to the case where each RHS is processed separately (last row of Table 4). This suggests that the most basic pseudo-block Krylov remains the most efficient one for seismic imaging applications, which involve moderate material contrast and simple wave propagation phenomena. We check that for groups of RHS larger than 20 no improvement is achieved during multi-RHS simulation.

Table 4: *Solver*: Intel MKL Pardiso versus MUMPS_{FR/BLR}. *arith*: single versus double precision. *ortho*: CGS versus MGS orthogonalization. #it: Number of iterations. $T_f(s)$: Elapsed time for local LU factorizations. $T_s(s)$: Elapsed time for all GMRES iterations for one RHS and 130 RHSs processed with a pseudo-block Krylov method. $T_{tot}(s) = T_f(s) + T_s(s)$: Total elapsed time for the simulation.

Solver	arith.	ortho.	$T_f(s)$	1 RHS			130 RHS		
				#it	$T_s(s)$	$T_{tot}(s)$	#it	$T_s(s)$	$T_{tot}(s)$
PARDISO	double	MGS	12.1	48	16.5	28.6	50	724.4	736.5
PARDISO	double	CGS	12.1	48	16.7	28.8	50	575.5	587.6
MUMPS _{FR}	double	MGS	11.3	48	17.2	28.5	50	654.9	666.2
MUMPS _{FR}	double	CGS	11.3	48	16.9	28.2	50	482.2	493.5
PARDISO	single	MGS	6.5	48	9.9	16.4	50	403.6	410.1
PARDISO	single	MGS	6.5	48	9.6	16.1	50	334.8	341.3
MUMPS _{FR}	single	MGS	6.2	48	8.9	15.1	50	345.8	352.0
MUMPS _{FR}	single	CGS	6.2	48	8.9	15.1	50	275.3	281.5
MUMPS _{BLR}	single	CGS	4.9	53	7.2	12.1	55	256.8	261.7

3.3 FEFD versus FDFD discretization: accuracy and computational cost

According to the results of the previous section, we assess the FDFD and FEFD schemes with $\varepsilon_{gmres} = 10^{-4}$. Computations are performed in single precision. The local solver relies on MUMPS_{BLR} with $\varepsilon_{BLR} = 10^{-3}$ while a CGS

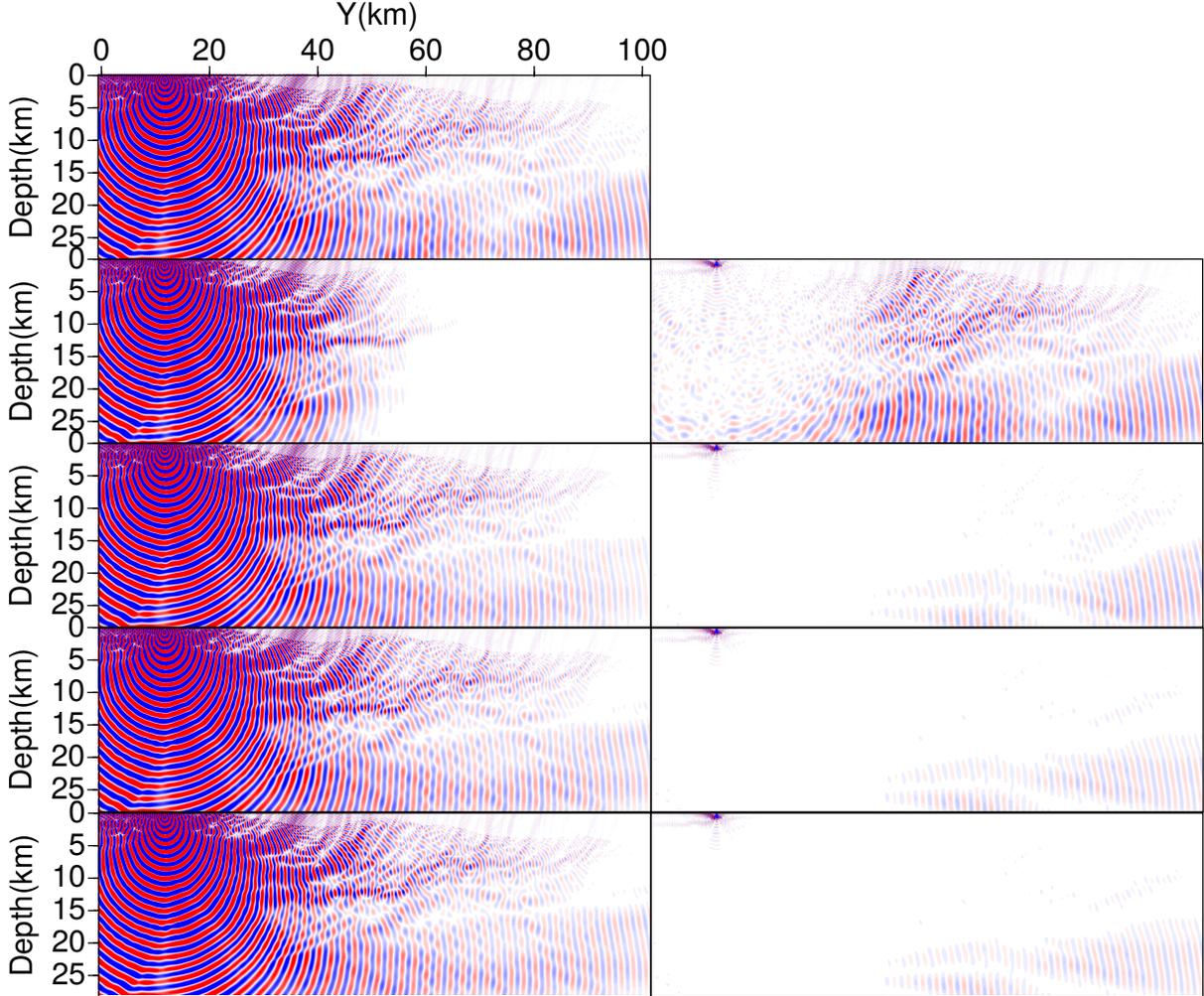


Figure 4: Setting the stopping criterion of GMRES iterations. Frequency is 3.75 Hz. In the caption of this figure, the panels are labeled with matrix notations: $a_{i,j}$ where i and j denote the row and column indexes, respectively. $a_{1,1}$: Section of the CBS wavefield (real part). $a_{2:5,1}$: From top to bottom, FDFD wavefields (real part) for $\varepsilon_{gmres} = 10^{-2}, 10^{-3}, 10^{-4}, 10^{-5}$. $a_{2:5,2}$: Differences between CBS and FDFD wavefields.

orthogonalization is used to build the Krylov basis. For FDFD simulation, we use the one-level ORAS preconditioner, while the two-level counterpart is used for the FEFD simulations. PML conditions are used between the subdomains in the FDFD simulation with an overlap of 3, while a Robin condition is used for the FEFD simulation. The results of the numerical tests for the four benchmarks are outlined in Table 5.

3.3.1 Homogeneous medium

The source is positioned at (5 km, 2 km, 5 km). We perform the FEFD and FDFD simulations with 2400 and 396 cores, respectively. The scalar time (the parallel run time \times the number of cores, where the parallel run time is the sum of the elapsed time required to perform LU factorization in the subdomains and the GMRES iterations) is 91.4 hours and 1.2 hour for the FEFD and FDFD simulations, respectively (Table 5). The analytical and the numerical FEFD and FDFD wavefields in the (x,y) plane across the source and the difference between the analytical and the numerical solutions are shown in Figure 6 as well as the direct comparison between the analytical and numerical wavefields along the y -profile. The results show a slightly better accuracy of the FDFD solution relative to the FEFD counterpart (Table 5). However, the two solutions are quite close, hence supporting the conclusions of the dispersion analysis (Fig 1).

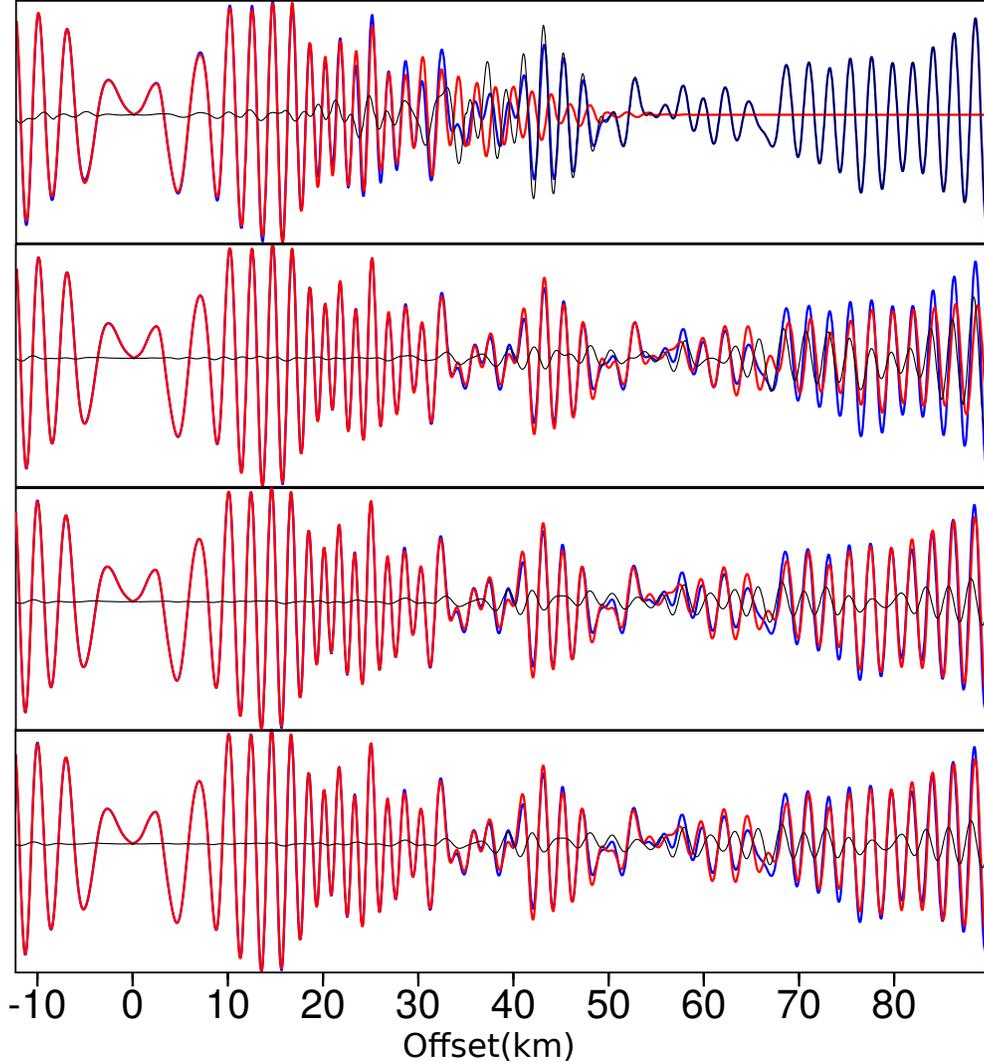


Figure 5: Stopping criterion of GMRES iterations. From top to bottom, direct comparison along a y-profile at 16 km depth ($x=10.5$ km) between CBS wavefield (blue) and FDFD wavefields (red) computed with $\varepsilon_{gmres} = 10^{-2}, 10^{-3}, 10^{-4}, 10^{-5}$ (Figure 4). The differences are plotted in black. A linear gain with offset is applied for amplitude balancing.

