ESAIM: M2AN 48 (2014) 433–447
DOI: 10.1051/m2an/2013114

ESAIM: Mathematical Modelling and Numerical Analysis
www.esaim-m2an.org

SWEEPING PRECONDITIONERS FOR ELASTIC WAVE PROPAGATION WITH SPECTRAL ELEMENT METHODS

Paul Tsuji[1], Jack Poulson[2], Björn Engquist[3] and Lexing Ying[4]

Abstract. We present a parallel preconditioning method for the iterative solution of the time-harmonic elastic wave equation which makes use of higher-order spectral elements to reduce pollution error. In particular, the method leverages perfectly matched layer boundary conditions to efficiently approximate the Schur complement matrices of a block LDL^T factorization. Both sequential and parallel versions of the algorithm are discussed and results for large-scale problems from exploration geophysics are presented.

Mathematics Subject Classification. 65F08, 65N22, 65N80.

Received August 3, 2013.
Published online February 20, 2014.

1. Introduction and motivation

This paper is concerned with the numerical solution of the elastic wave equation in the frequency domain,

$$-(C_{ijkl}u_{k,l})_{,j} - \omega^2 \rho u_i = f_i,$$

where C_{ijkl} is a component of the fourth-order elasticity tensor, u is the displacement vector, ρ is the material density, and ω is the angular frequency. Solving this forward wave problem is a necessary component in many applications, including several full waveform inversion (FWI) techniques, eigenvalue computations for structural dynamics, and non-destructive testing.

It is common practice to discretize wave equations with a minimum of ten points per wavelength, and domains of interest often span several hundred wavelengths in each direction. Thus, 3D problems can easily require a billion degrees of freedom, even when only qualitative solutions are desired. Asymptotic computational complexities are, of course, most relevant for large-scale problems, and so it is desirable to seek alternatives to the quadratic computational cost (and superlinear memory requirements) of 3D sparse-direct methods [17]. Furthermore, it is well-known that the efficiencies of classical iterative methods for time-harmonic wave equations

Keywords and phrases. Elastic wave, seismic wave, time-harmonic, frequency domain, spectral elements, parallel preconditioner, iterative solver, sparse-direct, perfectly matched layers, full waveform inversion.

[1] Sandia National Laboratories, Org. 1442: Numerical Analysis and Applications, Livermore, CA 94550, USA.

[2] Georgia Institute of Technology, School of Computational Science and Engineering, Atlanta, GA 30332, USA.

[3] University of Texas at Austin, Department of Mathematics, Austin, TX 78712, USA.

[4] Stanford University, Department of Mathematics, Stanford, CA 94305, USA. lexing@math.stanford.edu

Article published by EDP Sciences

degrade as the frequency is increased [9], and so several *sweeping* preconditioners [6,7] were recently introduced in order to alleviate this problem in the context of Helmholtz equations.

Both of the original sweeping preconditioner approaches approximate a block LDL^T factorization organized such that the inverse of each Schur complement is a discrete half-space Green's function restricted to the region near the half-space boundary. It was shown in [6] that, in 2D, these discrete restricted half-space Green's functions are approximately hierarchically low-rank and can therefore be accurately inverted with near-linear complexity using hierarchical matrix algebra [10]. In order to tackle challenging 3D problems, [7] introduced a *moving-PML* sweeping preconditioner, which approximates each Schur complement using a quasi-2D auxiliary problem which only takes into account the material properties near the relevant half-space boundary. Both methods were empirically demonstrated to require $O(1)$ iterations of GMRES [22], essentially independent of frequency, for several velocity models without large-scale resonances. Since their introduction, a highly scalable parallel version of the preconditioner was introduced [19], and extensions to curl-conforming finite element discretizations and other time-harmonic problems, such as Maxwell's equations, were investigated [24,25].

This paper makes two major contributions. The first is the extension of the moving-PML sweeping preconditioner to linear elasticity *via* higher-order finite element methods. In particular, we make use of spectral element methods because of their popularity for time-domain seismic problems [16]. Although the framework and structure of our new preconditioner is very similar to that of the original Helmholtz case, elasticity introduces additional complications, such as multiple propagation modes. The second major contribution is the extension of the recently developed parallel sweeping preconditioner [19] to large-scale 3D elastic wave equations. We demonstrate the performance of our parallel approach using elastic analogues of the SEG/EAGE Overthrust and Salt Dome models.

The rest of the paper is as follows. In Section 2, we review the elastic wave equation and spectral element methods. Continuing in Section 3, we discuss the main ideas behind the linear-elastic moving-PML preconditioner, and in Section 4, we describe how the parallel version of the preconditioner must be tailored for higher-order finite elements. Section 5 then presents numerical results for several test problems using a few thousand cores, and, finally, we conclude with some comments on future work.

2. Spectral element methods for the elastic wave equation

2.1. Perfectly matched layers for linear elasticity

Using index notation, the time-harmonic linear elasticity problem for the displacement field $u = (u_1, u_2, u_3)$ on an infinite domain is given as

$$\begin{aligned} -\left(C_{ijkl}u_{k,l}\right)_{,j} - \omega^2 \rho u_i &= f_i \ \text{ in } \mathbb{R}^3, \\ \lim_{r\to\infty} r\left(\frac{\partial u_\mathrm{s}}{\partial r} - \imath\kappa_\mathrm{s} u_\mathrm{s}\right) &= 0, \ \text{ and} \\ \lim_{r\to\infty} r\left(\frac{\partial u_\mathrm{p}}{\partial r} - \imath\kappa_\mathrm{p} u_\mathrm{p}\right) &= 0. \end{aligned}$$

The last two equations are the Kupradze–Sommerfeld radiation conditions [3], where the displacement field has been decomposed into its solenoidal part, u_s, and its irrotational part, u_p. We have assumed that in the far field, the material becomes homogeneous and isotropic, and the usual Helmholtz decomposition can be made. These two equations enforce the radition conditions for the S and P waves, whose asymptotic wavenumbers have respectively been denoted by κ_s and κ_p. Given the Lamé parameters λ and μ of the background medium, these quantities are

$$\kappa_\mathrm{s} = \frac{\omega}{\sqrt{\frac{\mu}{\rho}}}, \qquad \kappa_\mathrm{p} = \frac{\omega}{\sqrt{\frac{\lambda+2\mu}{\rho}}}.$$

In order to make the problem computationally feasible, we must truncate the domain and introduce an absorbing boundary to emulate the radiation conditions. In this work, we employ perfectly matched layers (PMLs), which were originally developed for electromagnetic waves [2,4], but have also been studied in the context of elastic waves [13]. We note that the stability of the PML formulation has been well studied in the works cited.

