Single frequency 2D acoustic full waveform inversion

Sjoerd de Ridder, Ali Almomin, and Musa Maharramov

ABSTRACT

Single frequency 2D acoustic full waveform inversion shows promise as an approach to inverting the acoustic, dispersive 2D acoustic wave equation for the underlying velocity cube as a function of frequency. A GPU friendly finite difference time domain kernel and an associated frequency domain optimization scheme are shown to retrieve a Gaussian anomaly via full waveform inversion for several acquisition geometries. Although the wave field is single frequency, the spatial distance of sources and receivers allows their sensitivity kernels to interfere and form a gradient that can recover the anomaly.

INTRODUCTION

Surface waves naturally propagate in two dimensions along the earth's surface and not in depth. However, du to their wavelength their waves' motion causes disturbances away from the interface, thus making surface waves sensitive to mechanical properties away from the interface. This is manifested in a variability of the wave propagation speed with frequency, i.e. dispersion. Dispersive Scholte waves can be extracted from ambient seismic noise by cross-correlation (de Ridder and Dellinger, 2011; de Ridder, 2012). Kimman (2011) found that higher modes are often not constructed or very weak, when the excitations are located at the surface. Furthermore, de Ridder (2012) shows how Scholte waves reconstructed from ambient seismic noise at Valhall field contain a single mode dispersive surface wave.

An approximate physical model for a single mode of surface waves is waves travelling in two dimensions given a phase velocity map, $c(\omega, \mathbf{x})$. The amplitudes and phases of surface waves, in that approximation, are governed by

$$\left(\nabla^2 + v^{-2}(\omega, \mathbf{x})\omega^2\right) V(\omega, \mathbf{x}) = F(\omega)\delta(\mathbf{x} - \mathbf{x}_s), \tag{1}$$

where $\mathbf{x} = (x, y)$ and $\nabla^2 = \partial_x^2 + \partial_y^2$. Equation 1 is very similar to a classical 2D acoustic wave equation, except that the velocity is now a function of frequency. Explicitly solving for a single frequency renders this difference moot. However, frequency domain solutions are slow to compute, and implementation of absorbing boundary conditions is not straightforward. Here we solve equation 1 by a time-domain equivalent, valid for individual frequencies and independent of boundary condition.

We want to use equation 1 to image the phase-velocity cube $v(\omega, \mathbf{x})$ by full waveform inversion (FWI) for each frequency separately. But single frequency data generally does not resolve anomalies in space, because sensitivity kernels have endless oscillatory behaviour. Conventionally, summing over a finite frequency band collapses the sensitivity kernels to the classic and familiar banana-doughnut kernels (Dahlen and Nolet, 2000). However, in this paper we show that a similar effect is achieved by having sources and receivers throughout and around the target of interest.

This paper starts by connecting a time domain kernel to solutions of the 2D frequency domain acoustic kernels with frequency dependant velocity. Then a non-linear optimization algorithm is developed to invert single frequency amplitude and phase data. Finally we explore the ability of various acquisition grids to invert for a Gaussian anomaly.

TIME-DOMAIN KERNEL

We seek to compute solutions of equation 1 for single frequencies. Thus assume that we have a time domain source function acting at a single frequency $F(t) = A\ \sin(\omega_o t + \phi)$. In a 2D acoustic system this would excite a wave equation as

$$\left(\nabla^2 + v^{-2}(\omega_o, \mathbf{x})\partial_t^2\right) d(t, \mathbf{x}) = A(\mathbf{x})\ \cos(\omega_o t - \phi(\mathbf{x})), \tag{2}$$

The phase and amplitude of the solution, $V(\omega, \mathbf{x})$, at frequency ω_0 is simply computed with a forward Fourier transform

$$d(\omega_o, \mathbf{x}) = \int_{-\infty}^{\infty} d(t, \mathbf{x})\mathrm{exp}\{i\omega_o t\}\ \mathrm{d}t. \tag{3}$$

If the solution of 2 is computed accurately, then $d(\omega, \mathbf{x}) = 0$ for $\omega \neq \pm\omega_o$. The sourcing terms $A(\mathbf{x})$ and $\phi(\mathbf{x})$ are only non zero where sources act.

