Finite-difference elastic modelling in Matlab, workbench version

Peter M. Manning

ABSTRACT

A CREWES 2D elastic finite-difference Matlab modelling program is offered which is particularly useful where near surface effects are important, because Rayleigh waves can be accurately represented. The program allows users to experiment freely with modelling parameters or geological models, and then continue with only those which are most likely to be valuable. Control choices are designed to be made through menu selection, but parameter choices are made by modification of the Matlab code. The present version of the program allows only energy initiation from a buried source. Geological models are defined in separate programs, as a series of interfaces where velocities and densities apply either above or below each interface. Because of its many display options and features, the program is especially useful in the analysis of various aspects of elastic wave propagation that are difficult to visualize.

INTRODUCTION

Because of the difficulty of analyzing elastic wave motion on field seismic records, it is very useful to have modeling software which can accurately simulate seismic wave propagation phenomena on known geological models. By studying the results of various modeling projects, geophysicists gain insight into interpretation of real seismic data. The software described in this report fills an important niche, particularly since it accurately models surface wave propagation, which few other programs are able to do.

This software was coded by the author, most of it during the process of obtaining his PhD. It was designed to test the concepts and display the results presented in the thesis. The original software has been rearranged to clarify its intent, and to make it more useful to others with an interest in experiments using finite-difference modelling. The software has been supplemented by three developments that replaced rudimentary parts of the original code: a formal way of building and storing geological models, a simple way of generating SEGY output, and a systematic means of storing computed model states. The model states are useful for storing intermediate results, and for storing final results ready for plotting in various ways.

The software package is called a workbench version because, while some tools are provided, some knowledge and skills are required for their use. Someone with a bit of experience with Matlab coding, and an understanding of boundaries and stability problems, should be able to use it. These users will be able to make most of their changes in the mainline program. More sophisticated users might have to change some subroutines, or even write their own.

FINITE-DIFFERENCE CONCEPTS

A finite-difference model consists of a spatial grid where at each nodes is specified various propagation parameters for the earth material represented. By imposing a properly constructed finite-difference wave equation (derived from a more exact differential equation governing wave propagation in a continuous medium), an elastic excitation, or source impulse, is propagated through the grid over small time increments, creating a series of “snapshots” of the energy propagation. It is the discrete nature of the finite-difference model, both in space and time, that makes it both useful and difficult to control.

The most important chosen parameters for a finite-difference model are the sample rates. On one hand, the highest quality results are obtained by using the finest sample rates. On the other hand, fine sample rates use much more computer resources. For the same size model, cutting a sample rate in half, for example, requires four times as much memory and eight times as much execution time. The relationship for stability between the spatial and time sample rates is given by

$$\Delta t = \Delta x / \left(v\sqrt{2} \right), \qquad 1$$

where Δt and Δx are the time and space sample rates, and v is the lowest velocity in the model. Other formulae that may be useful are found in the thesis Manning (2008), Appendix E.

Forward finite-difference modelling proceeds through a series of time steps, where the state throughout space at a given time is determined from the states of the two immediately preceding times. The two states are required to determine the momentum throughout space, which together with the distortions determine the propagation of waves. This style of computation is inherently inefficient, since operations are carried out throughout the model, whether energy is present there or not, but it provides natural points at which computations may be suspended and restarted. At these points Matlab options may be used to store or recover from disc, the contents of all the critical variables within a routine. Calculations may then proceed as they would without the interruption.

Computed model states are not saved automatically so that the saved disc files can be given descriptive names, and to ensure that useful older files are not overwritten.

Elastic modelling allows the simulation of Rayleigh waves, otherwise known as surface waves or ground roll. Rayleigh waves involve a continual exchange of shear and compression energy, and can only exist near a free surface. Amplitudes are highest right at the surface, and tend to decay with depth in an exponential fashion. The present program is focused on these waves, and on the detailed near surface layering, because layering has such a strong effect on them.

The effect of geological layers on Rayleigh wave propagation may be simply related to the frequency content of these waves. A high frequency component of a Rayleigh wave is always associated with a high wavenumber of the wave, which translates to a short wavelength. This short wavelength applies in the propagation direction (horizontal), but also in depth. High frequency components, then, tend to have short wavelengths, with little penetration into the earth, and are therefore affected mostly by the shallow layers. In contrast, low frequency waves are more strongly influenced by deeper layers. Since deeper layers tend to have higher pressure and shear velocities, it follows that lower frequency Rayleigh wave components usually travel at higher velocities.

