



## Module 10: Numerical Solutions to Hyperbolic PDEs

This module focuses on generating numerical solutions of **hyperbolic** partial differential equations (PDEs). Recall that these types of second-order PDEs

$$Au_{xx} + 2Bu_{xy} + Cu_{yy} + Du_x + Eu_y + Fu + G = 0 \quad (1)$$

are defined by the condition  $B^2 - AC > 0$  (where  $y$  is considered the  $t$  variable and thus  $A = 1$ ,  $C = -\frac{1}{c}$  and all others are equal 0).

Three interesting examples (with increasing complexity) are the following:

- **Acoustic wave equation (Homogeneous medium):** The acoustic wave equation (AWE) is normally used to model the pressure disturbance  $\phi = \phi(x, y, z, t)$  generated by an acoustic wave propagation through a fluid medium defined by constant velocity  $v = \sqrt{\frac{K}{\rho}}$  where  $K$  is the bulk compressibility and  $\rho$  is the density.

$$\frac{\partial^2 \phi}{\partial t^2} - v^2 \nabla^2 \phi = F(x, y, z, t) \quad (2)$$

where  $F(x, y, z, t)$  is an external force term. Note that the acoustic wave equation is commonly used in exploration seismology to approximate compressional (i.e., P-wave) propagation through complex heterogeneous media.

- **Acoustic wave equation (variable density):** This is like the example above, except now that the density,  $\rho$ , is assumed to spatially vary:

$$\frac{\partial^2 \phi}{\partial t^2} - K \nabla \cdot \left( \frac{1}{\rho} \nabla \phi \right) = F(x, y, z, t), \quad (3)$$

where  $K$  is the bulk compressibility. This equation is commonly used in computational ocean acoustics because it allows one to introduce variations in density caused by temperature and salinity concentration among others. This wave equation requires the same type of initial and boundary conditions as the homogenous equation above.

- **One-way wave equations:** We have actually looked at this hyperbolic equation in Module 7 of the course. One can recognize this to be the case if we **factorize** the 1D AWE given above in the following way

$$\left[ \frac{\partial^2}{\partial t^2} - v^2 \frac{\partial}{\partial x^2} \right] \phi = \left( \frac{\partial}{\partial t} - v \frac{\partial}{\partial x} \right) \left( \frac{\partial}{\partial t} + v \frac{\partial}{\partial x} \right) \phi = F(x, t). \quad (4)$$

Here we see that we have the produce of the left- and right-going advection equations. This was recognized in 1747 by French scientist [d'Alembert](#) ([https://en.wikipedia.org/wiki/D%27Alembert%27s\\_formula](https://en.wikipedia.org/wiki/D%27Alembert%27s_formula)). Note that one sometimes see the **d'Alembert operator**  $\square^2$  in various problems in mathematical physics, which is given by the following in Cartesian coordinates:

$$\square^2 \phi = \left[ \frac{\partial^2}{\partial t^2} - v^2 \frac{\partial^2}{\partial x^2} \right] \phi \quad (5)$$

### Initial and Boundary Conditions

Because this is a **second-order** temporal partial derivative, uniquely defining a wavefield solution requires implementing **two initial conditions** that define the wavefield at, say, time zero:

$$\phi(x, y, z, t = 0) = f(x, y, z) \quad (6)$$

as well as its derivative.

$$\frac{\partial \phi}{\partial t}(x, y, z, t = 0) = g(x, y, z). \quad (7)$$

One also has to prescribe the behavior of the wavefield solution on the domain boundary  $D$  (e.g.,  $\phi|_{\partial D} = B(x, y, z)$ ). In some circumstances, one may want to apply a Dirichelet BC

$$\phi(\mathbf{x}) = f(\mathbf{x}), \quad \forall \mathbf{x} \in \delta\Omega \quad (8)$$

where one sets  $f(x) = 0$  to obtain an inward moving reflected wave with the opposing polarity. This is commonly used for waves approaching the free surface of a fluid acoustic medium. However, in most other circumstances one wants to have perfectly absorbing boundary conditions such that no energy is reflected back into the finite-sized computational domain. These are the so-called (Sommerville) radiation boundary conditions which require the wave disturbance to vanish at infinity in an unbounded medium. This topic, though, remains an active area of research and goes well