For the unit cube $\Omega = [0,1]^3$ with boundary $\partial\Omega$ and PML width L, we can define the complex stretching function in the ith direction for a point in space $\mathbf{x} = (x_1, x_2, x_3)$ as

$$s_i(\mathbf{x}) = 1 + \imath\sigma(x_i),$$

where σ is the ramp function

$$\sigma(\xi) = \begin{cases} \eta\left(\frac{L-\xi}{L}\right)^2, & \xi \in [0, L] \\ 0, & \xi \in [L, 1-L] \\ \eta\left(\frac{\xi-1+L}{L}\right)^2, & \xi \in [1-L, 1] \end{cases}$$

and η is a constant parameter which optimizes the PML for mesh size. The modified fourth-order elasticity tensor and material density are then

$$\tilde{C}_{ijkl} = C_{ijkl}\frac{s_1 s_2 s_3}{s_j s_l}, \qquad \tilde{\rho} = \rho s_1 s_2 s_3,$$

and the original infinite domain problem becomes the truncated problem

$$\begin{aligned} -(\tilde{C}_{ijkl}u_{k,l})_{,j} - \omega^2\tilde{\rho}u_i &= f_i \quad \text{in } \Omega, \text{ and} \\ u &= 0 \quad \text{on } \partial\Omega. \end{aligned} \tag{2.1}$$

Since the PML damps the outgoing waves exponentially, it is not crucial to place a radiation condition on the outside of the PML; the zero Dirichlet boundary condition is a standard approximation.

Finally, the variational formulation of the problem is as follows. After integrating (2.1) against a test function $v \in (H_0^1(\Omega))^3$ and applying integration by parts to the left term, we arrive at the sesquilinear form and linear functional

$$\begin{aligned} B(u,v) &= \int_\Omega v_{i,j}\tilde{C}_{ijkl}u_{k,l}\mathrm{d}\Omega - \omega^2\int_\Omega \tilde{\rho}u_i v_i \mathrm{d}\Omega, \text{ and} \\ \ell(v) &= \int_\Omega f_i v_i \mathrm{d}\Omega. \end{aligned}$$

The weak form of the problem is thus to find $u \in (H_0^1(\Omega))^3$ such that

$$B(u,v) = \ell(v) \qquad \forall v \in (H_0^1(\Omega))^3. \tag{2.2}$$

2.2. Spectral element methods

The spectral element method (SEM) was first used for fluid dynamics [15, 18], though in the past decade, it has also become popular for discretizing the time-domain elastic wave equation [16]. We will carry over the same ideas to the frequency domain. First, we briefly review the ideas which are similar to the formulations in [14].

For polynomial order p, take $\{\xi_i\}_{i=1}^{p+1}$ and $\{w_i\}_{i=1}^{p+1}$ to be the Gauss–Lobatto quadrature nodes and weights on the interval $[-1,1]$. We define the ith Lagrangian interpolant on the 1D Gauss–Lobatto grid to be

$$\varphi_i(\xi) = \prod_{\substack{j=1 \\ i\neq j}}^{p+1} \frac{(\xi - \xi_j)}{(\xi_i - \xi_j)}.$$

Consider a master hexahedral element on the cube $[-1,1]^3$ and define $(p+1)^3$ tensor-product nodal basis functions as

$$\psi_{ijk}(\mathbf{x}) = \varphi_i(x_1)\varphi_j(x_2)\varphi_k(x_3), \qquad i,j,k = 1,\dots,p+1. \tag{2.3}$$

If we then discretize Ω with a uniform hexahedral mesh with element width h, and define K as the number of elements in each dimension, then there will be K^3 elements for the entire domain. For each of these elements, we associate a scaled Gauss–Lobatto grid with width h in each direction. Since the Gauss–Lobatto nodes include the endpoints of the interval, adjacent elements share some Gauss–Lobatto quadrature nodes. Let us order the Gauss–Lobatto nodes in the total mesh lexicographically, which is to say, if index (i_1, i_2, i_3) describes the node location on an $n \times n \times n$ grid, then it is given the global nodal number $j = j_1 + (j_2 - 1)n + (j_3 - 1)n^2$. Here, $n = pK - 1$ is the number of Gauss–Lobatto points in each direction on the whole mesh as a result of the zero Dirichlet boundary condition imposed on the boundary of Ω. With a little abuse of notation, we denote $\psi_j(x)$ to be the nodal basis function defined at global node number j, for $j = 1, \ldots, N = n^3$. These are standard conforming basis functions which ensure continuity of the solution across element boundaries. If a node with index j belongs to the boundary of multiple elements, then the basis function $\psi_j(x)$ is supported over all these elements. Otherwise, the support of $\psi_j(x)$ is equal to a single element. We can now expand each component of the displacement field as

$$u_i^h(x) = \sum_{j=1}^{N} c_{i,j}\psi_j(x).$$

In the global linear system, we order the degrees of freedom in the vector

$$(c_{1,1}, c_{2,1}, c_{3,1}, c_{1,2}, c_{2,2}, c_{3,2}, \ldots, c_{1,N}, c_{2,N}, c_{3,N})^T.$$

Using this ordering, the standard Galerkin formulation yields the $3N \times 3N$ system

$$\mathbf{Au} = \mathbf{f},$$

where the matrix and vector entries can be written as

$$\begin{aligned}\mathbf{A}_{(i,a)(j,b)} &= B(\psi_a \mathbf{e}_i, \psi_b \mathbf{e}_j),\\ \mathbf{u}_{(j,b)} &= c_{b,j}, \text{ and}\\ \mathbf{f}_{(i,a)} &= \ell(\psi_a \mathbf{e}_i),\end{aligned}$$

where $\mathbf{e}_i$ is the unit vector in the ith direction. To numerically evaluate these integrals, we reuse the Gauss–Lobatto quadrature rule over each element.

3. Moving PML sweeping preconditoners

In order to simplify the discussion of the preconditoner, we assume that the element width h is equal to the width of the PML L, *i.e.*, $K = 1/h = 1/L$. With these assumptions, we may perform a 1D decomposition of Ω in the $+z$ direction and construct the subdomains

$$\begin{aligned}\Omega_i &= [0,1]^2 \times [(i-1)L, iL], \quad i = 1, \ldots, K-1, \text{ and}\\ \Omega_K &= [0,1]^2 \times [(K-1)L, KL].\end{aligned}$$

Thus, each subdomain is the same thickness as the PML. Now, denote the vector of degrees of freedom contained in Ω_i as $\mathbf{u}_i$. By the definition of the subdomains Ω_i, the degrees of freedom on the boundary of two subdomains are contained in the top subdomain; we have illustrated this on a simple 6×6 quadrilateral mesh with fourth-order polynomials in Figure 1. Given these groupings, we can write the global linear system in block-tridiagonal form,

$$\begin{pmatrix} \mathbf{A}_{1,1} & \mathbf{A}_{1,2} & & & \\ \mathbf{A}_{2,1} & \mathbf{A}_{2,2} & \ddots & & \\ & \ddots & \ddots & \ddots & \\ & & \ddots & \mathbf{A}_{K-1,K-1} & \mathbf{A}_{K-1,K} \\ & & & \mathbf{A}_{K,K-1} & \mathbf{A}_{K,K} \end{pmatrix} \begin{pmatrix} \mathbf{u}_1 \\ \mathbf{u}_2 \\ \vdots \\ \mathbf{u}_{K-1} \\ \mathbf{u}_K \end{pmatrix} = \begin{pmatrix} \mathbf{f}_1 \\ \mathbf{f}_2 \\ \vdots \\ \mathbf{f}_{K-1} \\ \mathbf{f}_K \end{pmatrix},$$