DISC FILES

This finite-difference modelling system uses two types of disc files that are designed to save intermediate steps in the process of creating synthetic seismic data. The files have extensions of .gfd or .fdc., where .gfd indicates a geological model for finite-difference input, and .fdc indicates a model state after some number of finite-difference steps.

Disc files can only be created interactively, by typing in a base name after a prompt from the program. The extension is then added automatically. Before the user is prompted for a name, a list is given showing all files of that type in the disc folder. The reason for making this step interactive is (first) to ensure that a valuable older file is not accidently overwritten, and (second) to make sure that a file name is descriptive enough to distinguish it from the other files in the folder.

A menu containing all the appropriate disc files in the folder is used to select the input for continued processing. A file is considered appropriate if it has the proper extension.

GEOLOGICAL MODEL DEFINITION

The geological model can be defined by coding within either of the Matlab stand alone scripts fdModUp or fdModDown. This hard coding ensures that the geological horizons are specified exactly (if necessary) or at least specified in a repeatable way.

The first card (line) of the model specifies the P-wave and S-wave velocities, and the density, related to an external boundary (either the top or bottom). The following lines give the velocities and density related to a layer, followed by offset/depth pairs which locate the layer in space. All the lines which define layers must have the same number of entries, in order to fit Matlab’s idea of a matrix, so the shorter lines must be padded out with zeros. All layers will be extended as flat surfaces from ‘xMinG’ to the first coded point, and from the last coded point to ‘xMaxG’. The ‘UpDown’ variable must be defined as character ‘Up’ or ‘Down’ to indicate which program was used.

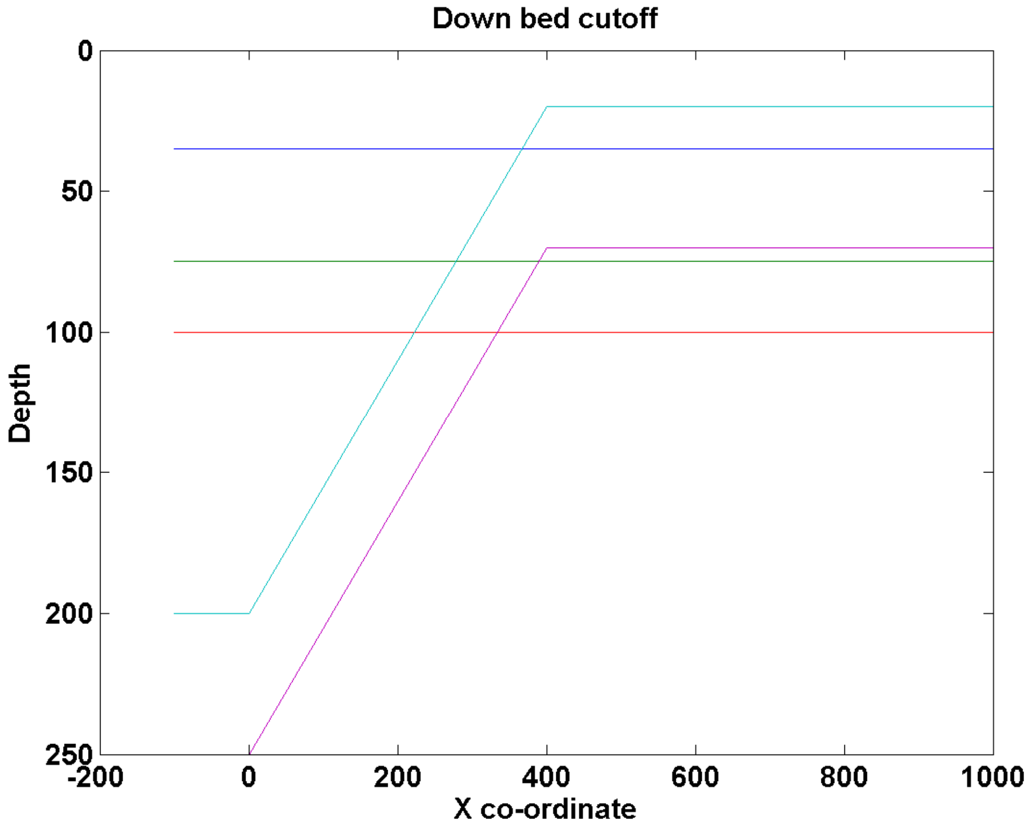

FIG. 1. An example of a geological model, which could be type ‘up’ or ‘down’. The type determines what happens at horizon crossings. All horizons have been extended flat to the limits of -100 and 1000. A part of this model will be selected for the actual finite-difference modelling.

