CFL CONDITION vs NYQUIST CONDITION

The LOD method’s main advantage is that it is not constrained by the CFL condition, which can be much more restrictive than the Nyquist condition. In order to understand why that is the case, consider that $\omega_{\max}$ is also the maximum frequency contained in the acoustic wavefield, i.e., the frequency above which the temporal Fourier transform becomes negligible. The corresponding maximum wave number occurs in regions where the wave speed is minimal and obeys $k_{\max} = \omega_{\max}/c_{\min}$, with corresponding wavelength $\lambda_{\min} = 2\pi/k_{\max}$. If $q$ points per wavelength are sufficient to resolve $\lambda_{\min}$, then $\Delta t \leq \lambda_{\min}/q$. However, the grid must also resolve the fine-grained heterogeneity of the medium $L$, which gives $\Delta t \leq L$. Together with the CFL condition, these restrictions on grid spacing imply

$$\Delta t \leq \min \left[ \frac{1}{\alpha} \frac{\epsilon_{\min}}{q}, \frac{2\pi}{\omega_{\max}}, \frac{\lambda_{\max}}{\epsilon_{\max}} \right].$$

Thus, the maximum time step due to the CFL condition may be substantially smaller than the maximum time step due the Nyquist rate, $\beta 2\pi/\omega_{\max}$. This situation happens when, for example:

- the velocity profile has a high contrast $c_{\max}/c_{\min}$,
- the fine-grained details in the velocity profile have length scale smaller than the representative wavelength $\lambda_{\min}$,
- $q$ is large, to ensure accuracy of the spatial discretization$^2$,
- a high-order spatial discretization, for example a $p$-order discontinuous Galerkin with, e.g., $p > 1$, which causes $\alpha$ to be small,

---

1. In practice, $q = 6$ is a good choice.
2. Finite differences of order $p$ require $O((\Delta t)^{-1/p})$ grid points per wavelength, per dimension, to reach a fixed accuracy level.
Let $m$ be the solution of the acoustic wave equation, with constant density, given by

$$m(x) \partial_t u(x,t) = \Delta u(x,t) + f(x,t), \quad (1)$$
$$u(x,0) = 0, \quad (2)$$
$$\partial_t u(x,0) = 0, \quad (3)$$

where $x = (x,z)$, $m(x) = 1/c^2(x)$ is the model with units of squared slowness, and $f(x,t) = g(x)w(t)$ is a separable source with point impulse $g$ and essentially bandlimited wavelet $w$. For now, we disregard spatial boundary conditions. Without loss of generality, we assume that $\Delta x = \Delta z$.

While there are many different approaches for discretizing (1)–(3), in this paper we use finite differences to approximate the derivatives both in time and in space. Let $r^n = n\Delta t$ and $h = \Delta x$, and let $u^n$ and $m$ be the discrete approximations of $u(x,t^n)$ and $m(x)$, respectively. Let $\delta_{xx}^n$, $\delta_{xz}^n$, $\delta_{xz}^n$, and $\delta_{z}^n$ be spatial finite difference operators, of order $p$, for the first $(x)$ and second derivatives $(x,z)$, respectively. Then, the simplest explicit time-stepping scheme for approximating solutions to (1) is

$$m u^{n+1}_{x} = m \left( 2u^n_{x} - u^{n-1}_{x} \right) + \Delta t^2 \left( \delta_{xx}^n + \delta_{xz}^n \right) u^n + f^n, \quad (4)$$

which is second order accurate in time when $u$ is smooth enough. Each time step in this scheme is computationally inexpensive, requiring only $O(N)$ floating point operations and $O(N)$ storage, where $N = N_x \times N_z$ is the number of unknowns. However, this scheme is limited by the CFL condition.

Implicit methods operate on the principle that the terms involving spatial differences can also be evaluated at time $t^{n+1}$, which results in a linear system of equations for $u^{n+1}$. ADI methods split the time step into two or more sub-steps and treat a single dimension implicitly in each sub-step. As a result, the linear systems due to the implicit part are simple, tightly banded systems.