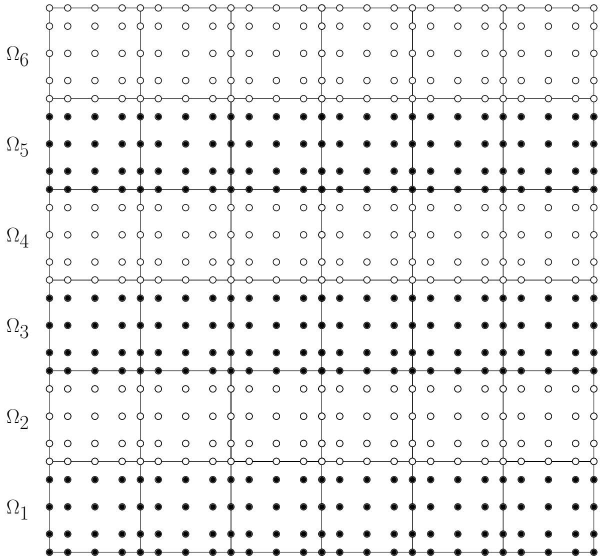


FIGURE 1. Grouping of degrees of freedom into elements of a 6×6 quadrilateral mesh. The black nodes belong to odd-numbered subdomains, while the empty nodes belong to even-numbered subdomains.

where $\mathbf{A}_{i,j}$ is the block corresponding to test functions supported in Ω_i and basis functions supported in Ω_j. Because of our PML scheme and the symmetries of the fourth-order elasticity tensor, the resulting matrix is complex symmetric, *i.e.*, $A = A^T$, *not* $A = A^H$. If we then factored this matrix with block Gaussian, we would have produced the decomposition

$$\mathbf{A} = \mathbf{L}_1 \dots \mathbf{L}_{K-1} \begin{pmatrix} \mathbf{S}_1 & & & \\ & \mathbf{S}_2 & & \\ & & \ddots & \\ & & & \mathbf{S}_K \end{pmatrix} \mathbf{L}_{K-1}^T \dots \mathbf{L}_1^T,$$

where the Schur complements take the form $\mathbf{S}_1 = \mathbf{A}_{1,1}$ and $\mathbf{S}_i = \mathbf{A}_{i,i} - \mathbf{A}_{i,i-1}\mathbf{S}_{i-1}^{-1}\mathbf{A}_{i-1,i}$ for $i = 2, \dots, K$, and the lower triangular matrix $\mathbf{L}_i$ is a block identity matrix which contains the elimination factor $-\mathbf{A}_{i+1,i}\mathbf{S}_i^{-1}$ in the $(i+1, i)$th block. However, naively running such a block elimination process would be far too expensive, as the Schur complements, S_i, would quickly become dense. For instance, in 2D, this leads to an $O(N^2)$ factorization algorithm, which is clearly more expensive than an $O(N^{3/2})$ multifrontal scheme.

In [7], it was explained that each inverse Schur complement, say S_m^{-1}, can be interpreted as the discrete Green's function of a particular half-space problem involving layers 1 through m. This interpretation can then be used to construct an approximation $\tilde{\mathbf{S}}_m^{-1}$ of $\mathbf{S}_m^{-1}$ which replaces the original half-space problem with one which moves the absorbing boundary condition from Ω_1 directly next to Ω_m (see Fig. 2). For 3D problems, these approximate half-space problems have a sparsity structure which is very close to that of a 2D problem, whereas, when the original problem is 2D, the resulting auxiliary problems are quasi-1D. Since the asymptotic complexity of multifrontal methods decreases with the dimension of the problem, individually factoring each subdomain problem is asymptotically faster than factoring the global problem. In particular, using nested-dissection orderings, the entire set of subdomain problems can be factored with $O(N)$ complexity in 2D and $O(N^{4/3})$ complexity in 3D, and an approximate inverse of an LDL^T factorization can then be applied to a vector with $O(N)$ and $O(N \log N)$ work in 2D and 3D, respectively.

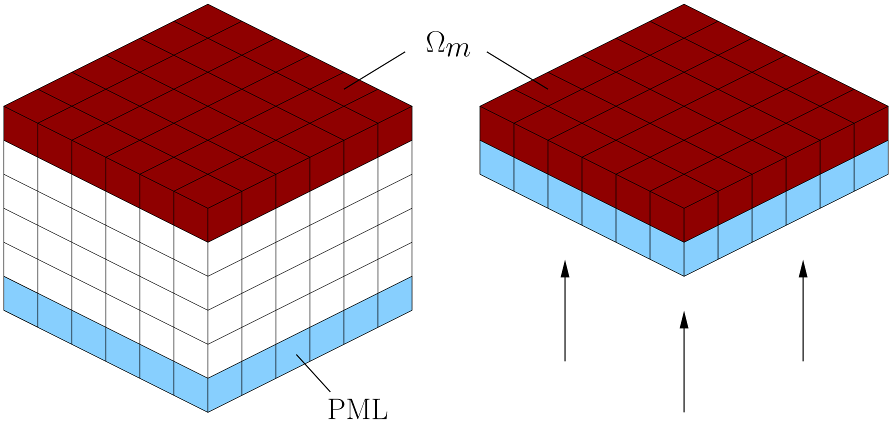


FIGURE 2. *Moving PML method*. Since the inversion only needs to be accurate on Ω_m, the original half-space problem (left) can be approximated by the 2-layer problem (right).

To illustrate how $\mathbf{S}_m^{-1}$ is approximated as an operator in the context of elasticity, define the shifted PML functions for subdomain m as

$$s_i^m(\mathbf{x}) = 1 + \imath\sigma^m(x_i),$$

where the shifted coordinate-stretching functions are

$$\sigma^m(\xi) = \begin{cases} \eta\left(\frac{(m-1)L-\xi}{L}\right)^2, & \xi \in [(m-2)L,(m-1)L], \\ 0, & \xi \in [(m-1)L, 1-L], \text{ and} \\ \eta\left(\frac{\xi-1+L}{L}\right)^2, & \xi \in [1-L,1]. \end{cases}$$

Given that we are sweeping in the $+z$ direction for 3D problems, denote the shifted elasticity tensor and material density as

$$\tilde{C}_{ijkl}^m = C_{ijkl}\frac{s_1 s_2 s_3^m}{s_j s_l}, \qquad \tilde{\rho} = \rho s_1 s_2 s_3^m,$$

where $s_j = s_j^m$ if $j = 3$ and $s_l = s_l^m$ if $l = 3$. Using spectral elements, we can now discretize the truncated half-space problem

$$\begin{aligned} -(\tilde{C}_{ijkl}^m u_{k,l})_{,j} - \omega^2 \tilde{\rho}^m u_i = f_i \quad &\text{in } \Omega_m \cup \Omega_{m-1}, \text{ and} \\ u = 0 \quad &\text{on } \partial(\Omega_m \cup \Omega_{m-1}), \end{aligned} \tag{3.1}$$

where the resulting discrete operator $\mathbf{H}_m$ has the 2×2 block form

$$\mathbf{H}_m = \begin{pmatrix} \tilde{\mathbf{A}}_{m-1,m-1} & \tilde{\mathbf{A}}_{m-1,m} \\ \tilde{\mathbf{A}}_{m,m-1} & \mathbf{A}_{m,m} \end{pmatrix}.$$

