Preconditioning the 2D Helmholtz equation with polarized traces

Leonardo Zepeda-Núñez[(*)], Russell J. Hewett, and Laurent Demanet
Department of Mathematics and Earth Resources Laboratory, Massachusetts Institute of Technology

SUMMARY

We present a domain decomposition solver for the 2D Helmholtz equation, with a special choice of integral transmission condition that involves polarizing the waves into one-way components. This refinement of the transmission condition is the key to combining local direct solves into an efficient iterative scheme, which can then be deployed in a high-performance computing environment. The method involves an expensive, but embarrassingly parallel precomputation of local Green's functions, and a fast online computation of layer potentials in partitioned low-rank form. The online part has sequential complexity that scales *sublinearly* with respect to the number of volume unknowns, even in the high-frequency regime. The favorable complexity scaling continues to hold in the context of low-order finite difference schemes for standard community models such as BP and Marmousi2, where convergence occurs in 5 to 10 GMRES iterations.

INTRODUCTION

Fixed-frequency wave scattering in a heterogeneous acoustic or elastic medium is a hard question in numerical analysis. It results in large, indefinite systems of equations, which are poorly handled by off-the-shelf preconditioners. Standard direct methods are not currently scalable to large 3D problems, while iterative methods require a number of iterations that often grows with frequency.

Recent progress was made on a few different fronts: Erlangga et al. (2006) showed how to implement very simple, although suboptimal, complex-shifted Laplace preconditioners; Engquist and Ying (2011a,b) proposed a class of sweeping preconditioners that have uniform asymptotic complexity scalings with respect to the frequency parameter in smooth 2D and 3D media; de Hoop et al. (2011) coupled direct multifrontal ideas with H-matrices to achieve very efficient although asymptotically suboptimal preconditioners in 2D and 3D; Vion and Geuzaine (2014) explored approximate transmission boundary conditions to improve on traditional domain decomposition methods; and most closely related to the topic of this paper, Stolk (2013) designed a domain decomposition method which realizes interface transmission with an ingenious forcing term, resulting in linear complexity scalings very similar to those of the sweeping preconditioners. Most of these methods were not only applied to geophysical imaging, but were designed chiefly for this purpose.

The picture that emerges is that the novel, sophisticated iterative methods (such as most cited above) can handle very large problems, but are still slower than old, simple, optimized sparse direct methods when the latter are able to accommodate problem size. This note seeks to bridge this gap by proposing a method that leverages existing sparse direct solvers, and at the same time scales to very large problems in highly parallel environments. The new point of view also invites to revise the lower complexity bound for the sequential part of the solver down from N, the total number of unknowns in the volume, to $O(\sqrt{N})$, the number of interface unknowns. While we discuss the 2D problem, there is in principle no obstruction to extending the method to 3D.

METHOD

Consider a partition of a 2D rectangular domain Ω into L horizontal layers $\{\Omega_i\}_{i=1}^{L}$ that we call slabs. Let $\mathbf{x} = (x,z)$ and consider the squared slowness $m(\mathbf{x}) = 1/c(\mathbf{x})^2$. The (constant-density, acoustic) Helmholtz equation is $\mathscr{H}u = f$, with the Helmholtz operator at frequency κ defined as

$$\mathscr{H} = \triangle + m\kappa^2, \tag{1}$$

with absorbing boundary conditions on the boundary $\partial\Omega$. Let f_i be the restriction of f to Ω_i, and denote by $\mathscr{H}_i$ the local Helmholtz operator $\mathscr{H}_i = \triangle + m_i\kappa^2$, now with absorbing boundary conditions on the boundary $\partial\Omega_i$. The corresponding local Green's function $G_i(\mathbf{x},\mathbf{x}')$ obeys $\mathscr{H}_i G_i(\mathbf{x},\mathbf{x}') = \delta(\mathbf{x}-\mathbf{x}')$. Using G_i, we can express the solution of $\mathscr{H}u = f$, using the Green's representation formula (GRF), locally in each layer as

$$\begin{aligned} u(\mathbf{x}) &= G_i f_i(\mathbf{x}) \\ &\quad + \int_{\partial\Omega_i} \left(G_i(\mathbf{x},\mathbf{x}') \frac{\partial u}{\partial n_{\mathbf{x}'}}(\mathbf{x}') d\mathbf{x}' - \frac{\partial G_i}{\partial n_{\mathbf{x}'}}(\mathbf{x},\mathbf{x}') u(\mathbf{x}') \right) dS_{\mathbf{x}'}, \end{aligned} \tag{2}$$

for $\mathbf{x} \in \Omega_i$, where $n_{\mathbf{x}'}$ is the exterior normal to Ω at $\mathbf{x}'$ and $G_i f_i(\mathbf{x}) = \int_{\Omega_i} G_i(\mathbf{x},\mathbf{x}') f_i(\mathbf{x}') d\mathbf{x}'$.