The ‘down’ type of model, using script ‘fdModDown’, is probably the most natural to code. The first line then gives the velocities, etc. of the zone directly below the surface, and the following lines define the properties and positions from shallow to deep. The significance of the ‘down’ model is that the properties associated with an horizon apply below the horizon until a lower boundary is reached. Where a boundary specified later in the coding crosses above an earlier layer, its properties override the earlier properties. This might be used, for example, to describe more recent layers deposited on, and truncating against, a sloping basement. An example is given in Figure 4.

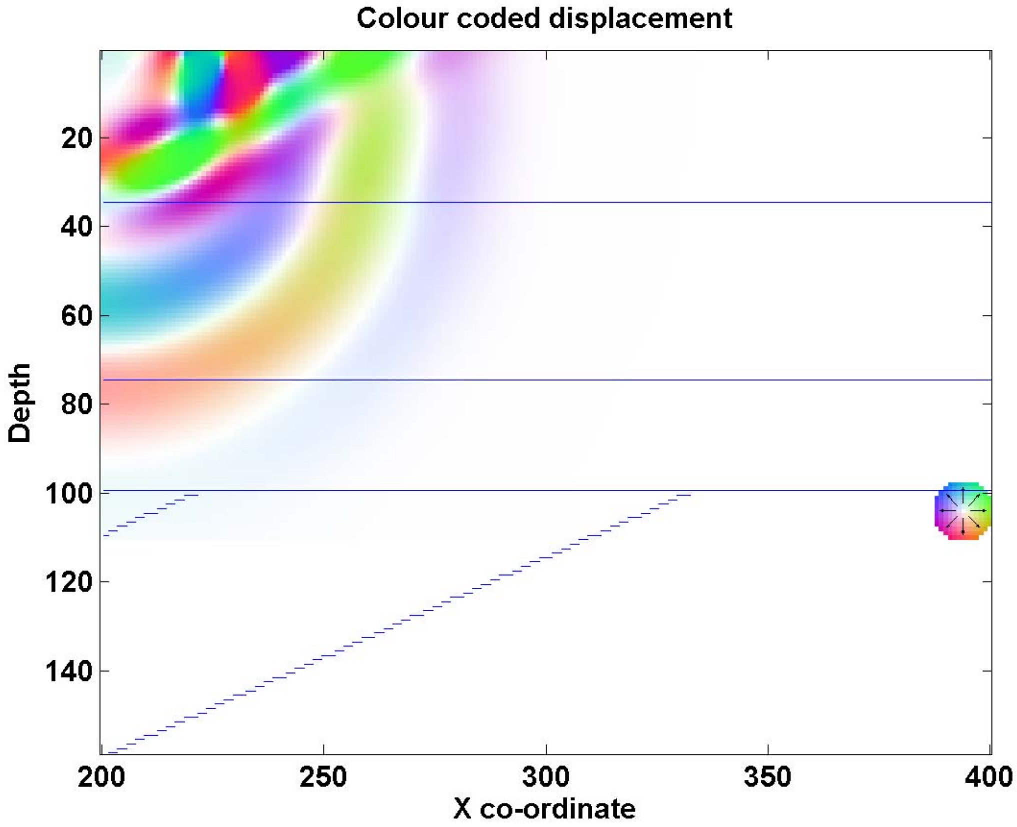

FIG. 2. The geological model defined with an 'up' series of layers from the structure in Figure 1. The first, lower horizons have been truncated by the higher horizons. The sloping layers are segmented because they mark the positions with vertical discontinuities between grid points. Note how a model section between 200 and 400 metres has been used. The energy is here coded by displacement direction.