## 1D Acoustic wave equation

Let's first look at the problem of an acoustic wave propagating on a 1D medium. Because of the easy connection with everyday experience, let's assume examine the propagation of a wave on a homogeneous taut string of length  $L$  where both of the string ends are fixed. In this case, we can assume that there is no external forcing function (i.e.,  $F(x, t) = 0$ ) and the string starts from some combination of an initial displacement disturbance  $\phi(x, t = 0) = f(x)$  and an initial velocity  $\frac{\partial \phi}{\partial t}(x, t = 0) = g(x)$ . Thus, the PDE we are solving is

$$\frac{\partial^2 \phi}{\partial t^2} = v^2 \frac{\partial^2 \phi}{\partial x^2}. \quad (9)$$

Let's now look at an **explicit** discretization method to solve equation 9 (as opposed to the **implicit** ones like Crank-Nicolson approach we studied in Module 9). Let's start by using a standard **centered in time centered in space** (CTCS)  $\mathcal{O}(\Delta x^2, \Delta t^2)$  discretization:

$$\frac{\phi_i^{n+1} - 2\phi_i^n + \phi_i^{n-1}}{\Delta t^2} = v^2 \left( \frac{\phi_{i+1}^n - 2\phi_i^n + \phi_{i-1}^n}{\Delta x^2} \right). \quad (10)$$

Let's now multiply through by  $\Delta t^2$  and use the square of the Courant number  $C^2 = \frac{v^2 \Delta t^2}{\Delta x^2}$  from the previous modules to write:

$$\phi_i^{n+1} - 2\phi_i^n + \phi_i^{n-1} = C^2 (\phi_{i+1}^n - 2\phi_i^n + \phi_{i-1}^n). \quad (11)$$

Rearranging terms to isolate the unknown quantity at time step  $n + 1$  on the left-hand side and keep known quantities on the right-hand side leads to:

$$\phi_i^{n+1} = 2\phi_i^n - \phi_i^{n-1} + C^2 (\phi_{i+1}^n - 2\phi_i^n + \phi_{i-1}^n). \quad (12)$$

You may note that one could combine the  $\phi_i^n$  terms into a more compact stencil. However, I am avoiding doing this because we will look at higher-order finite-difference expansions of the spatial second partial derivative in sections below

### Initial Conditions

Above we noted that two initial conditions are required for this PDE because of the second temporal derivative. However, based on the above discretization it may not be apparent how the derivative initial condition can be introduced. One thing to note is that the discretization now contains term  $\phi_i^{n-1}$ , which represents the wavefield at the previous time step. Thus, this can be used to set the second condition by:

$$\frac{\partial \phi}{\partial t} \approx \frac{\phi_i^n - \phi_i^{n-1}}{\Delta t} = g(x). \quad (13)$$

With this finite-difference approximation, we can specify wavefield at the previous time step as

$$\phi_i^{n-1} = \phi_i^n - \Delta t g(x). \quad (14)$$

Alternatively, it may also be possible that  $\phi_i^{n-1}$  is known analytically and can just be stated (e.g., it is a translation of the waveform at  $\phi_i^n$ ).

### Handling Boundaries with Ghost Points

We can also explore what happens at the left and right boundaries from equation 12. At the left hand boundary (i.e.,  $i = 0$ ) we can write

$$\phi_1^{n+1} = 2\phi_1^n - \phi_1^{n-1} + C^2 (\phi_2^n - 2\phi_1^n + \phi_0^n). \quad (14)$$

Note that there is a  $\phi_0^n$  term in this equation. Currently, we are able to handle this because we can use the Dirichelet boundary condition to set this point equal to zero. However, if we were to use higher-order finite-difference stencils for our second derivative, these would require using additional points such as  $\phi_{-1}^n$  which represent a point falling out of the grid! Let's assume that we can set  $\phi_{-1}^n = 0$  (as well those with an increasing negative index). This is termed using **ghost points** which are not formally defined, but are conjured up for numerical expediency. Similarly, on the right-hand side we will have to do the same at ghost point  $\phi_{N+1}^n = 0$  as well as those with greater indices.

## 1D AWE Solver

Given this machinery, we can now adapt the types of solvers we constructed in the previous modules to the current scenario. Let's first build a 1D AWE solver of  $\mathcal{O}(\Delta x^2, \Delta t^2)$  that is appropriate for homogeneous medium where  $v=\text{const}$ .