Denote by $\Gamma_{i,i+1}$ the interface between Ω_i and Ω_{i+1}. Supposing that Ω_i are thin slabs extending to infinity, we can approximate Eq. 2 via

$$\begin{aligned} u(\mathbf{x}) &\approx G_i f_i(\mathbf{x}) \\ &\quad - \int_{\Gamma_{i-1,i}} \left(G_i(\mathbf{x},\mathbf{x}') \partial_{z'} u(\mathbf{x}') + \partial_{z'} G_i(\mathbf{x},\mathbf{x}') u(\mathbf{x}') \right) dx' \\ &\quad + \int_{\Gamma_{i,i+1}} \left(G_i(\mathbf{x},\mathbf{x}') \partial_{z'} u(\mathbf{x}') - \partial_{z'} G_i(\mathbf{x},\mathbf{x}') u(\mathbf{x}') \right) dx'. \end{aligned} \tag{3}$$

As in many domain decomposition methods for the Helmholtz equation (see Després (1990), Hsiao et al. (2000), Hiptmair and Jerez-Hanckes (2012)) the solution is computed from information at the interfaces of the subdomains. In our formulation, the problem is reduced to finding the traces u and $\partial_z u$ on $\Gamma_{i,i+1}$ and then mapping them to $u(\mathbf{x})$ for $\mathbf{x} \in \Omega_i$ locally in each slab via Eq. 3. In the sequel, we write $u_i(\mathbf{x})$ for $u(\mathbf{x})$ when $\mathbf{x} \in \Omega_i$.

We now decompose u_i by polarization as $G_i f_i + u_i^{\uparrow} + u_i^{\downarrow}$, where $u_i^{\uparrow}$ approximates the field generated by sources located below slab Ω_i, and $u_i^{\downarrow}$ approximates the field generated by sources located above slab Ω_i. We characterize $u_i^{\uparrow}$ and $u_i^{\downarrow}$ from transmission conditions expressed as incomplete Green's integrals:

$$\int_{\Gamma_{i-1,i}} \Big(G_i(\mathbf{x},\mathbf{x}')\partial_{z'} u_i^{\uparrow}(\mathbf{x}') - \partial_{z'} G_i(\mathbf{x},\mathbf{x}') u_i^{\uparrow}(\mathbf{x}') \Big)\, dx' = 0,$$

$$\int_{\Gamma_{i,i+1}} \Big(G_i(\mathbf{x},\mathbf{x}')\partial_{z'} u_i^{\downarrow}(\mathbf{x}') - \partial_{z'} G_i(\mathbf{x},\mathbf{x}') u_i^{\downarrow}(\mathbf{x}') \Big)\, dx' = 0,$$

for $\mathbf{x} \in \Omega_i$. We then construct a preconditioner by assigning the four remaining terms of Eq. 3 in an upwind fashion to $u_i^{\uparrow}$ and $u_i^{\downarrow}$:

$$u_i^{\uparrow}(\mathbf{x}) = \int_{\Gamma_{i,i+1}} \Big(G_i(\mathbf{x},\mathbf{x}')\partial_{z'} u_i^{\uparrow}(\mathbf{x}') - \partial_{z'} G_i(\mathbf{x},\mathbf{x}') u_i^{\uparrow}(\mathbf{x}') \Big)\, dx', \tag{4}$$

$$u_i^{\downarrow}(\mathbf{x}) = -\int_{\Gamma_{i-1,i}} \Big(G_i(\mathbf{x},\mathbf{x}')\partial_{z'} u_i^{\downarrow}(\mathbf{x}') - \partial_{z'} G_i(\mathbf{x},\mathbf{x}') u_i^{\downarrow}(\mathbf{x}') \Big)\, dx', \tag{5}$$