The 'up' type of model, using script 'fdModUp', is defined from bottom to top (see Figure 2). The first line gives the velocities, etc. of the zone directly above the basement, and the following lines define the properties and positions from deep to shallow. The properties associated with an horizon apply above the horizon until a higher boundary is reached. A boundary specified later in the coding may be made to cross down through earlier layers, overriding their properties. This is a very natural way to code deposition on an erosion surface, for example.

The output from these scripts is a plot showing the location of the horizons, whether crossing or not (as in Figure 1), and disc files which describe all the conditions within the earth that affect elastic wave propagation. The files (.gfd files) can be picked up within the main finite-difference program, limited (sometimes) in horizontal extent, converted into a grid form, and illuminated with a seismic survey.

FINITE-DIFFERENCE MODELLING

The finite-difference modelling is set up by the script FDmodel. This drives the actual modelling routines, and most of the parameters should be set here by changing the Matlab code.

Operational control

Most control within the finite-difference program is by menu selection. The control menu appears when the program is started, and after an operation is selected it is executed, and the program ends. To execute another step the program must be restarted, but it normally will have available in memory most of the critical variables from the previous run.

The control options are:

1. **Save computations**. This menu item appears when finite-difference calculations have been made, but have not yet been saved to disc. Upon selection, a listing of previously saved states is presented, and a prompt appears. The file name the state will be saved under must appear between single quotes, and the .fdc extension is added automatically.

2. **Load prior**. This step will replace the finite-difference state in memory (if any) with a state previously saved to disc. A list of finite-difference saved states (.fdc files) is presented in the form of another menu where a selection may be made. From here a model may be continued (option 4) or some type of plotting or other output may be done (options 5, 6, 7, and 8).

3. **Start afresh**. This step requires a geological model, given in the form of a .gfd file, usually created by the scripts ‘fdModDown’ or ‘fdModUp’. The choice of model is again made from a menu containing all the appropriate files in the folder. The modelling parameters must be coded in the main script, in this ‘Key code 3 *****’ section. This is where the extent of the model is limited, the gridding is set up, and the seismic acquisition system is defined.

4. **Continue from previous**. This option builds on the present state of the variables in memory, either directly from a previous run, or as set up from a file on disc. The main value that must be updated here is the new total number of steps (nstep) following ‘Key code 4 *****’.

5. **Plot last frame**. This shows a snapshot of the latest stage of the finite-difference model, or the last frame of the movie. It can be used to try other plotting parameters, or perhaps to create a ‘.tif’ or ‘.esp’ file for external use. The default parameters are the same as were used in the real time and Matlab movie, but may be changed by code in this section, following ‘Key code 5 *****’. The options here are given in the **snapshots** part of this report.

6. **Plot seismic**. This option plots the accumulated surface (usually) seismic using the CREWES Matlab routine ‘plotseis’. A parameter in this section determines which component is plotted, and for now other parameters must be adjusted in a subroutine. The default is the standard wiggle variable area. An example is given in Figure 3.

7. **Play movie**. This is the Matlab movie. The only variable available for change is ‘nRepeats’, the number of times the movie is repeated from start to finish.

8. **Output SEGY**. This option uses the same data plotted with option 6, but sends it to a program that formats it into a standard SEGY file.