3.3.2 Velocity gradient model

Our motivation behind this test, beyond accuracy issue, is to gain a first insight whether the h -adaptivity implemented in the FEFD method can balance the highest number of dofs per element relative to the FDFD method in term of computational cost when a wide range of wavelengths are involved in the simulation. The results are shown in Figure 7 with the same showing as for the homogeneous case. As for the homogeneous case, the FDFD method remains marginally more accurate and remains one order of magnitude less expensive than the FEFD counterpart (Check field T_{hc} in Table 5). Figure 8 shows the results of the FEFD simulation in a regular mesh where the number of grid point per minimum wavelength is set to four. The accuracy is slightly improved at the expense of the computational cost, which increases by a factor 4.4 (Table 5).

3.3.3 3D SEG/EAGE overthrust velocity model

We now benchmark the FEFD/FDFD ORAS solver against the heterogeneous overthrust model involving complex compressive tectonics (Table 1). The source is positioned at (2500 m, 2500 m, 500 m). We perform the FEFD and FDFD simulations with 2400 and 396 cores, respectively. The CBS, FEFD and FDFD wavefields are shown in Fig-

Table 5: Results of the four benchmarks obtained with the FEFD and FDFD methods. $\#d$: Number of dofs (including PMLs). $\#cores$: Number of cores. $\#it$: Number of GMRES iterations (for FEFD, average number of inner iterations to solve the coarse problem in parentheses). $T_f(s)$: Elapsed time for local LU factorizations. $T_s(s)$: Elapsed time for all GMRES iterations. $T_{tot}(s) = T_f(s) + T_s(s)$: Total elapsed time for the simulation. $T_{hc}(s) = \#c \times T_{tot}$: Scalar time. ERR: Solution error as defined by equation 3.1. The specifications of the four benchmarks in term of size and frequency are outlined in Table 5.

FEFD method								
Benchmark	$\#d(M)$	$\#cores$	$\#it$	$T_f(s)$	$T_s(s)$	$T_{tot}(s)$	$T_{hc}(h)$	Err
Homogeneous	526.2	2400	31(13)	64.1	73.0	137.1	91.4	0.0891
Gradient	147.1	2400	5(14)	12.4	6.1	18.5	12.3	0.0485
Gradient ^(c)	526.2	2400	8(16)	59.8	22.3	82.1	54.7	0.0119
Overthrust	157.1	2400	6(17)	13.5	6.7	20.2	13.5	0.5215
Overthrust ^(c)	516.5	2400	6(20)	57.6	19.5	77.1	51.4	0.5156
GO_3D_OBS	597.7	2400	10(14)	62.3	26.1	88.4	58.9	0.4212
FDFD method								
Benchmark	$\#d(M)$	$\#c$	$\#it$	$T_f(s)$	$T_s(s)$	$T_{tot}(s)$	$T_{hc}(h)$	Err
Homogeneous	19.6	396	41	5.4	5.7	11.1	1.2	0.0317
Gradient	23.5	396	42	5.7	6.4	12.1	1.3	0.0114
Overthrust	19.1	363	33	5.0	4.5	9.5	0.96	0.1188
GO_3D_OBS	67.5	660	45	10.8	11.5	22.3	4.1	0.2096

ure 9. Direct comparison between these wavefields along selected profiles are shown in Fig. 10. In Figure 9, we show two depth slices of the wavefields at 500 m depth across the source and 3500 m depth just above the basement and two vertical sections at $y=2500$ m across the source and $y=15000$ m. In Figure 10, we show a direct comparison between the CBS and the FEFD/FDFD wavefields along four horizontal profiles at $(x,z)=(2000$ m, 500 m), $(x,z)=(15000$ m, 3500 m), $(y,z)=(2500$ m, 500 m), $(y,z)=(15000$ m, 3500 m). For this benchmark, the FDFD stencil clearly outperforms the FEFD counterpart in terms of accuracy. The scalar time of the FDFD simulation is also 14 times smaller than that of the h -adaptive FEFD simulation (Table 5). A FEFD simulation on a regular mesh using four points per minimum wavelength confirms that the adaptive meshing doesn't impact the accuracy of the simulation (Figures 11 and 12 and Table 5). The computational saving provided by the adaptive mesh is 3.8 for this benchmark (Table 5).

3.3.4 Deep crustal GO_3D_OBS model

We now move to the GO_3D_OBS model. The source is positioned on the sea bottom at (10500 m, 12300 m, 900 m). It is reminded that the simulated frequency is 3.75 Hz (Table 1). We perform the FEFD and FDFD simulations using 2400 and 660 cores, respectively. In Figure 14, a direct comparison between the CBS and FEFD/FDFD wavefields is performed along with three horizontal profiles in the y direction running across the source position and cross-cutting the accretionary wedge at 6 km depth, the subduction megathrust at 10 km depth and the Moho at 16 km depth. For this benchmark too, the FDFD solution outperforms the FEFD counterpart in terms of accuracy. However, the accuracy ratio between FEFD and FDFD (Err_{fe}/Err_{fd}) is smaller in the GO_3D_OBS model compared to the overthrust model (2 against 4.4). This probably result from the fact that the structures have a larger scale relatively to the wavelength in the GO_3D_OBS model relative to the overthrust counterpart. The computational cost of the FDFD scheme remains one order of magnitude higher than that of the FDFD counterpart.

3.4 Weak and strong scalability analysis

We conclude the numerical section with a weak and strong scaling analysis of the ORAS solver. We perform these analyzes with the GO_3D_OBS model and the FDFD method. The finite-difference grid is discretized with four grid points per minimum wavelength. According to the conclusions of the previous section, we use single precision arithmetic, we perform local factorizations with MUMPS_{BLR} using $\epsilon_{BLR} = 10^{-3}$ and CGS is used for the orthogonalization of the Krylov basis.

3.4.1 Weak scaling

For the weak scaling analysis, we use frequencies ranging between 2.5 Hz and 10 Hz (Table 6). The number of dofs ($\#dof$) in the Cartesian grid ranges between 21.4 and 1,160.6 million. Each simulation is performed such that the ratio $\#dof/\#cores$ is roughly constant where $\#cores$ denotes the number of cores. We define the weak-scaling parallel

