

Modelling from well logs with CREWES tools

Heather J.E. Lloyd and Gary F. Margrave

SUMMARY

In 1994, LOGSEC was written using the MATLAB programming language. Its original features included, importing logs, editing horizons, editing logs using LOGEDIT, propagating logs along the cross-section, converting the cross-section to time, and computing theoretical seismograms. Over time the code had to be modified to correct programming code changes and to add additional functionality to the program. Some of these new functions include computing a synthetic stack using the SYNGRAM algorithm and being able to export the section for further analysis with other tools. LOGEDIT also has new functions including a Gassmann Function which allows for fluid substitution in the logs, a Gardner Function and a Mudrock Line Function. A Blackfoot Channel model was created and used to illustrate the functions in both LOGEDIT and LOGSEC. Information on the functions in LOGSEC and LOGEDIT can be found in their respective User Guide or help files.

INTRODUCTION

LOGSEC is a program written with MATLAB, which allows the creation of cross-sections by importing wells and editing horizons. It was created in 1994, by Gary Margrave, however as MATLAB evolved the LOGSEC code required maintenance. In 2009, most of these issues were worked out making LOGSEC a valuable tool once again.

The following is a comprehensive list of the functions that are available in LOGSEC:

- Import LAS logs and control what portions of the file will be imported.
- Create, edit and delete horizons in the cross-section.
- Propagate logs (combine well logs to form one complete new log) as a section or individually
- Control how the logs will be propagated with respect to truncation or stretching at horizons
- Edit well logs or propagated logs using LOGEDIT
- Promote or add a propagated log to a well
- Shift the datum of the cross-section
- Compute time-depth curves from sonic logs
- View the section in time
- Load or create wavelets from a file or using WAVELETED
- Compute theoretical seismograms
- Compute synthetic stacks using the SYNGRAM algorithm
- Export log sections for further analysis with other tools

A model of a Blackfoot Channel system will be used to illustrate the major features of LOGSEC and highlight the features of supporting programs such as LOGEDIT and SYNGRAM. However, first we will explain the LOGSEC log propagation algorithm.

Appendix 2: The LOGSEC Propagation Algorithm

HOW LOGSEC WORKS

The “LOGSEC propagation algorithm” refers to the logical rules used to predict how a particular well log will appear some distance away from its measured location. This appendix describes that algorithm together with the “LOGSEC log synthesis algorithm”. The later is used to merge propagated logs from all wells on the cross section into a single predicted result or “synthesis”.

LOGSEC requires that a cross section be built with definite connectivity. That is, each horizon must be defined not only in terms of its geometry but also in how it connects to the rest of the cross section. Cross sections built in this way are said to be “topologically connected”. Horizons must not just “end” dangling in space but must connect to other cross sections or to the edge of the model. (See “Horizon Editing in Link Mode” for a description of the mechanics of making these connections.) The places where horizons connect are called “vertices” or “links” and are represented in LOGSEC as anchored, identical points on several horizons. The connectedness rule can be visualized as a requirement that each horizon must be mappable to the boundaries of the cross section by following it through any number of vertices and onto other horizons as needed until the boundary is reached. This means that isolated bodies (such as lenses) are currently not supported though they can be simulated. A summary of horizon requirements is:

- **No isolated bodies:** Though lenses can be easily simulated.
- **Horizons must intersect at vertices**
- **Horizons must be single valued in depth:** That is recumbent folds are not supported.
- **Faults are not supported:** Small faults can be simulated as small horizons though the distance along the bed will be measured incorrectly.

Given a topologically connected cross section, the algorithm next requires the specification of “propagation guide horizons” above and below each horizon. These guide horizons specify the trajectories along which samples just above and just below a particular horizon propagate. For example, if horizon B has its upper guide horizon specified as horizon A, the log sample immediately above horizon B moves along the cross section parallel to horizon A. This situation is illustrated in Figure 1.

Let the horizon whose guide curves are being set be called the primary horizon. Possible choices for guide horizons are any horizon on the cross section (including the primary) or horizontal. By default, all guide horizons are set to be equal to the primary horizon. To specify something else, you must use the horizon edit mode called “Set Propagation Specs” which is described elsewhere in this document. If the guide horizon does not extend over the same inline coordinate range as the primary, then the guide horizon is mapped onto other horizons at vertices as needed. The guide curves may not change with position along the primary except at vertices. Therefore if it is absolutely necessary to change guide curves, a primary may be segmented and then joined with a vertex at the segment boundaries.

