

Overview of a C++ Helmholtz solver library

Joseph Jennings and John Washbourne

ABSTRACT

SEP received a Helmholtz solver during a visit from Chevron researchers. We describe a high-level overview of the workings of the solver and provide simple examples of how this solver may be used for performing full-waveform inversion.

INTRODUCTION

Since the onset of research on full-waveform inversion (FWI) at SEP, the majority of our implementations have been performed in the time domain (Mora, 1987; Shen, 2014; Biondi and Almomin, 2014). Time-domain methods have the advantage that they are relatively straightforward to implement with explicit finite differences and thus require relatively less memory (Wang et al., 2011). In comparison, frequency domain techniques that solve the Helmholtz equation require factorizing a large, asymmetric, sparse matrix (the Helmholtz operator). In spite of this, frequency-domain methods offer the advantage that they solve the wave-equation for a single frequency, easily allowing for frequency continuation methods that have been used to mitigate cycle-skipping (Bunks et al., 1995). Computationally, they also lend themselves to multiple-source modeling as the factorized matrix can be reused for additional source vectors (Marfurt, 1984). In turn, this property becomes very useful for iterative inversion schemes such as Gauss-Newton and Full-Newton algorithms that require many forward propagations for primary and secondary sources at each iteration (Pratt et al., 1998). For these and other reasons, frequency-domain FWI is becoming more common in the geophysical exploration industry (Washbourne et al., 2013).

As part of the Chevron Center of Research Excellence (CoRE) at Stanford, Chevron researchers visited SEP this summer and shared a Helmholtz solver with SEP researchers. The solver is written in C++ in an object-oriented fashion and utilizes the UMFPACK factorization libraries from SuiteSparse (Davis et al., 2014). The solver consists of three operators: a non-linear forward operator for solving the non-linear wave equation, a linearized operator that maps a perturbation in velocity to a perturbation in the data, and lastly the adjoint of the linearized operator.

In this report, we first give a brief introduction to numerically solving the Helmholtz equation. We then describe the different operators of the Helmholtz solver. With each operator, we provide a simple example for how it might be used in an FWI context.

DISCRETIZATION OF THE HELMHOLTZ EQUATION

We begin with the constant-density acoustic wave equation,

$$\frac{1}{c(\mathbf{x})^2} \frac{\partial^2 u(\mathbf{x}, t)}{\partial t^2} - \nabla^2 u(\mathbf{x}, t) = f(\mathbf{x}, t). \quad (1)$$

If we then take the temporal Fourier transform of equation 1, we arrive at the well-known Helmholtz equation,

$$-\frac{\omega^2}{c(\mathbf{x})^2} u(\mathbf{x}, \omega) - \nabla^2 u(\mathbf{x}, \omega) = f(\mathbf{x}, \omega).$$

Discretizing the Laplacian with second-order finite differences, we write the discrete Helmholtz equation as:

$$\frac{\omega^2}{c_{i,j}^2} u(\omega)_{i,j} + \left[\frac{u(\omega)_{i-1,j} - 2u(\omega)_{i,j} + u(\omega)_{i+1,j}}{\Delta x^2} \right] + \left[\frac{u(\omega)_{i,j-1} - 2u(\omega)_{i,j} + u(\omega)_{i,j+1}}{\Delta z^2} \right] = -f_{i,j}(\omega).$$

Grouping terms by spatial index, we can write the system of equations as a matrix vector product:

$$\mathbf{A} \mathbf{u} = \mathbf{f}, \quad (2)$$

where the coefficients of \mathbf{A} depend on the spatial samplings ($\Delta x, \Delta z$), velocity (\mathbf{c}) and the frequency (ω). Note that \mathbf{A} is nonlinear in the velocity $c(\mathbf{x})$ and is independent for each frequency.

EXAMPLES

The Helmholtz solver contains four principal operators. These operators consist of three wave equation operators: the nonlinear modeling operator, the linearized modeling operator, the adjoint of the linearized modeling operator and a receiver restriction operator that interpolates the wavefield onto the receiver locations. Each of the wave-equation operators has the capability of performing conventional phase and amplitude, phase-only and normalized-amplitude modeling. In the following subsections, we provide examples of how to use these operators on simple synthetic models.