The bottom right block of $\mathbf{H}_m^{-1}$ will be denoted as $\tilde{\mathbf{S}}_m^{-1}$ and serve as an approximation to $\mathbf{S}_m^{-1}$. Since $\mathbf{H}_m$ has a quasi-2D sparsity pattern, it can be factored efficiently with a 2D multifrontal method. Once factored, we can apply $\tilde{\mathbf{S}}_m^{-1} \approx \mathbf{S}_m^{-1}$ to a vector $\mathbf{f}_m$ defined on the degrees of freedom of Ω_m *via* the equation

$$\mathbf{H}_m^{-1}\begin{pmatrix} 0 \\ \mathbf{f}_m \end{pmatrix} = \begin{pmatrix} * & * \\ * & \tilde{\mathbf{S}}_m^{-1} \end{pmatrix}\begin{pmatrix} 0 \\ \mathbf{f}_m \end{pmatrix} = \begin{pmatrix} * \\ \tilde{\mathbf{S}}_m^{-1}\mathbf{f}_m \end{pmatrix}. \tag{3.2}$$

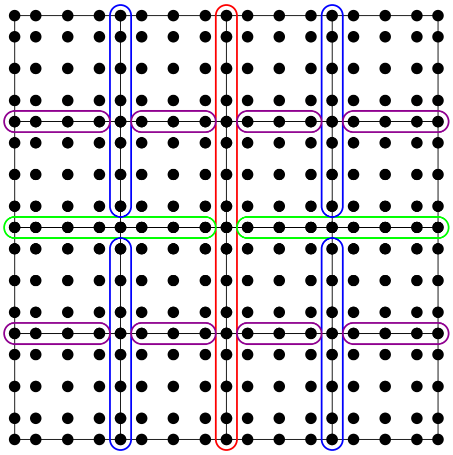


Figure 3. Separators for the multifrontal tree. The root separator is in red.

Here, the $*$ denotes the part of $\mathbf{H}_m^{-1}$ that we can ignore. Computationally, this amounts to extending the vector $\mathbf{f}_m$ by zero onto the artificial PML, applying H_m^{-1} using the multifrontal factorization, and then extracting $\tilde{\mathbf{S}}_m^{-1}\mathbf{f}_m$.

When performing the multifrontal factorization of H_m, we construct the separators so that they are all aligned with the element interface. Figure 3 illustrates this scheme on a 4×4 quadrilateral mesh with polynomial order $p = 4$. This minimizes the separator size and thus the cost of the multifrontal factorization. High-level descriptions of the setup and application phases of the moving-PML sweeping preconditioner are provided in Algorithms 3.1 and 3.2.

Algorithm 3.1. *Setup phase of sweeping preconditioner*

```
for $m = 2, \dots, K$ do
    Let $\mathbf{H}_m$ be the discretized operator of (3.1). Construct the sparse $LDL^T$ factorization of $\mathbf{H}_m$ using the multifrontal method with nested dissection.
end for
```

Algorithm 3.2. *Application of sweeping preconditioner*

```
Set $\mathbf{u}_i = \mathbf{f}_i$, for $i = 1, \dots, K$.
for $i = 1, \dots, K-1$ do
    Compute $\mathbf{u}_{i+1} = \mathbf{u}_{i+1} - \mathbf{A}_{i+1,i}\tilde{\mathbf{S}}_i^{-1}\mathbf{u}_i$, where $\tilde{\mathbf{S}}_i^{-1}$ is applied as in (3.2).
end for
for $i = 1, \dots, K$ do
    Compute $\mathbf{u}_i = \tilde{\mathbf{S}}_i^{-1}\mathbf{u}_i$, where $\tilde{\mathbf{S}}_i^{-1}$ is applied as in (3.2).
end for
for $i = K-1, \dots, 1$ do
    Compute $\mathbf{u}_i = \mathbf{u}_i - \tilde{\mathbf{S}}_i^{-1}\mathbf{A}_{i,i+1}\mathbf{u}_{i+1}$, where $\tilde{\mathbf{S}}_i^{-1}$ is applied as in (3.2).
end for
```

In order to avoid letting the Schur complement approximations introducing long-range phase errors, we introduce an $O(1)$ damping parameter α and compute the approximate inverses of

$$-(C_{ijkl}u_{k,l})_{,j} - (\omega + \imath\alpha)^2 \rho u_i = f_i,$$

instead of the original wave equation. This idea is similar to the complex shift introduced in the shifted-Laplacian preconditioner [8], with the difference being that their damping parameter grows as $O(\omega)$.

While the formulation of the preconditioner proposed does not require the use of a higher-order method, it must be noted that the superior dispersion properties of the spectral element method reduces the pollution error and improves the propagation of information when sweeping through the domain; it is generally observed that these discretizations lead to a more accurate preconditioner and faster convergence of the outer Krylov solver. As a corollary of the dispersion properties, higher-order basis functions require less degrees of freedom per wavelength to obtain the same accuracy over the domain as low order methods; this allows us to shrink the size of each subdomain problem which needs to be factorized. Of course, using a higher polynomial order also increases the number of nonzero entries in each subdomain matrix, increasing the memory and computational time of the direct solver; there is a delicate balance between accuracy, memory, and computational time which needs to be maintained. In practice, we found that using a higher polynomial order ($p = 5$ or $p = 6$) with one element per wavelength produced reasonable results.

4. Parallel sweeping preconditioners

As was previously mentioned, moderate to high-frequency 3D wave propagation problems often involve billions of degrees of freedom and therefore their solution requires large amounts of work and storage space. Both of these issues are best addressed with large distributed-memory machines, and a parallel version of the moving-PML sweeping preconditioner was therefore presented in [19] in the context of finite-difference discretizations of the Helmholtz equation. We will now briefly review the parallel sweeping preconditioner and discuss an extension to higher-order discretizations of time-harmonic linear elasticity.

The main difficulty with parallelizing the sweeping preconditioner should be clear from Algorithms 3.1 and 3.2: while the approximate auxiliary problems can be independently factored, their inverses must be applied one at a time. Storing and applying the preconditioner in a time and memory scalable manner therefore requires each quasi-2D multifrontal factorization to be effectively distributed and applied using the entire set of processors. While Gupta et al. [12] have demonstrated that sparse-direct factorizations can be weakly scaled to several thousand processors, an effective parallelization of the sweeping preconditioner requires that multifrontal triangular solves over small subproblems achieve a similar level of scalability (both in terms of memory usage and runtime). *Selective inversion* [21] is currently the best-known technique for parallelizing multifrontal triangular solves with few right-hand sides, and so a parallel multifrontal solver named *Clique* was implemented in a manner conceptually similar to [11], but incorporating selective inversion and using the element-wise matrix distributions of [20] instead of conventional block matrix distributions [5]. This multifrontal solver was then used to construct *Parallel Sweeping Preconditioner (PSP)*, which carefully exploits scalable multifrontal subdomain factorizations and solves in order to provide a scalable left-preconditioned version of GMRES(k) [22] which can exploit the existence of multiple right-hand sides.