Let's now build up a scenario that we are trying to model. I will assume that the length of the region to be modeled is  $L = 1000$  mm, that is discretized into  $nx = 201$  (and thus  $\Delta x = 5$  mm segments), and the velocity is  $v = 1000$  mm/s.

Let's now define the time stepping conditions. Because we have defined our  $\Delta x$  and  $v$ , we know that we need to obey the following Courant condition for stability (usually called the CFL condition):

$$C \equiv \frac{v\Delta t}{\Delta x} \leq 1. \quad (15)$$

Thus, let's be a bit conservative and choose  $C = 0.5$ . Thus, we can set our  $\Delta t$  time step parameter to satisfy the CFL condition:

$$\Delta t = \frac{0.8\Delta x}{v} = \frac{0.8 \times 5\text{mm}}{1000\text{mm/s}} = 0.004\text{ms}. \quad (16)$$

We also want to define our two initial conditions, which we assume to have the shape of a stationary Ricker wavelet with no prior velocity:

$$\phi(x, t = 0) = f(x) = \frac{2}{\sqrt{3}\sqrt{\pi}} \left(1 - ((x - 3L/4)/\sigma)^2\right) e^{-\frac{(x-3L/4)^2}{2\sigma^2}} \quad (17)$$

and

$$\left. \frac{\partial \phi}{\partial t} \right|_{t=0} = g(x) = 0 \rightarrow \phi_i^{n-1} = 0 \quad (18)$$

where the right arrow shows the implication that the  $n - 1$  time step is zero. For this simulation we will use  $\sigma = 40$ , though this will be varied in the section below. Let's now compute the solution

**Out[4]:**

0:00 / 0:19



**Figure 1. Illustration of wave equation solution when one only defines the wavefield  $\phi(x, t = 0)$  and  $\phi(x, t = -\Delta t) = 0$ . Note that there is a two-way solution because there we haven't specified a constraint forcing the solution to propagate one way or the other.**

Let's now look at what happens when we specify the previous wavefield state at  $\phi(x, t = -\Delta t)$ , which requires establishing what the wave looks like at the previous time step. Given the initial Ricker wavelet above, it is reasonable to expect that the wavefield solution at  $\phi(x, t = -\Delta t)$  will be given by some wavefield shift  $t_0$ :

$$\phi(x, t = -\Delta t) = \frac{2}{\sqrt{3}\sqrt{\pi}} \left(1 - ((x - 3L/4 - v\Delta t)/\sigma)^2\right) e^{-\frac{(x-3L/4-v\Delta t)^2}{2\sigma^2}}, \quad (19)$$

where  $t_0 = v\Delta t$  has been introduced to account for a spatial shift. Let's rerun the solution from above, but now where we set the value of  $\phi_i^{n-1}$  as non-zero:

**Out[5]:**

0:00 / 0:19



**Figure 2. Illustration of wave-equation solution when defining the wavefield at  $t = 0$  and  $t = -\Delta t$ . Note that the wave initially travels to the left instead of both directions like in Figure 10.1 above.**

Parameter  $\sigma$  is of course related to the **frequency** content of the Ricker wavelet. Let's now investigate what happens when we set  $\sigma = 15$  instead of using  $\sigma = 40$  as we did in the simulations above:

Out[6]:

0:00 / 0:12

**Figure 3. Illustration of how numerical dispersion can degrade the solution over time. The initial wave packet starts to develop a long "tail" that represents non-physical behaviour.**

## Higher-order derivatives

It's clear that the numerical solution behavior noted in Figure 3 is less than ideal. Thus, we need to look at different methods for trying to eliminate these artifacts. One strategy would be to use a smaller  $\Delta x$ , which is easy to do in a 1D solution because the computational cost scales according to  $\mathcal{O}(N)$  where  $N$  is the number of points. However, in 3D this approach would scale according to  $\mathcal{O}(N^3)$  since we would have to apply it in each dimension! This would be a significant impact on our computational efficiency.

