Stanford Exploration Project, Report 97, July 8, 1998, pages 147–157

Short Note

Interval velocity estimation with a null-space

Robert G. Clapp, Paul Sava, and Jon F. Claerbout [1]

keywords: *tutorial, velocity, null-space*

INTRODUCTION

Many powerful velocity estimation programs have been written based on intuition and exhaustive experience. It is not easy to improve such programs. Recently we presented a more formal approach showing how directional interpolation in interval velocity can exploit null-spaces in order to build more "geologically pleasing" velocity models (Clapp et al., 1997).

Here we build the velocity model from surface seismology while retaining the null-space aspect. We begin from fundamental concepts in Geophysical Estimation Theory (Claerbout, 1997b) and we define the simplest interval velocity estimation method we can that includes the notion of null-space. Generally, we minimize interval velocity "wiggliness" where we don't have good quality reflections.

The method presented here is linear (almost). Although this simple approach may perform badly on certain practical tests (such as with multiple reflections) it should be a useful beginning in areas where careful statistical averaging is essential. Since the approach is a systematic interpretation of a general theory, the pathways to further improvements rely more on logic than on intuition and experience. An attractive feature of this simple method is that it readily extends towards estimation of the lateral variation in interval velocity. It also leads towards a logically consistent way of combining well velocities and seismic velocities.

BASIC IDEA

Assuming a $v(z)$ earth with flat reflectors, we can define the theoretical RMS velocities by

$$\mathbf{C}\mathbf{u} \quad = \quad \mathbf{d}. \tag{1}$$

where

[1] **email:** bob@sep.stanford.edu, paul@sep.stanford.edu,jon@geo.stanford.edu,

$\mathbf{C}$ is the matrix of causal integration, a lower triangular matrix of ones.

$\mathbf{D}$ is the matrix of causal differentiation, namely, $\mathbf{D} = \mathbf{C}^{-1}$.

$\mathbf{u}$ is a vector whose components range over the vertical traveltime depth τ, and whose component values contain the interval velocity squared v_{interval}^2.

$\mathbf{d}$ is a data vector whose components range over the vertical travel time depth τ, and whose component values contain the scaled RMS velocity squared $\tau v_{\text{RMS}}^2/\Delta\tau$ where $\tau/\Delta\tau$ is the index on the time axis.

Start from a CMP gather $q(t,i)$ moveout corrected with velocity v. A good starting guess for our RMS velocity function is the maximum "instantaneous stack energy"

$$\text{stack(t, v)} = \sum_{i=0}^{n} q(t,i). \tag{2}$$

An alternate starting guess for our RMS velocity function is the conventional one of maximum semblance

$$\text{semb(t, v)} = \frac{\frac{1}{n}\sum_{i=1}^{n} q(t,i)}{\sqrt{\frac{1}{n}\sum_{i=1}^{n} q(t,i)^2}}. \tag{3}$$

In addition to the RMS velocities we need a diagonal weighting matrix $\mathbf{W}$, again found from stack energy or semblance, that differentiates between RMS velocities which we have confidence in (at reflectors) and ones that are more a function of noise in the data. Our data fitting goal is to minimize the residual

$$\mathbf{0} \quad \approx \quad \mathbf{W}\left[\mathbf{Cu} - \mathbf{d}\right]. \tag{4}$$

Because we are multiplying our RMS function by τ, we must must make a slight change in our weighting function to give early times approximately the same priority as later times.

$$\mathbf{W}' = \frac{\Delta\tau}{\tau}\mathbf{W} \tag{5}$$

To find the interval velocity where there is no data (where the stack power theoretically vanishes) we have the "model damping" goal to minimize the wiggliness $\mathbf{p}$ of the squared interval velocity $\mathbf{u}$ where $\mathbf{p}$ equals

$$\mathbf{0} \quad \approx \quad \mathbf{Du} \quad = \quad \mathbf{p}. \tag{6}$$

To speed convergence we "precondition" these goals (Fomel et al., 1997) by changing the optimization variable from interval velocity squared $\mathbf{u}$ to its wiggliness $\mathbf{p}$. Substituting $\mathbf{u} = \mathbf{Cp}$ gives the two goals expressed as a function of wiggliness $\mathbf{p}$.

$$\mathbf{0} \;\approx\; \mathbf{W}'\left[\mathbf{C}^2\mathbf{p} - \mathbf{d}\right] \tag{7}$$

$$\mathbf{0} \;\approx\; \epsilon\mathbf{p}. \tag{8}$$

SIMPLE EXAMPLE

As a simple example of the methodology, we started from a single CMP gather (Figure 1). We then scanned over velocities calculating semblance and instantaneous stack energy (Figure 2). We started with an initial RMS velocity function $\mathbf{d}$ by picking the maximum semblance within a fairway. We used the corresponding instantaneous stack energy for our weighting function, $\mathbf{W}$. We applied fitting goals (7) and (8) to the picked RMS function and obtained Figure 3 a smooth, reasonable velocity function.