To evaluate this inverse Fourier transform, we would need to perform infinite number of time steps and at minus infinity, practically impossible. However, the solution is periodic, thus we can suffice with having computed the wave field for as little as one period. However, we need to initialize the wave field first and wait for the energy of the source to have spread throughout the medium. The computational boundaries could be made absorbing, but this is a challenge, especially at the longer wavelengths that we are interested in modelling. A simpler solution is to extend the domain and omit updating the boundaries. The domain has to be extended large enough to avoid reflections to start modulating the phase throughout the inner area of the model domain that is of interest. This requires a much larger model space, but keeps coding simple and stable.

The kernel is started and run for enough time steps to have the energy of one period to propagate beyond all receivers, assuming one period is enough to stabilize the amplitude and phase of the wave field. This propagation time length is based on a minimum velocity. Then the code steps through two periods while the receiver wave field is kept. We do not need to have a memory variable for the source function or to collect the wave field at the receiver locations. Instead, we upload one sinusoid with a phase from -1π to 3.5π. A simple *mod* function statement reduces any absolute phase to fall within that range. The value for a sinusoid with a phase shift of up to $\pm\pi$ can easily be retrieved. And similarly the value of a cosine is available though a shift of .5π. The discrete inverse Fourier transform is performed on the fly. No memory transfer needs to occur between CPU and GPU while the modelling is running through the time steps.

An upside of this algorithm is that is is embarrassingly parallel over shots and frequencies, not memory intensive and boundary conditions can be neglected. The downside of this approach is that each frequency inversion requires solving a time-domain waveform inversion, hence this approach is suboptimal and is only implemented here because of the straightforward implementation on CUDA of the 2D waveform inversion with the special excitation source 15. An alternative approach to this problem is to solve the waveform inversion for equation 1 in the frequency domain using a Helmhotz solver that lends itself to efficient CUDA implementation.

OPTIMIZATION SCHEME

The FWI objective function J_{FWI} can be written as:

$$J_{\mathrm{FWI}}(\mathbf{v}) = \|\mathbf{d}(\mathbf{v}) - \mathbf{d}_{\mathrm{obs}}\|_2^2, \tag{4}$$

where $\mathbf{v}$ is the velocity model, $\mathbf{d}(\mathbf{v})$ is the computed data, and $\mathbf{d}_{\mathrm{obs}}$ is the observed data. $\mathbf{d}(\mathbf{v})$ is computed as:

$$d(\mathbf{x}_s, \mathbf{x}_r, \omega; \mathbf{v}) = f(\mathbf{x}_s, \omega) G(\mathbf{x}_s, \mathbf{x}, \omega; \mathbf{v}) \delta(\mathbf{x}_r - \mathbf{x}), \tag{5}$$

where $f(\mathbf{x}_s, \omega)$ is the source function, ω is frequency, $\mathbf{x}_s$ and $\mathbf{x}_r$ are the source and receiver coordinates, and $\mathbf{x}$ is the model coordinate. In the acoustic, constant-density case the Green's function $G(\mathbf{x}_s, \mathbf{x}, \omega; \mathbf{v})$ satisfies:

$$\left(\nabla^2 + v^{-2}(\mathbf{x})\omega^2\right) G(\mathbf{x}_s, \mathbf{x}, \omega) = \delta(\mathbf{x}_s - \mathbf{x}). \tag{6}$$

We then separate the model into a background and a perturbation:

$$v^{-2}(\mathbf{x}) = b(\mathbf{x}) + m(\mathbf{x}), \tag{7}$$

where $b(\mathbf{x})$ is the background component, which is the current model in slowness squared units, and $m(\mathbf{x})$ is the perturbation component. After this separation, we can use Taylor expansion on the data around the background component as follows:

$$\mathbf{d}(\mathbf{v}) = \mathbf{d}(\mathbf{b}) + \frac{\partial \mathbf{d}}{\partial \mathbf{v}}|_b \mathbf{m} + ... \tag{8}$$