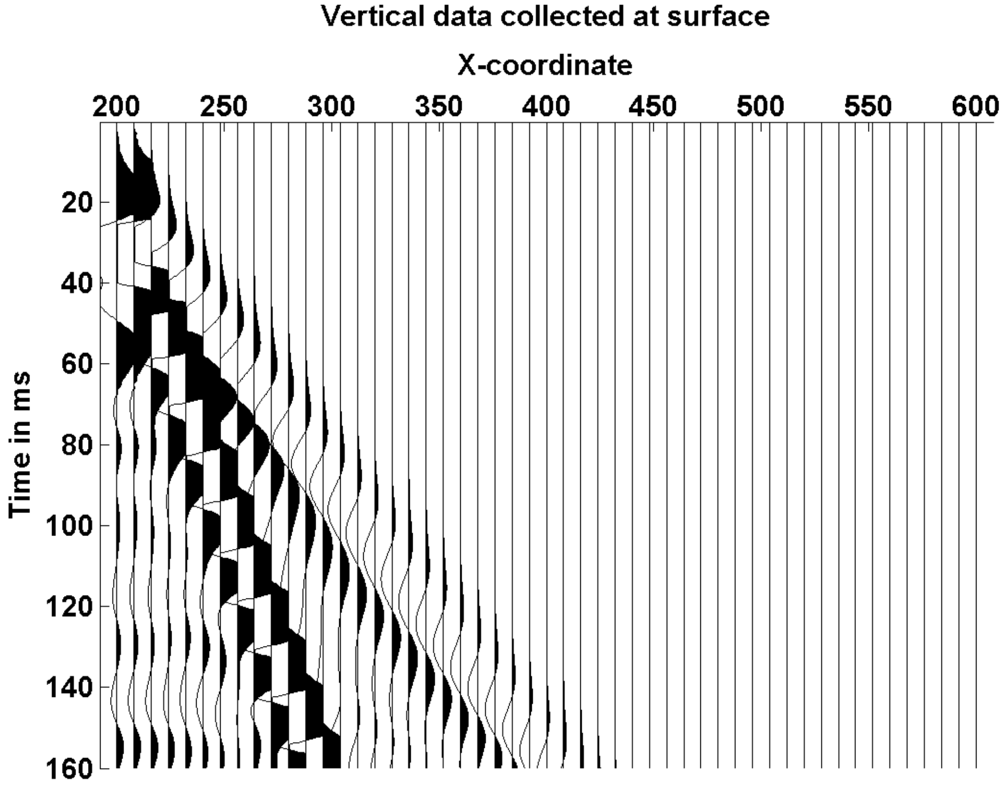

FIG. 3. Seismic data collected at the surface. The source in this case has an x-coordinate of 200 metres. The surface wave (ground roll) is the high amplitude, steeply dipping wave between the first breaks and the reflections.

General parameter selection

After the geological model is selected, finite-difference coding parameters are specified by changing the Matlab code in the mainline of the program. These parameters appear in the part of the code following section 'Key code 3 *****'. Most parameters have comments following them, with an explanation on their use. The most important general parameters are:

- **Dxz**. The spatial sample rate in the X and Z directions. This may be considered to be in metres, and is usually in the range of 1 to 5 metres.
- **Dt**. The time sample rate in seconds, usually in the range of .001 to .0002 seconds. This must be related to the spatial sample rate by the lowest velocity used in the model, see equation 1.
- **lengthX**. The length of the model in metres.

- **lengthZ**. The depth of the model in metres.

- **xMin**. The origin of the finite-difference model in metres from the origin of the geological model. These three parameters may be used to extract part of the geological model from the disc file, possibly to make a series of shot gathers

- **nstep**. The number of time steps to take in order to propagate energy through the model. nstep times Dt gives the time interval covered by the propagation.

- **nframes**. The number of movie frames to display in the course of propagating the model. These 'snapshots' are used to check on the stability of the wave propagation, how the boundary conditions are working, etc. This number also applies to the number of .tif files saved for use in later movie making.

- **shotDepth**. The depth of the explosive energy source used to generate the model. The X-coordinate is assumed to be zero, or on the left edge of the model at **xMin**.

- **centerFreq**. The center frequency of the Ricker zero phase wavelet used to generate the model.

External boundary conditions

The four sides of a finite-difference model may all be considered somewhat unique. The boundary conditions for this finite-difference program are:

- **Top** - A free surface at Z = 0. The conditions applied here are required to simulate a real seismic experiment under flat topography. They allow ground roll (Rayleigh waves) to be simulated, and give proper polarity and amplitudes to multiples.

- **Left side –** A mirror surface at offset X = 0. This is the perfect 'invisible' boundary and acts as if there was a continued geological model anti-symmetric about the zero X axis (or about **xMin**). It is suitable for most conditions, but if there is a strong vertical reflector near this axis, eventually energy will appear that comes from the equivalent anti-symmetric reflector in the 'mirror'. The default position of the energy source is on this boundary.

- **Bottom –** This is usually considered a rigid boundary, and therefore a strong reflector. These reflections must be anticipated, and if not wanted the only present option is to increase the depth of the model.

- **Right side –** This is usually coded as a transparent boundary, but it can be replaced by a rigid boundary (code iRrigid > 0). The transparent boundary will work for some time, but can eventually become unstable. The likely reason for this is that it depends on central first derivatives, which can be unstable at frequencies of more than one-half the Nyquist frequency.

