Improved shortest-path ray tracing with locally linear velocity variations

Hui Liu, Hua-wei Zhou, Zhihui Zou, and Fan Jiang, Texas Tech University*

Summary

Shortest-path algorithm has been used widely to calculate first arrival traveltimes and ray paths. It assumes that the velocity in each model cell is constant. When the velocity field varies continuously, models using too few constant-velocity cells will result in too much error, and models using too many constant-velocity cells will be inefficient. We made an improvement to this problem by allowing velocity increases linearly with depth at each model location. This improvement leads to a more accurate and faster ray tracing, and incorporates fewer model parameters for tomographic inversion at places of continuous velocity variation. The improved method is shown using 2D and 3D ray tracing examples.

Introduction

Calculations of traveltimes and ray paths are important in migration, tomography, and modeling. One of the widely used ray tracing methods is the shortest-path calculation introduced by Moser (1991). This algorithm assumes that velocity in each cell is constant. However, in the subsurface of real earth, velocity usually changes in the vertical direction with consequent lithological changes and increasing pressure with increasing depth. Among different relationship between velocity and depth, velocity increasing linearly with depth is of considerable practical importance, and the value of the velocity gradient is generally between 0.3/s and 1.3/s (Sheriff and Geldart, 1995). The typical velocity gradient found in the Gulf of Mexico is 0.5/s (Biondi, 2006). Large velocity gradient is likely present within near-surface materials and near the water table in unconsolidated material (Birkelo *et al*., 1987; Miller and Xia, 1997, 1998). Ettrich (2002) shows that using a constant velocity gradient may approximate the real velocity field much better than using constant velocities.

If we use shortest-path algorithm to model the subsurface where velocity varies continuously, the assumption of constant velocity in each cell cannot be effective. Using few cells, the error will be large. For increasing the accuracy, many constant-velocity cells are needed to approximate, which means a huge increase of computation expense, and the efficiency is always an evergreen issue. During the last several years, some modifications and improvements to shortest path algorithm for efficiency have been done by some researchers (Fischer and Lees, 1993; Urdanera and Biondi, 2001; Van Avendonk *et al*, 2001; Zhang and Li, 2005; Bai *et al*, 2007).

We developed a technique that incorporates velocity gradient into each cell, which means we don't need so many cells to model the subsurface with velocity gradient.

Method

It is well known that the ray path will be a part of a circle if the velocity gradient is constant, in other words, the ray will be circular if velocities increase linearly with depth (Slotnick, 1959). Sheriff and Geldart (1995) also give a very clear description about this. Figure 1 shows this idea in 2D case. Green area illustrates geological subsurface. The red curve is a ray leaving a source (red solid dot) at the angle i_0 to a receiver (blue triangle) and the center of this circular ray is **O**. The height of this center above the geologic surface is **v/a**, where **v** is the velocity at the surface and **a** is the velocity gradient. Because this height is independent of i_0, the centers of all circular rays lie on the same horizontal line -- the top horizontal red dashed line in Figure 1. This line is located where the velocity would be zero if the velocity function were extrapolated up into the air (Sheriff and Geldart, 1995). We call this line 'centers line' which will be used by following figures (Figure 2, 3, and 4). This important result makes it easier for us to take the velocity gradient into real applications.

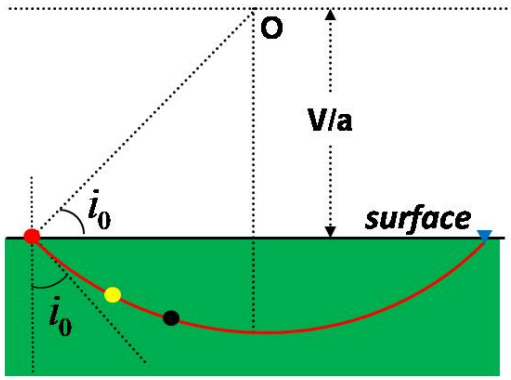

Figure 1: Green area illustrates geological subsurface. The red curve is a ray path from a source (red solid dot) to a receiver (blue triangle). If velocity gradient is constant, the ray path is a part of a circle centered at **O**. The height of this center above the surface is **v/a**, where **v** is the velocity at the surface and **a** is the velocity gradient. i_0 is the incidence angle of the ray and is used to calculate the ray parameter.