By neglecting the higher-order terms in the data series, we can define the linearized modeling operator $\mathbf{L}$ as:

$$\Delta\mathbf{d}(\mathbf{v}) = \frac{\partial\mathbf{d}}{\partial\mathbf{v}}|_b\mathbf{m} = \mathbf{L}(\mathbf{b})\mathbf{m}. \tag{9}$$

The first order Born approximation can be used to define the operator:

$$\Delta d(\mathbf{x}_s, \mathbf{x}_r, \omega; \mathbf{b}, \mathbf{m}) = -\omega^2 f(\omega)\sum_{\mathbf{x}} G(\mathbf{x}_s, \mathbf{x}, \omega; \mathbf{b})m(\mathbf{x})G(\mathbf{x}, \mathbf{x}_r, \omega; \mathbf{b}), \tag{10}$$

where the Green's functions now satisfy the acoustic wave equation as follows:

$$\left(\nabla^2 + b(\mathbf{x})\omega^2\right) G(\mathbf{x}_s, \mathbf{x}, \omega) = \delta(\mathbf{x}_s - \mathbf{x}), \tag{11}$$

$$\left(\nabla^2 + b(\mathbf{x})\omega^2\right) G(\mathbf{x}, \mathbf{x}_r, \omega) = \delta(\mathbf{x} - \mathbf{x}_r). \tag{12}$$

We can now compute the model gradient $g(\mathbf{x})$ as follows:

$$g(\mathbf{x}) = \frac{\partial J_{\mathrm{FWI}}}{\partial\mathbf{m}} = \mathbf{L}^*\Delta\mathbf{d}. \tag{13}$$

Finally, we can update the model with the gradient:

$$b_{\mathrm{new}}(\mathbf{x}) = b(\mathbf{x}) - \alpha g(\mathbf{x}), \tag{14}$$

where α is the step size. To estimate the step size, we first evaluate the objective function with the gradient scaled to have a maximum of 2% and 4% of the minimum value of the current model. Using these two points as well as the objective function value at the current model, which is already computed in the gradient calculation, we fit a parabola. If the parabola has positive-side minimum, i.e. both the curvature and the x-axis shift are positive, a new objective function evaluation is performed at the parabola minimum. Then, the two or three evaluations are compared and the scale that resulted in the smallest objective function is used as the step size given that the objective function decreases. Otherwise, the line search is repeated after shrinking the gradient by a factor of 4. The optimization scheme is implemented on the CPU in Fourier domain notations using frequency domain Green's function solutions computed on the GPU.

TEST GEOMETRIES

A series of nine examples of single frequency FWI are run for a Gaussian anomaly, each with different source and receiver geometry and with a 2 Hz wave field. The Gaussian shape has a standard deviation of 350 m and an amplitude of 10% on a background velocity of 1500 m/s (Figure 1).

Having just one source and one receiver (Figure 2) severely limits the ability of the acquisition to retrieve a Gaussian anomaly. The final model, achieved after only

4 iterations, basically resembles the sensitivity kernel of one source-receiver couple. Using two receivers and one source (Figure 3) does not improve the situation much. The two sensitivity kernels now interfere, but together they still don't resemble the Gaussian anomaly we attempt to retrieve. A survey using a line of receivers and a single source starts to do better (Figure 4) but the range resolution is still quite poor. However, an acquisition with one source and a field of receivers suffers from a range interference of the outer side lobes between individual source-receiver sensitivity kernels (Figure 5). Two sources and a line of receivers (Figure 6) drastically improves the range resolution over just one source (Figure 4). And similarly for a field of receivers using just two sources (Figure 7). The next two experiments show the result of a line of sources. The first is essentially a conventional borehole geometry, where a line of sources and a line of receivers flank the target (Figure 8). This is capable of retrieving the the velocity anomaly in the cross-line direction reasonably well, but fails to resolve the Gaussian anomaly in the in-line direction. It actually retrieves negative values as big side lobes of the anomaly. This problem is only partially overcome when there is a field of available receivers (Figure 9). When sources and receivers encircle the anomaly, we achieve excellent retrieval (Figure 10) in just 5 iterations. However, retrieval is limited by the wavelength of the single frequency data.