An example of the log propagation process is illustrated in Figure 1. For each layer, the process proceeds as follows:

- Determine the horizons which form the layer boundary at the well location, x_w . Let the log segment found between those horizons be called $ls(x_w)$.
- Determine how the layer bounding horizons at x_w map into horizons as the propagation location x_p .
- Transport the endpoints of $ls(x_w)$ along the appropriate assigned guide horizons from x_w to x_p to determine a propagated log interval at x_p .
- Stretch or compress $ls(x_w)$ into the propagated log interval at x_p (apply antialias filter if required.) to compute $ls(x_p)$.
- Finally window $ls(x_p)$ with the layer boundaries as they actually exist at x_p . That is, any points which lie outside the layer are rejected and, if $ls(x_p)$ does not span the layer it is padded with null values.

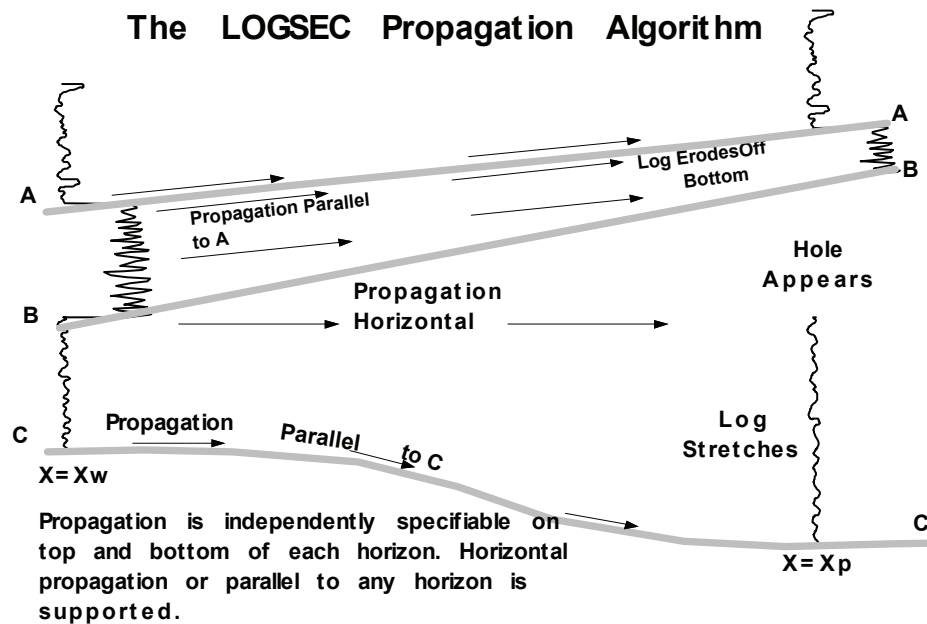


FIG. 1. The LOGSEC Propagation Algorithm

Consider how these steps work in figure A1 for the propagation of the log segment at x_w between horizons A and B. As noted on the figure, the guide horizons at the top and bottom of the layer are both horizon A. Thus, by the end of the fourth step above, the log segment has been transported parallel to horizon A, without stretching or compressing, to x_p . Since horizon B is trending upward relative to A, only the upper 40% or so of the log segment lies in the layer at x_p and the remainder is rejected. This effectively achieves truncation (or erosion) off the bottom of the log. This is thus propagation situation in which log information is lost.

A slightly different situation occurs for the log segment between horizons B and C. Here the upper guide horizon is horizontal and the lower one is parallel to C (which is the default). Thus, at the end of step three above, we determine that the propagated log interval is actually greater than the interval at x_w . Therefore, in the next step the log segment at x_w is stretched into the larger propagated interval at x_p . However, even this larger propagated interval is not sufficient to fill the layer, so a hole appears at the top of

the layer. In this situation, there is no loss of log information in the propagation process but the log at x_w contains insufficient information to fully fill the cross section at x_p . If the cross section contained another well further to the right, then this information could be used to complete the solution.