Nonlinear forward operator

The nonlinear forward wave equation operator maps an input source function \mathbf{f} to a data vector \mathbf{d} given a velocity model:

$$\mathbf{d} = \mathbf{R}\mathbf{A}^{-1}\mathbf{f}, \quad (3)$$

where \mathbf{A} is the Helmholtz matrix as defined in equation 2 and \mathbf{R} is a restriction operator that interpolates the wavefield to the receivers. Note that intermediately, we compute the wavefield $\mathbf{u} = \mathbf{A}^{-1}\mathbf{f}$. When we do this, we only need to factorize \mathbf{A} once and thus we can apply \mathbf{A}^{-1} to multiple sources.

As \mathbf{A} is quite large, sparse and asymmetric, sparse LU decomposition algorithms can be used to factorize \mathbf{A} into two lower and upper triangular matrices. In this Helmholtz solver, this is done with the UMFPACK libraries within SuiteSparse.

To factorize \mathbf{A} and perform the nonlinear modeling, we must construct both \mathbf{A} and \mathbf{f} . To construct \mathbf{A} we need to choose a velocity model and frequency. The velocity model for this particular example is shown in Figure 1 and the frequency of choice is 1.0 Hz. For all of these examples, the velocity contains no imaginary component. Note that another advantage of frequency domain modeling is that it is trivial to implement attenuation by providing an imaginary component to the velocity.

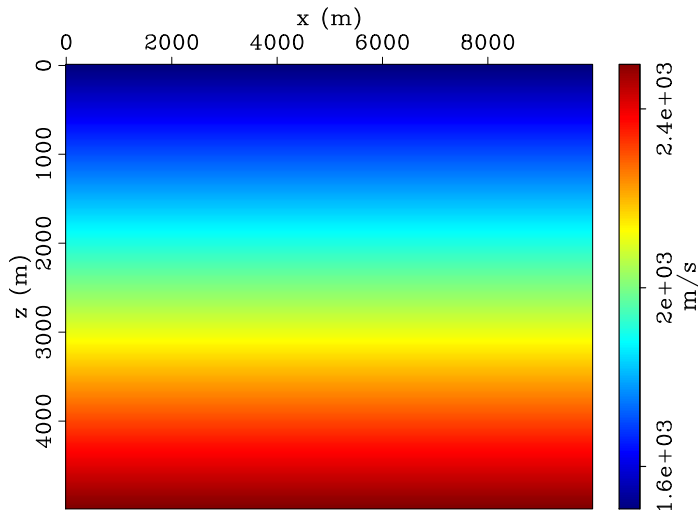


Figure 1: The velocity used in the nonlinear modeling. For this example it contains no imaginary component. [ER]

For velocity model shown in Figure 1 and a frequency of 1.0 Hz, the Helmholtz matrix \mathbf{A} is shown in Figure 2. Note that, \mathbf{A} is banded but not symmetric. This asymmetry is due to the use of one-sided derivatives used in the Engquist/Majda absorbing boundaries. (Ajo-Franklin, 2005; Engquist and Majda, 1977). The width of the bands depends on the length of the stencil used for computing the Laplacian. For higher frequencies and faster velocities, the stencil size must be increased to avoid numerical dispersion. This in turn increases the computational cost of factorizing \mathbf{A} . For this reason this solver implements a compact 9-point stencil that is based on a Padé-like approximation (Singer and Turkel, 1998).

In order to construct the source vector \mathbf{f} and the receiver restriction operator \mathbf{R} , the solver must be provided with the source and receiver coordinates. Note that this solver does not assume that the sources and the receivers are located on the same