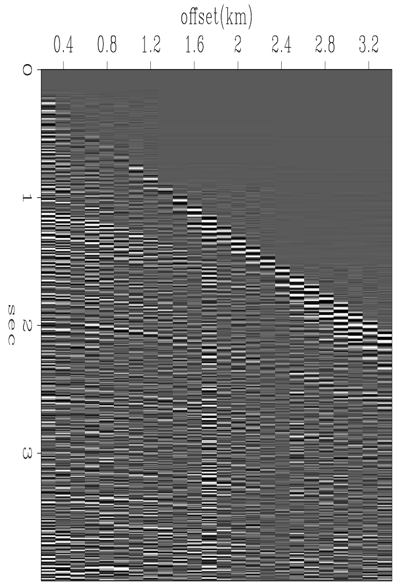


Figure 1: Input CMP gather from the Gulf of Mexico. bob1-data [ER]

Block Models

The methodology also nicely lends itself to producing block velocity models. For example we can produce a velocity model with linear velocity functions within the layers by rewriting our fitting goals as:

$$\begin{aligned} \mathbf{0} &\approx \mathbf{W}'(\mathbf{CCGp} - \mathbf{d}) \\ \mathbf{0} &\approx \epsilon \mathbf{p} \end{aligned} \tag{9}$$

where $\mathbf{G}$ is a diagonal matrix of 1s and 0s, 1s at layer boundaries and 0s everywhere else. As Figure 4 shows, our velocity function is approximately linear within the layers and has a reduced wiggliness compared to the smooth interval velocity curve.

ADDITIONAL CONCERNS

The methodology raises several additional questions:

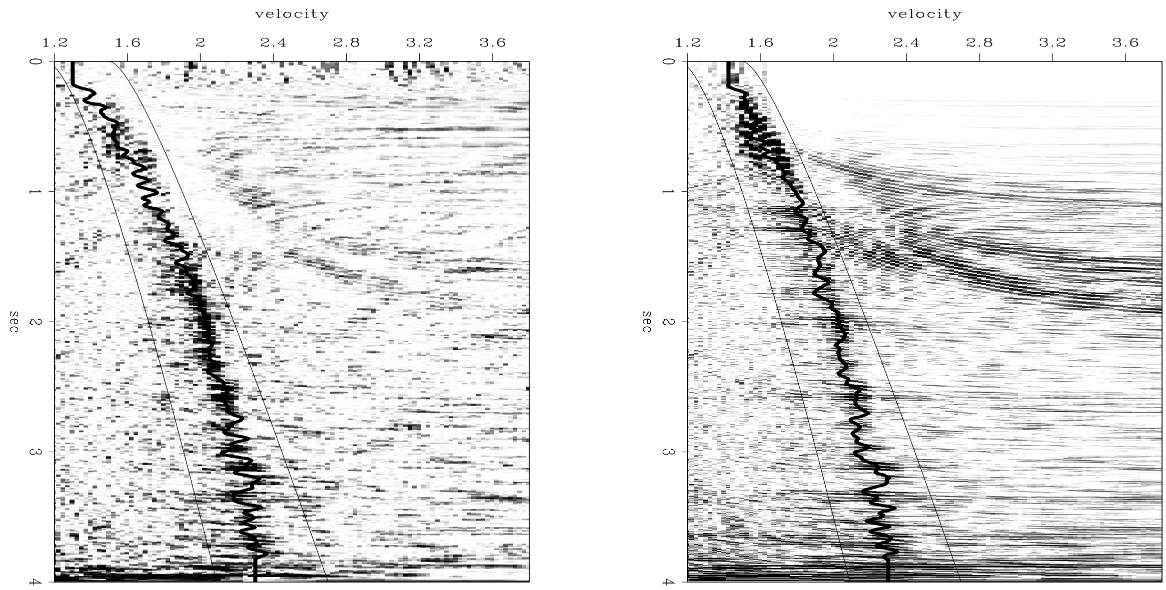
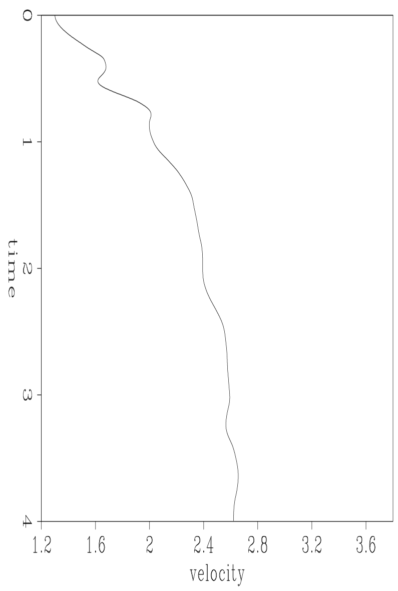