Given the velocity at surface and velocity gradient, the traveltime from one reference point (such as the red solid dot in Figure 1) at surface to another point at depth (such as the yellow solid dot in Figure 1) can be calculated directly by Equation 1 introduced by Slotnick (1959).

$$t(v,a)=\frac{1}{a}\ln\left(\frac{v+az}{v}\frac{1+\sqrt{1-p^2v^2}}{1+\sqrt{1-p^2(v+az)^2}}\right) \qquad (1)$$

where, v is the velocity at surface; a is the velocity gradient; z is the depth from surface; p is the ray parameter of the ray.

If the traveltime between any two points (such as the yellow dot and the black dot in Figure 1) in subsurface is to be calculated, then Equation 1 can be used to calculate the traveltime T_1 between the red point (reference point) to the yellow point, and traveltime T_2 between the red point to the black point respectively, and then the difference (T_2-T_1) is the traveltime from the yellow dot to the black dot. Based on this, we can calculate the traveltime between any two arbitrary points having the same surface velocity and subsurface gradient.

In the traditional model parameterization, each cell has a constant velocity value. The traveltime between two arbitrary nodes is calculated by multiplying the slowness value (inverse of velocity) with the length of the straight ray between the two nodes.

In our method, the ray path is not a straight line anymore in each cell. Figure 2 is an illustration of how we calculate the travetime and ray path. The black rectangle is the model cell used in shortest path calculation. The velocities are recorded at the four corners of the cells. The top horizontal red dashed line is what we called 'centers line' in Figure 1. The second red dashed line is considered as the geological surface with surface velocity $\mathbf{V}_T$ and the bottom red dashed line is the bottom of the subsurface with velocity $\mathbf{V}_B$. $\mathbf{V}_T$ is defined as the average of the velocities of the top two corners. $\mathbf{V}_B$ is defined as the average of the velocities of the bottom two corners. Given any two arbitrary nodes (Node 1 and Node 2) on the boundary of the cell, the midpoint **M** is got. Then draw a perpendicular line passing **M**, and this line will cross the top red dashed line (so-called 'centers line'). The crossing point **O** is got, which is the center of the circular ray between Node 1 and Node 2. Then an arc from node 1 to node 2 can be drawn (red curve) and this arc is the ray path. i_0 and surface velocity $\mathbf{V}_T$ will be used for calculating ray parameter, which is $p= \sin(i_0) / \mathbf{V}_T$. Velocity gradient of this cell is defined by $(\mathbf{V}_B-\mathbf{V}_T)/\mathbf{h}$. Traveltime is calculated using Equation 1.

In real earth, layer interfaces often have lateral variations. For modeling such kind of interfaces, we may let the top and bottom boundaries be dipping, shown in Figure 3. This time the cell is a trapezoidal. This figure is similar to Figure 2, and the only difference is that we define the average depth of top two corners as geological surface (second horizontal red dashed line) and the average depth of bottom two corners as bottom of subsurface (bottom red dashed line). Velocity gradient of this cell is defined by $(\mathbf{V}_B-\mathbf{V}_T)/\mathbf{h}$. Then the traveltime and ray path can be calculated with the same method shown in Figure 2.

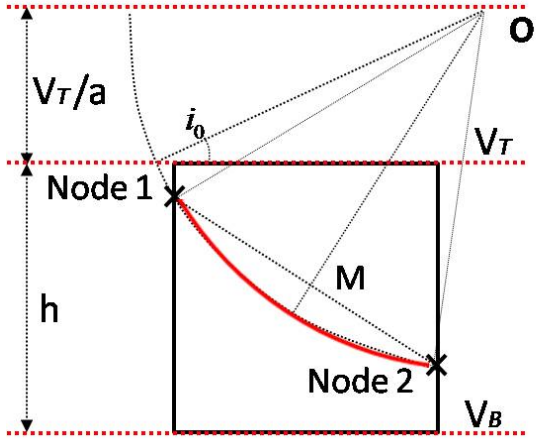

Figure 2: 2D illustration of calculating traveltimes and ray paths in each model cell with flat top and bottom boundaries. Red curve is ray path. Traveltime between Node 1 and Node 2 is calculated using Equation (1).

