*Stanford Exploration Project, Report 72, December 18, 1997, pages 1–73*

# Moving source and receiver position by slant stacks

*Martin Karrenbach*[1]

***keywords:*** *ray tracing, slant stack, traveltime, modeling, inversion, finite difference, algorithm, least squares, amplitudes*


## ABSTRACT

Slant stacks are used to move source or receiver positions of seismic data. A least squares forward slant stack incorporates irregular geometry-related effects and diminishes artifacts in the $p$-$\tau$ domain. This step is followed by a transpose stack to return to the $x$-$t$ domain with a different geometry. This very basic procedure can be improved by applying a weighting function in the slant stack domain. The weighting function is derived from a semblance panel computed during the slant stack. This is a nonlinear step and equalizes the importance of coherent events with large amplitude differences.


## INTRODUCTION

Many land data are recorded irregularly or in a pseudo-regular fashion in source and receiver coordinates. On the other hand, many data processing algorithms need regularly sampled data. A source equalization algorithm, which is based on reciprocity, requires that at each source location there is a geophone location. In reality this poses a problem: it is very common for an acquisition geometry to have interlaced source and receiver stations. Certain kinds of interpolation algorithms will handle that problem very efficiently. However, usually the geometry has additional twists to it, like unequal near offset distances. An algorithm which is to be applied to real data has to be able to handle such irregular spacings.

For my application, I chose to use slant stacks of shot gathers for two reasons. Source equalization operates under the assumption of varying source behaviour and quasi regular receiver properties. Under that assumption it is more convenient to reposition receivers. Slant stacks have the property that the data decomposition can be carried out in a least squares sense incorporating irregular geometry; Clement Kostov (1990) describes the linear properties of that operator in detail. The least

[1]**email:** not available

squares approach greatly reduces artifacts introduced by data boundaries. It can be shown that the least-squares slant stack operator easily handles irregular spacing in $x$-$t$.

## SLANT STACK AS GEOMETRY TRANSFORMATIONS

Most people don't see slant stacks as geometry manipulation tool, but rather as a tool for analyzing or filtering seismic data. However, as long as one has a regular sampling in ray parameter, one can have arbitrary irregular sampling in $x$, which makes slant stacks a convenient process for manipulating source and receiver positions. In this paper I will view slant stacks merely as a process for geometry transformation. It turns out that these geometry transforms are closely related to interpolation and extrapolation of data.

Given $(p, \tau)$ data, we know exactly how to transform from $p$-$\tau$ domain into the $x$-$t$ domain. It is just a straight stack or plane wave superposition. The operator $\mathbf{S}$ denotes the stack:

$$\mathbf{d}'(x,t) = \mathbf{S}\ \mathbf{d}(p,\tau). \tag{1}$$

Going from the $x$-$t$ domain to the $p$-$\tau$ domain represents some difficulties, if we are concerned with the finiteness of the domain. However, a least squares operation can be designed, which transforms the data $d(x,t)$ into the $p$-$\tau$ domain

$$\mathbf{d}(p,\tau) = (\mathbf{S^t}\ \mathbf{S})^{-1}\ \mathbf{S^t}\ \mathbf{d}(x,t). \tag{2}$$

In equation 1 and 2 we assumed that operator $\mathbf{S}$ incorporates in both instances the same geometry. This does not have to be necessarily the case. One can deliberately choose different geometries, which is reflected in the different structure of the operators. Taking $\mathbf{d}(p,\tau)$ as a good representation of the data in the $p$-$\tau$ domain, we can now change the geometry. Having a $p$-$\tau$ data estimate, that is free of geometrical effects, is the key for this process. Of course, real $p$-$\tau$ data (noisy) are "geometry free" only in the least-squares sense. We would guess that changing the geometry of the transform $\mathbf{S}$, on the way back, leads us into the null space of the operator. To see if this assertion is true I ran a few simple synthetic tests.

### Geometry transform operator

The geometry changing operation can simply be expressed as a cascade of operators. The operator $\mathbf{S_1}$ refers to the stack operation under a given data geometry 1. It might include knowledge about missing or dead traces. "$t$" denotes the transpose operation. $\mathbf{S_2}$ is the stack with a new arbitrary data geometry.

$$\mathbf{d}'(x,t) = \mathbf{S_2}\ (\mathbf{S_1^t}\ \mathbf{S_1})^{\mathbf{-1}}\ \mathbf{S_1^t}\ \mathbf{d}(x,t) \tag{3}$$

### In terms of known and missing data

Alternatively, changing geometrical bases can be viewed as estimating missing data from known data. Subscripts $_m$ denote missing and $_k$ denote known data or model values. Identifying $\mathbf{S_1}$ with the known portion and $\mathbf{S_2}$ with the unknown portion of the data, we see directly that equation (3) is an estimation of unknown data (at the new geometry).

$$\mathbf{d_k}(x,t) = \mathbf{S_k}\ (\mathbf{S_k^t}\ \mathbf{S_k})^{-\mathbf{1}}\ \mathbf{S_k^t}\ \mathbf{d_k}(\mathbf{x},\mathbf{t}) \tag{4}$$

$$\mathbf{d_m}(x,t) = \mathbf{S_m}\ (\mathbf{S_k^t}\ \mathbf{S_k})^{-\mathbf{1}}\ \mathbf{S_k^t}\ \mathbf{d_k}(\mathbf{x},\mathbf{t}). \tag{5}$$

Given a line in $x$-$t$ with a gap in the middle, it is easy to demonstrate the null space problem. We run into that problem when the domain is not finely enough sampled in $p$, or in other words the model space is underdetermined.