The algorithm described above can propagate a log from any position on a cross section to any other position. If the goal is to predict log measurements at any location and if multiple wells exist on the cross section, then it is obvious that each well provides a possible prediction at the output location and some means of combining this multiplicity of solutions is required. Figure 2 shows the method used in LOGSEC which is called the log synthesis algorithm.

The LOGSEC Log Synthesis Algorithm

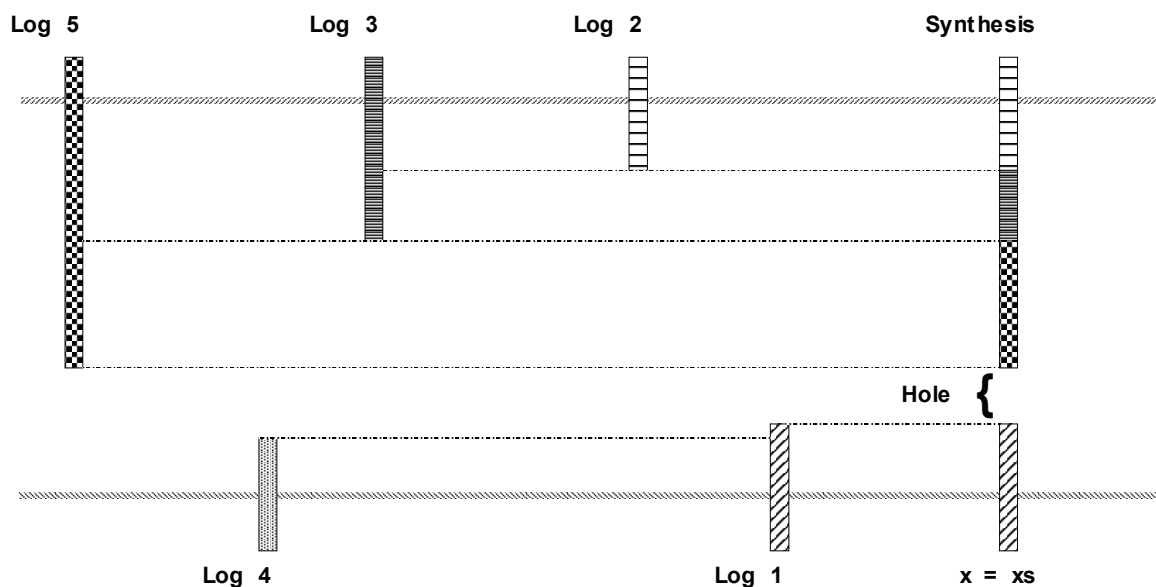


FIG. 2. The LOGSEC synthesis algorithm, used to combine various logs at a single location, is illustrated.

As with log propagation, log synthesis proceeds layer by layer. For each layer, left and right sided composite propagated logs together with sample-by-sample inverse distance weighting are determined. Either one-sided composite propagated log is formed by:

- Beginning with the closest log, propagate each available log segment to the synthesis location.
- Each sample of each propagated log segment may contribute to the synthesis if, and only if, a closer log segment has not already contributed. Thus the closer logs screen out more distant contributions.

When both right and left sided composites are available, they are combined using inverse distance weighting. Holes in the final synthesis represent regions where no log information on the cross section was sufficient for specification. This is because the final

synthesis only allows a hole to appear where both right and left sided composites lack information.

THE MODEL

The Blackfoot Channel model was built by modifying the dry Blackfoot-1409 well. This well did not have a channel top picked and had little porosity where the channel should have been located. To create the two channel wells, Blackfoot-1009 and Blackfoot-0809, a portion of the Blackfoot-0808 density log was copied, using the copy/paste function in LOGEDIT, and then pasted into the producing interval of both 1009 and 0809. The Mudrock Line Function in LOGEDIT was then used to calculate a shear sonic for all three wells. The mudrock line is a linear relation between a shear velocity log and a compressional velocity log; more information can be found in the paper by Castagna et al, 1985. The Gassmann Function in LOGEDIT was applied to wells 1009 and 0809 to change the fluid in the two wells. In Blackfoot-1009 the well was adjusted to contain 30% oil, 10% gas and 60% water. The Blackfoot-0809 well was adjusted to contain 75% gas and 25% water. The Gassmann equation uses the bulk modulus and the density of the rock and fluid to compute the response of the compressional sonic and shear sonic logs; more information can be found in the paper by Smith et al., 2003. The copy and paste features in LOGEDIT were also used to move the top of the channel in 0809 up by 10m and thus increasing the thickness of the channel by 10m.