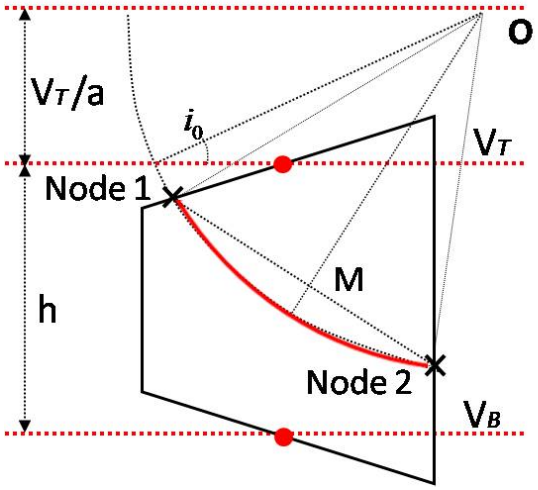

Figure 3: 2D illustration of calculating traveltimes and ray paths in each model cell with dipping top and bottom boundaries. Red curve is ray path. Traveltime between Node 1 and Node 2 is calculated using Equation (1).

For 3D cases (Figure 4), the illustration of how a velocity gradient is defined in a triangular prism is similar to 2D cases. Nodes for shortest-path algorithm are defined on the five surfaces of the triangular prism. All the ray paths between two arbitrary nodes will be circular and all the centers of these circular rays are in the same surface -- the top red

surface. Also similar to the 2D cases shown in Figure 3, because the layer interfaces have undulations, the top and bottom surfaces of the prism are often defined as not parallel with each other. Like the two red lines where the $\mathbf{V}_T$ and $\mathbf{V}_B$ are defined in Figure 3, there are two surfaces with velocities of $\mathbf{V}_T$ and $\mathbf{V}_B$ are shown in Figure 4. Velocity gradient for the whole prism will be $(\mathbf{V}_B-\mathbf{V}_T)/\mathbf{h}$. Then the traveltimes and ray paths can be calculated by the same method of 2D cases.

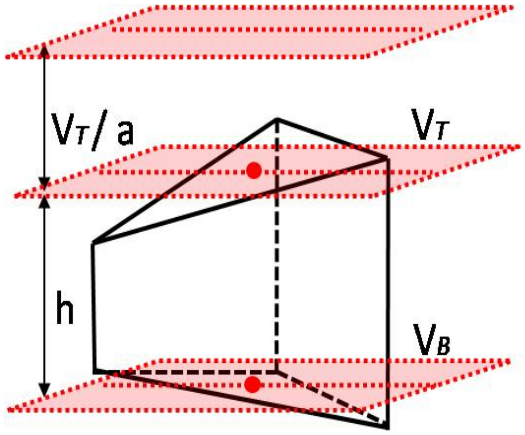

Figure 4: 3D illustration of triangular prisms with dipping top and bottom surfaces used in the 3D ray tracing algorithm.

An example comparing computation time and accuracy

This example made a test of ray tracing using our method and traditional shortest-path algorithm to compare computation time and accuracy. For comparing easily, a simple model (Figure 5a) is used here. It is 1000 m long and 500 m deep. The velocity is 500m/s at the surface and 1500m/s at the bottom. This means there is a velocity gradient. Different color shows different velocities (see color bar at bottom of Figure 5). The source (pink circle) is at (0.0, 0.0) and the receiver (blue triangle) is located at (1000.0, 0.0). Red curve is the first arrival ray path.