The LOD method we describe in this article has similar structure, but the sub-steps are conceptually different. Disregarding boundary conditions, the LOD scheme is

$$m - \eta \Delta t^2 \delta_{xx}^h \tilde{u} = m \left( 2u^n - u^{n-1} \right) + \Delta t^2 \left( \delta_{xx}^h \eta u^n + P \right), \quad (5)$$

$$m - \eta \Delta t^2 \delta_{zz}^h u^{n+1} = m \tilde{u} + \Delta t^2 \left( \delta_{zz}^h u^n \right), \quad (6)$$

where $\eta$ is a constant, $u^n = ((1 - 2\eta)u^n + \eta u^{n-1})$. This LOD scheme is unconditionally stable if $\eta > 0.25$ (Geiser, 2008).

In contrast to traditional ADI methods (Lees, 1962; Fairweather and Mitchell, 1965), the sub-step solution $\tilde{u}$ is not a solution at $r^{n+1/2}$, rather, it is a full step approximation of the solution at $r^{n+1}$, which is then corrected by (6). Moreover, the LOD method uses information from a single derivative term at each step, whereas ADI methods usually involve information from both derivative terms at each half-time step.

The LOD scheme is easily extended to 3D, by adding a third equation to (5) and (6), to account for the correction in $z$. The stability condition on $\eta$ remains unchanged in the 3D case.

**LOD formulation with PML.**

Open boundary conditions for wave equations are approximated by an absorbing condition on the boundary of the computational domain. A naive truncation of the domain introduces unphysical reflections that pollute the solution and cause severe imaging artifacts. One approach for mitigating unphysical reflections is to pad the domain with an absorption layer. An efficient realization of such a scheme is the perfectly matched layer (PML), introduced by Bérenger (1994). The PML approach involves complex coordinate stretching, which results in the introduction of auxiliary wavefields to absorb the energy of the outgoing wave. Because it requires fewer auxiliary wavefields, we consider the PML method for the scalar wave equation introduced by Sim (2010),

$$m (\partial_t u + \sigma_{z} \partial_z u + \sigma_{x} \partial_x u) = \Delta u + \sigma_x \partial_x u, \quad (7)$$

$$\sigma_x \partial_x u = -\sigma_x \partial_x u, \quad (8)$$

$$\sigma_z \partial_z u = -\sigma_z \partial_z u, \quad (9)$$

$$\phi_x(x,0) = 0, \quad \phi_z(x,0) = 0, \quad u(x,0) = 0, \quad \partial_x u(x,0) = 0, \quad (10)$$

where $\sigma_x$ and $\sigma_z$ are the PML profile functions, $\sigma_{z} = \sigma_x$, $\sigma_{x} = \sigma_z$, etc., and $\phi_x$ and $\phi_z$ are the auxiliary wavefields.

**LOD solver for the wave equation with PML.**

We extend the LOD solver in (5) and (6) to the PML-augmented scalar wave equation in (7)–(10). Discretizing (7)–(10) as before leads to a system of equations for $u$ and the the discrete auxiliary wavefields, $\phi_x$ and $\phi_z$. In the PML formulation, the evolution equations for $\phi_x$ and $\phi_z$ involve spatial operators only in $x$ and in $z$, respectively. Thus, with the auxiliary fields already decoupled and the property that the LOD method treats dimensions independently, we formulate a decoupled system defined by the operators

$$L^1 = \begin{bmatrix} m - \eta \Delta t^2 \delta_{xx}^h & -\Delta t^2 \delta_{xx}^h \\ -\Delta t \eta \sigma_{z} \partial_z \delta_{xx}^h & I + \gamma_z \end{bmatrix}, \quad (11)$$

$$L^2 = \begin{bmatrix} m(I + \gamma_{z} \partial_z) - \eta \Delta t^2 \delta_{zz}^h - m \sigma_{z} & -\Delta t^2 \delta_{zz}^h \\ -\Delta t \eta \sigma_{z} \partial_z \delta_{zz}^h & I + \gamma_z \end{bmatrix}, \quad (12)$$
in which $L^s$ and $L^c$ contain only spatial differences in $x$ and in $z$ respectively, and where $\gamma_k = \Delta t \sigma / 2$, $\gamma_z = \Delta t \sigma_z / 2$, and $\gamma_{k+z} = \Delta t \sigma_{k+z} / 2$.