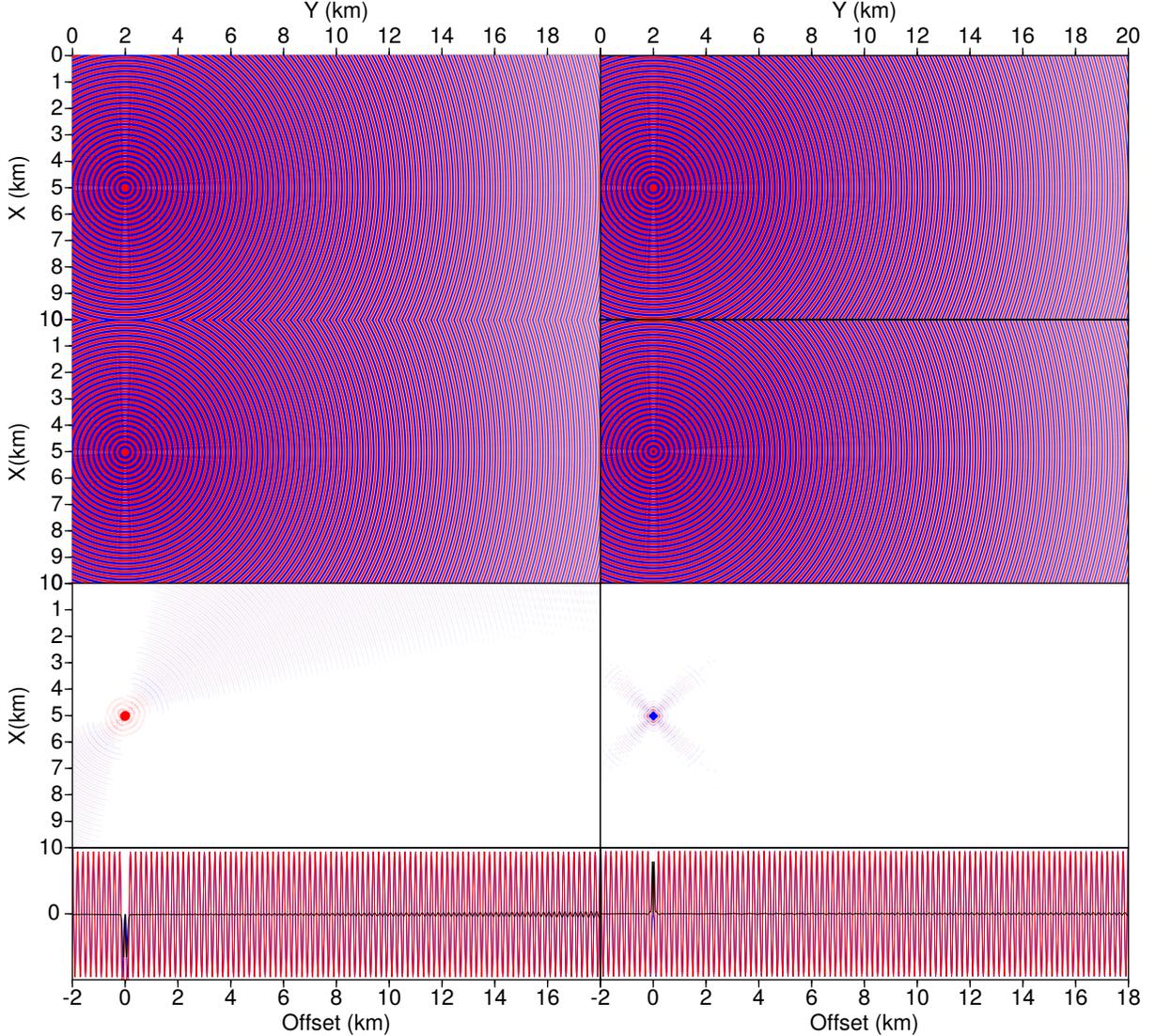


Figure 6: FE and FD monochromatic wavefields in infinite homogeneous medium. $a_{1,1:2}$: Analytical solution (real part). $a_{2,1}$: FEFD wavefield (real part). $a_{2,2}$: FDFD solution (real part). $a_{3,1}$: Difference between Analytical solution and FEFD wavefield. $a_{3,2}$: Same as $a_{3,1}$ for the FDFD wavefield. $a_{4,1}$: Direct comparison between analytical (blue) and FEFD solution (red) along a Y profile cross-cutting the source position. Difference is plot in black. $a_{4,2}$: Same as $a_{4,1}$ for the FDFD wavefield.

efficiency as

$$E_w = \frac{[T_{tot}^{(ref)} \times \#cores^{(ref)} / \#dof^{(ref)}]}{[T_{tot} \times \#cores / \#dof]}, \quad (3.3)$$

where the subscript ref refers to the smallest tackled problem.

The results are highlighted in Table 6. The number of iterations scales linearly with frequency, which is consistent with previous studies based upon multigrid preconditioner with shifted Laplacian (e.g., Plessix, 2017) (Figure 15).

3.4.2 Strong scaling

We perform a strong scaling analysis of the two-level ORAS preconditioner in the finite element case for the 3.75 Hz frequency. The number of cores goes from 2400 to 7200 (Table 7).

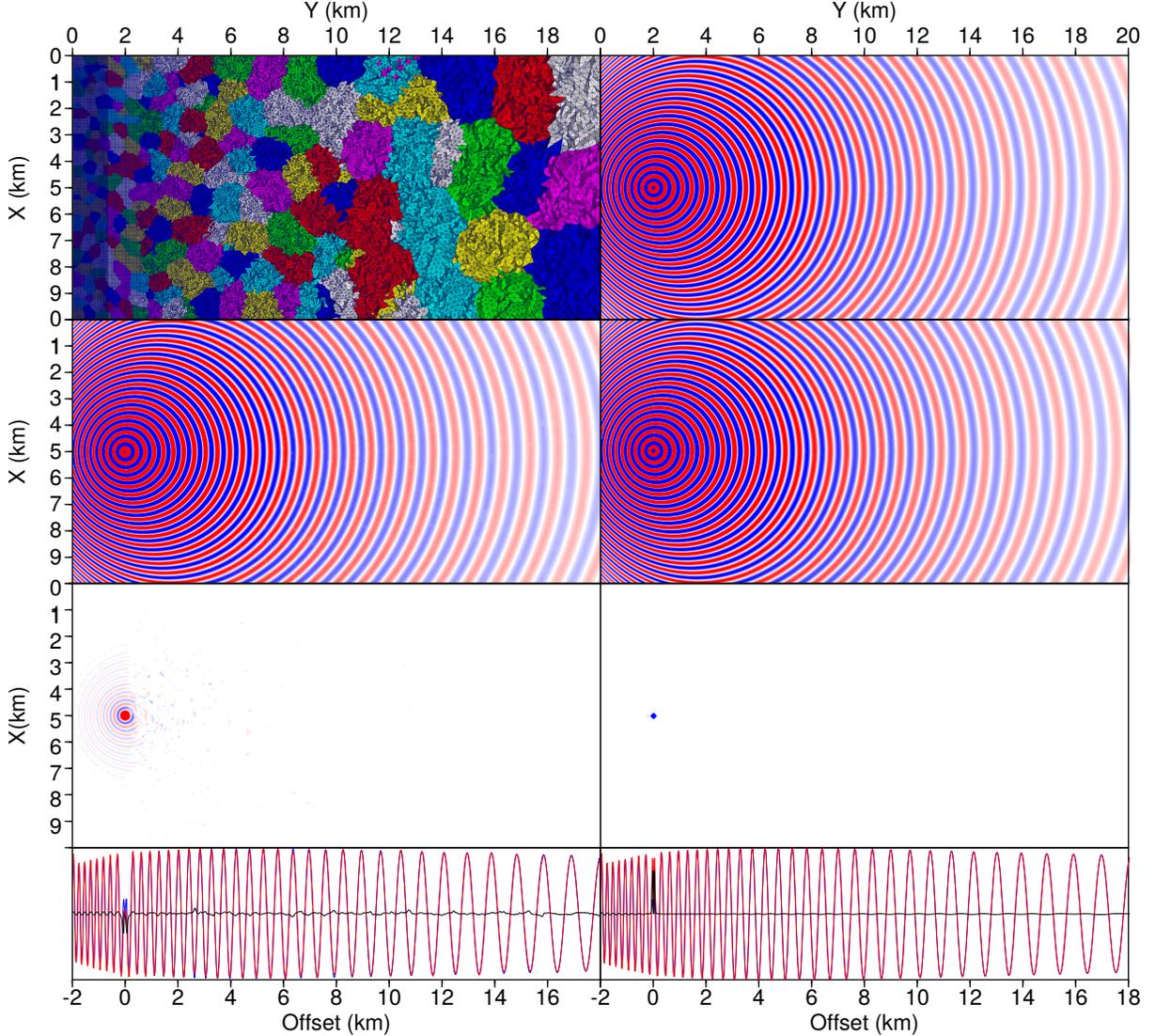


Figure 7: h -adaptive FE and λ -adaptive FD monochromatic wavefields in linear velocity model. $a_{1,1}$: h -adaptive tetrahedral mesh for FDFD simulation. Each subdomain is plotted with a different color. $a_{1,2}$: Analytical solution (real part). $a_{2,1}$: FEFD wavefield (real part). $a_{2,2}$: FDFD wavefield (real part). $a_{3,1}$: Difference between analytical and FEFD solution. $a_{3,2}$: Same as $a_{3,1}$ for the FDFD solution. $a_{4,1}$: Direct comparison between analytical (blue) and FEFD (red) solution along a Y profile cross-cutting the source position. Difference is plot in black. $a_{4,2}$: Same as $a_{4,1}$ for FDFD solution.

The parallel efficiency is defined by

$$E_s = \frac{[T_{tot}^{(ref)} \times \#cores^{(ref)}]}{[T_{tot} \times \#cores]}, \quad (3.4)$$

The results outlined in Table 7 highlight the high scalability of the ORAS solver. Figure 16 also illustrates the scalability of the GMRES solution time for the FDFD and FEFD methods for this test case (results for the FDFD method are extracted from Table 6 for the 3.75 Hz frequency). We can see that the number of GMRES iterations for the FEFD method is more stable thanks to the coarse grid component of the preconditioner.

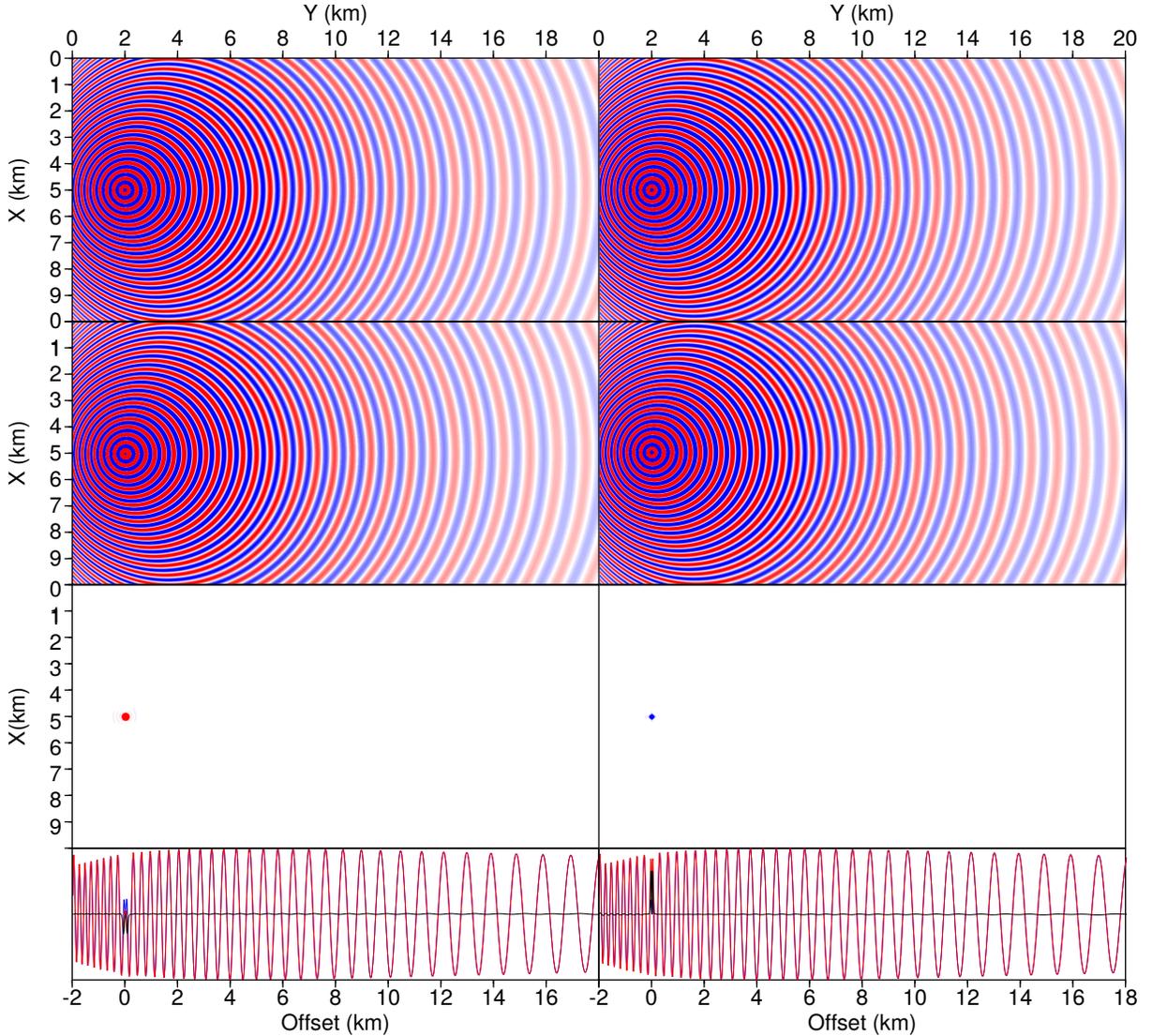


Figure 8: Same as Figure 7 when FEFD is performed on a regular mesh with a discretization rule of four grid point per minimum wavelength.