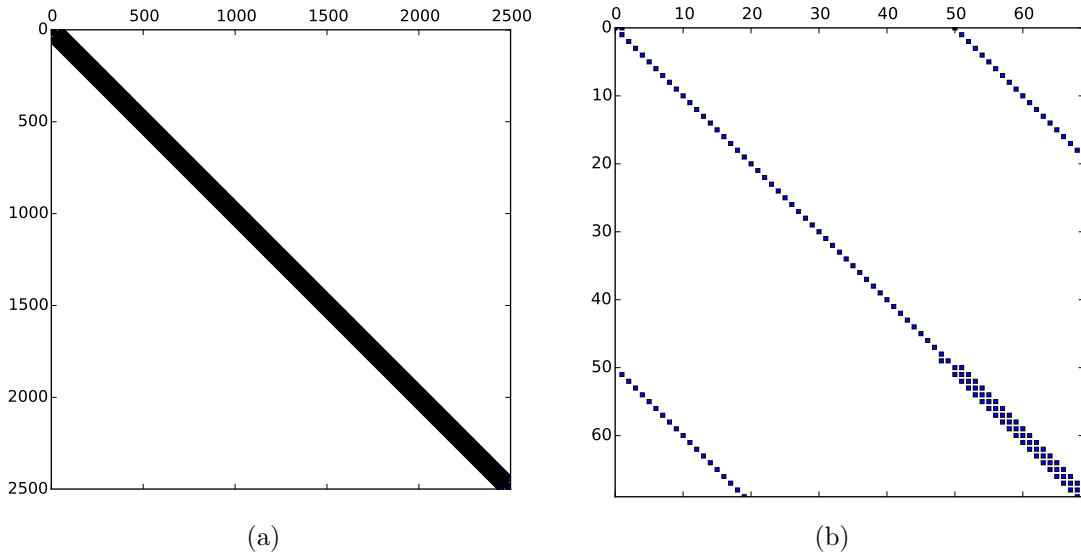


Figure 2: (a) The sparsity pattern of \mathbf{A} . (b) A zoom-in on the upper diagonal of the matrix (first 70 elements). Note the three bands on the main diagonal and the outer fringes. Also note two stray points near the main diagonal. These stray points appear due to the one-sided derivative at the boundary and are the cause of the asymmetry of \mathbf{A} . [ER] [joseph2/. A-0-2499,A-0-69](#)

grid as the velocity. Figure 3 shows the overlay of the acquisition geometry for the nonlinear modeling and also that the sources and receivers are placed on a different grid than the velocity grid.

With each of the components of equation 3 built, we can perform the nonlinear modeling. The real part of the result of the modeling performed is shown in Figure 4(a) and the real part of the recorded data is shown in Figure 4(b).

Linearized operator

We can express the linearized operator in terms of the Helmholtz matrix by first expressing \mathbf{A}^{-1} as a first-order Taylor-series expansion around \mathbf{m}_0 :

$$\mathbf{A}^{-1}(\mathbf{m})\mathbf{f} \approx \mathbf{A}^{-1}(\mathbf{m}_0)\mathbf{f} + \left(\frac{\partial \mathbf{A}^{-1}(\mathbf{m}_0)}{\partial \mathbf{m}} \Delta \mathbf{m} \right) \mathbf{f},$$

where \mathbf{m} is a vector of length nm containing the velocity $c(\mathbf{x})$ and $\mathbf{m} = \mathbf{m}_0 + \Delta \mathbf{m}$. \mathbf{m}_0 is the initial (background) model and $\Delta \mathbf{m}$ is our model perturbation. We can further expand the derivative as:

$$\Delta \mathbf{u} = \left(\frac{\partial \mathbf{A}^{-1}(\mathbf{m}_0)}{\partial \mathbf{m}} \right) \mathbf{f} = \left(-\mathbf{A}^{-1} \frac{\partial \mathbf{A}(\mathbf{m}_0)}{\partial \mathbf{m}} \mathbf{A}^{-1} \Delta \mathbf{m} \right) \mathbf{f}, \quad (4)$$