Snapshots

Snapshots are plots of the displacements throughout the model at a given time, or the 'state' of the model. They are useful for showing where energy is going, and the form that it is taking. A series of these snapshots can be presented as a movie, and these can make the propagation of energy very clear.

Snapshots are usually made at regular intervals while a finite-difference model is being generated, to ensure that the program is working properly. The same interval choice can also be used to make a series of .tif files, which must then be processed with other software to make another form of movie, .mpg files. These files are often used in PowerPoint and other presentations.

There are several choices in the format of finite-difference snapshots:

- **Wiggle traces.** (mvPlot = 0 or 1) These look like seismic traces, with all samples plotted in depth, but at some limited sampling in offset in order to fit on the page. mvPlot = 0 gives the Uz component, and mvPlot = 1 gives the Ux component.

- **Colour coded displacement.** (mvPlot = 2) This display combines the displacements in X and Z to give a total displacement direction which is colour coded. A key in the lower/right corner shows the colour shade for each displacement direction. A more intense colour indicates a greater displacement.

- **Colour coded wave type.** (mvPlot = 3) This display does not depict displacement directly, but by using the derivatives of the displacements, interprets whether the energy displayed is shear or compression energy. The key in the lower/right corner shows how compression energy (P) appears as red/green shades, and shear energy (T) appears as blue/yellow shades.

- **Colour coded absolute displacement.** (mvPlot = 4) This display is the same as for mvPlot = 2, except that the scaling is specified without reference to the amplitude of the displacements within the plot. This can make a significant difference within a movie, where the larger amplitudes near the energy source do not affect the plotted amplitudes throughout the volume. Scaling is controlled by the variable mvAmp.

- **Vector displacement.** (mvPlot > 4) This display is the most natural way to show vector displacements, and uses the Matlab routine 'quiver'. An arrow is plotted at the grid points in the direction of the displacement, and with a length proportional to the amplitude. The dynamic range is not very good, but it shows the higher amplitude parts of a frame very clearly, and especially the central parts of a Rayleigh wave.