4 Discussion

We propose a forward engine to perform 3D frequency-domain FWI of datasets collected by long-offset seabed acquisitions implemented with sparse layout of ocean bottom nodes. In particular, we design carefully the experimental setup allowing us to achieve the best compromise between computational efficiency and simulation accuracy, the latter being quantitatively controlled by the solutions of the CBS method (Osnabrugge et al., 2016). We also compare two discretization schemes based on the recently-proposed 27-point wavelength-adaptive finite-difference (FDFD) method on regular Cartesian grid (Aghamiry et al., 2021) and an h -adaptive finite-element method (FEFD) on unstructured tetrahedral mesh.

4.1 Which solver for FWI of sparse long-offset data?

The pros and cons of the ORAS solver for FWI applications can be discussed relative to sparse direct solvers and explicit time marching methods. Theoretical complexities of each method for a dense 3D fixed-spread acquisition are basically identical and hence are useless to rate them in a relative sense (Kostin et al., 2019; Nihei and Li, 2007; Plessix, 2007, 2017; Sourbier et al., 2011; Virieux et al., 2009). Instead, periodic assessment of the methods carried

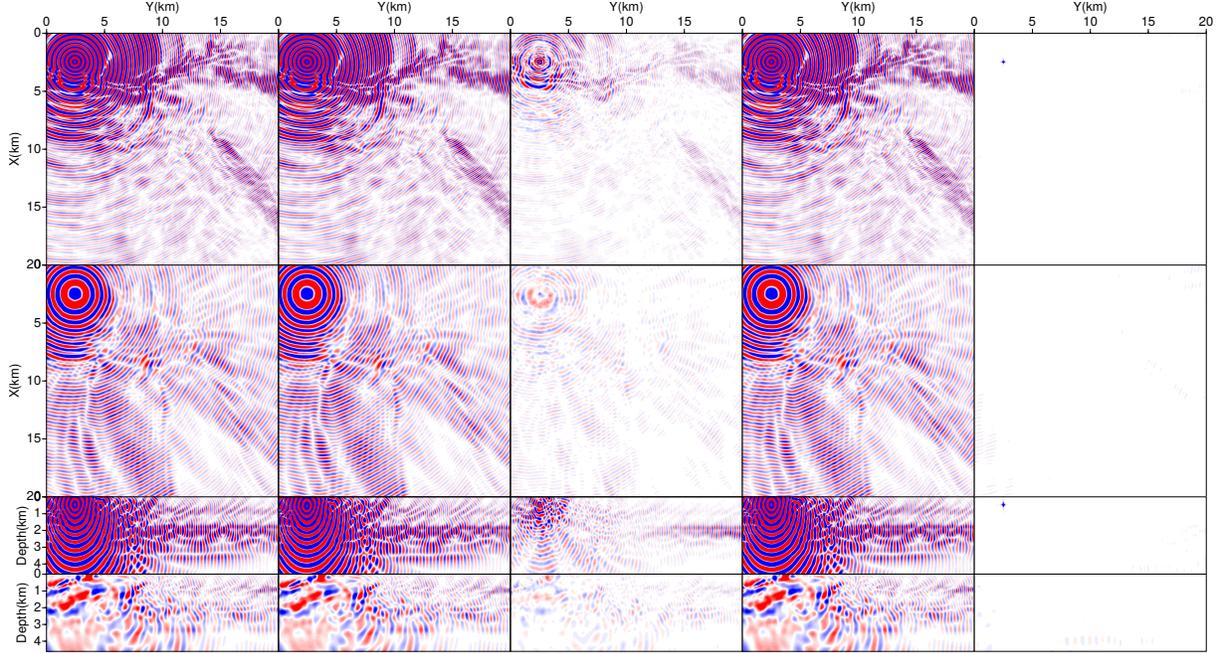


Figure 9: 3D EAGE/SEG overthrust model. Comparison between the wavefields computed with the CBS, FEFD and FDFD methods. The rows shown from top to bottom two depth slices at 500 m depth (across the source) and 2 km depth, and two vertical sections at $x=2.5$ km (across the source) and 15 km. From left to right, the columns show the CBS wavefield, the FEFD wavefield, the differences between the two, the FDFD wavefield and its differences with the CBS wavefield.

Table 6: Weak scaling. The GO_3D_OBS model and the FDFD method are used to perform this analysis. Same nomenclature as that of Table 4. f : Frequency in Hz. ovl : Size of the optimal overlap in terms of computational time.

f (Hz)	#dof(10^6)	#cores	#it	ovl	T_f (s)	T_s (s)	T_{tot} (s)	E_w
2.5	21.4	60	26	3	52.5	24.8	77.3	1
2.5	21.4	168	30	3	13.7	8.9	22.6	1.222
3.75	67.5	360	40	3	21.9	18.6	40.5	1.003
3.75	67.5	660	45	3	10.8	11.5	22.3	0.994
3.75	67.5	1344	53	3	4.4	7.1	11.5	0.947
3.75	67.5	2380	59	3	2.5	4.8	7.3	0.842
5	153.5	875	61	3	20.9	26.0	46.9	0.811
5	153.5	1344	63	3	12.4	18.6	31.0	0.798
7.5	500.1	2450	98	4	26.7	60.8	87.5	0.506
7.5	500.1	4752	108	4	12.2	39.7	51.9	0.439
10	1160.6	9856	142	5	15.1	70.8	85.9	0.297

Table 7: Strong scaling. The GO_3D_OBS model and the FEFD method are used to perform this analysis. Same nomenclature as that of Table 6. #it: Number of GMRES iterations, with average number of inner iterations to solve the coarse problem in parentheses.

f (Hz)	#dof(10^6)	#cores	#it	ovl	T_f (s)	T_s (s)	T_{tot} (s)	E_s
3.75	597.7	2400	10(14)	2	62.3	26.1	88.4	1
3.75	597.7	3600	13(15)	2	38.4	22.1	60.5	0.974
3.75	597.7	4800	11(17)	2	28.4	18.2	46.6	0.948
3.75	597.7	6000	12(17)	2	19.4	16.7	36.1	0.980
3.75	597.7	7200	11(17)	2	18.1	13.6	31.7	0.930

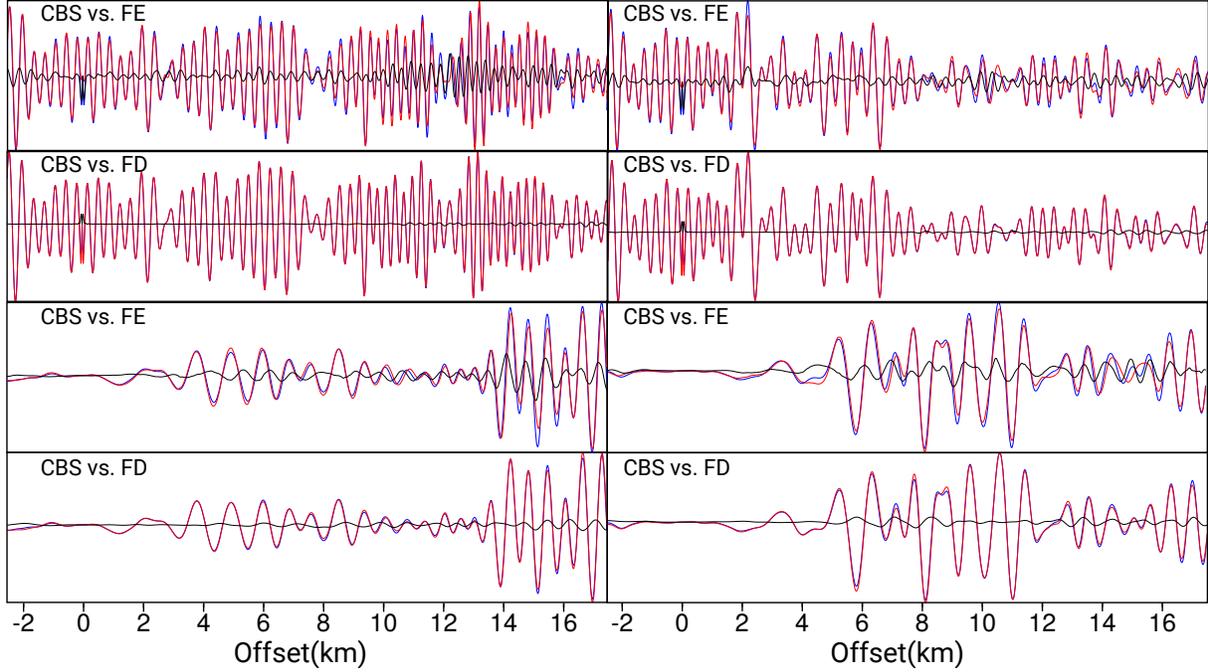


Figure 10: 3D EAGE/SEG overthrust model. Direct comparison between the CBS (blue) and FEFD/FDFD wavefields (red). The differences are plotted in black. $(a_{1,2,1})$: Profile at $(x,z)=(2000 \text{ m}, 500 \text{ m})$. $(a_{1,2,2})$: Profile at $(y,z)=(2500 \text{ m}, 500 \text{ m})$. $(a_{3,4,1})$: Profiles at $(x,z)=(15000 \text{ m}, 3500 \text{ m})$. $(a_{3,4,2})$: Profile at $(y,z)=(15000 \text{ m}, 3500 \text{ m})$. $(a_{1,1:2}, a_{3,1:2})$: FEFD solution. $(a_{2,1:2}, a_{4,1:2})$: FDFD solution.