for $\mathbf{x} \in \Omega_i$. These equations imply interface-to-interface propagation rules, obtained by using $\mathbf{x} \in \Gamma_{i,i+1}$ for $u_i^{\downarrow}$ in Eq. 5, and $\mathbf{x} \in \Gamma_{i-1,i}$ for $u_i^{\uparrow}$ in Eq. 4. Moreover, by superposition, the respective traces of $G_i f_i$ on $\Gamma_{i-1,i}$ and $\Gamma_{i,i+1}$ become additional contributions to the polarized traces $u_{i-1}^{\uparrow}$ and $u_{i+1}^{\downarrow}$ of the neighboring slabs. Algorithm 1 embeds these propagation rules in a sweeping-like iteration.

From the knowledge of the polarized traces, we can then recombine the field everywhere in Ω_i as $u_i = G_i f_i + u_i^{\downarrow} + u_i^{\uparrow}$, where the polarized components obey Eq. 4 and Eq. 5. The following figure helps keep track of the various indices and quantities.

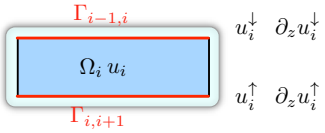

Figure 1: The light-shaded layer around Ω_i represents the absorbing boundary condition.

ALGORITHM

We now describe a preconditioner, i.e., an approximation of $\mathscr{H}^{-1}$. It involves the following layer potentials:

$$S_i^{\downarrow} w(\mathbf{x}) = \int_{\Gamma_{i-1,i}} G_i(\mathbf{x},\mathbf{x}') w(\mathbf{x}') dx', \ \mathbf{x} \in \Gamma_{i,i+1}, \tag{6}$$

$$D_i^{\downarrow} w(\mathbf{x}) = \int_{\Gamma_{i-1,i}} \partial_{z'} G_i(\mathbf{x},\mathbf{x}') w(\mathbf{x}') dx', \ \mathbf{x} \in \Gamma_{i,i+1}, \tag{7}$$

$$(D_i^{\downarrow})^* w(\mathbf{x}) = \int_{\Gamma_{i-1,i}} \partial_{z} G_i(\mathbf{x},\mathbf{x}') w(\mathbf{x}') dx', \ \mathbf{x} \in \Gamma_{i,i+1}, \tag{8}$$

$$N_i^{\downarrow} w(\mathbf{x}) = \int_{\Gamma_{i-1,i}} \partial_{z}\partial_{z'} G_i(\mathbf{x},\mathbf{x}') w(\mathbf{x}') dx', \ \mathbf{x} \in \Gamma_{i,i+1}. \tag{9}$$

We also write $S_i^{\uparrow}, D_i^{\uparrow}, (D_i^{\uparrow})^*$, and $N_i^{\uparrow}$ for the corresponding four operators where the roles of $\mathbf{x}$ and $\mathbf{x}'$ are reversed, namely $\mathbf{x} \in \Gamma_{i-1,i}$ and $\mathbf{x}' \in \Gamma_{i,i+1}$.

We now define the preconditioner $\widetilde{v} = Pg$, for any source g, as follows.