Figure 5b is a five-layer model with constant layer velocities (600, 800, 1000, 1200, 1400 m/s) chosen to best fit the velocity gradient in Figure 5a. Figure 5c is a ten-layer model with constant layer velocities (550, 650, 750, 850, 950, 1050, 1150, 1250, 1350, 1450 m/s) chosen to best fit the velocity gradient in Figure 5a. The 1D velocity profile is shown on the right side of each model.

Table 1 compiled the computation times and errors calculated by ray tracing in the three models shown in Figure 5. From this table, we can see that our method calculates the traveltime analytically without any approximation in this case. The computation time is less than 0.1 s because, in this case, the model using our method only needs 1 cell. This table also tells us that, to make the error smaller than 10ms which is usually the noise level for offset of 1000m, many layers are needed. However, the computation expense will increase.

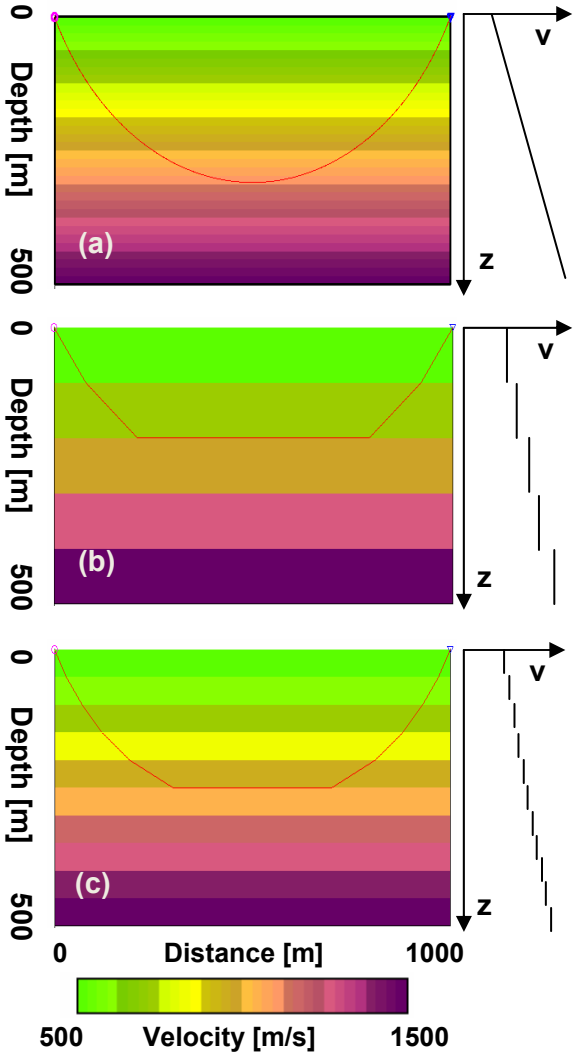

Figure 5: (a) A single-layer model with a constant velocity gradient. (b) A five-layer model with constant layer velocities (600, 800, 1000, 1200, 1400 m/s) chosen to best fit the velocity gradient in (a). (c) A ten-layer model with constant layer velocities (550, 650, 750, 850, 950, 1050, 1150, 1250, 1350, 1450 m/s) chosen to best fit the velocity gradient in (a). The red curve in each panel denotes the ray path.

Model	Time	Error
Constant velocity gradient	< 0.1 s	0.00 ms
5-layer constant-velocity	3.9 s	26.93 ms
10-layer constant-velocity	7.1 s	8.96 ms

Table 1: Comparison of computation time and accuracy of the models shown in Figure 4.

A 2D ray tracing example

A 2D ray tracing example using our method is shown here. The 1D velocity-depth profile in Figure 6a is at the location denoted by the red arrow in Figure 6b. There are two sources (pink circles) and 26 receivers (blue triangles) at or near surface. Red curve is the ray path from the first source to all receivers. Green curve is the ray path from the second source to all receivers.