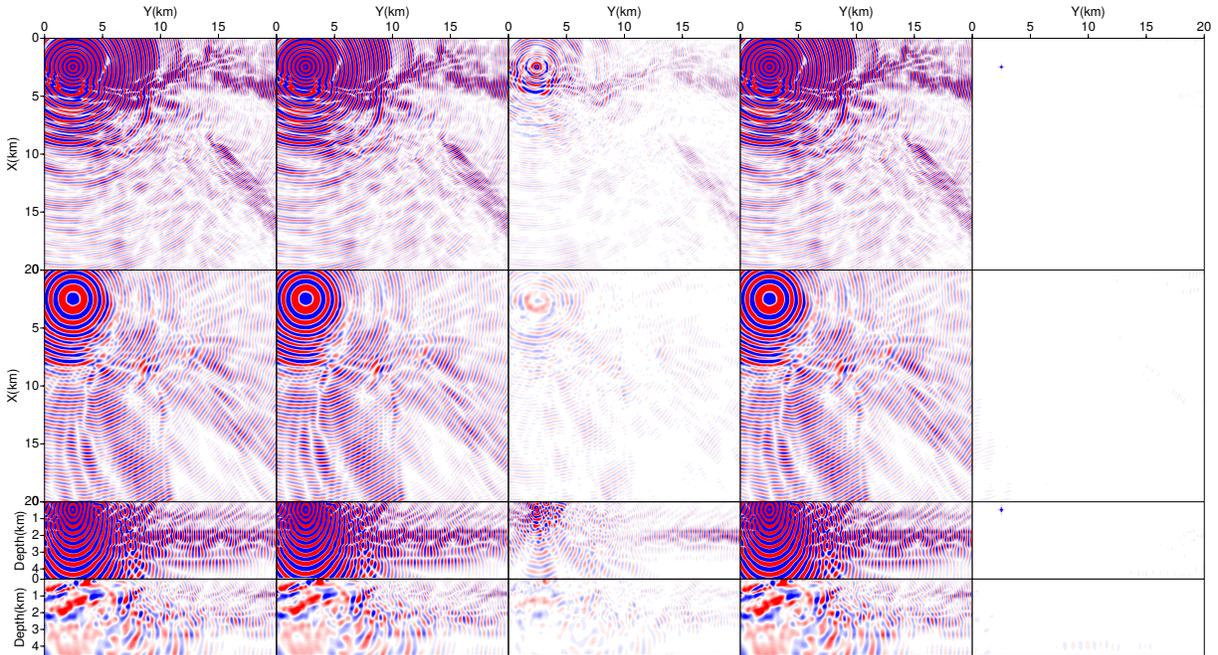


Figure 11: 3D EAGE/SEG overthrust model. Same as Figure 9 but the FEFD simulation is performed on regular mesh with a discretization rule of four grid point per minimum wavelength.

out with realistic numerical experiments that incorporate the latest advances in the field of linear algebra may be more instructive. Such a comparative analysis is presented in Plessix (2017) where a Krylov-subspace iterative solver based

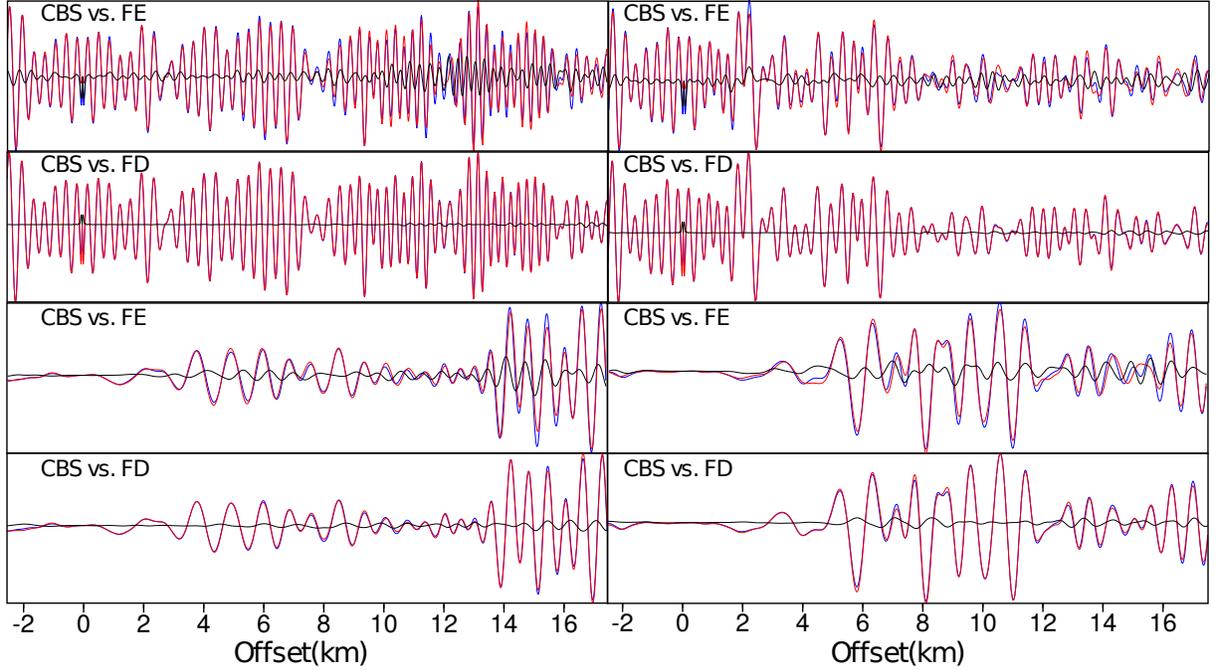


Figure 12: 3D EAGE/SEG overthrust model. Same as Figure 10 but the FEFD simulation is performed on regular mesh with a discretization rule of four grid point per minimum wavelength.

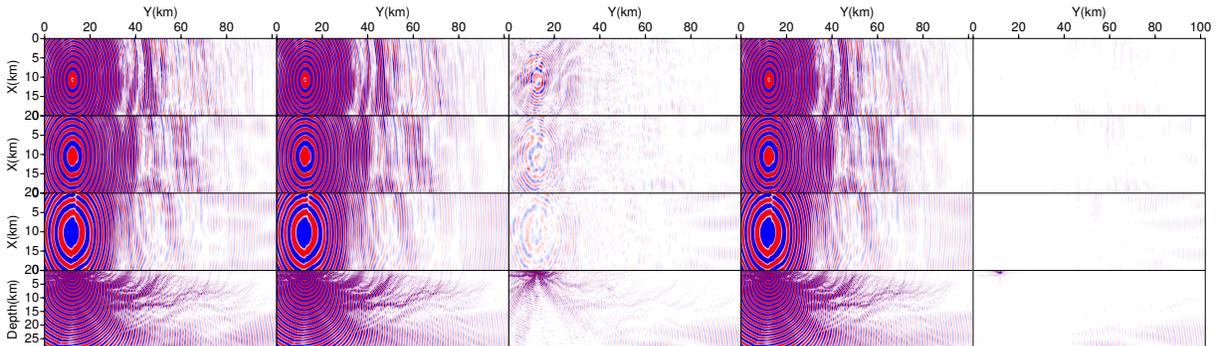


Figure 13: 3D GO_3D_OBS model. Comparison between the wavefields computed with the CBS, FEFD and FDFD methods. The rows show from top to bottom three depth slices at 6 km, 10 km and 16 km depth, and one vertical section at $x=10$ km. From left to right, the columns show the real part of the CBS wavefield, the FEFD wavefield, the differences between the two, the FDFD wavefield and its differences with the CBS wavefield.

upon BI-CGSTAB (Van der Vorst, 1992) and multigrid preconditioner with complex shifted-Laplacian (Riyanti et al., 2007) is assessed against an explicit time marching method. To gain an up-to-date insight on the relative performances of the three possible approaches (namely, the two frequency-domain approaches based upon either a preconditioned iterative solver or a pure direct solver and the time marching method), we compare the results obtained with the ORAS solver for the (acoustic) GO_3D_OBS model with 130 RHSs with those obtained with MUMPS_{BLR} and an explicit time-stepping finite-difference time-domain solver based on a classical $\mathcal{O}(\Delta t^2, \Delta t^8)$ staggered-grid stencil using a grid interval of 100 m. In all cases, we use the Occigen supercomputer.

For the FDFD simulation with MUMPS_{BLR}, we use $\varepsilon_{BLR} = 10^{-5}$, which is required to reach the desired accuracy (Amestoy et al., 2021). Note that a relaxed value of $\varepsilon_{BLR} = 10^{-3}$ was used to solve the local problems with ORAS since a high accuracy is not required to build the preconditioner. We perform the simulation with MUMPS_{BLR} on 80 nodes using one MPI process per node and 24 threads per process to minimize memory overhead while exploiting multithreading for computational efficiency. The elapsed time to perform LU factorization and compute 130 solutions

is 1000 s and 28.3 s, respectively, leading to a total elapsed time of 1028.3s and a scalar time of 548 scalar hours. The FDTD simulation relies on two-level MPI parallelism with an embarrassing parallelism implemented by distributing the sources over the processes and a domain decomposition of the computational domain. We use a total of 12,480 computer cores to implement this two-level parallelism. Two sequential paths over the RHSs are performed to process 65 RHSs in parallel while the domain is decomposed into $2 \times 24 \times 4 = 192$ subdomains. The parallel runtime to perform one simulation is 132s. Hence, the parallel run time to perform 130 simulations is 264 s and the scalar time is 915 hours. These numbers can be compared with those obtained with $\text{ORAS}_{\text{FDFD}}$ with a scalar time of 98 core hours. This number is inferred from the last line of Table 4 by multiplying the parallel run time for 130 RHSs ($T_{\text{tot}} = 261.7$ s) by the number of cores (1344).