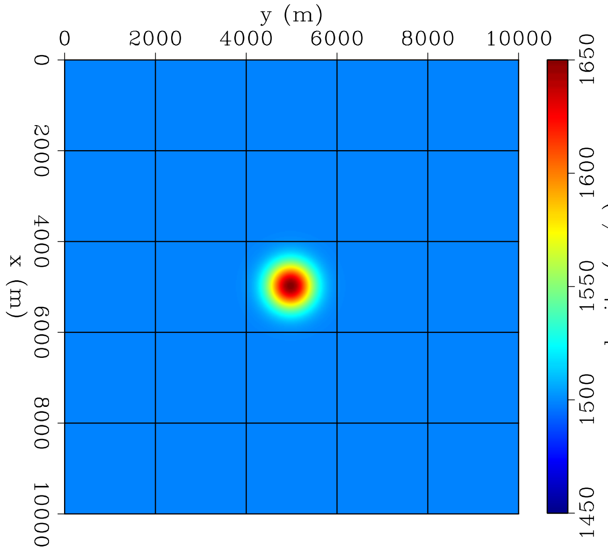


Figure 1: True model. A Gaussian shaped anomaly, with a standard deviation of 350 m and an amplitude of 10% on a background velocity of 1500 m/s. [**ER**]

CONCLUSIONS

A time domain kernel to compute a frequency domain solution of the Green's function is successfully implemented on GPU's. If the source receiver distribution illuminates the target from sufficient geometries, individual sensitivity kernels interfere and construct gradients for updates in a FWI scheme. A single frequency full waveform inversion scheme is not applied in a multi-scale velocity refinement as is the standard for the waveform inversion process, but the approach is still able to recover Gaussian anomalies in space.

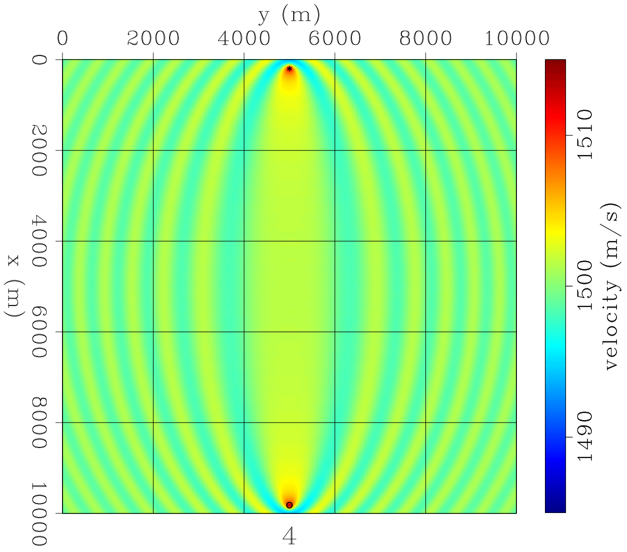


Figure 2: Model 1: One source and one receiver. FWI converged after 4 iterations. A classical single frequency sensitivity kernel. [**CR**]

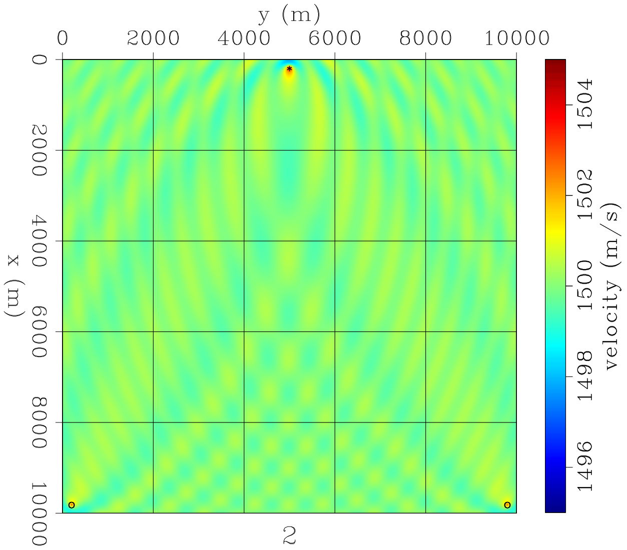


Figure 3: Model 2: One source and two receivers. FWI converged after 2 iterations. The two sensitivity kernels interfere, but together still do not resemble the Gaussian anomaly we attempt to retrieve. [**CR**]