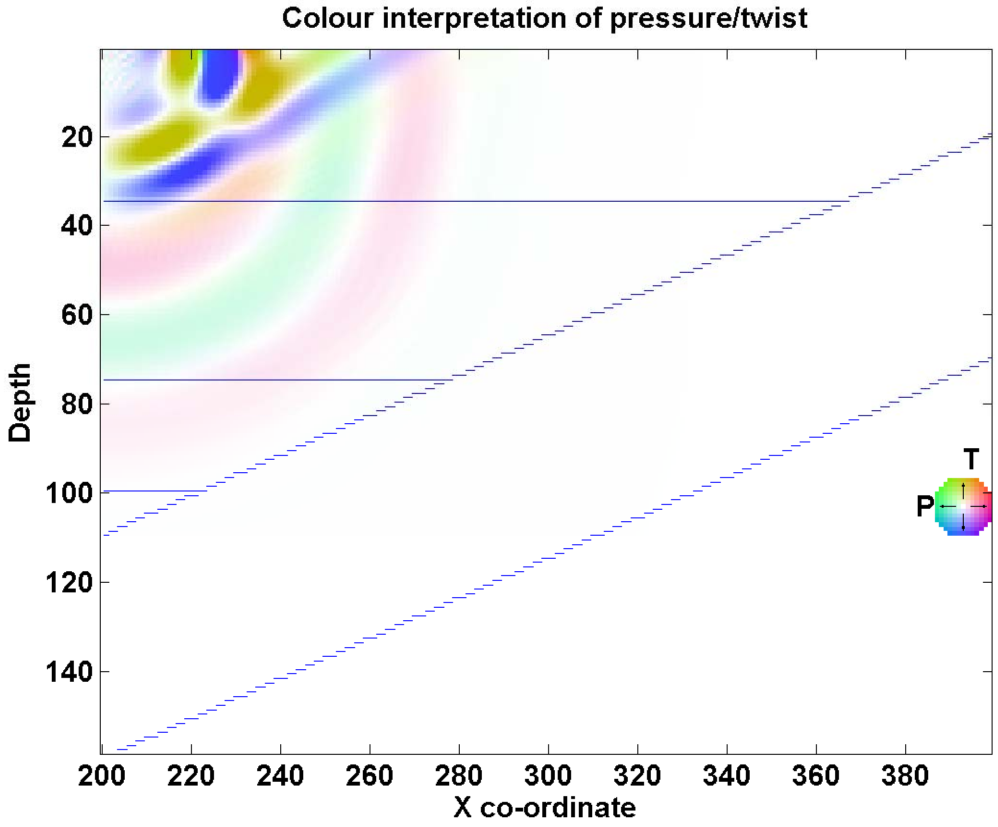

FIG. 4. The colour interpretation plot (mvPlot = 3) of the finite-difference model after 350 time steps. This is the 'down' version of the geological model shown in Figure 1. The source is at 200 metres, and the model is shown from 200 to 400 metres. The pressure energy may be seen in red and green, and the shear energy in blue and yellow.

The variable mvClip is useful in the plots 2, 3, and 4, where it prevents a limited number of high amplitude events from dominating the whole plot. It takes the form of a fraction, which is multiplied by the highest amplitude in the frame to give a new maximum value. All the amplitudes higher than this new maximum are then 'clipped' back to this level.

Plots 2, 3, and 4 have some layer boundaries marked. The method that was chosen to mark boundaries was to plot a short horizontal line between every pair of grid points where a change in the P-wave velocity was more than 1 (m/sec) in the vertical direction. This could potentially be changed to another value (S-wave or density), another increment threshold, or to vertical lines. The advantage to marking horizons in this way is that it may be used on a model generated by any means.

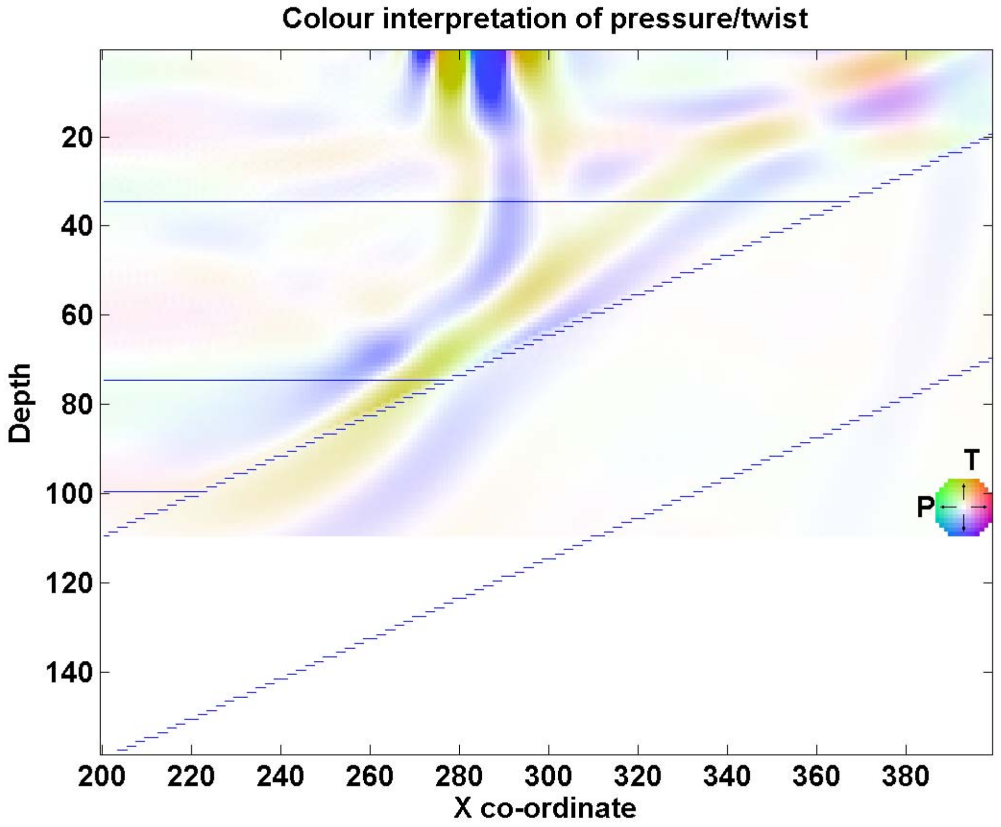

FIG. 5. The colour interpretation plot of the finite-difference model after 800 time steps. The model is the same as that used for Figure 4. The body waves are mostly the blue/yellow shear waves, and the Rayleigh wave consists mainly of shear energy near the surface, with small amounts of red/green pressure energy between the shear bands.