Before comparing these numbers, it is worth reminding that, during a FWI iteration, a time-domain method generally requires to recompute the incident wavefields backward in time from the boundaries during the adjoint simulation unlike frequency-domain methods where the incident and adjoint wavefields associated with a block of RHSs can be kept in memory. Alternatively, check-pointing strategies can be used to perform this re-computation, in particular in attenuating media where time-reverse modeling of the incident wavefields are unstable (Anderson et al., 2012; Symes, 2007; Yang et al., 2016). Moreover, attenuation significantly increases the cost of FDTD simulation (by a factor of around 3 according to Figure 13 of Plessix (2017)), while physical attenuation will improve the convergence rate of the ORAS solver. As an illustrative example, when a 3.75-Hz simulation is performed in the GO_3D_OBS model using the Q_P model shown in the Figure 6d of Górszczyk and Operto (2021), the iteration count drops from 45 to 39 and the solution step for 1 RHS from 11.5 s to 9.8 s, hence leading to a computational saving of around 15%. However, it is worth reminding that small patches of frequencies (typically, two to four) are generally inverted during frequency-domain FWI in particular when attenuation reconstruction is targeted (e.g. Operto and Miniussi, 2018). Therefore, the scalar time of frequency-domain modelling should be multiplied by the number of frequencies involved in the patch (these frequencies are ideally processed in parallel with an embarrassing parallelism) for a fair comparison with time-domain methods, the forward engine of which provides the full bandwidth of the wavefield in one go. Concerning the two frequency-domain approaches, we note that the speedup achieved during the block processing of multiple RHSs is 18 and 3.6 for MUMPS and ORAS, respectively. Indeed, direct solvers are more suitable for dense acquisitions to really exploit their ability to process a large number of sparse RHSs.

Lumping all these specifications together, we conclude that the ORAS-FDFD solver should be highly competitive relative to a time-domain solver in particular when attenuation effects should be taken into account. It allows for compact volume of data to be managed, attenuation speeds up the convergence rate of the solver and the good strong scalability provides the necessary flexibility to optimally tune the two-level parallelism based on frequency distribution and domain decomposition according to the acquisition specifications, target size and available computational resources. The solver MUMS_{BLR} further contributes to mitigate the memory cost of the method and the computational efficiency of the local subproblems. Overall, the amount of resources used to perform the simulations (#cores in Table 8) shows that frequency-domain approaches are more suitable when a more limited amount of resources are available, which is consistent with the conclusions of Kostin et al. (2019).

Table 8: Cost of the FDTD, MUMPS-FDFD and ORAS-FDFD simulations in the GO_3D_OBS model for a frequency of 3.75 Hz. #cores: Number of computer cores. $T_s^{(130)}(s)$: Elapsed time to perform 130 simulations in parallel. $T_{hc}(hr)$: scalar hours (elapsed time times the number of cores). $T_f(s)$: Elapsed time to perform parallel LU factorization with MUMPS-FDFD. $T_s^1(s)$: Elapsed time to process one RHS with MUMPS-FDFD.

FDTD			MUMPS-FDFD				ORAS-FDFD			
#c	$T_s^{(130)}(s)$	$T_{hc}(hr)$	#cores	$T_f(s)$	$T_s^1(s)$	$T_s^{130}(s)$	$T_{hc}(hr)$	#cores	$T_s^{130}(s)$	$T_{hc}(hr)$
12480	264	915	1920	1000	3.9	28.3	548	1344	256.8	96

4.2 Finite elements versus finite differences

Our results suggest that the FDFD method outperforms the FEFD counterpart in terms of computational efficiency and accuracy for the four acoustic benchmarks tackled in the previous section. This results because the recently proposed wavelength-adaptive 27-point FDFD stencil (Aghamiry et al., 2021) leads to a quite significant improvement of accuracy compared to the non-adaptive version (Operto et al., 2007) without introducing computational overhead. In the frame of FWI, a clear advantage of the FEFD method is however the flexibility to comply the mesh with known boundary such as bathymetry. Generally, FWI is performed with a frequency continuation strategy to mitigate cycle skipping. The grid interval can be matched to the frequency to regularize the inversion and mitigates the computational cost (Operto et al., 2015). However, when FDFD method on regular grid is used, this grid adaptation may lead to inac-

curate representation of the bathymetry when low frequencies are processed on coarse grids. This can inject shallow inaccuracies in the velocity model near the seabed, which are propagated deeper in the velocity model and which are difficult to correct at higher frequencies, hence making the inversion stall at high frequencies. The FEFD method will clearly give the necessary flexibility to comply the mesh with the bathymetry whatever the size of the elements, hence facilitating the implementation of multiscale approaches. Moreover, elastic physics will further increase the value of the h -adaptive meshing used by FEFD methods to account for the low shear wavespeeds in the near surface. A re-assessment of the two discretization schemes is clearly necessary for elastic physics.

5 Conclusion

We have proposed an efficient, accurate and versatile forward engine for 3D frequency-domain FWI from ultra-long offset stationary-recording survey. The solver relies on an iterative solver and a domain-decomposition based preconditioner where the local subproblems are tackled with sparse direct solver. Two different discretization schemes based on wavelength-adaptive finite differences on regular Cartesian grid and h -adaptive finite elements on unstructured tetrahedral meshes can be interfaced with the solver. We perform a careful parametric analysis that allows us to define the experimental setup providing the best compromise between computational efficiency and accuracy. The strong scalability of the solver provides the necessary flexibility to implement two levels of parallelism by frequency distribution and domain decomposition according to the specifications of the acquisition. The perspective of this work is to use this solver as a forward engine for FWI to perform 3D visco-acoustic case studies from sparse long-offset node acquisitions. Extension to visco-elastic physics is another objective.

Acknowledgments

This study was granted access to the HPC resources of SIGAMM (<http://crimson.oca.eu>) and CINES/IDRIS under the allocation 0596 made by GENCI. This study was partially funded by the WIND consortium (<https://www.geoazur.fr/WIND>) sponsored by Chevron, Shell and Total. The finite-difference time-domain simulation of the discussion section has been performed by L. Combe (Geoazur) with the GeoInv3D_fwkw software. We thank the MUMPS team (P. Amestoy (Mumps Technologies, ENS Lyon), A. Buttari (CNRS, IRT), J.-Y. L'Excellent (Mumps Technologies, ENS Lyon), T. Mary (Sorbonne Université, CNRS, LIP6) and C. Puglisi (Mumps Technologies, ENS Lyon)) for proving the MUMPS solver (<https://mumps-consortium.org>) and for fruitful discussions.

References

- Aghamiry, H., A. Gholami, L. Combe, and S. Operto, 2021, Accurate 3D frequency-domain seismic wave modeling with the wavelength-adaptive 27-point finite-difference stencil: a tool for full waveform inversion: *Geophysics*, **Submitted**, [arXiv:2108.08730](https://arxiv.org/abs/2108.08730).
- Ainsworth, M., and H. Wajid, 2010, Optimally blended spectral-finite element scheme for wave propagation and nonstandard reduced integration: *SIAM Journal on Numerical Analysis*, **48**, 346–371.
- Alappat, C., G. Hager, O. Schenk, J. Thies, A. Basermann, A. Bischof, H. Fehske, and G. Wellein, 2020, A recursive algebraic coloring technique for hardware-efficient symmetric sparse matrix-vector multiplication: *ACM Transactions on Parallel Computing*, **7**, 1–37.
- Amestoy, P., R. Brossier, A. Buttari, J.-Y. L'Excellent, T. Mary, L. Métivier, A. Miniussi, and S. Operto, 2016, Fast 3D frequency-domain FWI with a parallel Block Low-Rank multifrontal direct solver: application to OBC data from the North Sea: *Geophysics*, **81**, R363 – R383.
- Amestoy, P., A. Buttari, J. Y. L'Excellent, and T. Mary, 2018, On exploiting sparsity of multiple right-hand sides in sparse direct solvers: *SIAM Journal on Scientific Computing*, **41**, A269–A291.
- Amestoy, P. R., A. Buttari, L. Combe, M. Gerest, J.-Y. L'Excellent, T. Mary, S. Operto, and C. Puglisi, 2021, Up to date assessment of 3D frequency-domain full waveform inversion based on the sparse multifrontal solver MUMPS: Presented at the Fifth EAGE Workshop on High Performance Computing for Upstream.
- Amestoy, P. R., A. Buttari, J.-Y. L'Excellent, and T. Mary, 2019a, Performance and Scalability of the Block Low-Rank Multifrontal Factorization on Multicore Architectures: *ACM Transactions on Mathematical Software*, **45**, 2:1–2:26.
- Amestoy, P. R., S. de la Kethulle de Ryhov, J.-Y. L'Excellent, G. Moreau, and D. V. Shantsev, 2019b, Efficient use of sparsity by direct solvers applied to 3D controlled-source EM problems: *Computational Geosciences*, **23**, 1237–1258.
- Aminzadeh, F., J. Brac, and T. Kunz, 1997, 3-D Salt and Overthrust models: SEG/EAGE 3-D Modeling Series No.1.
- Anderson, J. E., L. Tan, and D. Wang, 2012, Time-reversal checkpointing methods for RTM and FWI: *Geophysics*, **77**, S93–S103.