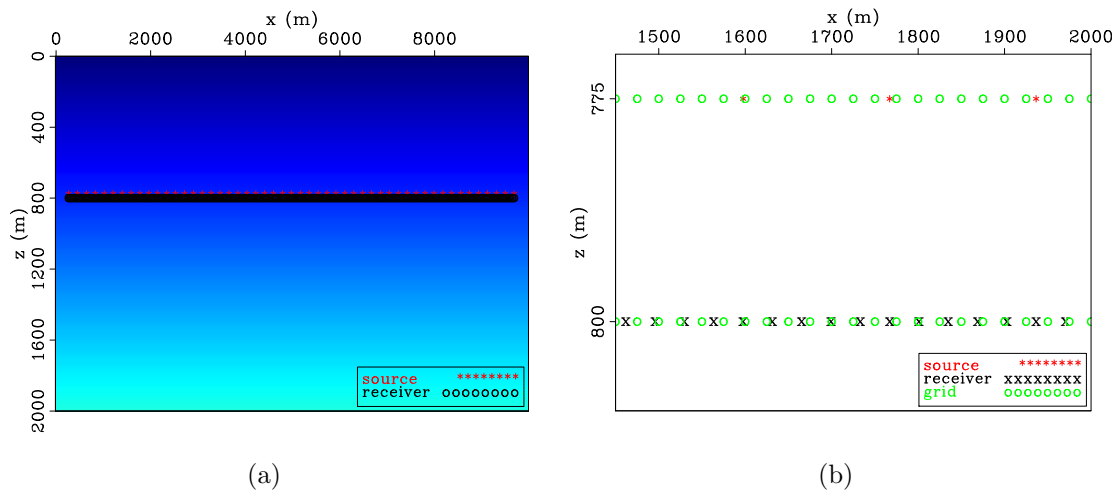


Figure 3: (a) The acquisition geometry for the nonlinear modeling overlain on the velocity model shown previously in Figure 1. Note that the velocity here is shown only to 2000 m in depth. (b) A zoom-in on the source, receiver and velocity grid. Note that the source, receiver and velocity grid all have different spatial sampling intervals and are not co-located. [ER] joseph2/. vel-geom-ovrly,srsrecmod

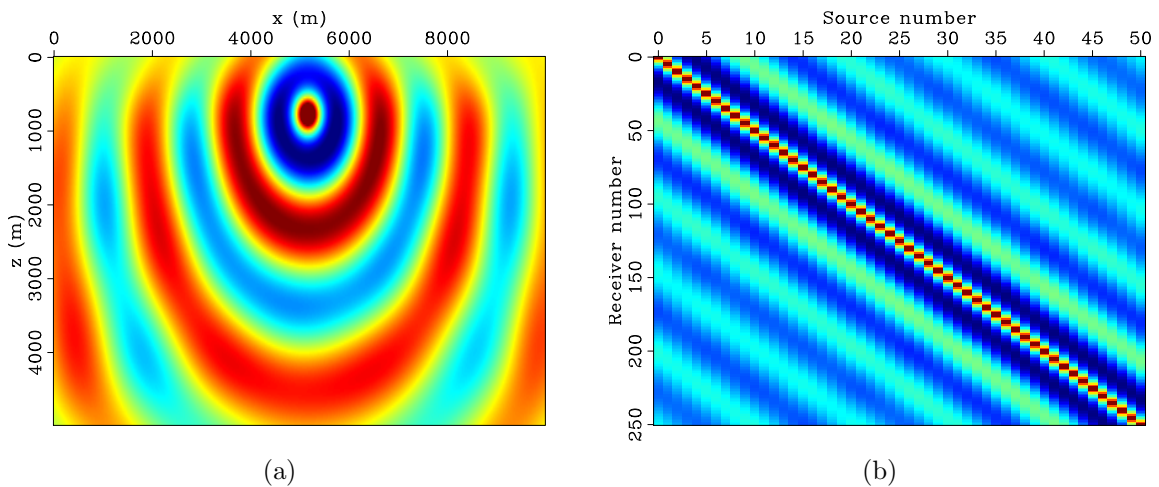


Figure 4: (a) The real part of the wavefield for a wavelet of 1Hz. (b) The real part of the data obtained after modeling at all source locations and the application of \mathbf{R} . [ER]

where $\Delta \mathbf{u}$ is the linearized Born wavefield. As $\partial \mathbf{A} / \partial \mathbf{m}$ is a tensor, we need to sum over the different components of \mathbf{m} to write equation 4 as a chain of operators:

$$\Delta \mathbf{u} = -\mathbf{A}^{-1}(\mathbf{m}_0) \sum_i^{nm} \frac{\partial \mathbf{A}(\mathbf{m}_0)}{\partial m_i} \Delta m_i \mathbf{A}^{-1}(\mathbf{m}_0) \mathbf{f}. \quad (5)$$