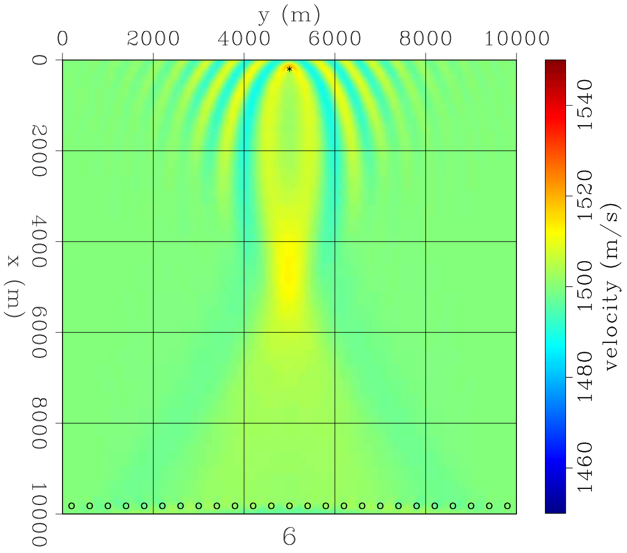


Figure 4: Model 3: One source and a line of receivers. FWI converged after 6 iterations. Poor range resolution, but a velocity anomaly starts to appear in the right location. [**CR**]

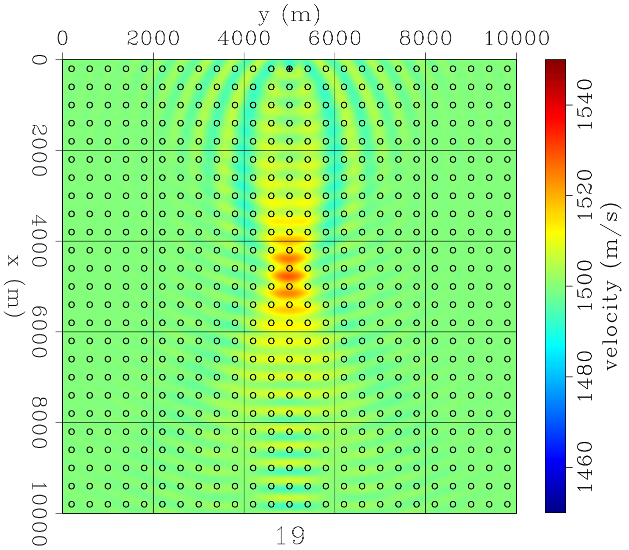


Figure 5: Model 4: One source and a field of receivers. FWI converged after 19 iterations. Although the retrieved anomaly is located correctly, side lobes of individual sensitivity kernels interfere and dominate the result. [**CR**]

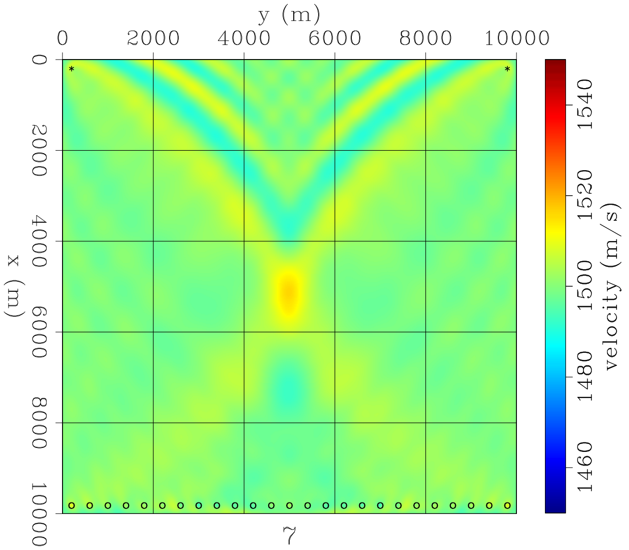


Figure 6: Model 5: Two sources and a line of receivers. FWI converged after 7 iterations. Two sources and a line of receivers drastically improve the range resolution over just one source. [**CR**]