- Babuska, I. M., and S. A. Sauter, 1997, Is the pollution effect of the FEM avoidable for the Helmholtz equation considering high wave numbers?: *SIAM Journal on Numerical Analysis*, **34**, 2392–2423.
- Ben Hadj Ali, H., S. Operto, and J. Virieux, 2011, An efficient frequency-domain full waveform inversion method using simultaneous encoded sources: *Geophysics*, **76**, R109.
- Bérenger, J.-P., 1994, A perfectly matched layer for absorption of electromagnetic waves: *Journal of Computational Physics*, **114**, 185–200.
- Böllhofer, M., A. Eftekhari, S. Scheidegger, and O. Schenk, 2019, Large-scale sparse inverse covariance matrix estimation: *SIAM J. Sci. Comput.*, **41(1)**, A380–A401.
- Böllhofer, M., O. Schenk, R. Janalik, S. Hamm, and K. Gullapalli, 2020, State-of-the-art sparse direct solvers: *Parallel Algorithms in Computational Science and Engineering*, Birkhäuser, 3–33.
- Bonazzoli, M., V. Dolean, I. G. Graham, E. A. Spence, and P.-H. Tournier, 2019, A 2-level domain decomposition preconditioner for the time-harmonic Maxwell’s equations: *Math. Comp.*, **88**, 2559–2604.
- Bootland, N., and V. Dolean, 2019, On the Dirichlet-to-Neumann coarse space for solving the Helmholtz problem using domain decomposition: Presented at the Numerical Mathematics and Advanced Applications ENUMATH 2019.
- Bootland, N., V. Dolean, P. Jolivet, F. Nataf, S. Operto, and P.-H. Tournier, 2021a, Several ways to achieve robustness when solving wave propagation problems.
- Bootland, N., V. Dolean, P. Jolivet, and P.-H. Tournier, 2021b, A comparison of coarse spaces for Helmholtz problems in the high frequency regime: *Computers and Mathematics with Applications*, **98**, 239–253.
- Brossier, R., V. Etienne, S. Operto, and J. Virieux, 2010, Frequency-domain numerical modelling of visco-acoustic waves based on finite-difference and finite-element discontinuous galerkin methods, *in* *Acoustic Waves: SCIO*, 125–158.
- Carcione, J. M., and E. M. Robinson, 2002, On the acoustic-electromagnetic analogy for the reflection-refraction problem: *Studia Geophysica et Geodaetica*, **46**, 321–346.
- Chen, Z., D. Cheng, and T. Wu, 2012, A dispersion minimizing finite difference scheme and preconditioned solver for the 3D Helmholtz equation: *Journal of Computational Physics*, **231**, 8152–8175.
- Chew, W. C., and W. H. Weedon, 1994, A 3-D perfectly matched medium from modified Maxwell’s equations with stretched coordinates: *Microwave and Optical Technology Letters*, **7**, 599–604.
- Conen, L., V. Dolean, R. Krause, and F. Nataf, 2014, A coarse space for heterogeneous Helmholtz problems based on the Dirichlet-to-Neumann operator: *J. Comput. Appl. Math.*, **271**, 83–99.
- Dapogny, C., C. Dobrzynski, and P. Frey, 2014, Three-dimensional adaptive domain remeshing, implicit domain meshing, and applications to free and moving boundary problems: *Journal of computational physics*, **262**, 358–378.
- Dolean, V., P. Jolivet, and F. Nataf, 2015, An introduction to domain decomposition methods: *Society for Industrial and Applied Mathematics (SIAM)*, Philadelphia, PA. (Algorithms, theory, and parallel implementation).
- Dolean, V., P. Jolivet, P.-H. Tournier, and S. Operto, 2020a, Iterative frequency-domain seismic wave solvers based on multi-level domain-decomposition preconditioners: Presented at the 82th Annual EAGE Meeting (Amsterdam).
- , 2020b, Large-scale frequency-domain seismic wave modeling on h -adaptive tetrahedral meshes with iterative solver and multi-level domain-decomposition preconditioners: Presented at the SEG2020 Annual Meeting. (Houston).
- Duff, I. S., A. M. Erisman, and J. K. Reid, 2017, *Direct methods for sparse matrices*, second edition: Oxford University Press, Inc.
- Erlangga, Y., 2008, Advances in iterative methods and preconditioners for the Helmholtz equation: *Archives of Computational Methods in Engineering*, **15**, 37–66.
- Ernst, O. G., and M. J. Gander, 2012, Why it is difficult to solve Helmholtz problems with classical iterative methods, *in* *Numerical analysis of multiscale problems*: Springer, 325–363.
- Gander, M., and F. Nataf, 2000, AILU for Helmholtz problems: a new preconditioner based on an analytic factorization: *Comptes Rendus de l’Académie des Sciences - Series I - Mathematics*, **331**, 261–266.
- Gander, M. J., and H. Zhang, 2019, A class of iterative solvers for the Helmholtz equation: Factorizations, sweeping preconditioners, source transfer, single layer potentials, polarized traces, and optimized Schwarz methods: *SIAM Review*, **61**, 3–76.
- Giraud, L., Y.-F. Jing, and Y. Xiang, 2021, A block minimum residual norm subspace solver for sequences of multiple left and right-hand side linear systems: Technical report, RR-9393, Inria Bordeaux Sud-Ouest, hal-03146213v2.
- Górszczyk, A., and S. Operto, 2021, GO_3D_OBS: the multi-parameter benchmark geomodel for seismic imaging method assessment and next-generation 3d survey design (version 1.0): *Geoscientific Model Development*, **14**, 1773–1799.
- Górszczyk, A., S. Operto, and M. Malinowski, 2017, Toward a robust workflow for deep crustal imaging by FWI of OBS data: The eastern nankai trough revisited: *Journal of Geophysical Research: Solid Earth*, **122**, 4601–4630.
- Gosselin-Cliche, B., and B. Giroux, 2014, 3D frequency-domain finite-difference viscoelastic-wave modeling using weighted average 27-point operators with optimal coefficients: *Geophysics*, **79**, T169–T188.

- Graham, I. G., E. A. Spence, and E. Vainikko, 2017, Recent results on domain decomposition preconditioning for the high-frequency helmholtz equation using absorption: Lahaye D., Tang J., Vuik K. (eds) *Modern Solvers for Helmholtz Problems*. Geosystems Mathematics. Birkhäuser, Cham, 3–26.
- Jolivet, P., and P. H. Tournier, 2016, Block iterative methods and recycling for improved scalability of linear solvers: SC16: International Conference for High Performance Computing, Networking, Storage and Analysis, 190–203.
- Karypis, G., 2013, METIS - A software package for partitioning unstructured graphs, partitioning meshes, and computing fill-reducing orderings of sparse matrices - version 5.1.0. University of Minnesota.
- Kostin, V., S. Solov'yev, A. Bakulin, and M. Dmitriev, 2019, Direct frequency-domain 3D acoustic solver with intermediate data compression benchmarked against time-domain modeling for full-waveform inversion applications: *Geophysics*, **84(4)**, T207–T219.
- Kuvshinov, B. N., and W. A. Mulder, 2006, The exact solution of the time-harmonic wave equation for a linear profile: *Geophysical Journal International*, **167**, 659–662.
- Li, Y., R. Brossier, and L. Métivier, 2020, 3D frequency-domain elastic wave modeling with the spectral element method using a massively parallel direct solver: *GEOPHYSICS*, **85**, T71–T88.
- Li, Y., L. Métivier, R. Brossier, B. Han, and J. Virieux, 2015, 2D and 3D frequency-domain elastic wave modeling in complex media with a parallel iterative solver: *Geophysics*, **80(3)**, T101–T118.
- Marfurt, K., 1984, Accuracy of finite-difference and finite-element modeling of the scalar and elastic wave equations: *Geophysics*, **49**, 533–549.
- MUMPS team, 2021, Multifrontal Massively Parallel Solver (MUMPS 5.4.1) Users' guide (August, 2021). Mumps Technologies, <http://mumps-solver.org>.
- Nihei, K. T., and X. Li, 2007, Frequency response modelling of seismic waves using finite difference time domain with phase sensitive detection (TD-PSD): *Geophysical Journal International*, **169**, 1069–1078.
- Operto, S., R. Brossier, L. Combe, L. Métivier, A. Ribodetti, and J. Virieux, 2014, Computationally-efficient three-dimensional visco-acoustic finite-difference frequency-domain seismic modeling in vertical transversely isotropic media with sparse direct solver: *Geophysics*, **79(5)**, T257–T275.
- Operto, S., and A. Miniussi, 2018, On the role of density and attenuation in 3D multi-parameter visco-acoustic VTI frequency-domain FWI: an OBC case study from the North Sea: *Geophysical Journal International*, **213**, 2037–2059.
- Operto, S., A. Miniussi, R. Brossier, L. Combe, L. Métivier, V. Monteiller, A. Ribodetti, and J. Virieux, 2015, Efficient 3-D frequency-domain mono-parameter full-waveform inversion of ocean-bottom cable data: application to Valhall in the visco-acoustic vertical transverse isotropic approximation: *Geophysical Journal International*, **202**, 1362–1391.
- Operto, S., J. Virieux, P. Amestoy, J.-Y. L'Écellent, L. Giraud, and H. Ben Hadj Ali, 2007, 3D finite-difference frequency-domain modeling of visco-acoustic wave propagation using a massively parallel direct solver: A feasibility study: *Geophysics*, **72**, SM195–SM211.
- Osnabrugge, G., S. Leedumrongwatthanakun, and I. M. Vellekoop, 2016, A convergent born series for solving the inhomogeneous helmholtz equation in arbitrarily large media: *Journal of computational physics*, **322**, 113–124.
- Parks, M., E. de Sturler, G. Mackey, D. Johnson, and S. Maiti, 2006, Recycling krylov subspaces for sequences of linear systems: *SIAM Journal of Scientific Computing*, **28**, 1651–1674.
- Pellegrini, F., 2018, PT-SCOTCH and libPTSCOTCH 6.0 user's guide - version 6.0.5. Université Bordeaux.
- Plessix, R. E., 2007, A Helmholtz iterative solver for 3D seismic-imaging problems: *Geophysics*, **72**, SM185–SM194.
- , 2017, Some computational aspects of the time and frequency domain formulations of seismic waveform inversion, *in* *Modern solvers for Helmholtz problems*, Geosystems Mathematics: Springer, 159–187.
- Pratt, R. G., Z. M. Song, P. R. Williamson, and M. Warner, 1996, Two-dimensional velocity models from wide-angle seismic data by wavefield inversion: *Geophysical Journal International*, **124**, 323–340.
- Riyanti, C. D., A. Kononov, Y. A. Erlangga, C. Vuik, C. Oosterlee, R. E. Plessix, and W. A. Mulder, 2007, A parallel multigrid-based preconditioner for the 3D heterogeneous high-frequency Helmholtz equation: *Journal of Computational physics*, **224**, 431–448.
- Saad, Y., 1993, A flexible inner-outer preconditioned gmres algorithm: *SIAM Journal on Scientific Computing*, **14**, 461–469.
- , 2003a, *Iterative methods for sparse linear systems*, 2nd ed.: Society for Industrial and Applied Mathematics.
- , 2003b, *Iterative Methods for Sparse Linear Systems*: SIAM.
- Sirgue, L., and R. G. Pratt, 2004, Efficient waveform inversion and imaging : a strategy for selecting temporal frequencies: *Geophysics*, **69**, 231–248.
- Sourbier, F., A. Haiddar, L. Giraud, H. Ben-Hadj-Ali, S. Operto, and J. Virieux, 2011, Three-dimensional parallel frequency-domain visco-acoustic wave modelling based on a hybrid direct/iterative solver: *Geophysical Prospecting*, **59**, 834–856.
- Symes, W. W., 2007, Reverse time migration with optimal checkpointing: *Geophysics*, **72**, SM213–SM221.