At this point in our derivation, the physical interpretation is quite clear. Starting from the rightmost side of equation 5, we have the background wavefield from the application of \mathbf{A}^{-1} to the source \mathbf{f} . We denote this as \mathbf{u}_0 . This is then scaled by a perturbation and a second time derivative is applied with the application of $\frac{\partial \mathbf{A}}{\partial m_i}$. The output of these two operations ($\frac{\partial \mathbf{A}}{\partial m_i} \mathbf{u}_0 \Delta m_i$) is commonly known as the secondary source in linearized modeling, and serves as input to another propagation \mathbf{A}^{-1} , thus resulting in the single-scattered or Born wavefield ($\Delta \mathbf{u}$). If we consider only perturbing one single point in the model space (i.e. $\Delta \mathbf{m}$ contains only one non-zero component), at model point j then we can drop the summation and rewrite equation 5 as:

$$\Delta \mathbf{u} = \mathbf{A}^{-1} \mathbf{D} \mathbf{u}_0 \Delta m,$$

where $\mathbf{D} = \frac{\partial \mathbf{A}}{\partial m_j}$ and $\Delta m = \Delta m_j$. With the application of \mathbf{R} , we now have a linear relationship between a perturbation in the data and a perturbation in the model parameters:

$$\Delta \mathbf{d} = \mathbf{R} \Delta \mathbf{u} = \mathbf{R} \mathbf{A}^{-1} \mathbf{D} \mathbf{u}_0 \Delta m. \quad (6)$$

To perform the linearized modeling in the solver, we use the layered velocity which consists of the smooth background and the perturbation. These are shown in Figure 5. The real part of the wavefield that resulted from applying the chain of operators in equation 5 to the perturbation shown in Figure 5(c) is shown in Figure 6(a). Figure 6(b) shows the recorded data.

Adjoint of linearized operator

With a simple conjugation and reversal of the operators in equation 6 we obtain the expression for the application of the adjoint Born operator that relates a perturbation in the data to a perturbation in the model parameters:

$$\Delta m = \mathbf{u}_0^* \mathbf{D}^* (\mathbf{A}^{-1})^* \mathbf{R}^* \Delta \mathbf{d}.$$

Using the solver, we first attempt to map the data perturbation computed in Figure 6(b) back to the model space. This is shown in Figure 7(a). In this migrated image, we can faintly see the first velocity contrast that occurs at 1500 m depth (Figure 5(c) is included for reference). As a second example, we compute a monochromatic sensitivity kernel for homogeneous velocity that is shown in Figure 8.