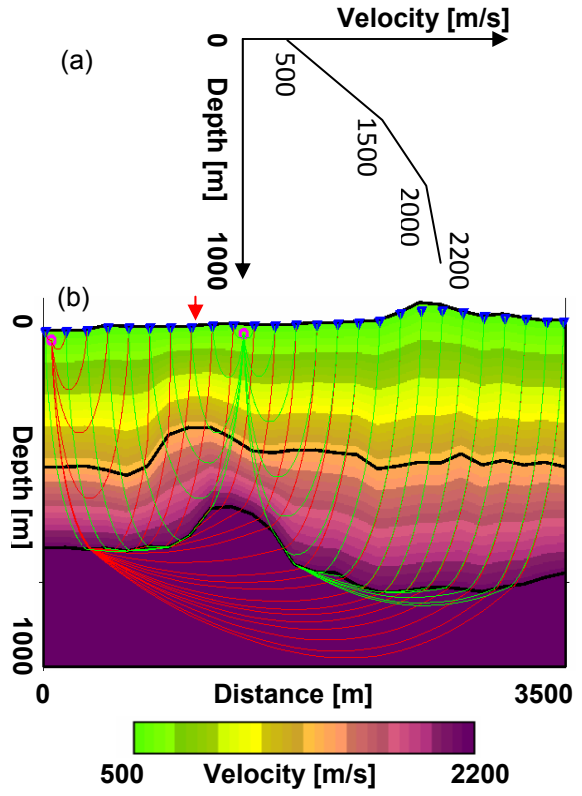

Figure 6: A test of our method for tracing rays in a velocity model with gradient. (a) 1D velocity profile at the location denoted by the red arrow in (b). (b) Ray tracing in the velocity model. There are two sources (pink circles) and 26 receivers (blue triangles). Red curve is the ray path from the first source to all receivers. Green curve is the ray path from the second source to all receivers.

A 3D VSP ray tracing example

A 3D ray tracing example is shown in Figure 7, which is an application of the method shown in Figure 4. The survey setup is the same as the real one already done at Vinton Salt Dome, located in the southwestern corner of Louisiana, 200 km east of Houston, and has been discovered more than 100 years. Some primary reservoirs are found near the crest of the dome which is also the location of the drilled well for this VSP survey. The red stars are the sources, and the 11 blue triangles are one subset of all the receivers in the well (The total number of receivers is about 60). Zhou (2006) gives much more details about this project and his tomographic inversion results. There are also some well log data at that area, and those data shows that there are two layers in large scale and, within each, there are prominent velocity gradients. So, in this ray tracing example, two layers with different gradients are shown. It will be interesting to use our new method to apply to the tomographic inversion and compare the result with the constant-velocity method in previous work.

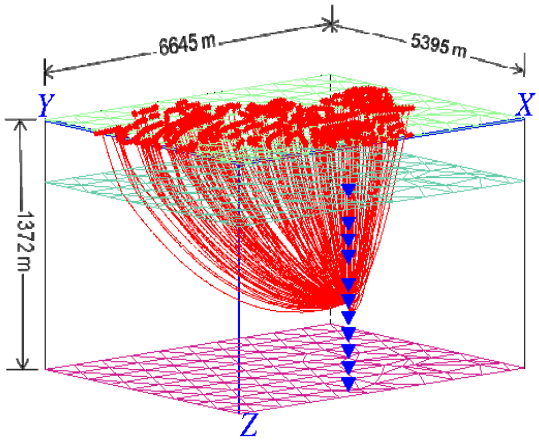

Figure 7: 3D VSP Ray tracing example. Red curves are ray paths. The red stars are the sources, and the blue triangles are the receivers. Rays drawn are from all sources to one of the receivers.

Discussion and conclusions

Traditional shortest-path algorithms for ray tracing assume constant velocity in each model cell. When the velocity field varies continuously, it will be erroneous to approximate it using few constant-velocity cells. On the other hand, using many constant-velocity cells will lead to more model parameters that will be harmful to ray tracing computation and inversion. The truth is that velocity gradient exists widely in the real world. Our improved method introduces a linear velocity increase with depth at each model location. Tests of the new method indicated improvement in both solution accuracy and computation efficiency. Our method uses fewer model parameters, hence it is more suitable for tomographic inversion because the solutions will be more robust for fewer unknowns.