Figure 2: Two different approaches to picking starting RMS and weighting functions. Both start by scanning over velocity and then picking maximum amplitude within a fairway. To avoid edge problems early and late time velocities are pre-determined. The left frame shows a semblance scan and the resulting RMS function; the right panel, same information for instantaneous stack energy bob1-rms-picks [ER]

Figure 3: Estimated interval velocity obtained from fitting goals (7) and (8). bob1-scale-residuals [ER]



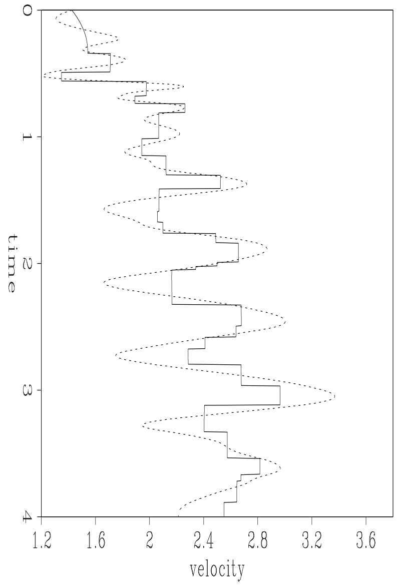


Figure 4: The result of the inversion for a blocky velocity model using the fitting goals (9). bob1-bvint.stack [ER]