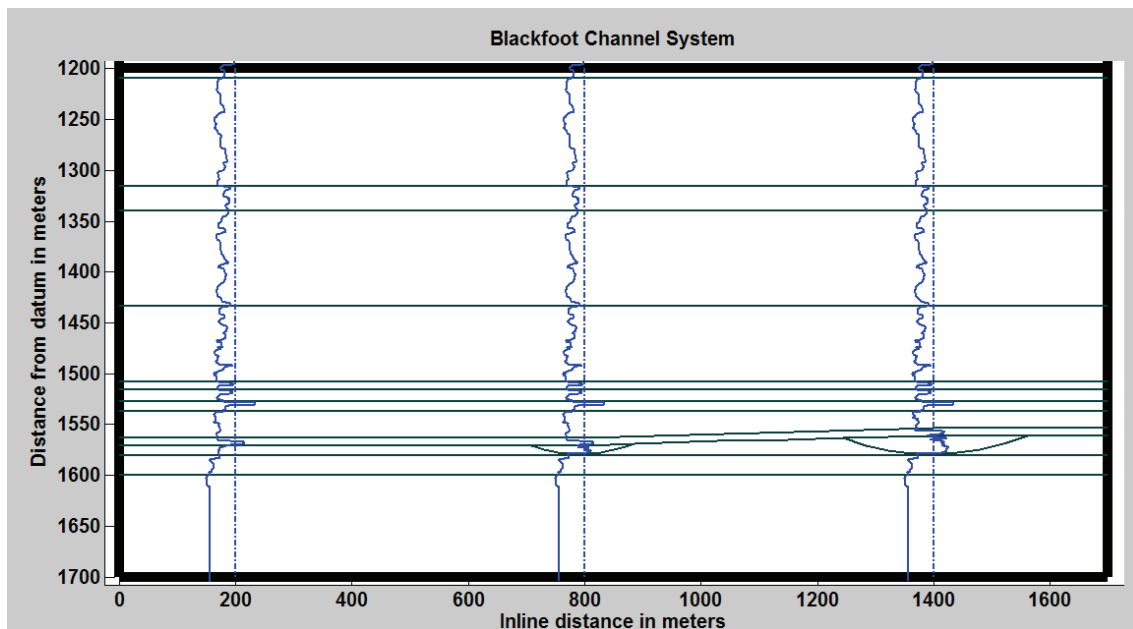


FIG 3. This cross-section shows the position of the wells as well as the extent of the channels.

These wells were then used to create a cross-section placing the 1409 well at an inline measurement of 200m, the 1009 well at an inline of 800m and the 0809 well at an inline of 1400. The channel in both wells 0809 and 1009 were created by linking the channel bottom to the Sunburst horizon. The Blackfoot Channel model can be seen in Figure 3.

FUNCTION OVERVIEW

LOGSEC has the ability to combine logs in different locations to create a new 'propagated log'. Figure 4 shows a section of density propagated logs. Using these log sections a time display of the data can be computed (Figure 5), as well as theoretic seismogram can be produced (Figure 6). The time section uses a section of sonic logs and an overburden function to calculate a T-Z curve (Time-Depth curve) which is used to convert the data from depth to time. The theoretic seismogram is computed using a sonic section and a density section to create an impulse response log which is then convolved with a wavelet. The theoretic seismogram only models normal incidence waveforms.