At first glance, the parallelization of the spectral element sweeping preconditioner looks identical to that of [19], but there are a few important subtleties which much be handled:

1. The analytical nested dissection algorithm used to define the subdomain elimination trees [17, 23] should ensure that separators are chosen along element boundaries (consider Fig. 3).
2. Due to the wider support of spectral element basis functions relative to those of second-order finite-difference stencils, the amount of work required for the matrix assemblies (both for the global problem and for the damped subdomain problems) becomes nontrivial and must be handled with care.

This first point is extremely important, as it allows for our spectral element discretizations to have separators which are the same size as those of second-order finite-difference stencils. When performing analytical nested dissection with second-order finite differences, it is not particularly crucial where the separators are placed, as the stencil of each node only overlaps with nodes directly adjacent to it; in higher order finite elements, however, the stencil of each basis function overlaps with all basis functions contained in the element. Thus, if a one-gridpoint thick separator is chosen along a plane which cuts across the interior of an element, there is interaction between two branches in the subdomain elimination tree (a nonzero off-diagonal block), and nested dissection breaks down. To ensure that the interaction between two branches in the elimination tree does not occur, either the separator width must be increased to the size of the element, or the separator must be carefully placed along the element boundaries in the mesh. Choosing the former option becomes unmanageable in terms of memory; the latter option results in a multifrontal scheme which has essentially equivalent computational costs as finite differences for the same grid size.

The second issue is handled by decomposing the global domain in two dimensions rather than three. Because the application of the subdomain solves cannot be parallelized like an additive Schwarz method, the setup stage is also performed in a sequential fashion, where all processes are employed simultaneously for each subdomain factorization, one after the other. To accelerate the matrix construction and provide each process with the rows of the matrix it will need without excessive communication, the global domain is decomposed in the 2D plane orthogonal to the sweeping direction. This way, each process will own a small subset of the elements roughly equal in size from each subdomain (and correspondingly, the global domain).

5. Numerical results

In this section, we present some preliminary results displaying the effectiveness of the sweeping preconditioner for linear elasticity problems. The sequential code was implemented in C++ using MUMPS as the direct solver, and the experiments were performed on a server with 2.2 GHz AMD Opteron 6174 processors. The parallel spectral element code was also implemented in C++, and its benchmarks made use of TACC's Lonestar cluster. Each node of Lonestar contains two hex-core Intel Xeon 5680 processors and 24 GB of DDR3-1333MHz memory, and in each of our experiments, we launched 8 MPI processes per node.

In many of the test problems, anisotropic elasticity versions of the materials are not available. Instead of artificially adding an anisotropic tensor, we have modeled the provided velocity data as the S-wave velocity. The speed of the S-wave is $\sqrt{\frac{\mu}{\rho}}$, and we vary the density ρ so that the S-wave velocity matches the acoustic model. The following examples model isotropic elasticity, which restricts the fourth-order elasticity tensor C_{ijkl} to the form

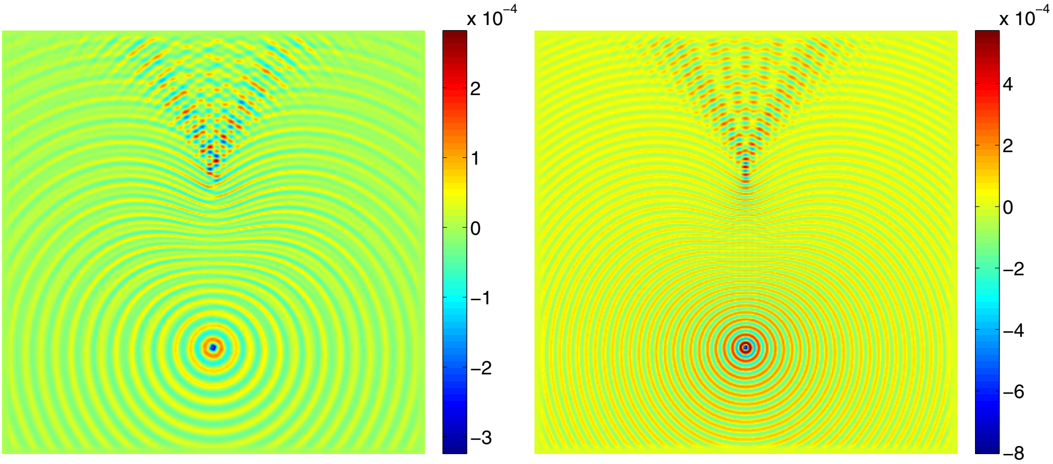
$$C_{ijkl} = \lambda\delta_{ij}\delta_{kl} + \mu(\delta_{ik}\delta_{jl} + \delta_{il}\delta_{jk}).$$

For our tests, we have chosen the homogeneous background material to be $\mu = 1$, $\lambda = 1$, and $\rho = 1$. We note that when we discretize, we consider the gridpoints-per-wavelength ratio in terms of the S-wave only; since the P-wave has a faster velocity with our parameters, it is naturally overresolved when compared to the S-wave.

In both codes, the moving PML method was used as a left preconditioner for GMRES with a relative residual tolerance of 1e-4. For the 2D examples, the polynomial order of the basis functions was set to 6, whereas for 3D the order was chosen as 3. The domains were discretized using only one element per wavelength in 2D and two elements per wavelength in 3D. In other words, there are 6 grid points per wavelength in both 2D and 3D. The thickness of each PML was set to one wavelength for each frequency, the maximum amplitude of the complex ramping function was chosen as 2, and the damping parameter, α, was set to 1. Similar PML parameters were chosen in [13] and were observed to produce stable solutions; for a detailed account of the convergence and stability of PML in elastodynamic problems, we refer the reader to the cited work.

5.1. 2D and 3D sequential tests

For the sequential code, we chose a few examples with smoothly varying materials. The domain is set to be the unit square in 2D and unit cube in 3D, *i.e.* $\Omega = [0,1]^d$. The right hand side is produced by a point source; in 2D, the source is located at $(0.5, 0.25)$ and oriented in the $\hat{y}$-direction, while in 3D, the source is located at $(0.5, 0.25, 0.5)$ and oriented in the $\hat{z}$-direction. The first example is a converging lens with an S-wave velocity



N_w	N_{dof}	T_{setup}	T_{solve}	N_{iter}
16	2.554e + 04	3	1	4
32	1.012e + 05	12	3	4
64	4.032e + 05	53	10	4
128	1.609e + 06	230	41	4

FIGURE 4. 2D sequential numerical results for the converging lens medium, with figures for $N_w = 64$. Left: displacement in the radial direction away from the source. Right: displacement in direction tangential to wavefronts.

profile given by

$$c(\mathbf{r}) = \frac{4}{3}\left(1 - \frac{1}{2}e^{-32|\mathbf{r}-\mathbf{r}_c|^2}\right),$$

where $\mathbf{r}_c$ is the center of the domain and $|\mathbf{r} - \mathbf{r}_c|$ is the distance from the center in both 2D and 3D. This profile is characterized by the slow wave speed in the center, which causes outgoing rays to form caustics behind the lens. The second example was a periodic medium with the S-wave velocity function