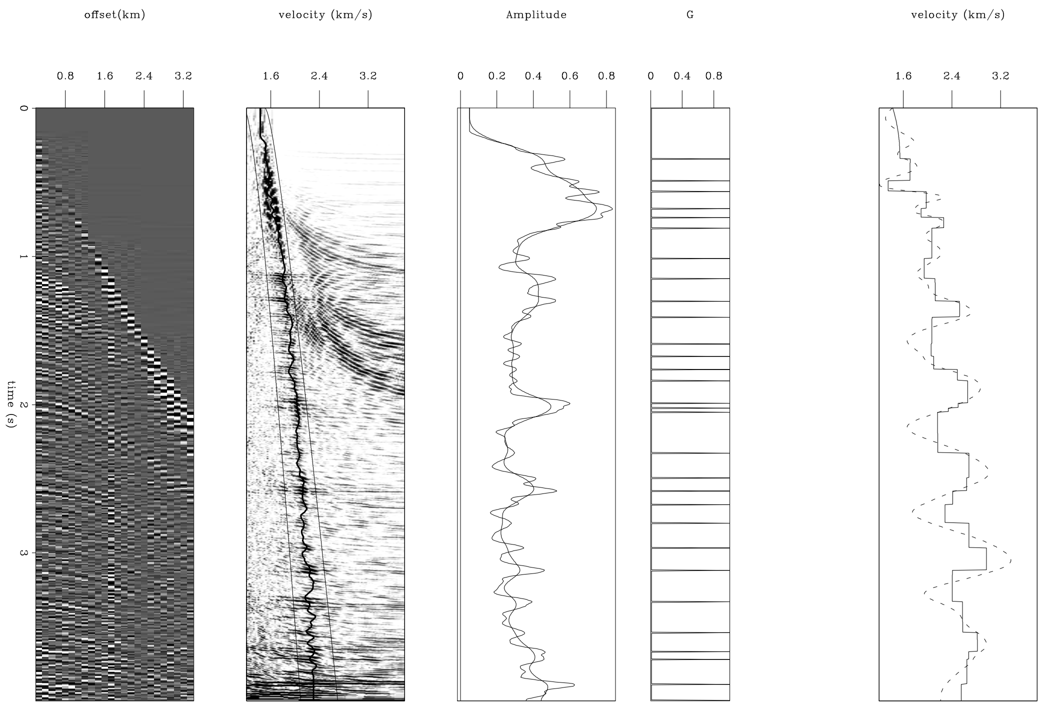


Figure 5: From left to right: CMP gather, velocity scan with picked RMS velocity within a fairway, smoothed and unsmoothed stacking power, the **B** function, continuous (dashed) and blocky (solid) interval velocity. bob1-composite.stack [ER]

- **How well does the method recover the correct interval velocity?**

  We constructed a synthetic interval velocity model, and then modeled a CMP gather using a finite difference code, Figure 6. We then picked RMS velocities and attempted to estimate our interval velocity. As Figure 7 shows, we did a very good job of recovering the correct interval velocity.

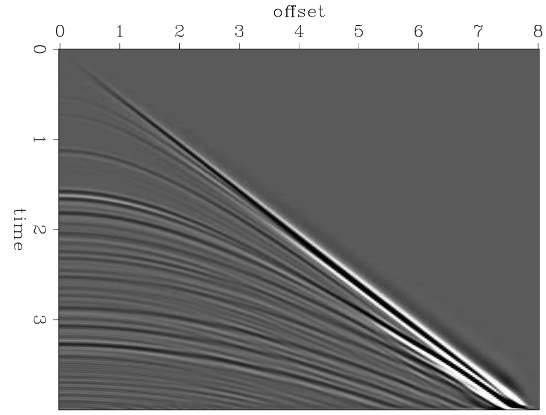


Figure 6: Modeled synthetic data. bob1-synthetic-data [ER]

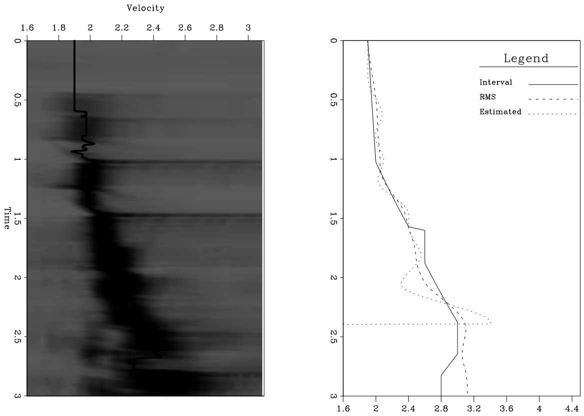


Figure 7: Initial interval velocity, picked RMS curve, and inverted interval velocity curve. bob1-synthetic [ER]

- **How well does the method handle bad data?**

  Using the same dataset (Figure 6) we introduced some low-energy noise to our velocity scan and then attempted to invert back to our interval velocities. As Figure 8 shows, even with the grossly inaccurate RMS picks at approximately 1.5 seconds we were still able to recover a reasonable interval velocity curve.

- **Is our preconditioned variable zero mean and approximately white?** To obtain maximum benefit from preconditioning we want our preconditioning variable ($\mathbf{p}$) to be white (Nash and Sofer, 1996). Not surprisingly the preconditioned variable for this problem is not quite zero-mean. Velocity generally increases with depth, which means the slope of velocity will on average be greater than one. Considering the limited resolution of the seismic method, it is not surprising that our preconditioned variable does not meet the second whiteness criteria either.

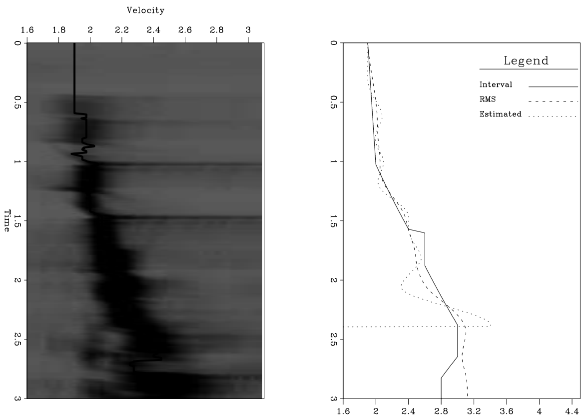


Figure 8: Left panel, RMS picks overlaying semblance scan with noise introduced at approximately 1.5 seconds. Right panel, the resulting interval velocity curve, the curve without the bad data, and our bad-data RMS curve. bob1-gap [ER]