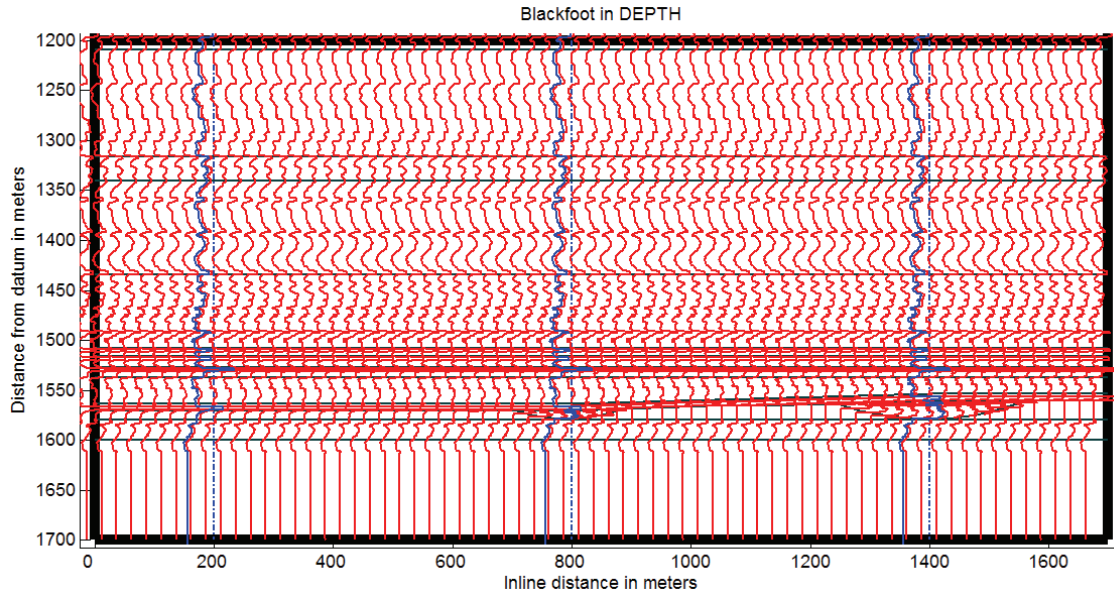


FIG. 4. This is a density log section. The red logs have been propagated by combining the logs at the wells indicated by the blue lines. The way the logs are propagated shows how the channels would look like if looking at a section of traces.

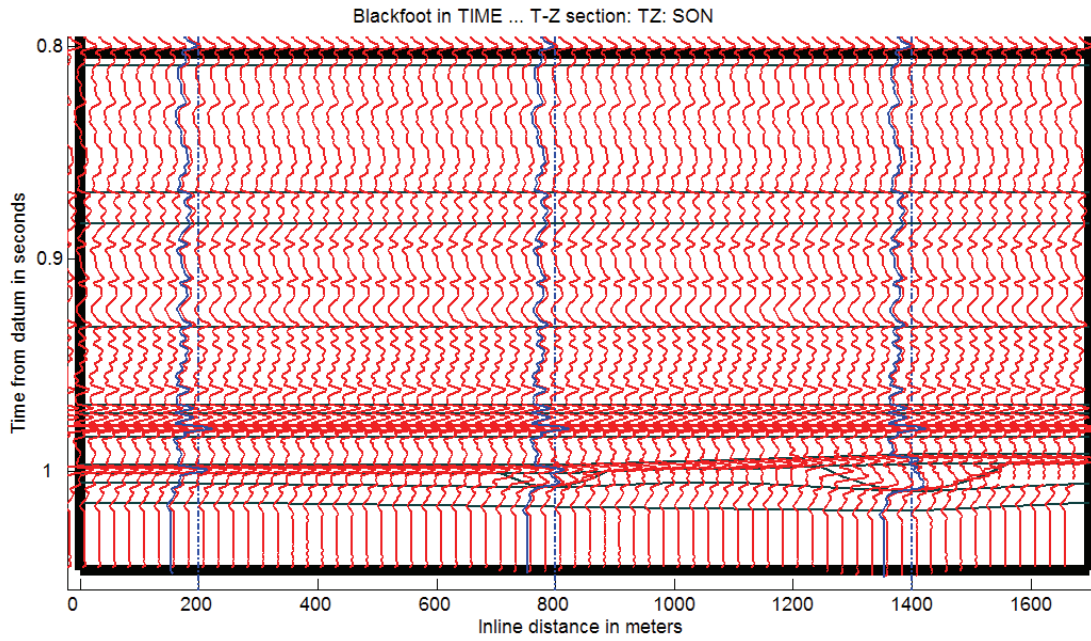


FIG. 5. The time section is a interpretation of what the cross-section would look like in respect to time. It is calculated using T-Z curves that are derived by a section of sonic logs.