Another strategy would be to use higher-order spatial derivatives, which we know lead to a more accurate representation of numerical derivatives. To do this, let's rewrite our 1D solver for homogeneous media from above to incorporate an  $\mathcal{O}(\Delta x^8)$  approximation of the spatial second partial derivative. This requires using a nine-point stencil that incorporates four points to the left and to the right of the stencil center:

$$\phi_i^{n+1} = 2\phi_i^n - \phi_i^{n-1} + C^2 \left( -\frac{1}{560}\phi_{i+4}^n + \frac{8}{315}\phi_{i+3}^n - \frac{1}{5}\phi_{i+2}^n + \frac{8}{5}\phi_{i+1}^n - \frac{205}{72}\phi_i^n + \frac{8}{5}\phi_{i-1}^n - \frac{1}{5}\phi_{i-2}^n + \frac{8}{315}\phi_{i-3}^n - \frac{1}{560}\phi_{i-4}^n \right), \quad (20)$$

where these [finite-difference coefficients](https://en.wikipedia.org/wiki/Finite_difference_coefficient#Central finite difference) ([https://en.wikipedia.org/wiki/Finite\\_difference\\_coefficient#Central finite difference](https://en.wikipedia.org/wiki/Finite_difference_coefficient#Central finite difference)) are for a **centered second derivative of eighth order**.

Note that we will again have issues at the boundaries where we will need **ghost points**. However, because we have longer stencil, we will now need to have four ghost points instead of just one. Let's see how this works in our solver:

Let's now rerun our numerically dispersive example where we set  $\sigma = 15$  from above but using our  $\mathcal{O}(\Delta x^8)$  solver for homogeneous media.

Out[8]:

0:00 / 0:12

**Figure 4.** As in Figure 3, but using the 8th-order solver for homogeneous media. Note that the previously noted "tail" has largely - but not completely - disappeared. This indicates that the higher-order finite-difference approximation has helped to reduce numerical dispersion, but is not a panacea for this issue.

It is also illustrative to plot the two previous solutions over top of each other. Note that because of the different number of ghost points used, the location of the boundary reflection will be at slightly different locations; thus, I'll only plot it up to the point where the waves reach the boundary.

Out[9]:

0:00 / 0:12

**Figure 5.** Comparison of the solutions generated by the eighth- and second-order solvers. The higher-order method clearly produces a more accurate result - though it is itself not perfect.

## 2D Acoustic wave equation

Let's now look at the problem of an acoustic wave propagating through a homogeneous 2D domain. In this case, we can assume that there is an external forcing function  $F(x, y, t) \neq 0$  and the wave disturbance is initially zero in a **quiescent** state. Thus, the 2D PDE we are solving is

$$\frac{\partial^2 \phi}{\partial t^2} = v^2 \left( \frac{\partial^2 \phi}{\partial x^2} + \frac{\partial^2 \phi}{\partial y^2} \right) + F(x, y, t). \quad (21)$$

One common thing that people use is to model the acoustic wavefield emanating from a spatially localized point source. In these scenarios, one can write the following for the force source

$$F(x, y, t) = W(t) \delta(x - s_x) \delta(y - s_y), \quad (22)$$

where point  $s = [s_x, s_y]$  represents the coordinates of point source and  $W(t)$  represents the wavelet to be **injected** into the domain.

### Discretizing the 2D AWE

Let's now look at an **explicit** discretization method. Let's use a  $\mathcal{O}(\Delta x^8, \Delta t^2)$  discretization:

$$\frac{\phi_{i,j}^{n+1} - 2\phi_{i,j}^n + \phi_{i,j}^{n-1}}{\Delta t^2} = \frac{v^2}{\Delta x^2} \sum_{k=-4}^4 c_k \phi_{i+k,j}^n + \frac{v^2}{\Delta y^2} \sum_{k=-4}^4 c_k \phi_{i,j+k}^n + F_{i,j}, \quad (23)$$