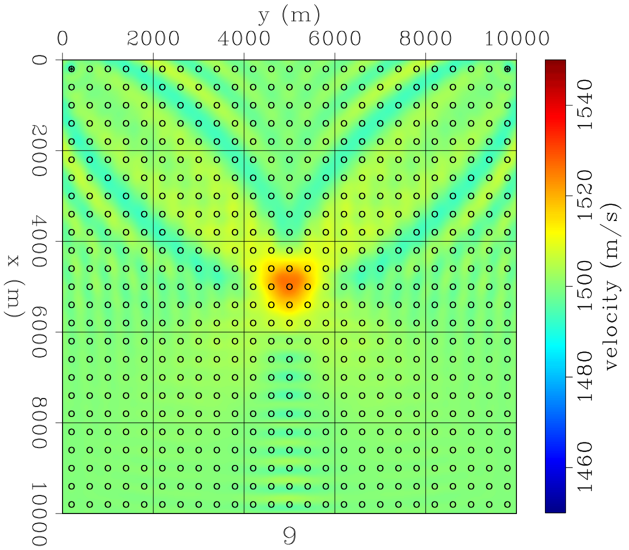


Figure 7: Model 6: Two sources and a field of receivers. FWI converged after 9 iterations. The anomaly is very well shaped in space. [**CR**]

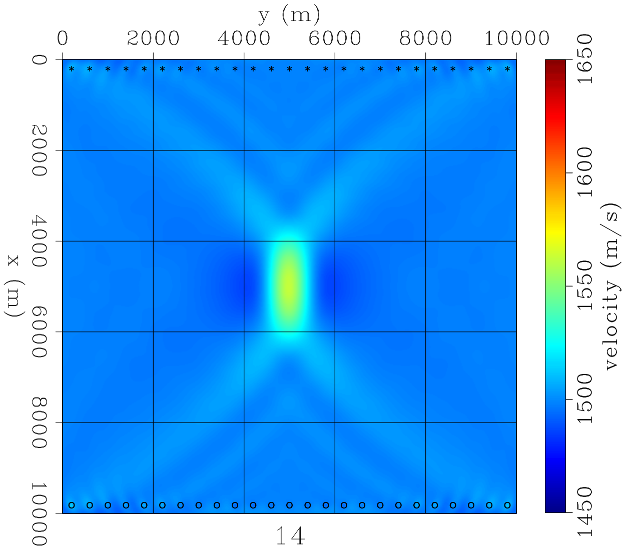


Figure 8: Model 7: A line of sources and a line of receivers. FWI converged after 14 iterations. The anomaly is poorly resolved in range, but there are very few side lobes. [**CR**]

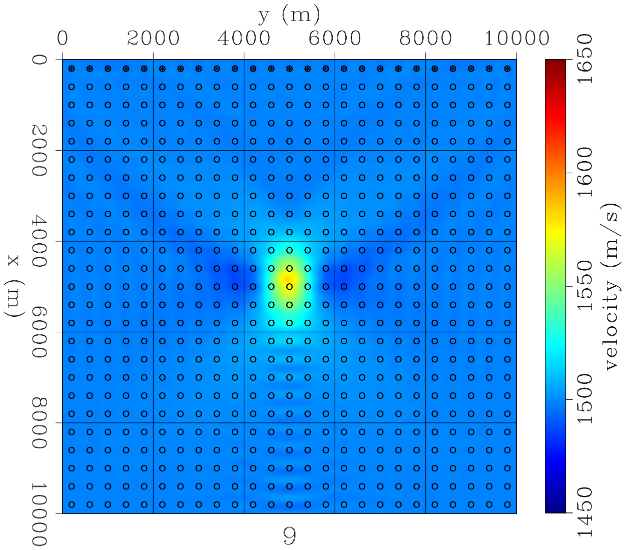


Figure 9: Model 8: A line of sources and a field of receivers. FWI converged after 9 iterations. A field of receivers improves the range resolution. [**CR**]