- Tournier, P.-H., I. Aliferis, M. Bonazzoli, M. de Buhan, M. Darbas, V. Dolean, F. Hecht, P. Jolivet, I. E. Kanfoud, C. Migliaccio, F. Nataf, C. Pichot, and S. Semenov, 2019, Microwave tomographic imaging of cerebrovascular accidents by using high-performance computing: *Parallel Computing*, **85**, 88 – 97.
- Van der Vorst, H. A., 1992, BI-CGSTAB: a fast and smoothly converging variant of bi-CG for the solution of nonsymmetric linear systems: *SIAM Journal on Scientific and Statistical Computing*, **13**, 631–644.
- Virieux, J., and S. Operto, 2009, An overview of full waveform inversion in exploration geophysics: *Geophysics*, **74**, WCC1–WCC26.
- Virieux, J., S. Operto, H. Ben Hadj Ali, R. Brossier, V. Etienne, F. Sourbier, L. Giraud, and A. Haidar, 2009, Seismic wave modeling for seismic imaging: *The Leading Edge*, **28**, 538–544.
- Yang, P., R. Brossier, L. Métivier, and J. Virieux, 2016, Wavefield reconstruction in attenuating media: A checkpointing-assisted reverse-forward simulation method: *Geophysics*, **81**, R349–R362.
- Zienkiewicz, O. C., R. L. Taylor, and J. Z. Zhu, 2005, *The Finite Element Method: Its basis and fundamentals*: Elsevier. (6th edition).

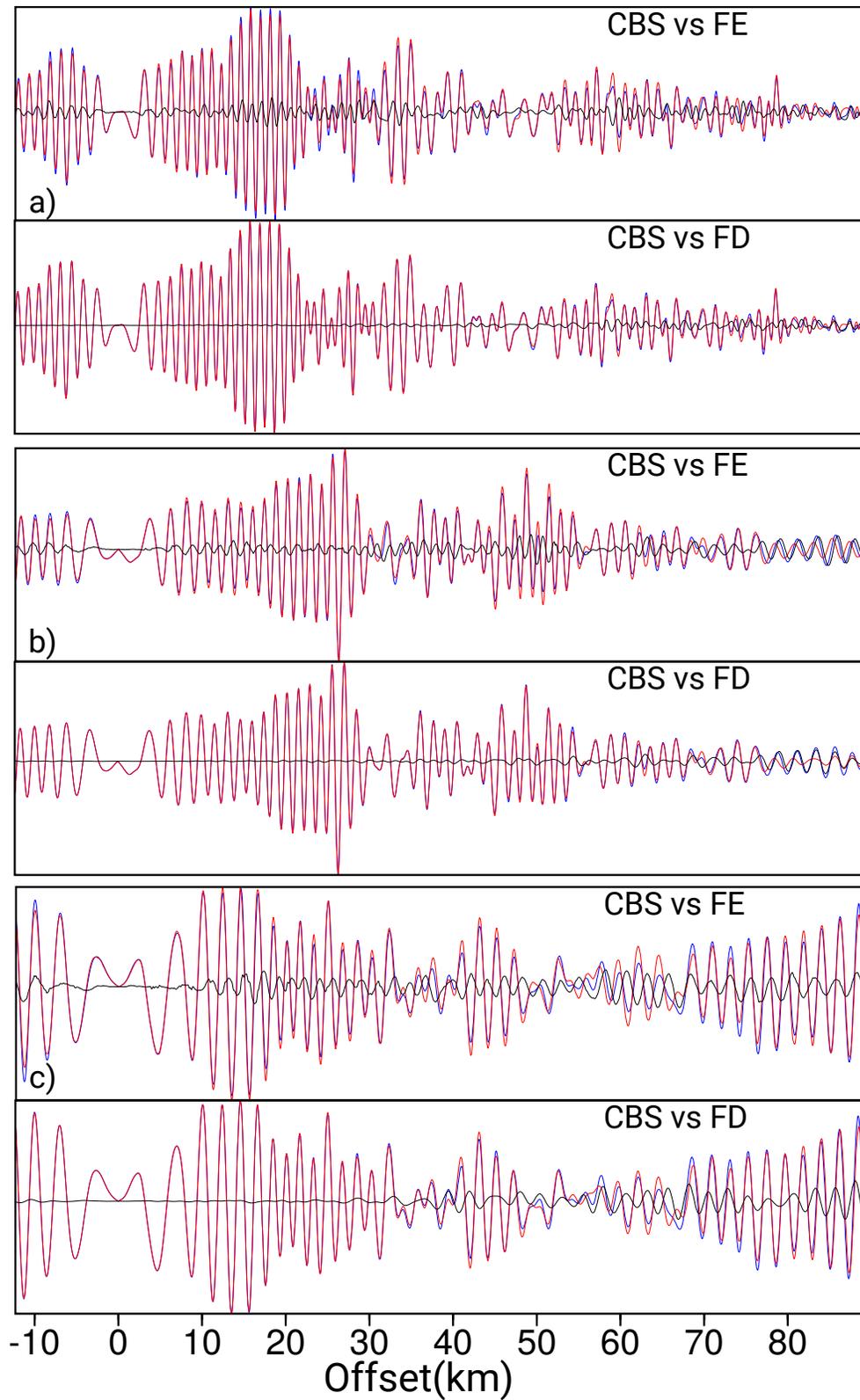


Figure 14: 3D GO_3D_OBS model. Direct comparison between the wavefields computed with the CBS (blue), FEFD/FDFD (red) methods. The differences are plotted in black. ($a_{1:2}$): Profile at $(x,z)=(10\text{k m}, 6\text{ km})$. ($a_{3:4}$): Profile at $(x,z)=(10\text{ km}, 10\text{ km})$. ($a_{5:6}$): Profile at $(x,z)=(10\text{ km}, 16\text{ km})$. a_1, a_3, a_5 : FDFD solution. a_2, a_4, a_6 : FEFD solution.

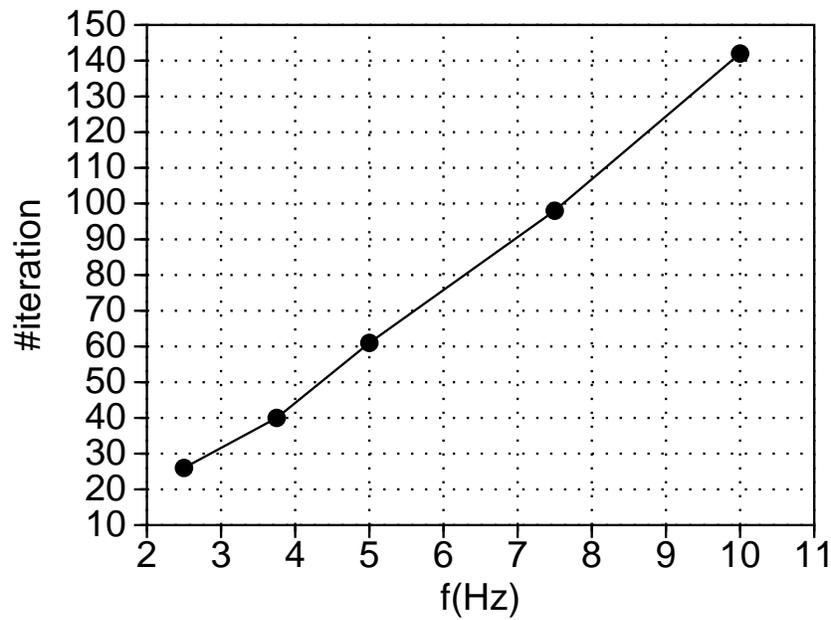


Figure 15: GMRES iteration count versus frequency extracted from Table 6. For each frequency, the number of iterations performed for the smallest number of cores is plot.

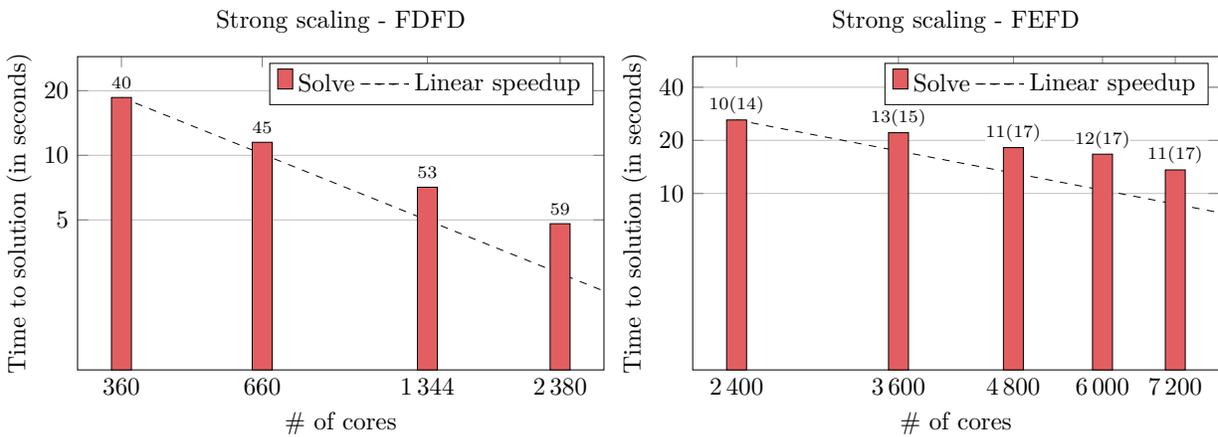


Figure 16: Strong scaling test for the FDFD (left) and FEFD (right) methods on the GO_3D_OBS model. Scalability of the GMRES solution time $T_s(s)$. Number of GMRES iterations is also reported, with average number of inner iterations to solve the coarse problem in parentheses for the FEFD method.