Algorithm 1. *Preconditioner associated to* $\{\Omega_i\}_{i=1}^{L}$

function $\widetilde{v} = P(g)$
  for `i = 1:L` *do* ▷ *Local solves*
    g_i = *restriction of* g *to* Ω_i
    $v^{g_i} = G_i g_i$
  end for
  $v_1^{\downarrow} = \partial_z v_1^{\downarrow} = v_L^{\uparrow} = \partial_z v_L^{\uparrow} = 0$
  for `i = 2:L` *do on* $\Gamma_{i-1,i}$ ▷ *Downward Sweep*
    $v_i^{\downarrow} = -S_{i-1}^{\downarrow} \partial_z v_{i-1}^{\downarrow} + D_{i-1}^{\downarrow} v_{i-1}^{\downarrow} + v^{g_{i-1}}|_{\Gamma_{i-1,i}}$
    $\partial_z v_i^{\downarrow} = -D_{i-1}^{*,\downarrow} \partial_z v_{i-1}^{\downarrow} + N_{i-1}^{\downarrow} v_{i-1}^{\downarrow} + \partial_z v^{g_{i-1}}|_{\Gamma_{i-1,i}}$
  end for
  for `i = L-1:1` *do on* $\Gamma_{i,i+1}$. ▷ *Upward Sweep*
    $v_i^{\uparrow} = S_{i+1}^{\uparrow} \partial_z v_{i+1}^{\uparrow} - D_{i+1}^{\uparrow} v_{i+1}^{\uparrow} + v^{g_{i+1}}|_{\Gamma_{i,i+1}}$
    $\partial_z v_i^{\uparrow} = D_{i+1}^{*,\uparrow} \partial_z v_{i+1}^{\uparrow} - N_{i+1}^{\uparrow} v_{i+1}^{\uparrow} + \partial_z v^{g_{i+1}}|_{\Gamma_{i,i+1}}$
  end for
  for `i = 1:L` *do* $\mathbf{x} \in \Omega_i$ ▷ *Recombination*
    $$v_i = v^{g_i} - \int_{\Gamma_{i-1,i}} \Big(G_i(\mathbf{x},\mathbf{x}')\partial_{z'} v_i^{\downarrow}(\mathbf{x}') + \partial_{z'} G_i(\mathbf{x},\mathbf{x}') v_i^{\downarrow}(\mathbf{x}') \Big)\, dx' + \int_{\Gamma_{i,i+1}} \Big(G_i(\mathbf{x},\mathbf{x}')\partial_{z'} v_i^{\uparrow}(\mathbf{x}') - \partial_{z'} G_i(\mathbf{x},\mathbf{x}') v_i^{\uparrow}(\mathbf{x}') \Big)\, dx'$$
  end for
  $\widetilde{v} = [v_1, \ldots, v_L]^T$ ▷ *Concatenation*
end function

KRYLOV ITERATION

We use Generalized Minimum Residual (GMRES) as an iterative solver. Within this framework, we right-precondition the system as

$$\mathscr{H} P v = f, \qquad u = Pv, \tag{10}$$

i.e., solve the first system for v, then get u as Pv. After the first iteration, the only nonzero contribution to the residual $f - \mathscr{H} P f$ occurs at the interface and is due to the concatenation of the v_i to form $\widetilde{v}$. This observation enables crucial computational savings: (i) the differential operator $\mathscr{H}$ needs only be applied at these interfaces to obtain the residual, hence (ii) the result of the preconditioner is needed only in a very small neighborhood of the slab boundaries.

To ensure well-posedness of the local problems, the residuals must be supported on the slab interiors only, which we accommodate by generating a second partition, staggered with respect to the first partition. Staggered partitions are used in the context of overlapping Schwarz methods, and were applied in the scope of Helmholtz solvers in Stolk (2013).

Define the second layered partition of the domain Ω as $\{\widetilde{\Omega}\}_{i=1}^{L}$, and define the interface $\widetilde{\Gamma}_{i,i+1}$ in the same way as before. The

slabs are staggered (or interlaced) when they have different interfaces, namely, $\Gamma_{i,i+1} \cap \widetilde{\Gamma}_{j,j+1} = \phi$, $\forall i, j$. We extend the definition of the interface operators to the new partition in the obvious way and, similarly, we define the preconditioner $\widetilde{P}$. Following the same reasoning, $f - \mathscr{H}\widetilde{P}f$ is supported on $\{\widetilde{\Gamma}_{i,i+1}\}_{i=1}^{L-1}$. To retain the property that residuals are defined on the slab interiors, subsequent iterations alternate between P and $\widetilde{P}$.

It is convenient to present one iteration of the full preconditioner as two steps involving P and $\widetilde{P}$ in succession. The method is initialized by computing Pf. Thus, the problem is reduced to solving

$$\mathscr{H}\psi = f - \mathscr{H}Pf, \tag{11}$$

where the right-hand-side is supported on $\{\Gamma_{i,i+1}\}_{i=1}^{L-1}$. The solution is then obtained as $u = Pf + \psi$. To solve Eq. 11, we define the dual-partition preconditioner in Alg. 2.