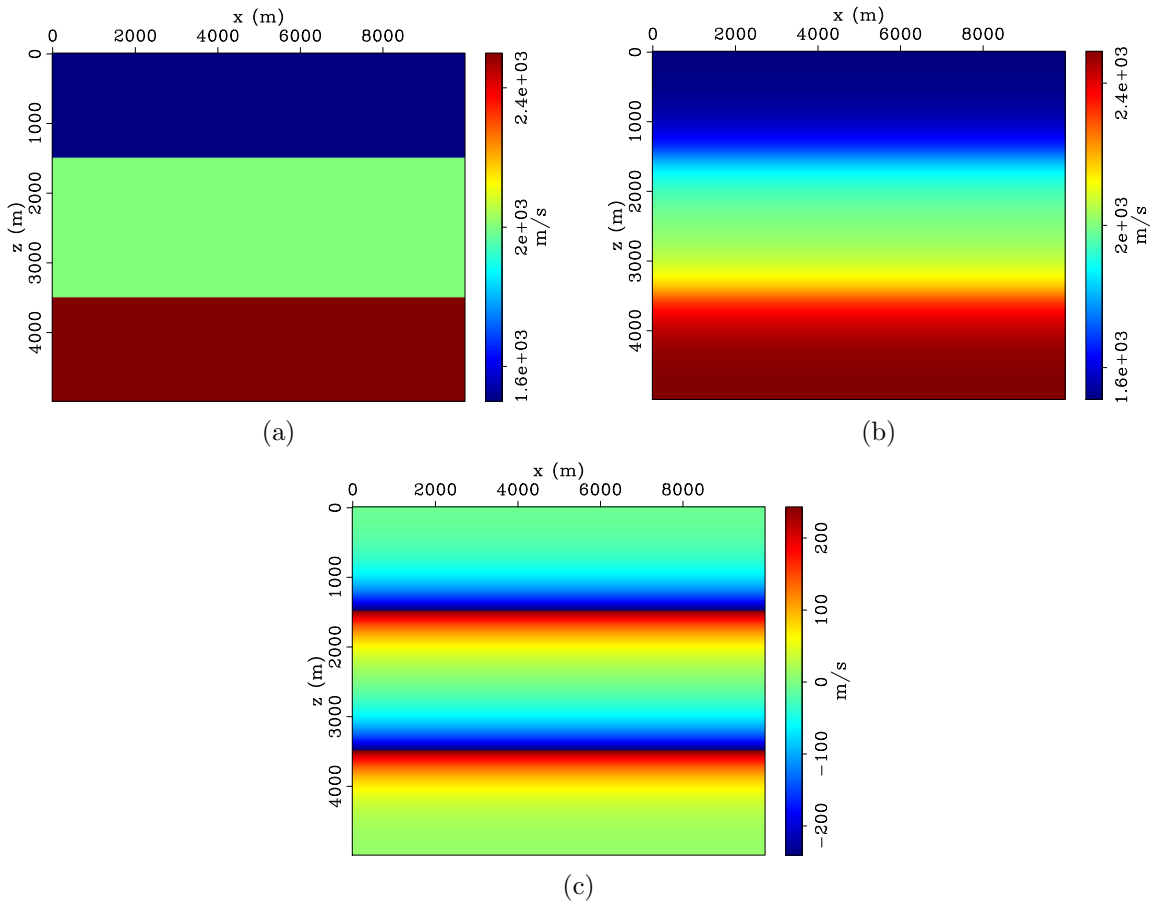


Figure 5: (a) Total velocity model ($\mathbf{m} = \mathbf{m}_0 + \Delta\mathbf{m}$), (b) background velocity \mathbf{m}_0 (low-pass filtered version of (a)) and (c) velocity perturbation $\Delta\mathbf{m}$. [ER]

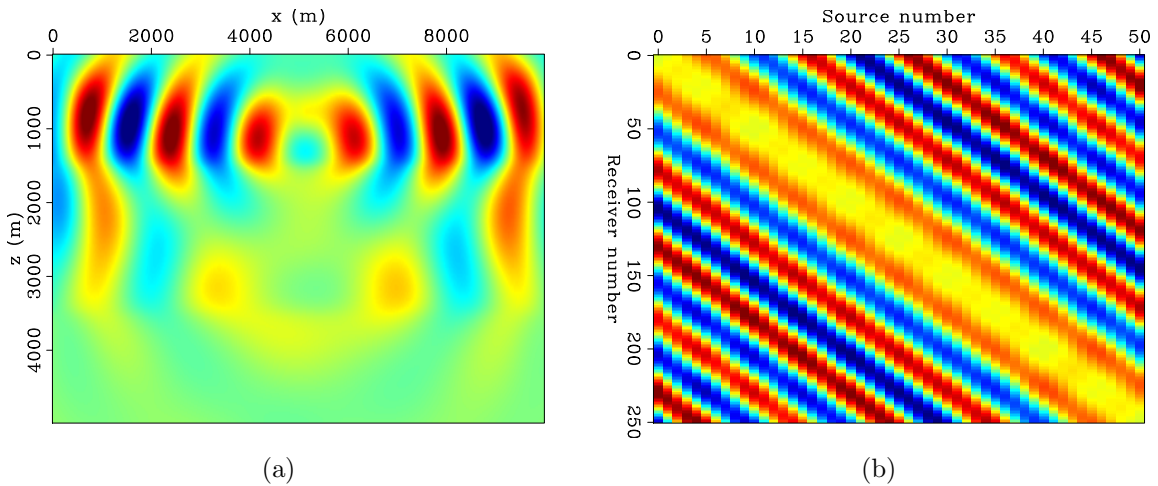


Figure 6: (a) The real part of the Born wavefield modeled $\Delta\mathbf{u}$ as result of applying the chain of operations in equation 5 to to the perturbation shown in Figure 5(c). (b) The real part of the data perturbation obtained after Born modeling at all source locations and the application of \mathbf{R} . [ER]