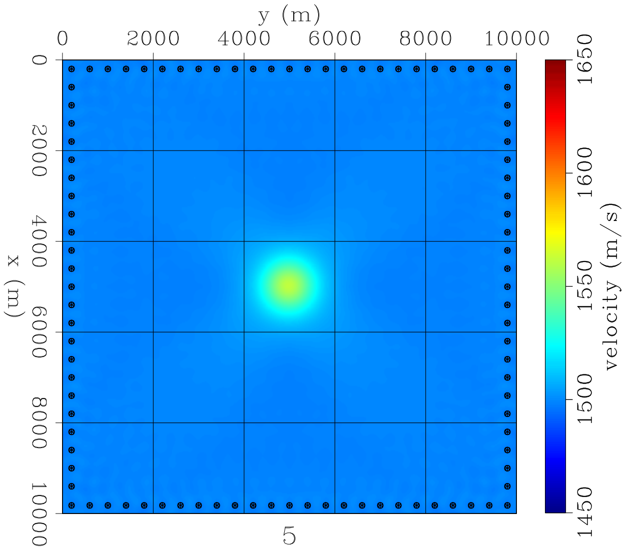


Figure 10: Model 9: A square of sources and receivers. FWI converged after 5 iterations. The anomaly is nearly perfectly reconstructed and without signatures of the individual sensitivity curves. [**CR**]

APPENDIX

Our objective is to invert the frequency-dependent velocity function $c(\omega, \mathbf{x})$ from observed solutions to our *postulated* surface wave propagation equation 1 in the frequency domain, where $F(\omega)$ is an arbitrary excitation source waveform. As has been noted above, 2D waveform inversion can be directly applied to this problem, with the only exception that inversion results for individual frequencies are treated as separate values of the frequency-dependent velocity $c(\omega, \mathbf{x})$ and not applied in a multi-scale velocity refinement of the standard for the waveform inversion process. While the applicability of 2D the waveform inversion is conceptually obvious, technical challenges arise in adapting specific implementations of the waveform inversion to equation 2. Our *time domain* CUDA implementation of the 2D acoustic equation solves equation 2 with an excitation source of the form

$$A \, \sin(\omega_o t + \phi)\delta(\mathbf{x} - \mathbf{x}_s). \tag{15}$$

The choice of the source 15 is dictated by the considerations of our CUDA implementation. We will now show how the time-domain kernel working with sources 15 can be adapted to solve the waveform inversion problem in the frequency domain for an arbitrary waveform source $F(\omega)$. First, Fourier-transforming 2, we can see that recovering the velocity from a solution to 2 with the source 15 is equivalent to recovering the velocity from a solution to

$$\Delta u + \frac{\omega^2 u}{c^2} = \frac{A}{2i}\left[e^{i\phi}\delta(\omega - \omega_0) - e^{-i\phi}\delta(\omega + \omega_0)\right]. \tag{16}$$

By choosing $\phi = \pi/2$ we get

$$\frac{A}{2}\left[\delta(\omega - \omega_0) + \delta(\omega + \omega_0)\right]$$

in the right-hand side of 16. Now assuming that the velocity is an even function of the frequency, and that we attempt to reconstruct the velocity from 2 with $F(\omega) = F(-\omega)$, we can see from equation 16 that we can equivalently run 2D waveform inversion 2 for frequencies $\omega \in [0, Nyq]$ with $A = 2F(\omega), \phi = \pi/2$ in 15. For an odd waveform (but still even velocity) for each ω we use $A = 2F(\omega), \phi = 0$. An arbitrary waveform can be represented as the sum of an even and odd components, and equation 2 will have a linear combination of two source terms 15 in the right-hand side.