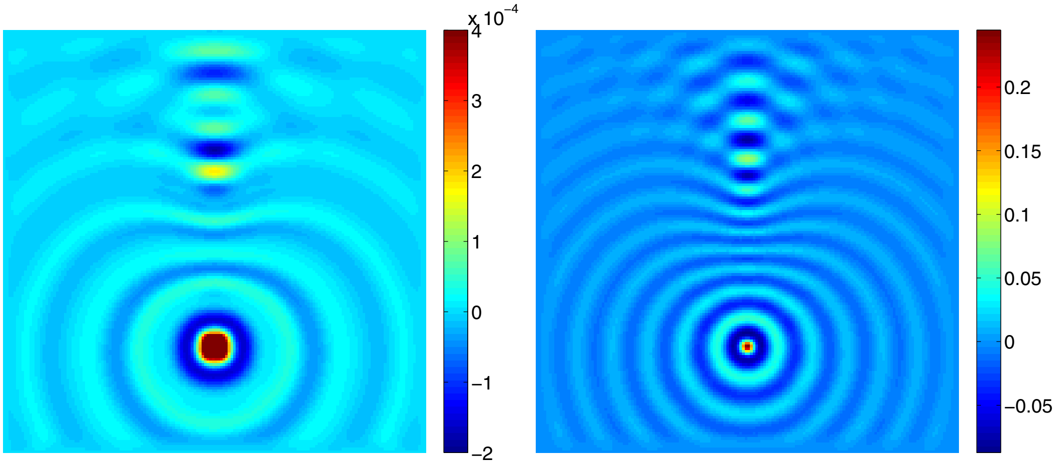
$$c(x, y) = \left(1 + \frac{1}{4}\cos\left(10\sqrt{2}x + 10\sqrt{2}y\right) + \frac{1}{4}\cos\left(10\sqrt{2}x - 10\sqrt{2}y\right)\right)^{-\frac{1}{2}}. \tag{5.1}$$

Note that, in the 3D case, this profile is invariant in the z-direction. For each problem, we list the number of wavelengths across the domain N_w, the number of degrees of freedom N_{dof}, the setup time T_{setup}, the GMRES solve time T_{solve}, and the number of iterations N_{iter}.

The 2D results are provided in Figures 4 and 6 and agree with our linear complexity estimates, as doubling the frequency increases both the total number of degrees of freedom and setup time (roughly) by a factor of 4. Since the number of iterations remains constant over the range of frequencies, it is also observed that the solve time increases approximately by a factor of 4, which implies a linear complexity solver for this class of problems. The 3D results are shown in Figures 5 and 7 and also agree with our complexity analysis. Since doubling the frequency increases the total number of degrees of freedom by a factor of 8, the setup time should increase by a factor of roughly $8^{4/3} = 16$, whereas the solve time should only increase by a factor slightly larger than 8.

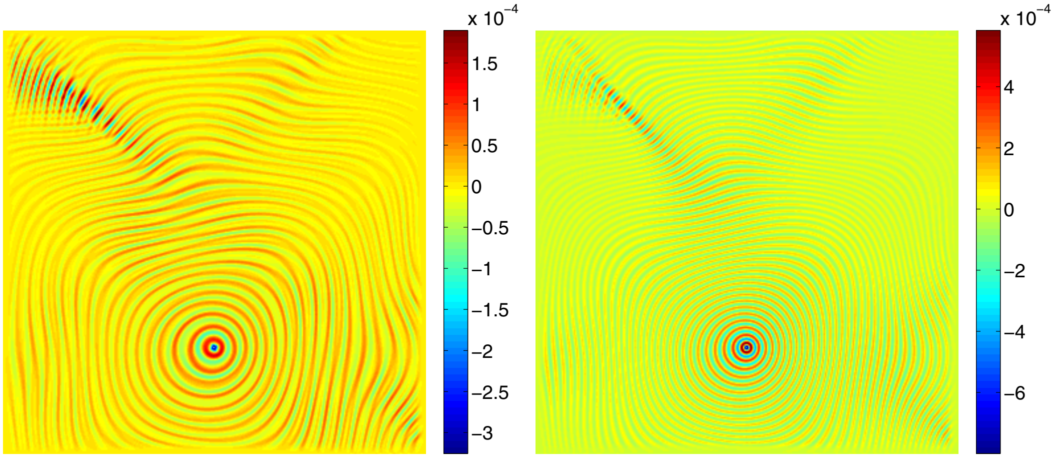
5.2. 3D parallel example: Overthrust model

The SEG/EAGE Overthrust model [1] is now a standard benchmark for 3D seismic problems and is characterized by discontinuous layers of material with varying wave speeds, as shown in the slice plot of Figure 8.



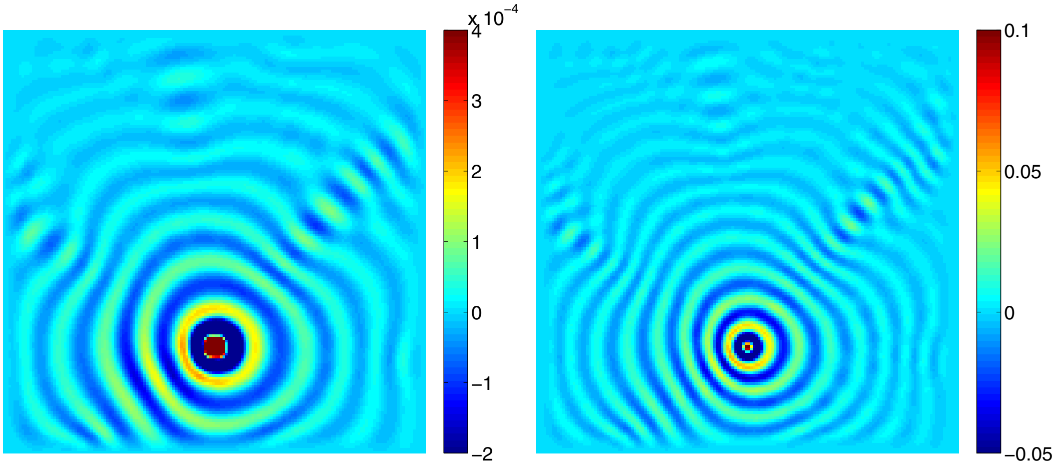
N_w	N_{dof}	T_{setup}	T_{solve}	N_{iter}
4	$4.687\text{e}+04$	20	2	4
8	$3.529\text{e}+05$	295	22	4
16	$2.738\text{e}+06$	4424	231	4

FIGURE 5. 3D sequential numerical results for the converging lens, with figures for $N_w = 16$. Left: displacement in the radial direction away from the source. Right: displacement in direction tangential to wavefronts.



N_w	N_{dof}	T_{setup}	T_{solve}	N_{iter}
16	$2.554\text{e}+04$	3	1	4
32	$1.012\text{e}+05$	13	3	4
64	$4.032\text{e}+05$	54	10	4
128	$1.609\text{e}+06$	233	42	4

FIGURE 6. 2D sequential numerical results for the periodic medium, with figures for $N_w = 64$. Left: displacement in the radial direction away from the source. Right: displacement in direction tangential to wavefronts.