Algorithm 2. *Dual partition Preconditioner*

1: ***function*** $u = \mathscr{P}(f)$
2: $u_1 = \widetilde{P}(f)$
3: $r = f - \mathscr{H}\widetilde{P}f$ ▷ *Compute the residual*
4: $u_2 = P(r)$
5: $u = u_1 + u_2$ ▷ *One step of iterative refinement*
6: ***end function***

Finally, we solve the preconditioned reduced system

$$\mathscr{H}\mathscr{P}w = f - \mathscr{H}Pf, \qquad \psi = \mathscr{P}w, \tag{12}$$

using GMRES, where w is supported on $\{\Gamma_{i,i+1}\}_{i=1}^{L-1}$. To stress the fact that all the operations are interface-to-interface, each application of $\mathscr{H}\mathscr{P}$ is called an "inner iteration" in this note.

NUMERICAL DISCRETIZATION

We discretize the Helmholtz operator using the usual second order, five point stencil. The absorbing boundary conditions are imposed via a PML, following Bérenger (1994). It is very important that the realization of the ∂_z derivatives at the interfaces is consistent with the discrete Helmholtz equation, hence we resort to a discrete version of Green's representation formula to obtain the correct formulas for the discrete layer potentials.

Precomputation

To extract the Green's functions and to compute the local solutions, a pivoted sparse LU factorization is performed for each slab, and the LU factors were saved in memory. The LU factors are independent across slabs, so they can be stored in different cluster nodes (within an MPI environment). The local solves require no communication between nodes. This step is independent of f and is performed once.

Compression

Call N the total number of unknowns in the volume. All operations within the inner loop of the preconditioner are interface-to-interface. Once discretized, they are reduced to matrix-vector multiplications where the vector has size $O(\sqrt{N})$. Although the discretized counterparts of Eq. 6 – 9 involve dense matrices, for which direct summation requires $O(N)$ operations, it is known from the work of Demanet and Ying (2012) that these kinds of matrices can be compressed and applied fast in partitioned low-rank (PLR) form (see Jones et al. (1994)).

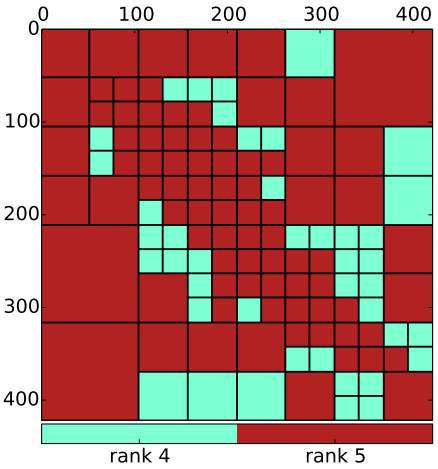

Figure 2: Illustration of the compressed Green's matrix under PLR form ($\text{rank}_{\text{sub-blocks}} \leq 5$, $\varepsilon = 10^{-6}$). Each color represent the numerical rank of each block.

For a desired accuracy ε and maximum rank $R_{\max}$, the matrices resulting from the discretization of the interface-to-interface operators are compressed using an adaptive block partitioning along a quadtree until the ε-rank of the leaf blocks is less than or equal to $R_{\max}$.

Fig. 2 illustrates a PLR matrix for the discrete operator $S_i^{\uparrow}$. We observe that the blocks tend to be larger outside the diagonal, in part because of the presence of the PML. Once the matrices are compressed in PLR form, the matrix-vector multiplication is simple and fast.

COMPLEXITY

We summarize the complexity of each step in Table 1, in which L is the number of slabs in the partition, N is the total number of unknowns, and $n = \sqrt{N}$. For the five-point discretization of the 2D Laplacian, the sparse LU factorization with nested dissection is known to have $\alpha = 3/2$, and back-substitution is known to have linear complexity up to logarithmic factors (see George (1973)), so $\beta \approx 1$. The value for γ depends on the scaling $\kappa(n)$. Indeed, it can be shown that if $\kappa \sim n$ (constant number of points per wavelength) then a perfectly discretized Green's matrix can be compressed so that $\gamma = 3/4$. However, we only have access to an approximation of the Green's function; the numerical errors present in that approximation hinder the compressibility of the operators for large κ.