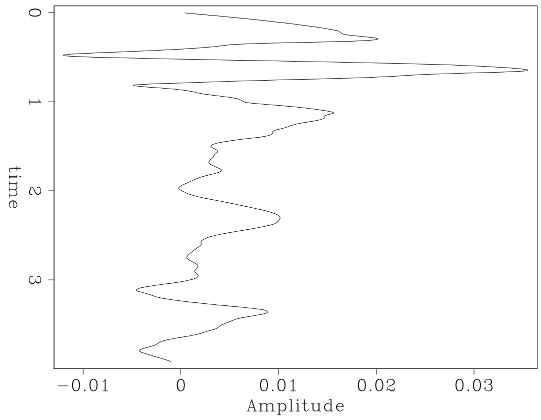


Figure 9: Our preconditioned variable for the continuous case. bob1-xvals [ER]

2-D VELOCITY ESTIMATION

An attractive feature of the velocity-estimation procedure defined in this paper is that it is linear. This means there should be no unpleasant surprises associated with multiple minima or with convergence speed. If bad results are found, they should arise directly from bad RMS velocity picks or from the possibly inappropriate association of stack power with the statistical weighing function.

Another attractive feature of the velocity-estimation procedure defined here is that it leads directly to a rational approach to finding interval velocity as a function of midpoint. Traditionally, velocity at each midpoint is estimated separately. Thus one midpoint cannot help neighboring estimates, nor are unrealistically rapid variations with midpoint suppressed in a natural way. For example, suppose we redefine our fitting goals as:

$$0 \approx \mathbf{W}' \left[\mathbf{C}^2 \mathbf{L}^{-1} \mathbf{p} - \mathbf{d} \right] \tag{10}$$

$$\mathbf{0} \approx \epsilon \mathbf{p}. \tag{11}$$

where $\mathbf{L}$ is a roughening operator such as the Laplacian (in the space of midpoint and traveltime depth) Starting with picked RMS velocities (Figure 10) from a North Sea salt structure, attempted to invert for interval velocity. As Figure 11 shows, for small, long wavelength variations in lateral velocity (Dix being approximately correct) we obtain a smooth, reasonable interval velocity model with minimal effort. For a

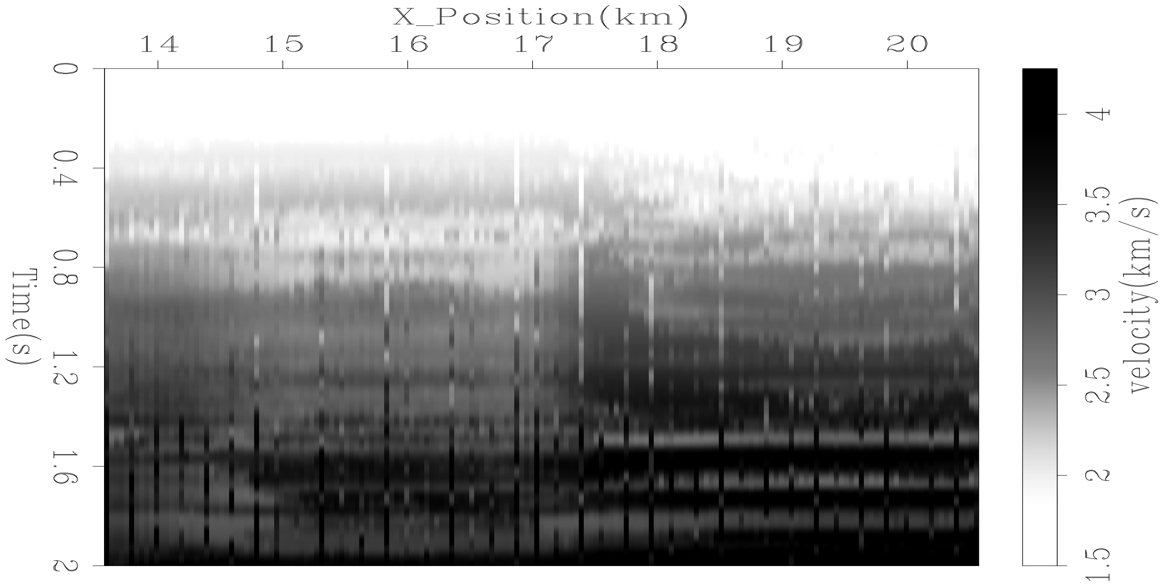


Figure 10: 2-D RMS velocity function picked by finding the maximum semblance within a fairway. Note that for some CMP locations, the picked velocity is completely inaccurate. The inversion algorithm takes these inaccuracies into account and leads to a smooth model (11). bob1-data2d [ER]

further test we replaced the Laplacian of equation (11) with steering filters (Clapp et al., 1997) built from dips off an interval interval velocity model (Figure 12) built with a more sophisticated scheme (nearest offset section could just have easily been used). We then applied the 2-D inversion and obtained Figure 13. Generally, we did a good job recovering the salt structure, but are still susceptible to multiple infected velocity picks.

CONCLUSIONS

We can write our interval velocity estimation from stacking velocity as a simple linear inverse. The method proves effective in quickly finding smooth and blocky interval velocity models from automatically picking stacking velocity. Early work in extending it to 2-D shows promise.

REFERENCES

Claerbout, J. F., 1997, Geophysical exploration mapping: Environmental soundings image enhancement: Stanford Exploration Project.

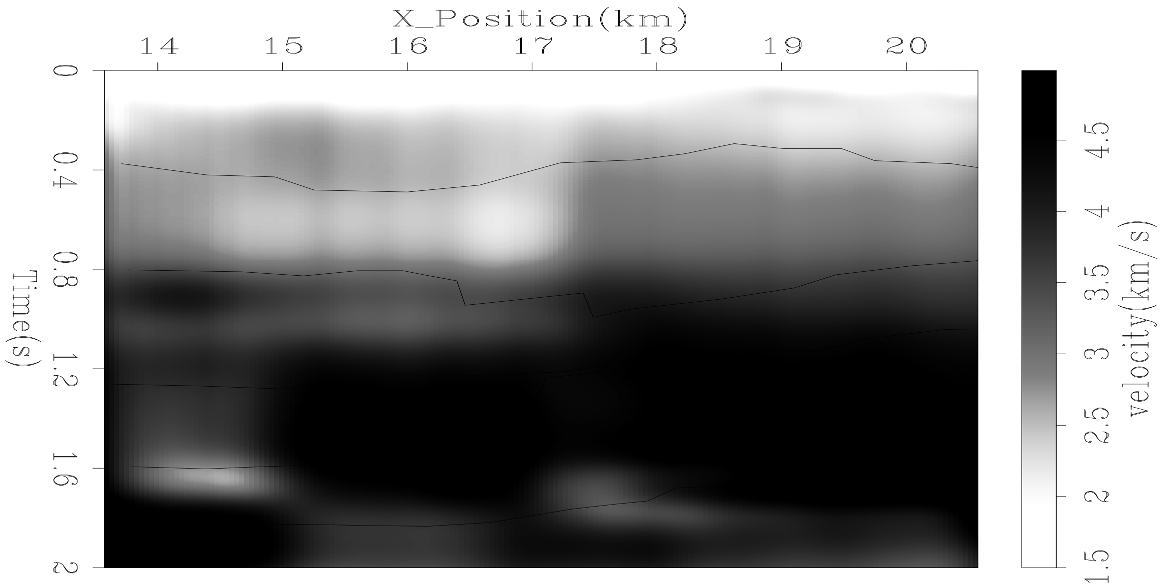


Figure 11: 2-D interval velocity function obtained from applying fitting goals (10) and (11) after 30 iterations. bob1-vint2d [ER]