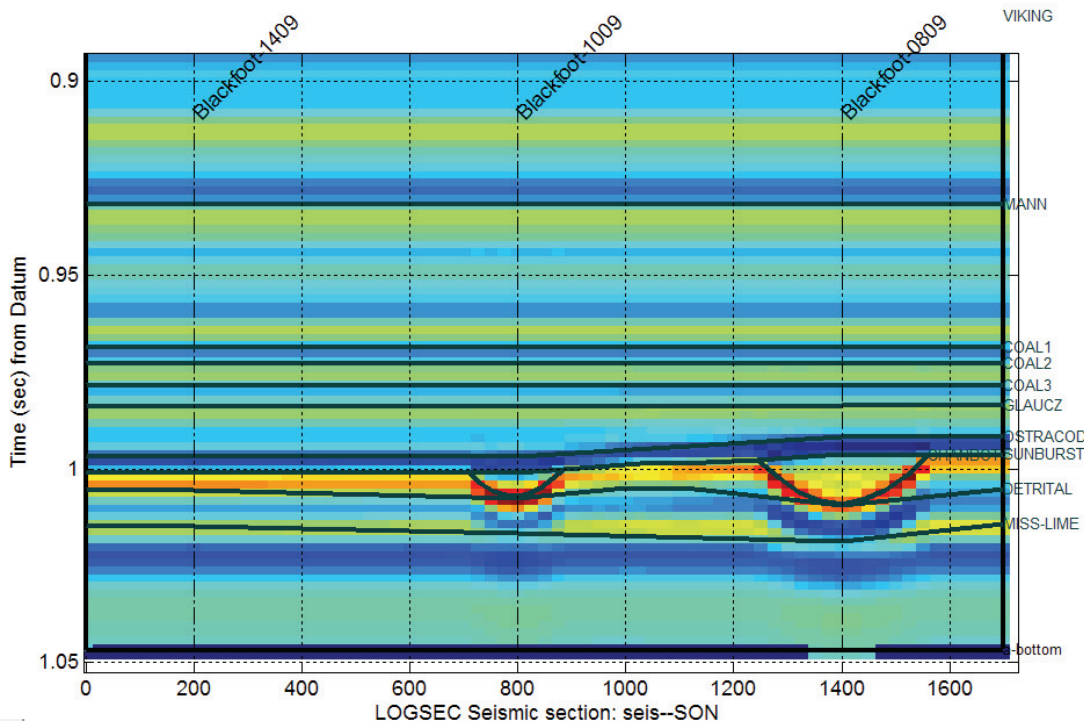


FIG. 6. This display shows a theoretical seismogram produced using the seismic section, the density section, and a 40 Hz minimum phase impulse wavelet.

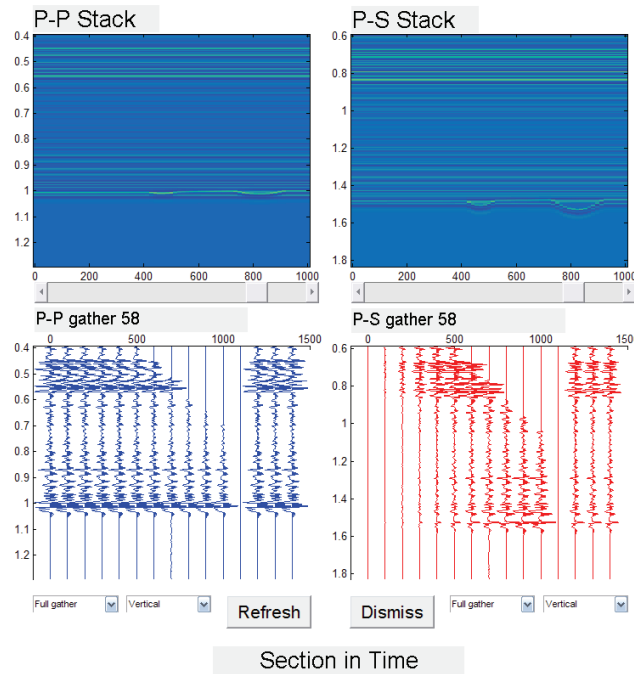


FIG. 7. This display shows the vertical stacks at the top for both the P-P and P-S situations as well as gathers for each shot at the bottom. This display is available in both depth and time but is only shown in time here.