There are still limitations for our method. In the presence of large undulation of the layer interfaces, the approximation shown in Figures 3 and 4 will require to use more model columns in order to constrain the numerical error. However, this error will be much smaller than that of using constant-velocity cells.

EDITED REFERENCES

Note: This reference list is a copy-edited version of the reference list submitted by the author. Reference lists for the 2009 SEG Technical Program Expanded Abstracts have been copy edited so that references provided with the online metadata for each paper will achieve a high degree of linking to cited sources that appear on the Web.

REFERENCES

Bai, C., S. Greenhalgh, and B. Zhou, 2007, 3D ray tracing using a modified shortest-path method: Geophysics, **72**, no. 4, T27–T36.

Bickel, S. H., 1990, Velocity-depth ambiguity of reflection traveltimes: Geophysics, **55**, 266–276.

Birkelo, B. A., D. W. Steeples, R. D. Miller, and M. A. Sophocleous, 1987, Seismic reflection study of a shallow aquifer during a pumping test: Ground Water, **25**, 703–709.

Biondi, B. L., 2006, 3D Seismic Imaging: SEG.

Clayton, R. W., and G. A. McMechan, 1981, Inversion of refraction data by wave field continuation: Geophysics, **46**, 860–868.

Ehinger, A., F. Jurado, and C. Monesti, 1998, Calculation of multi-valued traveltimes with a ray bending method: 68th Annual International Meeting, SEG, Expanded Abstracts, 1893–1896.

Ettrich, N., 2002, Offset-dependent geometrical spreading in isotropic laterally homogeneous media using constant velocity gradient models: Geophysics, **67**, 1612–1615.

Fischer, R., and J. M. Lees, 1993, Shortest path ray tracing with sparse graphs: Geophysics, **58**, 987–996.

Lines, L., 1993, Ambiguity in analysis of velocity and depth?: Geophysics, **58**, 596–597.

Miller, R., and J. Xia, 1997, High resolution seismic reflection survey to map bedrock and glacial/fluvial layers in Fridley, Minnesota: Symposium on Application of Geophysics To Engineering And Environmental Problems, 1, 281–290.

Miller, R., and J. Xia, 1998, Large near-surface velocity gradients on shallow seismic reflection data: Geophysics, **63**, 1348–1356.

Moser, T., 1991, Shortest path calculation of seismic rays: Geophysics, **56**, 59–67.

Paige, C. C., and M. A. Saunders, 1982, LSQR: An Algorithm for Sparse Linear Equations And Sparse Least Squares: ACM Transactions on Mathematical Software, **8**, 43–71.

Sheriff, R. E. and L. P. Geldart, 1995, Exploration Seismology. 2nd ed.: Cambridge University Press.

Slotnick, M. M., 1959, Lessons in seismic computing: SEG.

Thouvenot, F., 1986, Drawing seismic rays in layers with velocity gradients using only ruler and compass: The Leading Edge, **5**, 28.

Van Avendonk, H. J., A. J. Harding, J. A. Orcutt, and W. S. Holbrook, 2001, Hybrid shortest path and ray bending method for traveltime and raypath calculations: Geophysics, **66**, 648–653.

White, R. S., D. McKenzie, and R. K. O'Nions, 1992, Oceanic crustal thickness from seismic measurements and rare Earth element inversions: Journal of Geophysical Research, **97**, 19,683–19,715.

Zhang, M., and X. Li, 2005, Ray tracing with the improved shortest path method: 75th Annual International Meeting, SEG, Expanded Abstracts, 1795–1799.

Zhou, H., 2003, Multiscale traveltime tomography: Geophysics, **68**, 1639–1649.

Zhou, H., 2006, Multiscale deformable-layer tomography: Geophysics, **71**, R11–R19.

Zhou, H., 2006, First-break vertical seismic profiling tomography for Vinton Salt Dome: Geophysics, **71**, U29–U36.