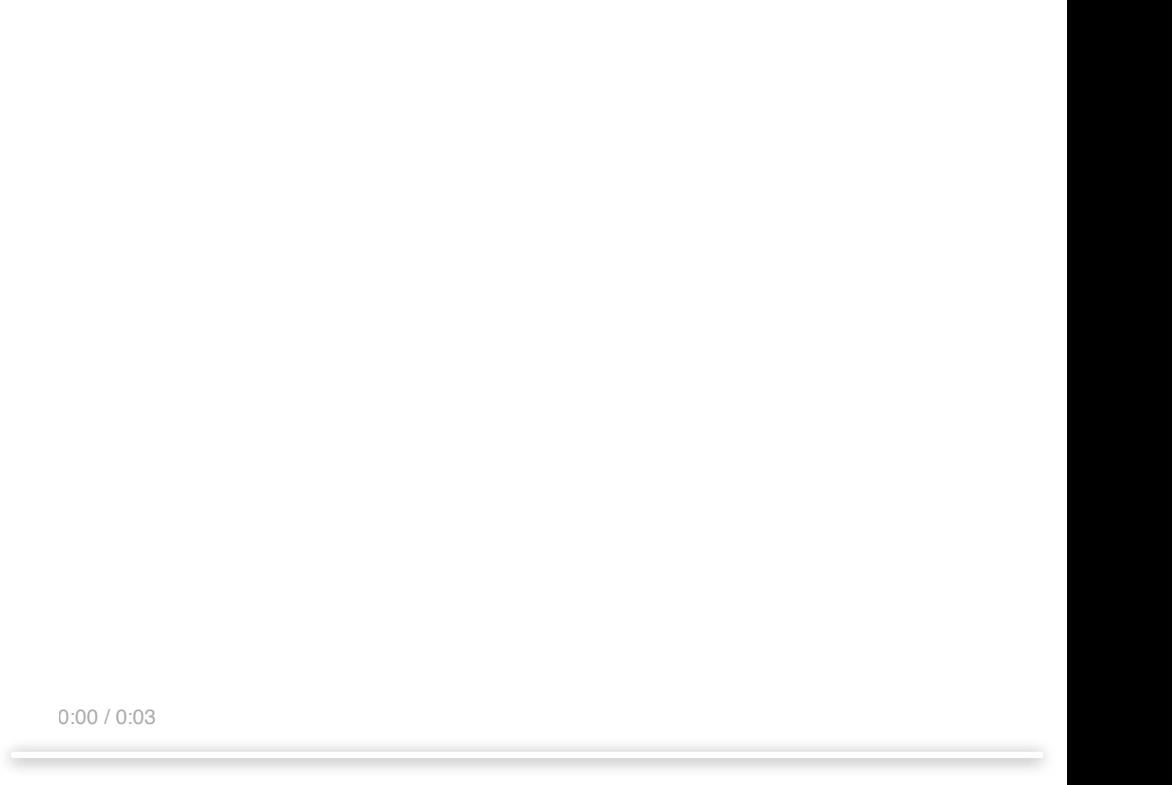
where  $c_k$  are the finite-difference coefficients discussed above. Let's now multiply through by  $\Delta t^2$  and use the square of the Courant number  $C_x^2 = \frac{v^2 \Delta t^2}{\Delta x^2}$  and  $C_y^2 = \frac{v^2 \Delta t^2}{\Delta y^2}$  from the previous modules to write

$$\phi_{i,j}^{n+1} = 2\phi_{i,j}^n - \phi_{i,j}^{n-1} + C_x^2 \sum_{k=-4}^4 c_k \phi_{i+k,j}^n + C_y^2 \sum_{k=-4}^4 c_k \phi_{i,j+k}^n + \Delta t^2 F_{i,j}. \quad (24)$$

This is pretty much the same as in 1D, so let's just move right to the 2D solver!

You'll note that I've changed things a bit here since I am now **injecting the wavelet** into the 2D domain at the current time step  $it$ . Here, I've calculated the location of the incidiencies  $[isx, isy]$  at which to inject the wavelet. I could have also created a force source term  $F_{i,j}^n$  as a 3D array; however, this would be wasteful of memory since I'm only injecting at one location.

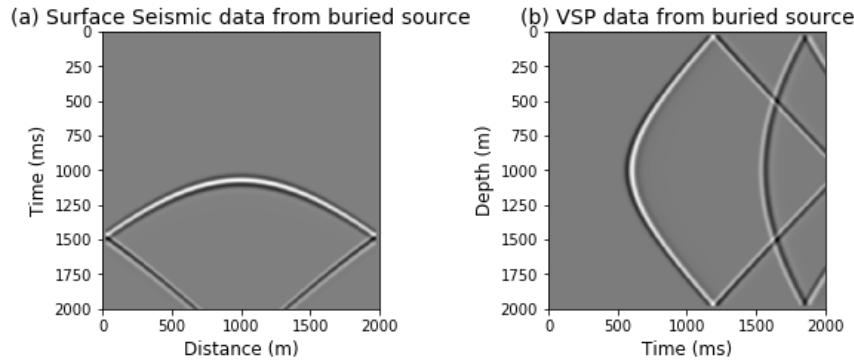
Out[12]:



**Figure 6.** Propagating 2D acoustic wavefield generated by the 8th-order finite-difference AWE solver. Note that the reflections from the boundary are with the opposite sign.