The second-order five point stencil scheme has a truncation error dominated by $h^2(\partial_x^4 + \partial_y^4)u \sim (\kappa/c)^4 h^2$, and given that $h \sim 1/n$, we need $\kappa \sim \sqrt{n}$ in order to have a bounded error in the approximation of the Green's function. In general, the scaling needed for a p-th order scheme to obtain a bounded

Step	N_{nodes}	Complexity per node
LU factorizations	$O(L)$	$O((N/L)^{\alpha})$
Green's functions	$O(nL)$	$O((N/L)^{\beta})$
Local solves	$O(L)$	$O((N/L)^{\beta})$
Sweeps	1	$O(LN^{\gamma})$
Recombination	$O(L)$	$O((N/L)^{\beta})$

Table 1: Complexity of the different steps of the preconditioner. $\alpha > 1$ and $\beta > 1$ are parameters depending on the factorization algorithm used and in the sparsity pattern of the matrix. $\gamma < 1$ for all of our experiments.

truncation error is $\kappa \sim n^{\frac{p}{p+2}}$. If $\kappa \sim \sqrt{n}$ then it can be shown that $\gamma = 5/8$. In that case, the estimate $O(N^{0.625})$ for the sequential complexity is sublinear with respect to N.

Within an HPC environment, it is possible to achieve sublinear complexity for the online part of the solver (i.e., ignoring the precomputation of the LU factors). If L scales as a fractional power of N, i.e., if $L \sim N^{\delta}$, then the overall execution time is $O(N^{\max(\delta+\gamma,(1-\delta)\beta)})$. For $\gamma = 5/8$ and $0 < \delta < 3/8$, the algorithm runs in sublinear time.

NUMERICAL EXPERIMENTS

Smooth Media

For a random smooth medium with contrast $c_{\max}/c_{\min} = 5$, we benchmark the solver implemented in Matlab. The number of iterations required for a reduction of the residual by 10^{-4} remained between 4 and 5 independent of $\kappa/c_{\min}$ and L (keeping the physical width of the PML constant).

We recorded the execution time of the inner iteration, for different scalings of the frequency with respect to n. Fig. 4 shows that for $\kappa \sim \sqrt{n}$ the execution time scales almost as $O(\sqrt{N})$, which is sublinear. For $\kappa \sim n^{2/3}$ the scaling is higher although it remains sublinear. However, for $\kappa \sim n$ we can observe that as N increases, the complexity deteriorates. For large N the scaling becomes almost linear, due to the inaccurate approximation of the Green's functions, which introduces high frequency numerical errors, hindering the compressibility of the blocks.

Rough Media

For experiments in rough media we chose the left side of the BP model (Billette and Brandsberg-Dahl (2005)). The original model was downsampled without any smoothing. Table 2 shows the number of iterations required to reduce the relative residual to 10^{-6} and the execution time for one inner iteration. Table 2 shows that in rough media the number of iterations is weakly dependent on the frequency and on the number of subdomains; and that the execution time scales sublinearly with respect to the number of unknowns. The same experiment was performed for the Marmousi2 model (G. Martin and Marfurt (2006)), obtaining the same behavior: sublinear execution time and no more than 5 iterations for convergence, for a large range of frequencies and number of sub-domains.

N	$\omega_{\max}$[Hz]	$L = 5$	$L = 15$	$L = 25$
263^2	1.89	**(5)** 1.299	**(7)** 3.727	**(7)** 6.438
526^2	2.64	**(5)** 2.501	**(7)** 5.854	**(7)** 11.39
1051^2	3.77	**(5)** 5.779	**(8)** 7.841	**(8)** 15.03
2100^2	5.40	–	**(7)** 19.22	**(9)** 24.59
2100^2	10.9	–	**(9)** 32.72	**(11)** 38.23

Table 2: Number of inner iterations (bold) required to reduce the relative residual to 10^{-6}, along with execution time (in seconds) for different N and L. Solver is applied to the left side of BP model and frequency is scaled such that $\omega \sim \kappa \sim \sqrt{n}$. The matrices are compressed using $\varepsilon = 10^{-6}$ and $\text{rank}_{\text{rmax}} = 5$.