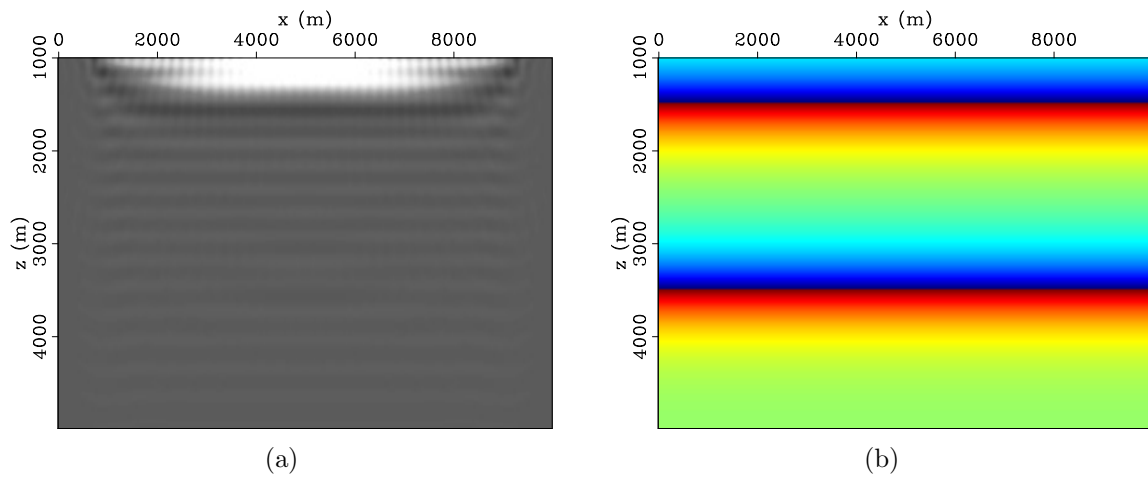


Figure 7: (a) The adjoint of the linearized operator applied to the data perturbation shown in Figure 6(b). (b) The model perturbation in Figure 5(c) shown again for comparison. Note that the image and perturbation are shown from 1000 m depth. Also note the faint reflector at 1500 m depth in the image that corresponds to the first reflector in the model. [ER]