Some recent additions to the LOGSEC program include a synthetic stack function, and an export section function. The synthetic stack function uses SYNGRAM to compute a synthetic seismogram for each trace and then stacks them to produce a synthetic stacked section. The synthetic stack algorithm calculates both the normal incidence and angled incidence waveforms. The synthetic stack display shows the vertical P-P stack and vertical P-S stack at the top as well as each gather at a specific inline location for both P-P and P-S. This display is available for both depth and time but is only shown in time in Figure 7.

Another new feature of LOGSEC is the ability to export a propagated log section for use with other tools. Figure 8 shows a sonic section that was converted to velocity and then used with the finite differencing tool AFD_SHOTREC to create a shot record.

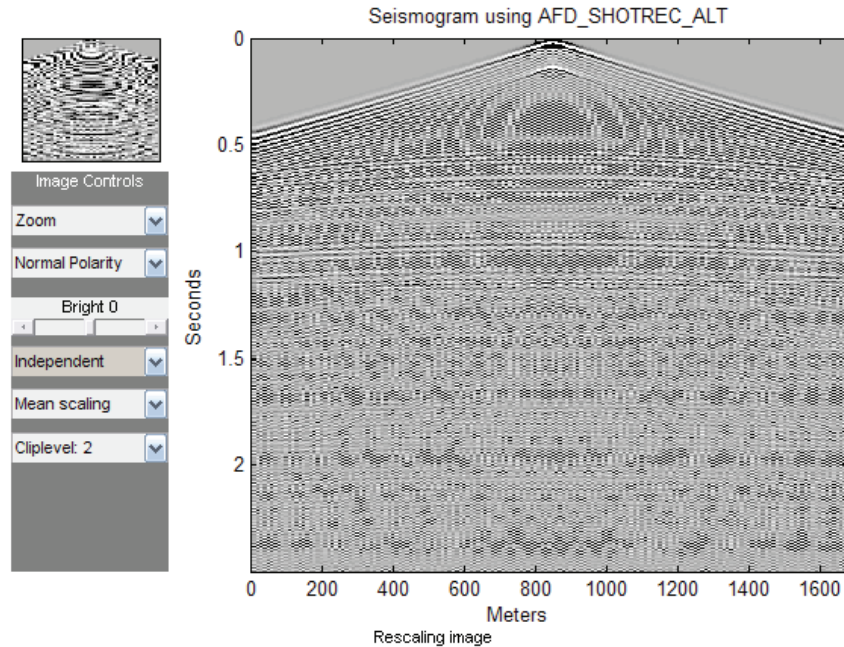


FIG. 8. This is an exported velocity section from the Blackfoot model that has been resampled with a grid spacing of 10 meters and then run through the AFD_SHOTREC_ALT program in the finite differencing toolbox.

CONCLUSION

LOGSEC is a powerful tool to model cross-sections. Wells can be added and horizons can be modified to create a host of different cross-sectional types. The tools in LOGEDIT further the advantage of LOGSEC, as they allow the user to change the logs including, copying and pasting sections from a different log, replacing the fluids in logs and creating a new log using the log algebra tools. LOGSEC can then be used to create theoretical seismograms, synthetic stacks and the data can be exported for use with other toolboxes.

REFERENCES

- Castagna, J.P., Batzle, M.L. and Eastwood, R.L., 1985, Relationships between compressional wave and shear-wave velocities in clastic silicate rocks: *Geophysics*, **50**, 571-581.
- Smith, Tad M., Sondergeld, Carl H., and Rai, Chandra S., 2003, Gassmann fluid substitutions: A tutorial *Geophysics* **68**, 430.
- Margrave, Gary, 2009, LOGSEC Users Guide, CREWES Document.
- Margrave, Gary, 2009, LOGEDIT Users Guide, CREWES Document.