N_w	N_{dof}	T_{setup}	T_{solve}	N_{iter}
4	4.687e + 04	19	2	4
8	3.529e + 05	295	22	4
16	2.738e + 06	4431	345	6

FIGURE 7. 3D sequential numerical results for the periodic medium, with figures for $N_w = 16$. Left: displacement in the radial direction away from the source. Right: displacement in direction tangential to wavefronts.

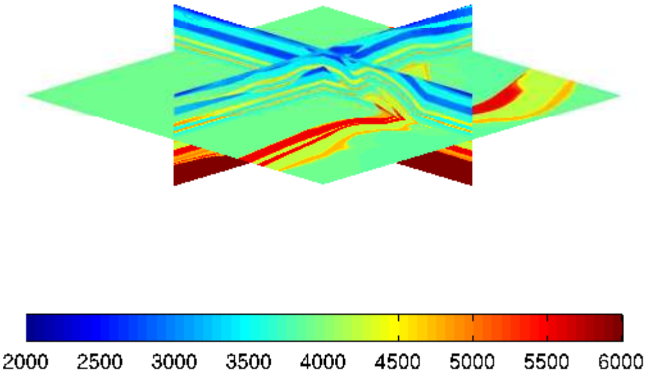
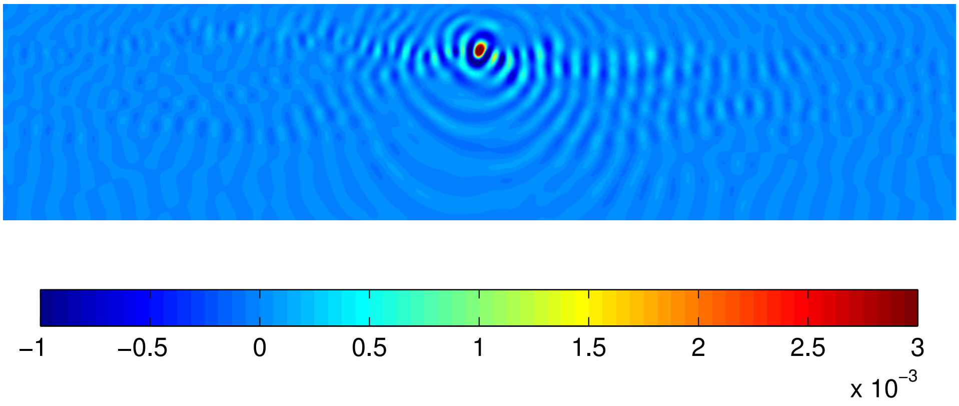


FIGURE 8. SEG/EAGE Overthrust model. Velocity data is given in meters/second.

The dimensions of the model are 20 km × 20 km × 4.65 km, and the original data for the wave speed is given on a 801 × 801 × 187 grid. The minimum wave speed in the model is 2.179 km/s, while the maximum wave speed is 6 km/s. Because the data is given on a uniform grid, linear interpolation was performed in order to evaluate the velocity field over the Gauss–Lobatto grids. The point source is located at $(10, 10, 3.4875)$ in kilometers and oriented in the $\hat{x}$-direction.

For this problem, we have enforced the sweeping direction to be in the x-direction. In our setup, there are PML layers on all sides surrounding the domain and it is feasible to sweep from the top layer downward; however, since the majority of the reflections are oriented in the direction of the discontinuities, sweeping in this direction would kill some of these fields when approximating the half-space problem. It is more prudent to sweep orthogonal to the reflected rays, as pushing the PML to the domain of interest would not remove the



f	N_{dof}	T_{setup}	T_{solve}	N_{iter}	N_{proc}
5	$4.511e+06$	164	98	11	128
10	$3.548e+07$	670	468	12	1024

FIGURE 9. Results for the elastic wave equation with the Overthrust model. The $\hat{x}$-component of the displacement field in the yz-plane at 10 Hz is shown.

contribution of fields that are returning to the domain. As a general guideline, the sweeping direction should be chosen so that sharp contrasts in the medium between adjacent subdomains are avoided.

Figure 9 shows results for the Overthrust model. Here, the relevant quantities listed are the frequency f, number of degrees of freedom N_{dof}, setup time T_{setup}, GMRES time T_{solve}, number of iterations N_{iter}, and the number of cores N_{proc}. The largest problem solved was the 10 Hz case. It should be mentioned that using the same spectral element code with the Helmholtz equation and polynomial order $p = 5$, the largest problem solved on 2048 cores was the 20 Hz example, with about 116 million degrees of freedom; we did not list these results here, since they have been included in another paper. Because the elastic wave equation contains three times the number of degrees of freedom as the Helmholtz equation and has a much denser sparsity pattern, more memory and processory power is required to push to higher frequencies.

5.3. 3D parallel example: Salt Dome model

Another example of a commonly used velocity model is the SEG/EAGE Salt Dome. The model is identified by the large salt body in the middle of sedimentary layers; inside the salt body, the wave speed is very fast relative to the surrounding material. The dimensions of the model are 13.5 km × 13.5 km × 4 km, and the velocity data is given on a 676 × 676 × 210 grid. The minimum velocity is 1500 m/s, while the maximum velocity is 4482 m/s. The point source is located at $(6.25, 6.25, 3)$ in kilometers and oriented in the $\hat{x}$-direction. Figure 10 shows a slice plot of the model.

Figure 11 shows the results for the Salt Dome model. Once again, the memory requirements are very demanding for the elasticity case, so only the 10 Hz example is presented, with about 33 million degrees of freedom on 2048 processors. In this example, the sweeping direction is taken to be the z-direction; regardless of this choice, however, it was observed that the number of iterations increased with frequency. One reason this happens is because of the large difference in the velocity between the salt body and surrounding area; it is difficult to resolve the smallest wavelength with enough grid points while keeping the PML wide enough to dampen the largest wavelength. Secondly, because the salt body is significantly large compared to the wavelength in all directions, the secondary reflections inside the structure are not restricted to a particular dimension as in a layered medium like the Overthrust case. Because there are reflected rays returning to subdomains in both the