A realistic model

An example of a more realistic set of traces from a ground roll model is given in Figure 6. In this case the shear velocity at the surface is 600 m/sec, and at 7 metres the shear velocity increases to 1000 m/sec. This scenario is much closer to what will be found in the field, although still over simplified. The ground roll here is not confined to a narrow zone, as in Figure 3, where the surface wave travelled in a layer thick enough to hold almost all of it. Here the velocity dispersion may be seen, as the lower frequency component increases its lead over the steep dipping higher frequency component. The same plot shows how refracted energy takes over the first breaks from the direct arrivals, because of the same near surface conditions.

The data modelling done in Figure 3 is a good test of the parameters, showing whether the sampling has sufficient resolution, and even whether the software represents a free surface accurately.

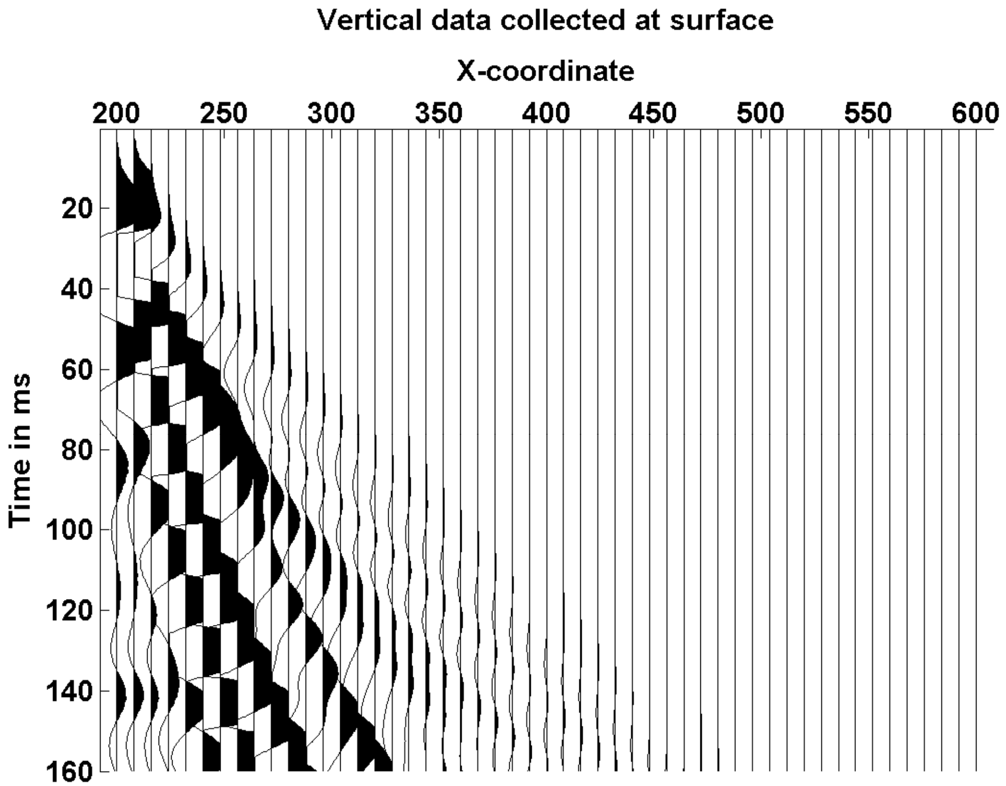

FIG. 6. This set of traces is similar to Figure 3, except the surface wave spans a zone where two velocities apply. The wave then disperses over a wider cone, with the lower frequency components leading the higher frequencies. The first breaks are also affected.

CONCLUSIONS

The software described here has been proved to have considerable value in simulating realistic seismic recordings.

FUTURE WORK

The boundary conditions available for these models have often been found to be less than ideal. Some effort will be directed to improving the right side and bottom boundaries.

REFERENCES

Manning, P. M., 2008, Techniques to enhance the accuracy and efficiency of finite-difference modelling for the propagation of elastic waves: PhD thesis, The University of Calgary.