## Wavefields, surface seismic and VSP data

Since we have the full 3D volume of  $[nx, ny, nt]$  it is possible to slice it in different ways. Figure 7 shows the 2D slice at a constant depth (a) representing a surface seismic data set as well as a 2D slice taken a constant x-location (b) which would represent a vertical seismic profile (VSP) data set.



**Figure 8.** (a) Constant depth slice extracted 5m down from the first ghost point representing surface seismic data. (b) Constant X-location slice extracted at 500m distance representing a vertical seismic profile (VSP).

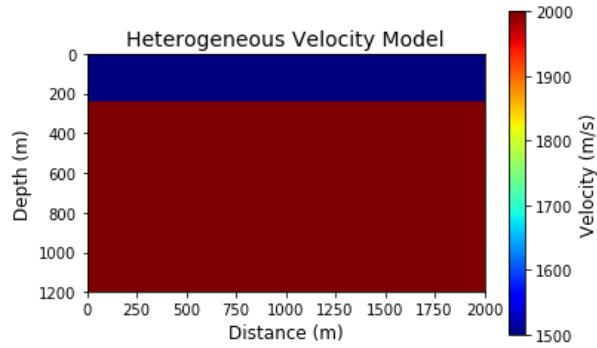
## 2D Acoustic Wave Equation for Heterogeneous Media

While solutions of the 2D acoustic wave equation for homogeneous media are important for evaluating our finite-difference scheme, they have quite limited use in practice. This is because velocity models usually are heterogeneous:  $v = v(x, y)$ . Thus, we need to account for this spatial heterogeneity in our numerical solvers. To do so, let's adapt our eight-order solver above to include heterogeneity. The most common practice in this situation would be to introduce a spatially varying velocity function into equation 24 above:

$$\phi_{i,j}^{n+1} = 2\phi_{i,j}^n - \phi_{i,j}^{n-1} + \left( \frac{v_{i,j}\Delta t}{\Delta x} \right)^2 \sum_{k=-4}^4 c_k \phi_{i+k,j}^n + \left( \frac{v_{i,j}\Delta t}{\Delta y} \right)^2 \sum_{k=-4}^4 c_k \phi_{i,j+k}^n + \Delta t^2 F_{i,j}, \quad (25)$$

which leads to only a modest change in the `awe_2d_explicit_solver_heterogeneous_8th_order` solver below:

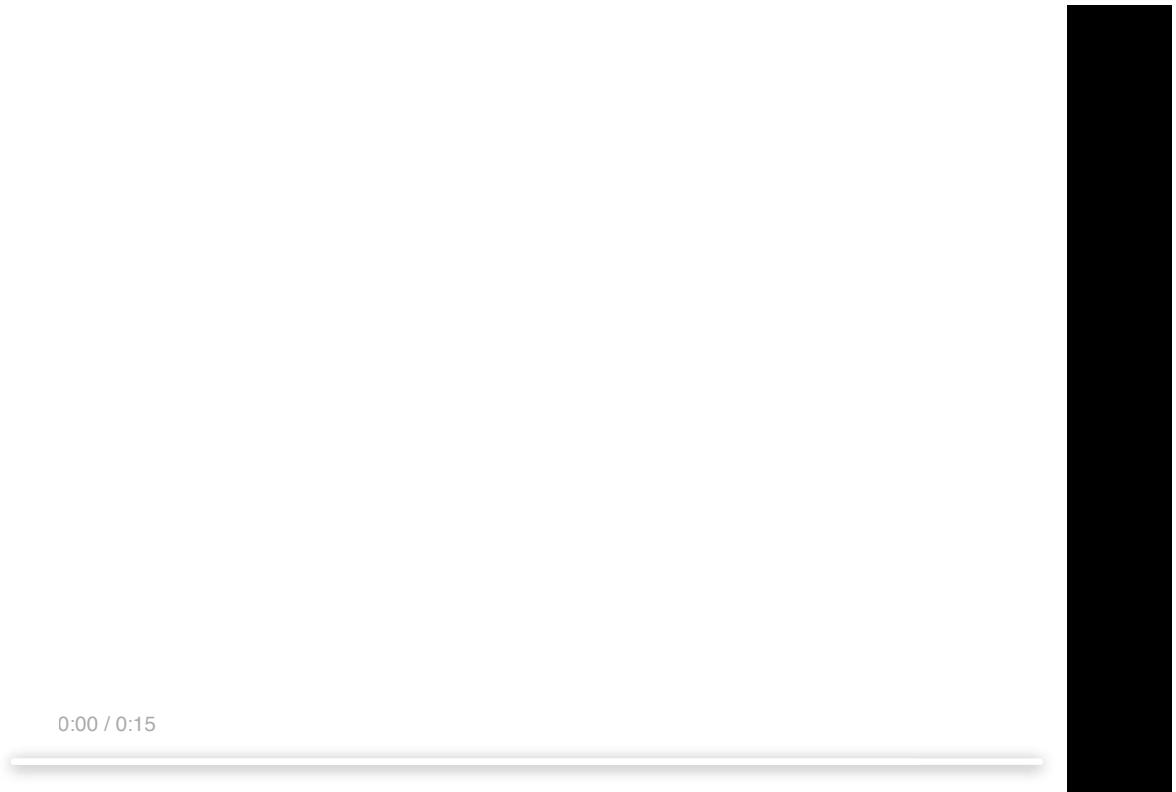
Let's now generate a simple 1D velocity model that has a single reflector located at 250m depth.



**Figure 9.** Heterogeneous velocity model with a single discontinuity located at 250 m depth.

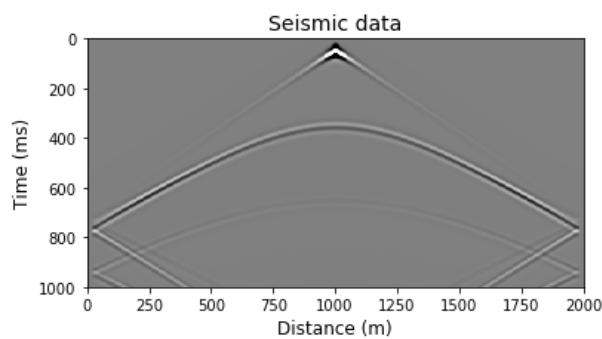
Let's now call our solver for heterogeneous media:

Out[17]:



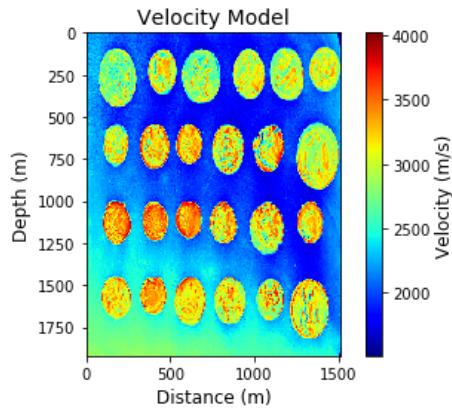
**Figure 10.** Animation of wavefield propagation in layered half space. For illustration purposes, I have added in a scaled version of the velocity model to the (scaled) wavefield solution so that it is easy to see where all of the wavefield scattering occurs.

We can also look at what the surface seismic data look like by taking a slice through the second coordinate of the 3D wavefield solution volume.



**Figure 11. Surface seismic data from the wavefield animation shown in Figure 10. We observe the direct and reflected arrivals.**

Let's now create a velocity model with a lot more spatial variations. For this example, I'm going to import an image from the `skimage` package, convert it to float, scale it by 1000x, and add it to a background velocity of 1500 m/s. (**Note: you may need to install the scikit-image package to run this example, the command for which is given below.**) The result is shown in Figure 12 below:



**Figure 12. A whimsical velocity model made from the `coins` image in the `skimage` tool kit.**

Let's now rerun the wave propagation simulation for heterogeneous media from above, but using this new velocity model!

Out[ ]:

