

2D finite-difference modeling in Matlab, version 1

Peter M. Manning

ABSTRACT

An updated CREWES 2D elastic finite-difference modeling program is offered for general use. It has many of the features of the original workbench version, but it may be set up so that very little Matlab coding is required. The program runs from two ascii files which must be named and stored in the working directory. The first file describes the geology, and is easiest to use for layer cake cases. The second file specifies how the finite-difference gridded data is obtained from the geology file, and gives the parameters of the finite-difference operations. The main controls of the program are by menu, and besides the main command to start the computations, there are commands to save or continue computations, and to output in various ways. An essential feature of this version is the capacity to apply a wavenumber correction filter.

INTRODUCTION

A preliminary version of this program was introduced to the CREWES group in 2008. It was used to a limited extent: by students at the Universities of Calgary and Houston, and by some CREWES staff. The program was also used by the author in several studies, and the features added to enable these studies are included with the newer version.

It had become obvious that the almost random spread of critical parameters throughout the clutter of a working program was not productive, so most were removed to an independent ascii file. The description of the geological model was also removed from a Matlab coded file, and specified in a second ascii file.

The structure of the program remains the same as that of the original version. Model calculations may be started and monitored by watching the snapshots which come up on the computer screen. After the number of time steps specified has been calculated, several quality control plots may be made. If the modeling appears successful, it may be continued through a further series of time steps.

FINITE-DIFFERENCE CONCEPTS

A finite-difference model consists of a spatial grid where each node has a set of specified 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 requirement for stability of the finite-difference calculation places a limit on the ratio of the sample rates in time and space. It may be given as

$$\Delta t \leq \Delta x / (v\sqrt{2}), \quad 1$$

where Δt and Δx are the time and space sample rates, and v is the highest 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.

RAYLEIGH WAVES

The program 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 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 three main types of disc files: two which provide input conditions, and one which stores calculation results after a given number of steps.

Geological model definition (.geo files)

The geological definition file (.geo file) must be used to describe the geology of the section of the earth to be modelled. The first data line of the model gives the compressional velocity, the shear velocity, and the density of the rocks at and below the model top. The second data line gives another triplet of velocities and density, and one or more pairs of offsets and depths which define the top of a second layer. Deeper layers may be added in the same fashion. When a layer later in the file is defined to cross above an earlier layer, its parameters replace those defined earlier.

This type of model definition is most natural with layer cake geology (where only a set of depths is required), or layers with gradual thickness changes (where each may require as few as two offset/depth pairs). An example of the first type of model is given in Figure 1.

After the layer definition lines is one line which gives the left and right limits of the model. All layers are then projected to these limits at the same level as are specified at the edges of the layer definitions above.

A final data line gives a title that the model will have.

A ‘%’ sign on a line marks the start of comments, and everything to the right of this mark is ignored within the program. A ‘%’ sign at the start of a line means that the whole line is ignored. This convention is carried over from Matlab.

These definition files may be coded with the Matlab editor, or another ascii editor. The .geo extension must be specified. A distinctive name should be used, partly because the name is usually used as part of the named .tif (plot) files that the program may generate.

```

1 % Vp, Vs, rho, X1, Z1, X2, Z2, X3, Z3, X4, Z4, X5, Z5,
2 4000 2500 2.70 0 0
3 3000 1875 2.50 0 115
4 4000 2500 2.70 0 135
5 4500 2600 2.75 0 250
6 -20 1210 % Left and right X limits to which the model will be extended
7 Microseismic model % The plot title

```

FIG. 1. An example of a simply layered geological definition file. This describes a 20 metre layer of 3000 m/sec between layers of 4000 m/sec. The input data values are at the beginning of a line; after the ‘%’ characters are the comments.

Parameter definition (.parm files)

The parameter definition file may be used to specify how the finite-difference modelling will be done, and how data will be collected in the course of the finite-difference modelling. An example of a .parm file is shown in Figure 2. The data lines of the parameter definition file all have the same format:

- At the left are the variable names within the program. These can not be changed because the program recognizes only these names.

- In the second column is the value assigned to the parameter on the left. Numeric values must be coded as numbers; character strings must be coded within single quotes, and must be preceded by an '=' sign.
- To the right of the first two columns, a '%' character marks the beginning of comments about the parameters to the left. Entire lines may also be marked as comments.

```

1 | #Parameter file for 2D finite-difference program
2 | Dt          .0001  %Sample rate in seconds
3 | Dxz         0.75   %Sample rate in metres
4 | xMin        0      %Starting X point within geological model
5 | lengthX     400   %X samples of the calculated model
6 | lengthZ     250   %Z
7 | nstep       1600   %2000      %101      %Number of time steps
8 | nframes     50     %Number of movie frames (< nstep)
9 | shotDepth   125   %Depth of the energy source
10 | shotX       0      %200      %X position of the energy source
11 | centreFreq  300   %Frequency in Herz at centre of Ricker wavelet
12 | energySrc   31     %Code number of the energy source (10,11,12,13,21,22,23)
13 | %           10-explosion, 11-Z_rupture, 12-double couple, 13-squeeze bulge
14 | %           21-Z_monopole, 22-X_monopole, 23-external XZ twist
15 | %           31 - vertExpl
16 | trX         250   %X position of the well in the case of 'Z' acquisition
17 | trZ         0      %Z level of flat 'X' acquisition
18 | iLbnd       7      %Boundary code left    <= 0 for rigid, =7 mirror
19 | iRbnd       0      %Boundary code right   <= 0 for rigid, =8 free
20 | iTbnd       0      %Boundary code top     <= 0 for rigid, =8 free
21 | iBnd        0      %Boundary code bottom  <= 0 for rigid, =8 free
22 | mvXmax      1000   %X length of the displayed model
23 | mvZmax      1000   %Z
24 | mvPlot      2      %Code number of the movie plot (2,3,4,5)
25 | %           2-displacement, 3-pressure/twist, 4-abs amp displacement, 5-arrows
26 | mvAmp       300   %Amplitude for movie code 4 (larger is higher amplitude)
27 | mvClip      .5    %Amplitude clip level - fraction of 1 (1 is unclipped)
28 | gfdFile =    'spSimp.geo' %Geological definition file
29 | wncvar =    'c75p40s24min' %Wave number correction file ('' indicates no corrections)
30 | %wncvar =    'c57p40s20' %Wave number correction file ('' indicates no corrections)
31 | %wncvar =    'c57p40s20min' %Wave number correction file ('' indicates no corrections)
32 | %wncvar =    '' %Wave number correction file ('' indicates no corrections)
33 | mvTif =     '' %Character string included in the movie tiff files ('' indicates no tiff file)
34 |

```

FIG. 2. An example of a parameter definition file (.parm file). The names on the left specify variables within the program, and cannot be changed. The numbers to the right of the variable names are assigned to that variable. The comments (marked by '%' characters) describe the variables. Note that for this case, line 28 specifies the geological (.geo) model.

The usual way to use the program is to bring a functioning .parm file into the editor and modify the parameter values for the purposes of the new model. The variable names are then preserved, and the type of variable and a realistic value for them may be seen.

Computed finite-difference (.cfd files)

The computed finite-difference files are designed to save all the computed results from a completed series of modelling steps. These files are not created automatically, but can be generated any time before a new model run is started. The name of one of these files is chosen by the user after reviewing the list of .cfd files in the folder. Then the newly named file can be chosen to replace an un-needed earlier file, or provide a completely new computed model.

The saved file may be loaded into the program again later when needed for output plots, or to continue further wavefield propagation.

CODING FOR THE FINITE-DIFFERENCE COMPUTATION

The main parameters which specify how the finite-difference computations will be done are stored in one file (the .parm file) and an example is shown in Figure 2. This file can be kept to match with its propagation results.

Many of the parameters here may be debugged by coding a limited number of time steps to see if the early propagation works as expected. It is then easy to change a parameter, restart, and compare the results. In a case where everything looks good, the process may be continued (using the same parameters) without restarting.

The parameters do not have to be in a particular order. The ones which can now be picked up from the .parm file 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 (m) of the energy source used to generate the model.
- **shotX** The distance (m) from the model edge to the energy source. A zero here puts the source at $X = 0$, or at **xMin**.
- **centerFreq.** The center frequency of the Ricker zero phase wavelet used to generate the model.

- **trX** The X distance from the model edge to a ‘well’ where the displacements at all depths will be saved at each time step.
- **trZ** The Z distance from the model top to a ‘spread’ where the displacements at all offsets will be saved at each time step. If $trZ = 0$, the spread will be the usual surface spread. If $trZ > 0$, the spread will be simulate recording in a horizontal well.
- **energySrc** The variable is used to specify a number of well known 2D energy source geometries, and a few rather arbitrary ones. The codes used are 10, 11, 12, 13, 21, 22, 23 & 31, and are displayed in Figures 3 and 4.

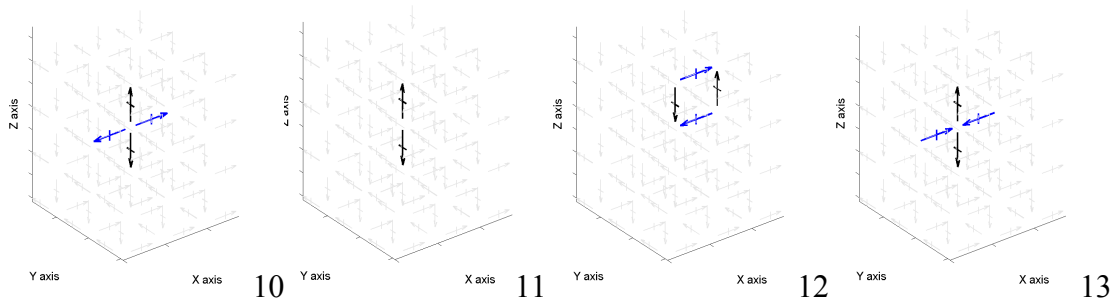


FIG. 3. Energy source pattern types: 10 - Explosion, 11 - Vertical dipole, 12 - Double-couple 13 - Squeeze-bulge

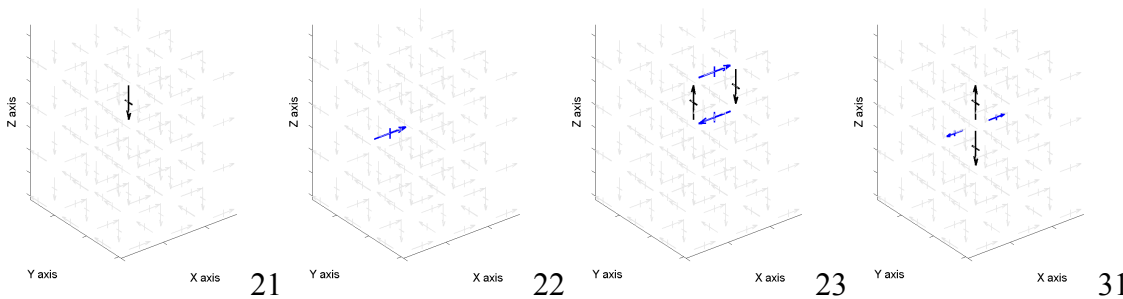


FIG. 4. Energy source pattern types: 21 – Z_monopole, 22 – X_monopole, 23 – External twist 31 – Vertical enhanced explosion

- **iLbnd** The left boundary code: ≤ 0 for rigid, $= 7$ for a mirror. A mirror boundary may be used with a symmetric source right on the boundary to reduce the execution time for a symmetric model.
- **iRbnd** The right boundary code: ≤ 0 for rigid, $= 8$ for a free surface
- **iTbnd** The top boundary code: ≤ 0 for rigid, $= 8$ for a free surface. This is the natural position for a physical free surface.
- **iBbnd** The bottom boundary code: ≤ 0 for rigid, $= 8$ for a free surface

The following parameters apply to the snapshots that appear in the course of the finite-difference propagation (determined by ‘nframes’), and to a movie that will be made from .tif files. All of these choices have more elaborate descriptions in the ‘SNAPSHOT PLOT MENU’ section.

- **mvPlot** The code number for the type of movie plot: 2 – displacement, 3 – pressure/twist, 4 – absolute displacement, 5 – arrows
- **mvXmax** The maximum X value to include in the snapshots. This may be less than the size of the propagation area, otherwise it is ignored.
- **mvZmax** The maximum Z value to include in the snapshots. This may be less than the size of the propagation area or it will be ignored.
- **mvAmp** The absolute amplitude, which applies only to the ‘absolute displacement’ option (4).
- **mvClip** The level of amplitude clipping
- **mvTif** serves as a switch. Coding only as ‘ ’ will indicate that no movie will be made. If one or more characters appear in the quote, they will be part of the root name of a collection of ‘.tif’ files used to make the wave propagation movie.

There are three character variables that may be coded for the basic finite-difference modelling. They are: gfdFile, wncvar, and mvTif. Unlike most of the parameters in the parameter file, the character variables must be given within single quotes, and preceded by an = sign. mvTif is described above. The other two are:

- **gfdFile** must specify a valid root name for a ‘.geo’ file.
- **wncvar** specifies the wave number correction file. Not coding or null coding this file name indicates that that wave number corrections will not be made.

MAIN OPERATIONS MENU

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 the 'plot last frame' routine (5) is started, so the user can make sure the right file was loaded.
3. **Start afresh.** This step requires a parameter file, given in the form of a .parm file. The choice of file is again made from a menu containing all the appropriate files in the folder.
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 value that must be updated here is the new total number of steps (nstep).
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. A *snapshot plot menu* follows for plot parameters.
6. **Plot seismic.** This option plots the horizontal or vertical seismic data accumulated with each time step using the CREWES Matlab routine 'plotseis'. A *seismic plot menu* follows for parameters.
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 SEG Y.** This option uses the same data plotted with option 6, but sends it to a file where a program may format it into a standard SEG Y file.

SNAPSHOT PLOT MENU

The menu presents a series of options with the default values for the options listed in the menu. The last item of the menu is labeled 'All OK'. The earlier options may be changed one at a time, and the menu will change to show the new value for that option. Changes may be made until they are all satisfactory, and then the 'All OK' button is hit to make the plot.

The buttons in the main menu are:

- **Displacement** is the default for the first button because it is the most commonly used option. If this button is hit, another menu appears which lists the display types available. They are:
 - **Wiggle traces** are represented by two buttons: 'Uz' and 'Ux'. These displays look like seismic traces, with all samples plotted in depth, but at some limited sampling in X in order to fit on the page.

- **Colour coded displacement**, the ‘Displacement’ button, is the default display. 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**, the ‘Press/Twist’ button, is an interpreted plot. 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**, the ‘Abs displace’ button, is a version of the ‘Displacement’ button. The display is the same 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**, or ‘Arrows’ 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.
- **Clip** is useful to prevent 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 value in the frame to give a new maximum value. All the amplitudes higher than this new maximum are then ‘clipped’ back to this level.
- **Xmin** sets the new left edge of the plot frame.
- **Xmax** sets the new right edge of the frame.
- **Zmax** sets the new bottom edge of the frame.
- **Tiff tag** has a default equivalent to coding ‘’, and indicates that no .tif file will be created. One or more alphanumerics between the quotes indicates that a .tif file must be created, and these characters will be included within the name of the .tif file. Other characters will be included in the file name to help distinguish it from other plots generated.
- **All OK** closes the menu and begins the plot.

SEISMIC PLOT MENU

Seismic plots are made from either of two lines through the model, where the displacements have been collected at every time step. A plot may be made of the 'surface' line, collected at depth 'trZ', or from the 'well' line, collected at an offset of 'trX'. The menu works just as the 'SNAPSHOT PLOT MENU' does, and the choices are:

- **Plot x vs t** is the top button default, indicating the 'surface' traces. This may be toggled to 'z vs t', or the 'well' traces.
- **Uz** is the default displacement direction for the plot, and it may be toggled to 'Ux'.
- **Amplitude** is currently defaulted to 1.5. A larger number here gives higher amplitudes; a smaller number gives lower amplitudes.
- **Xmax** applies to the 'surface' plot, and it may be made less than the maximum X for the model.
- **Zmax** applies to the 'well' plot, and it may be made less than the maximum Z for the model.
- **Tiff tag** works like the same variable in the 'SNAPSHOT' menu. It has a default equivalent to coding '', and indicates that no .tif file will be created. A significant entry in the text string will save the seismic plot as a '.tif' file with this string included in the file name.
- **All OK** closes the menu and begins the plot.

EXAMPLE

Figure 5 is a snapshot from a microseismic study done with this program. It shows the complexity of the wavefields that are often generated from comparatively simple geological models.

The model consisted of a 20 m. 3000 m/sec. bed with a background of 4000 m/sec. The Vp/Vs ratio was 1.6, and the source wavelet was a 300 Hz zero-phase Ricker from a 'vertical enhanced explosion'. The spatial sample rate was 0.75 m. and the time sample rate was 0.1 msec. Wavenumber corrections were applied, and were designed for the 4000 m/sec. zones.

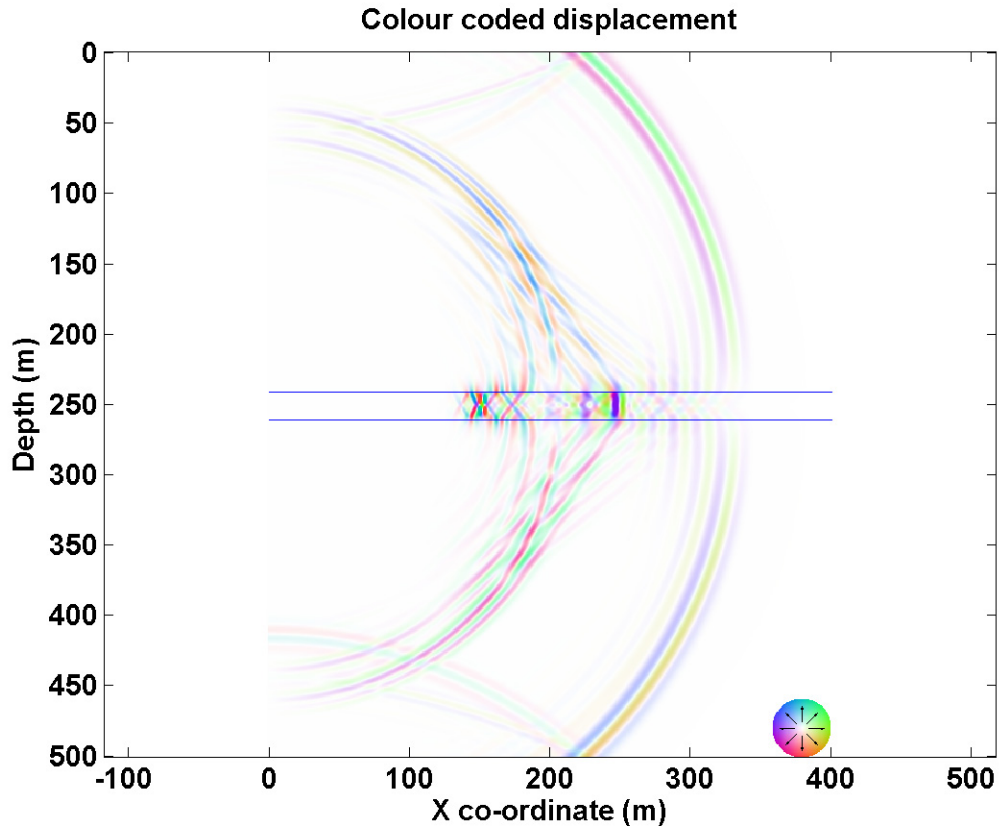


FIG. 5. The program was used to model this seismic response of a 300 Hz source within a 20 m. low velocity zone. The headwaves create complex wavefields from a relatively simple geological model.

CONCLUSIONS

The current version of the CREWES finite-difference modelling code shows several improvements over the 2008 version.

- Most of the program may be run without coding Matlab. This allows a broader set of users and reduces the chance of damage to the original code.
- Placing critical data values in ascii files makes their importance more obvious and user understanding more complete. These compact files are named, and can be kept as a reference to the model wavefields that they generated.
- The control and parameter selection is much easier where they are available within a series of menus. This is especially true where input and output disc files are concerned.

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.