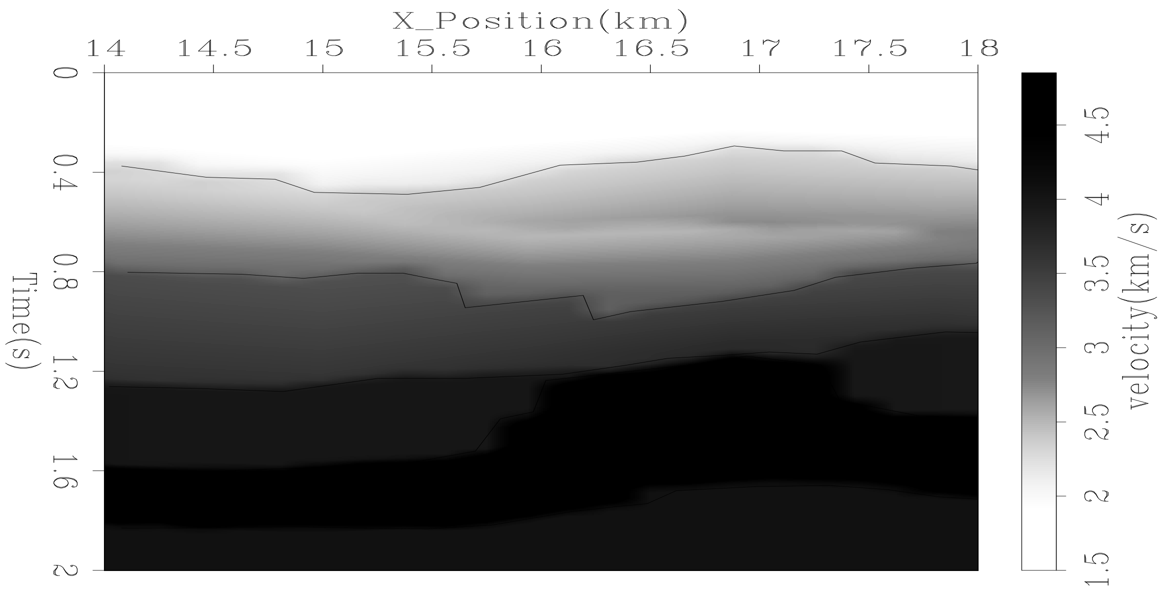


Figure 12: Supplied interval velocity model with picked reflectors superimposed. bob1-reflectors-cor [ER]

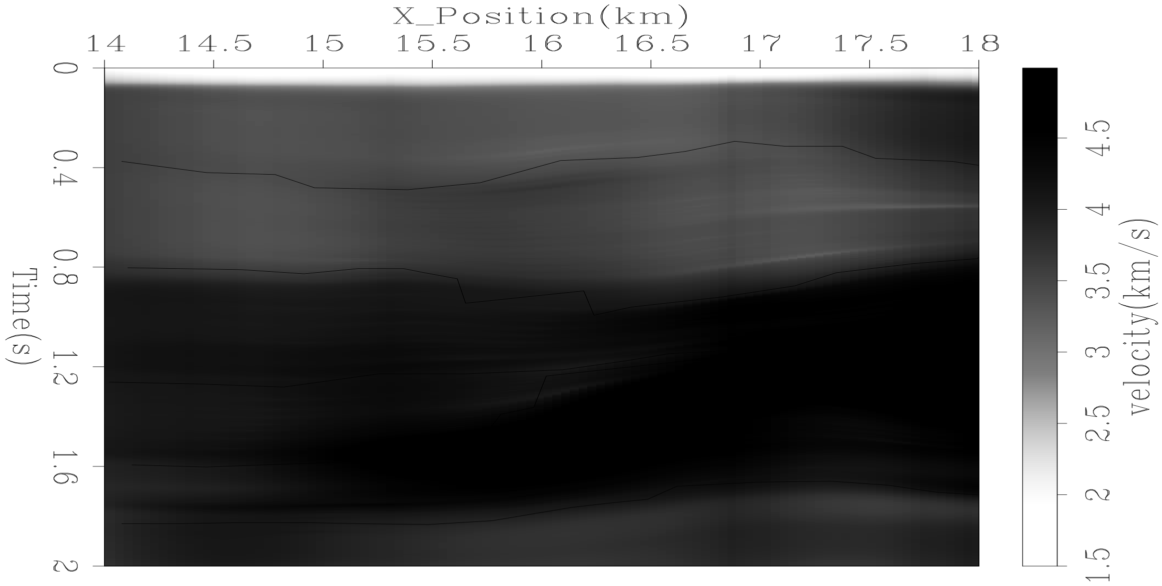


Figure 13: Estimated interval velocity model with picked reflectors superimposed. bob1-reflectors-est [ER]

Clapp, R. G., Fomel, S., and Claerbout, J., 1997, Solution steering with space-variant filters: SEP–**95**, 27–42.

Fomel, S., Clapp, R., and Claerbout, J., 1997, Missing data interpolation by recursive filter preconditioning: SEP–**95**, 15–25.

Nash, S. G., and Sofer, A., 1996, Linear and nonlinear programing: McGraw-Hill.