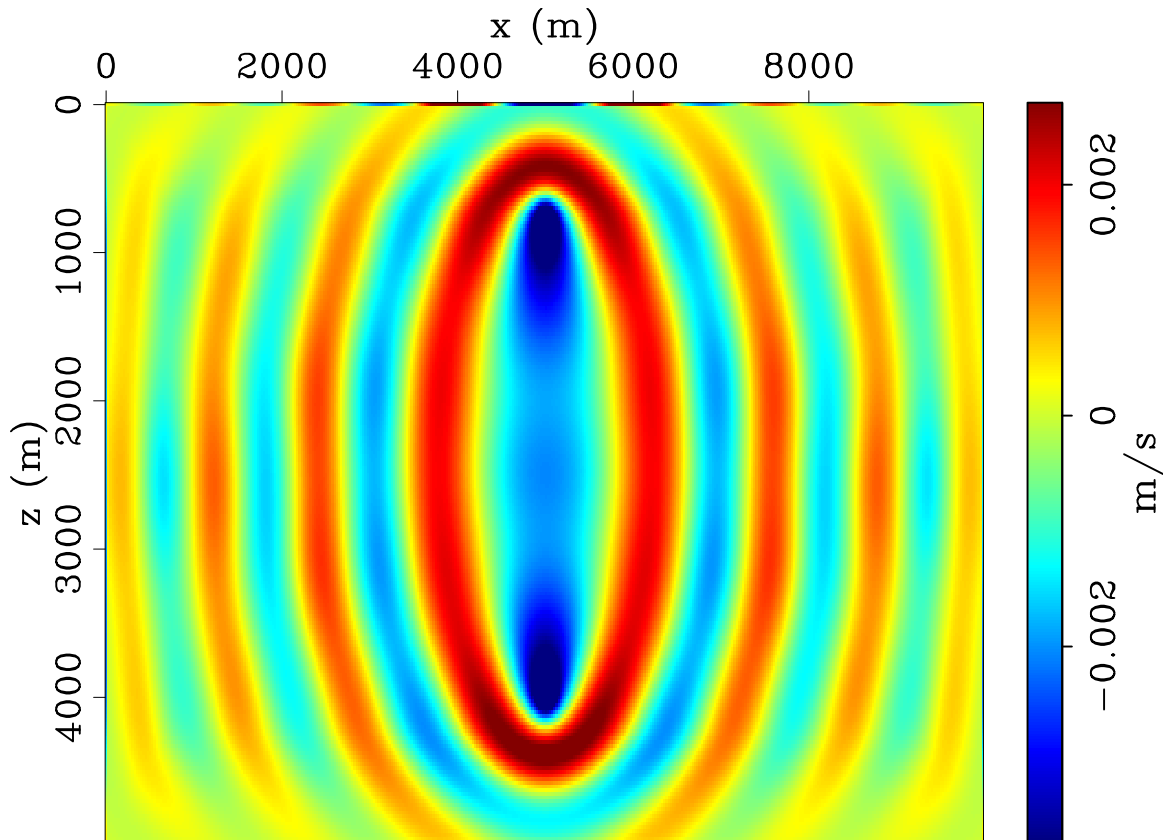


Figure 8: FWI gradient (sensitivity kernel). Note that this example used only one source and one receiver located at 775 m and 4000 m depth respectively. [ER]

CONCLUSION

We have presented a Helmholtz solver written in C++ that can be used for performing frequency domain FWI. This solver contains three main operators that implicitly solve the Helmholtz equation to compute the monochromatic non-linear, linearized and adjoint wavefields and data. While in this report only conventional amplitude plus phase modeling was performed, this solver has the machinery to also perform phase-only and normalized-amplitude modeling as well. While factorizing the Helmholtz matrix is more involved than solving wave equation via explicit finite differences, the frequency domain implementation has many advantages such as frequency-continuation, straight-forward implementation of attenuation and cheap multi-source modeling. We have presented examples of performing forward modeling and computing gradients in hopes that this software may be used among SEP researchers.

ACKNOWLEDGEMENT

The first author would like to thank the sponsors of the SEP for financial and intellectual support. He would also like to thank Rahul Sarkar and Huy Le for their help in deriving the linearized operator.

REFERENCES

- Ajo-Franklin, J. B., 2005, Frequency-domain modeling techniques for the scalar wave equation: An introduction.
- Biondi, B. and A. Almomin, 2014, Simultaneous inversion of full data bandwidth by tomographic full-waveform inversion: *Geophysics*, **79**, WA129–WA140.
- Bunks, C., F. M. Saleck, S. Zaleski, and G. Chavent, 1995, Multiscale seismic waveform inversion: *Geophysics*, **60**, 1457–1473.
- Davis, T., W. Hager, and I. Duff, 2014, Suitesparse: <http://faculty.cse.tamu.edu/davis/suitesparse.html>.
- Engquist, B. and A. Majda, 1977, Absorbing boundary conditions for numerical simulation of waves: *Proceedings of the National Academy of Sciences*, **74**, 1765–1766.
- Marfurt, K. J., 1984, Accuracy of finite-difference and finite-element modeling of the scalar and elastic wave equations: *Geophysics*, **49**, 533–549.
- Mora, P., 1987, Elastic wavefield inversion: PhD thesis, Stanford University.
- Pratt, R. G., C. Shin, and G. Hick, 1998, Gauss–newton and full newton methods in frequency–space seismic waveform inversion: *Geophysical Journal International*, **133**, 341–362.
- Shen, X., 2014, Early-arrival waveform inversion for near-surface velocity estimation: PhD thesis, Stanford University.
- Singer, I. and E. Turkel, 1998, High-order finite difference methods for the helmholtz equation: *Computer Methods in Applied Mechanics and Engineering*, **163**, 343–358.

Wang, C., J. Park, R. Bloor, J. Leveille, and P. Farmer, 2011, Application of time domain and single frequency waveform inversion to real data: Presented at the 73rd EAGE Conference and Exhibition incorporating SPE EUROPEC 2011.

Washbourne, J., N. Shah, and K. Bube, 2013, System and method for data inversion with phase extrapolation. (WO Patent App. PCT/US2012/039,077).