The complete LOD scheme for the PML formulation is

$$
L^s \begin{bmatrix}
\tilde{u}^n_{z + 1} \\
\phi^n_{x + 1}
\end{bmatrix} = \begin{bmatrix}
\mathbf{m}(2u^n - u^{n-1}) + \Delta t^2 (\delta^h_{x} w^n_{x} + \Phi^t) \\
(1 - \gamma_k) \phi^n_x + \Delta t \sigma_{k,z} \delta^h_{x} u^n_{x}
\end{bmatrix},
$$

and

$$
L^c \begin{bmatrix}
u^{n+1} \\
\phi^{n+1}_{z + 1}
\end{bmatrix} = \begin{bmatrix}
\bar{m} \bar{u} + \Delta t^2 (\delta^h_{z} - m \sigma_{z}) u^n_{z} + \gamma_{k+z} u^{n-1} \\
(1 - \gamma_z) \phi^0_z + \Delta t \sigma_{k,z} \delta^h_{z} u^n_{z}
\end{bmatrix},
$$

where $\gamma_{k+z} = \Delta t \sigma_{k+z} / 2$.

For $\eta > 0.25$, we claim that this scheme is stable, provided that $\Delta t < 1 / \max (\|\sigma_k\|_{L_c}, \|\sigma_z\|_{L_c})$. The first restriction is from the boundary-free LOD scheme and the second condition is due to the stiffness of the evolution equations ((8) and (9)) for $\phi$ and $\phi_z$. As a result, the LOD method for the wave equation with PML is no longer unconditionally stable, but the stability requirement due to the PML does not involve the grid spacing $h$, hence the restriction is very mild when compared to the original CFL condition.

Efficient inversion

The matrices defined in (11) and in (12) are expensive to invert as is. Precomputing the inverse requires a large amount of memory because of fill-in. To overcome this difficulty, we precompute the inverse of the Schur complement as a sparse banded matrix. For

$$
L = \begin{bmatrix}
A & B \\
C & D
\end{bmatrix},
$$

we have

$$
L_{\text{Schur}} = A - BD^{-1}C.
$$

Using (16), a closed form expression for the inverse of an operator in the form of (15) (e.g., (11) and in (12)) is

$$
L^{-1} = \begin{bmatrix}
I \\
-D^{-1}C
\end{bmatrix}
\begin{bmatrix}
-L^{-1}_{\text{Schur}} & 0 \\
0 & D^{-1}
\end{bmatrix}
\begin{bmatrix}
I \\
0
\end{bmatrix}.
$$

The block involving $L^{-1}_{\text{Schur}}$ is not necessarily sparse and may be difficult to evaluate, but when $D$ is diagonal and $A$, $B$, and $C$ are sparse and tightly banded, as in (11) and in (12), the Schur complement is also sparse and tightly banded. Thus, for our formulation, the LU factorization of (16) is sparse for both $L^s$ and $L^c$. In practice, solving the Schur complement is done by two back-substitutions using the precomputed LU factors, which are stored in $O(N)$ memory.

As a result, the complexity of solving (13) and (14) in each time step is $O(N)$ and the new method has the same asymptotic complexity as the explicit scheme. In the next section, we examine complexity in more detail.

RESULTS

Complexity

The explicit time-stepping scheme (with PML) requires three sparse matrix-vector products, of size at most $N + M \times N + M$, where $M$ is the number of new degrees of freedom added due to the PML and auxiliary wavefields, and a negligible number of vector arithmetic operations. $M$ is usually a percentage of $N$, so we consider the cost of the explicit method to be $O(N)$, per time step$^3$.

The dominant cost for the LOD method is in the $2N_x + 2N_z$ back substitutions, each of which has a respective cost of $O(N_x)$ and $O(N_z)$. When combined with a non-negligible number of vector arithmetic operations and a few smaller sparse matrix-vector products, the overall cost is still $O(N)$, but the constant is larger.

The explicit time-stepping scheme (with PML) requires three wavefields per time index ($u$, $\phi$, and $\phi_z$), and three time snapshots for second order accuracy in time. Additionally, the three sparse update matrices must be stored, at $O(N)$ storage cost. The LOD method requires one additional wavefield ($\tilde{u}$) and 16 sparse matrices, with a maximum number of nonzeros per row that depends on the spatial accuracy of the finite difference approximation (e.g., 5 for fourth-order accuracy).