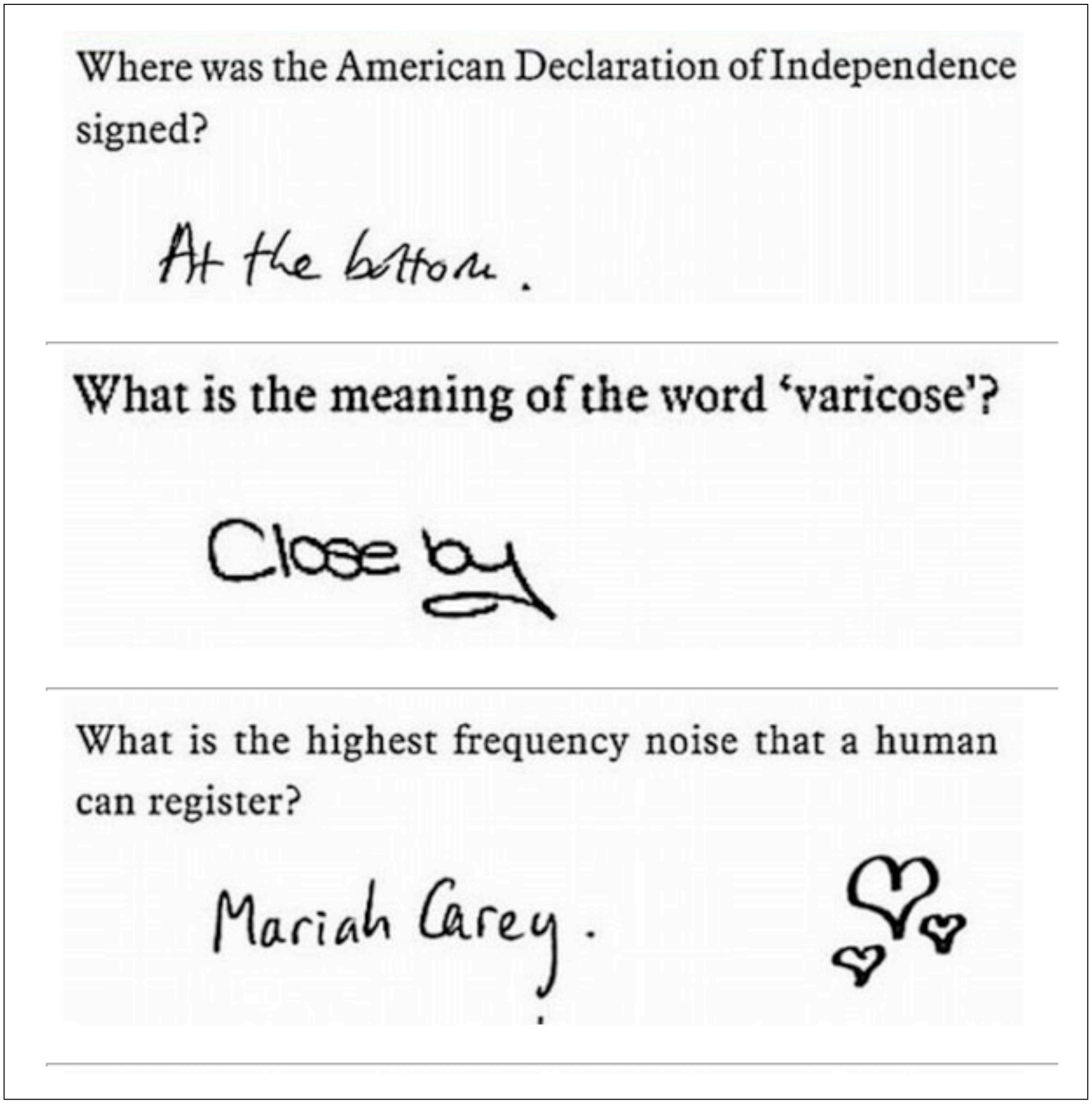
REFERENCES

Dahlen, S.-H. H. F. A. and G. Nolet, 2000, Fréchet kernels for fnite-frequency traveltimes - ii. examples: Geophys. J. Int., **141**, 175–203.

de Ridder, S., 2012, Continuous monitoring by ambient-seismic noise tomography: SEP-Report, **147**, 165–182.

de Ridder, S. and J. Dellinger, 2011, Ambient seismic noise eikonal tomography for near-surface imaging at valhall: The Leading Edge, **30**, 506–512.

Kimman, W. P., 2011, Higher mode surface wave interferometry: PhD thesis, Stanford University.



Where was the American Declaration of Independence signed?

At the bottom.

What is the meaning of the word 'varicose'?

Close by

What is the highest frequency noise that a human can register?

Mariah Carey.