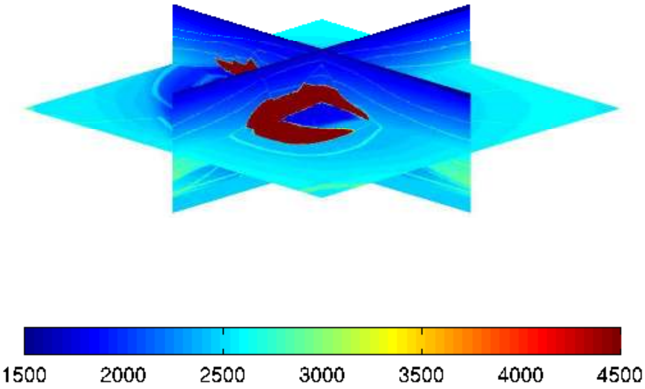
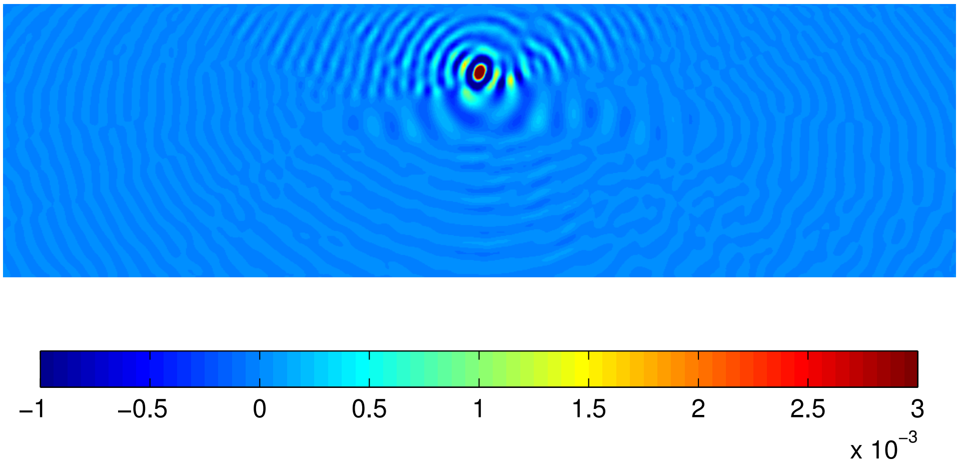


FIGURE 10. SEG/EAGE Salt Dome model. Velocity data is given in meters/second.



f	N_{dof}	T_{setup}	T_{solve}	N_{iter}	N_{proc}
5	4.198e + 06	150	37	7	256
10	3.305e + 07	431	221	11	2048

FIGURE 11. Results for the elastic wave equation with the Salt Dome model. The $\hat{x}$-component of the displacement field in the yz-plane at 10 Hz is shown.

horizontal and vertical directions, the moving PML does not approximate the Green's function as well for this problem.

6. Conclusions

In this paper, we have presented a moving-PML sweeping preconditioner for the time-harmonic elastic wave equation using higher-order spectral element discretizations. We have shown the complexity of the sequential algorithm in 2D and 3D on examples with smoothly varying media. In addition, we have detailed a parallel version of the algorithm, and have demonstrated its effectiveness in solving 3D geophysics problems. Currently, the spectral element code for elasticity is not fully scalable, and is in the process of being modified to minimize communication and memory usage. This will allow the solution of the 20 Hz problem for both the Overthrust and Salt Dome models.

References

[1] F. Aminzadeh, J. Brac and T. Kunz, 3-D Salt and Overthrust models. In *SEG/EAGE 3-D Modeling Series 1*. Tulsa, OK (1997).

[2] J.P. Berenger, A perfectly matched layer for the absorption of electromagnetic waves. *J. Comput. Phys.* **114** (1994) 185–200.

[3] J. Bramble and J. Pasciak, A note on the existence and uniqueness of solutions of frequency domain elastic wave problems: *a priori* estimates in H^1. *J. Math Anal. Appl.* **345** (2008) 396–404.

[4] W.C. Chew and W.H. Weedon, A 3D perfectly matched medium from modified Maxwell's equations with stretched coordinates. *Microwave Optical Tech. Lett.* **7** (1994) 599–604.

[5] J. Choi, J.J. Dongarra, R. Pozo and D.W. Walker, ScaLAPACK: A scalable linear algebra library for distributed memory concurrent computers, in *Proc. of the Fourth Symposium on the Frontiers of Massively Parallel Computation, IEEE Comput. Soc. Press* (1992) 120–127.

[6] B. Engquist and L. Ying, Sweeping preconditioner for the Helmholtz equation: hierarchical matrix representation. *Commun. Pure Appl. Math.* **64** (2011) 697–735.

[7] B. Engquist and L. Ying, Sweeping preconditioner for the Helmholtz equation: moving perfectly matched layers. *Multiscale Model. Simul.* **9** (2011) 686–710.

[8] Y.A. Erlangga, C. Vuik and C.W. Oosterlee, On a class of preconditioners for solving the Helmholtz equation. *Appl. Numer. Math.* **50** (2004) 409–425.

[9] O.G. Ernst and M.J. Gander, Why it is difficult to solve Helmholtz problems with classical iterative methods, in vol. 83 of *Numerical Analysis of Multiscale Problems*. Edited by I. Graham, T. Hou, O. Lakkis and R. Scheichl. Springer-Verlag (2011) 325–361.

[10] L. Grasedyck and W. Hackbusch, Construction and arithmetics of $\mathcal{H}$-matrices. *Computing* **70** (2003) 295–334.

[11] A. Gupta, G. Karypis and V. Kumar, A highly scalable parallel algorithm for sparse matrix factorization. *IEEE Transactions on Parallel and Distributed Systems* **8** (1997) 502–520.

[12] A. Gupta, S. Koric and T. George, Sparse matrix factorization on massively parallel computers, in *Proc. of the Conference on High Performance Computing, Networking, Storage and Analysis*. Portland, OR (2009).

[13] I. Harari and U. Albocher, Studies of FE/PML for exterior problems of time-harmonic elastic waves. *Comput. Methods Appl. Mech. Eng.* **195** (2006) 3854–3879.

[14] T. Hughes, *The finite element method: linear static and dynamic finite element analysis*. Prentice-Hall, Inc. (1987).

[15] G. Karniadakis, *Spectral/hp element methods for CFD*. Oxford University Press (1999).

[16] D. Komatitsch and J. Tromp, Introduction to the spectral element method for three-dimensional seismic wave propagation. *Geophys. J. Int.* **139** (1999) 806–822.

[17] J. Liu. The multifrontal method for sparse matrix solution: theory and practice. *SIAM Review* **34** (1992) 82–109.

[18] A. Patera, A spectral element method for fluid dynamics: laminar flow in a channel expansion. *J. Comput. Phys.* **54** (1984) 468–488.

[19] J. Poulson, B. Engquist, S. Li and L. Ying, A parallel sweeping preconditioner for heterogeneous 3D Helmholtz equations `e-prints ArXiv` (2012).

[20] J. Poulson, B. Marker, R.A. van de Geijn, J.R. Hammond and N.A. Romero, Elemental: A new framework for distributed memory dense matrix computations. *ACM Trans. Math. Software* **39**.

[21] P. Raghavan, Efficient parallel sparse triangular solution with selective inversion. *Parallel Proc. Lett.* **8** (1998) 29–40.

[22] Y. Saad and M.H. Schultz, A generalized minimum residual algorithm for solving nonsymmetric linear systems. *SIAM J. Sci. Stat. Comput.* **7** (1986) 856–869.

[23] R. Schreiber, A new implementation of sparse Gaussian elimination. *ACM Trans. Math. Software* **8** (1982) 256–276.

[24] P. Tsuji, B. Engquist and L. Ying, A sweeping preconditioner for time-harmonic Maxwell's equations with finite elements. *J. Comput. Phys.* **231** (2012) 3770–3783.

[25] P. Tsuji and L. Ying, A sweeping preconditioner for Yee's finite difference approximation of time-harmonic Maxwell's equations. *Frontiers of Mathematics in China* **7** (2012) 347–363.