To understand the range of grid spacing $\Delta t$ and frequency $\omega_{max}$ at which the LOD method would provide an advantage over traditional explicit time-stepping, we apply both methods to the left 22km of the 2004 BP Velocity-Analysis Benchmark (Billette and Brandsberg-Dahl, 2005) with $\Delta t = 12.5$, 25, 50, and 100 m and compare the runtime per time step. The data shown are for $\Delta t$ determined by the CFL condition with $\alpha = 1 / 6$. The same $\Delta t$ was used for both methods so the number of time steps remained constant and both methods are fourth-order accurate in space. All results in this paper are obtained using the authors’ Python Seismic Imaging Toolbox (PySIT). The results in Figure 1 are for final time $T = 5$ s and averaged over 25 runs per grid scale. Empirically, our implementation of the LOD is approximately $r = 3.25$ times slower than the explicit method per time step.

When to use LOD

The LOD method is favored over an explicit method when the smaller overall number of time steps compensates for the additional stepwise complexity. This situation is illustrated in Figure 2, where for $c_{min} = 1500$ m/s, $c_{max} = 4500$ m/s, $\alpha = 1$, $\beta = 1 / 6$, $q = 6$, and $r = 3.25$ we consider the valid ranges for $\Delta t$ and $\Delta x$ for the explicit and LOD methods at $\omega_{max} = 15 \pi$ (resp. $\omega'_{max} = 6 \pi$). In region A (resp. A’), the explicit method is stable and in region B (resp. B’) the LOD method is valid, but the explicit method is more efficient, even at smaller $\Delta x$. In region C (resp. C’), the LOD method is faster than the explicit method. When

$$
r \leq \beta q \frac{1}{\alpha} \frac{c_{max}}{c_{min}},
$$

region C extends below the diagonal and we have that the LOD method is always favored over an explicit method.

$^3$Sparse matrix-vector products have complexity $O(N + n_{mc})$, for matrices with $N$ rows and $n_{mc}$ non-zeros. For this problem, both are on the order of $N$.
LOD Methods

Figure 3 shows the solution to the Helmholtz equation, computed by the Fourier transform of the wavefields determined by the explicit and LOD methods. We used a single 2.5 Hz Ricker wavelet source in the left 22 km of the BP model, with PML boundaries on all sides and a free surface boundary on top. The LOD experiment had equivalent computational cost but required over a third fewer time steps. At higher resolution, the cost reduction is magnified. At low frequency, the two solutions are virtually indistinguishable. At higher frequency, there are differences that we attribute to differing levels of numerical dispersion in the two methods.

Application to Imaging

For this numerical experiment, we use our LOD solver to recover a 370 \times 170 patch of the Marmousi2 p-wave velocity model (G. Martin and Marfurt, 2006). We use 24 sources and 360 receivers, equispaced at the top of the domain with a Ricker wavelet centered at 7.5 Hz. The time-domain least-squares misfit objective function was reduced by approximately two orders of magnitude in 50 L-BFGS iterations from a linearly increasing gradient initial model. The simulated measured data are computed via the explicit method with a smaller time step to prevent the so-called inverse crime. The true model, initial model, and final reconstruction are shown in Figure 4.

CONCLUSION

We present a locally one-dimensional (LOD) solver for the scalar time-dependent wave equation with PML boundary conditions. The method has very favorable stability properties not restricted by the CFL condition. It can be an efficient alternative to explicit time-stepping, e.g., in settings of low frequencies or high velocity contrasts. We demonstrate the potential of the LOD method by contrasting it to an explicit method for the numerical solution of the Helmholtz equation. Full waveform inversion tests also validate the solver.

ACKNOWLEDGEMENTS

The authors thank Total SA for supporting this research. LD is also grateful to the National Science Foundation and the Alfred P. Sloan Foundation. The 2004 BP Velocity-Analysis Benchmark model was provided courtesy of BP and Frederic Billette.

Figure 1: Run time per time step for explicit (green) and LOD (blue) as function of number of unknowns.

Figure 2: Boxes for \( \omega = 15\pi \) (resp. \( \omega' = 6\pi \)) correspond to sampling restrictions in time and space. Region A (resp. A') satisfies the CFL stability condition \( \Delta t \leq \alpha \Delta x/c_{\text{max}} \), region B (resp. B') has LOD method valid, but slower than explicit method with smaller \( \Delta t \), region C (resp. C') has LOD method more efficient than explicit method.

Figure 3: Real part of solution to the Helmholtz equation at 2 Hz, computed by discrete Fourier transform of temporal wavefields from (a) explicit method and (b) LOD method. LOD method is computed with 3 times fewer time steps.

Figure 4: (a) True, (b) initial, and (c) recovered model from time-domain full waveform inversion, using LOD solver, after 50 L-BFGS iterations.
REFERENCES