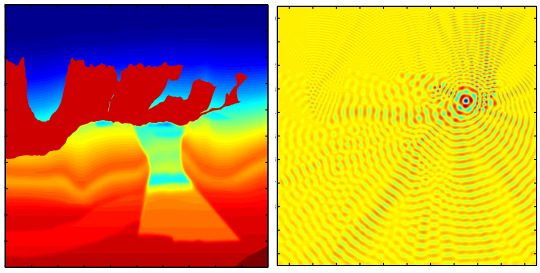

Figure 3: (left) Patch of the BP model used, (right) wavefield generated at 10.9[Hz] with $L = 25$ and $N = 2100^2$.

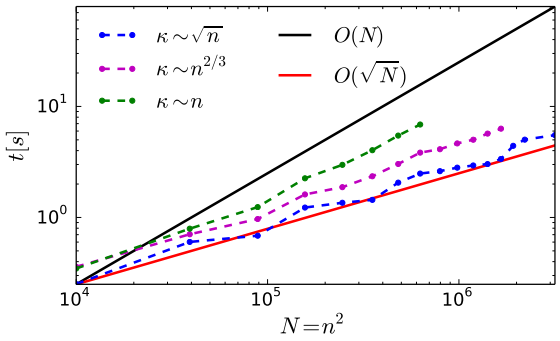

Figure 4: Execution time for one application of the preconditioner, for different scalings $\kappa \sim n^{1/2}$, $\kappa \sim n^{2/3}$ and $\kappa \sim n$.

DISCUSSION

The numerical method described in this note handles interface data in a manner loosely analogous to other methods such as Engquist and Ying (2011a,b), and most importantly Stolk (2013), but results in a preconditioner that effectively switches the computational burden from a sequential iteration to a completely parallel precomputation of local Green's functions. The sublinear complexity for the online step is unlike that of any other method, as far as the authors are aware.

ACKNOWLEDGMENTS

The authors thank Total SA for support. LD is also supported by AFOSR, ONR, and NSF.

REFERENCES

Bérenger, J.-P., 1994, A perfectly matched layer for the absorption of electromagnetic waves: Journal of Computational Physics, **114**, 185 – 200.

Billette, F., and S. Brandsberg-Dahl, 2005, The 2004 BP velocity benchmark.: EAGE.

de Hoop, M. V., S. Wang, and J. Xia., 2011, On 3D modeling of seismic wave propagation via a structured parallel multifrontal direct Helmholtz solver: Geophysical Prospecting, **59**, 857–873.

Demanet, L., and L. Ying, 2012, Fast wave computation via Fourier integral operators: Mathematics of Computation, **81**, 1455–1486.

Després, B., 1990, Décomposition de domaine et problème de Helmholtz: C.R. Acad. Sci. Paris Ser. I Math., **311**, 313–316.

Engquist, B., and L. Ying, 2011a, Sweeping preconditioner for the Helmholtz equation: Hierarchical matrix representation: Comm. Pure Appl. Math., **64**, 697–735.

———, 2011b, Sweeping preconditioner for the Helmholtz equation: moving perfectly matched layers: Multiscale Model. Simul., **9**, 686–710.

Erlangga, Y. A., C. W. Oosterlee, and C. Vuik, 2006, A novel multigrid based preconditioner for heterogeneous Helmholtz problems: SIAM J. Sci. Comput., **27**, 1471–1492.

G. Martin, R. W., and K. Marfurt, 2006, An elastic upgrade for Marmousi: The Leading Edge, Society for Exploration Geophysics, **25**.

George, A., 1973, Nested dissection of a regular finite element mesh: SIAM Journal on Numerical Analysis, **10**.

Hiptmair, R., and C. Jerez-Hanckes, 2012, Multiple traces boundary integral formulation for Helmholtz transmission problems: Advances in Computational Mathematics, **37**, 39–91.

Hsiao, G., O. Steinbach, and W. Wendland, 2000, Domain decomposition methods via boundary integral equations: Journal of Computational and Applied Mathematics, **125**, 521 – 537.

Jones, P., J. Ma, and V. Rokhlin, 1994, A fast direct algorithm for the solution of the laplace equation on regions with fractal boundaries: J. Comput. Phys., **113**, 35–51.

Stolk, C., 2013, A rapidly converging domain decomposition method for the Helmholtz equation: Journal of Computational Physics, **241**, 240 – 252.

Vion, A., and C. Geuzaine, 2014, Double sweep preconditioner for optimized Schwarz methods applied to the Helmholtz problem: Journal of Computational Physics, **266**, 171 – 190.