Figure **??** shows a line with a bandlimited spike. The least squares forward transform incorporates the information about the gap into the operator. The slant stack domain shows clearly a single spike for the properly sampled data.

### Arbitrary (irregular) location shifting

One of the first tests was to see how simple shifting of source positions by integral and non-integral amounts of the spatial grid spacing could lead to artifacts on the resulting data $\mathbf{d}'(x,t)$. In Figure **??** we see the input data for different amounts of shifts. The only noticeable artifact is on the lower right picture, which had a shift about one third of the total section width applied. On the right hand side of the linear event, energy is dispersing near the border of the section. Shifting smaller amounts did not creat significant artifacts for this simple example.

### Interlacing

Interlacing a data set is equivalent to shifting to new location by 1/2 a gridpoint. This assumes a regular starting geometry. Figure **??** fills in missing data at every other grid point by actually projecting the model space $p$-$\tau$ onto a data geometry with twice as many traces. Much more common than this data interlacing problem, is the need for filling gaps in the data, due to trace editing or irregular acquisition. Figure **??** showed an example of an acquisition gap and its reconstruction. Figure **??** fills in missing traces by estimating the $p$-$\tau$ domain in a least squares sense for the given geometry and then projecting the model back onto the new geometry.

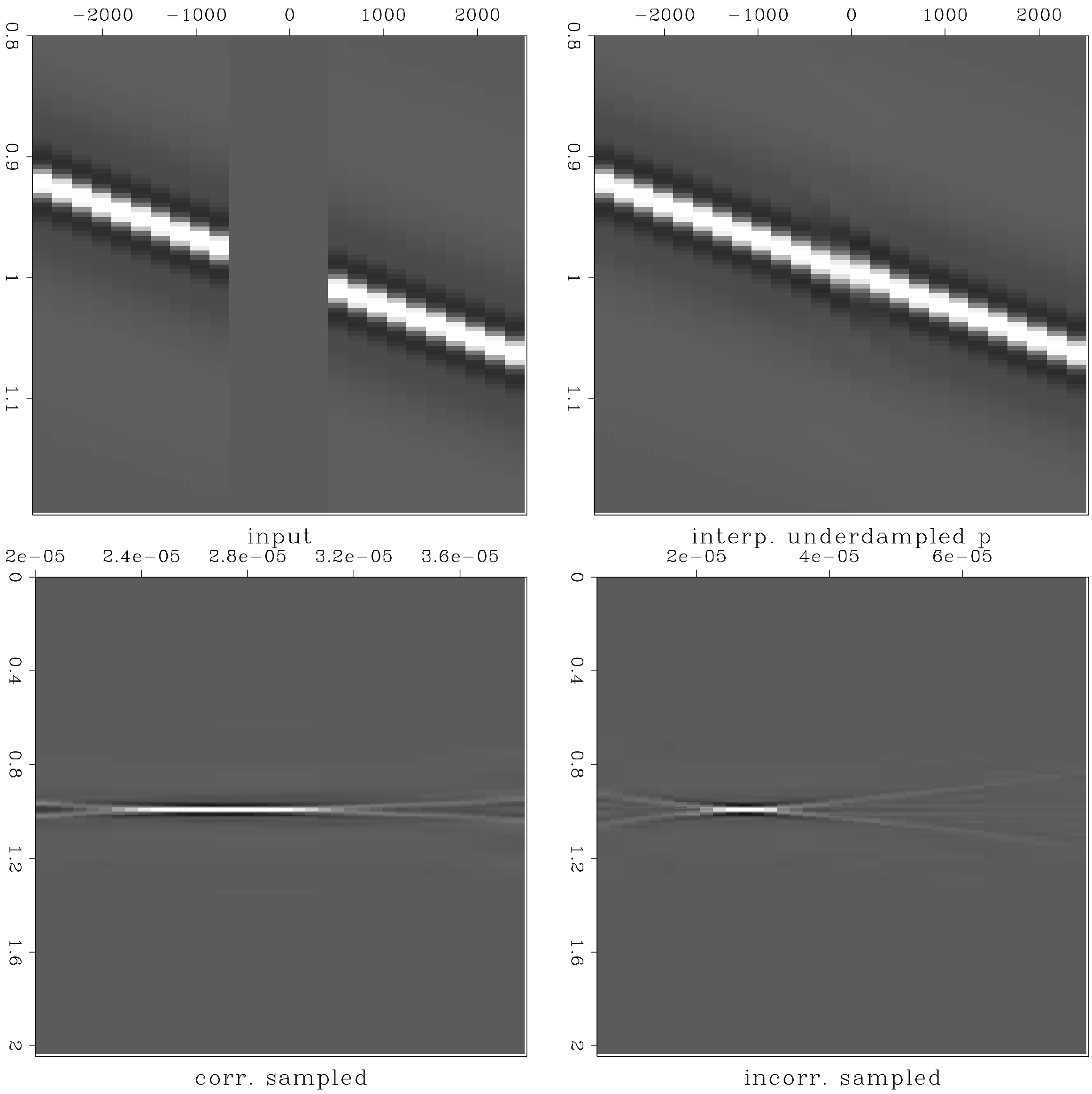


Figure 1: Inadequate sampling in ray parameter leaves problem underdetermined. The least squares approach still recovers original data very well but looks for a minimum length solution in the underdetermined area. It might be hard to see but known amplitudes are reconstructed perfectly, while amplitudes that lie in the gap are smoothly decreasing towards the middle of the gap. A perfect reconstruction would place constant known amplitudes into the gap. martin1-inadequate [ER]

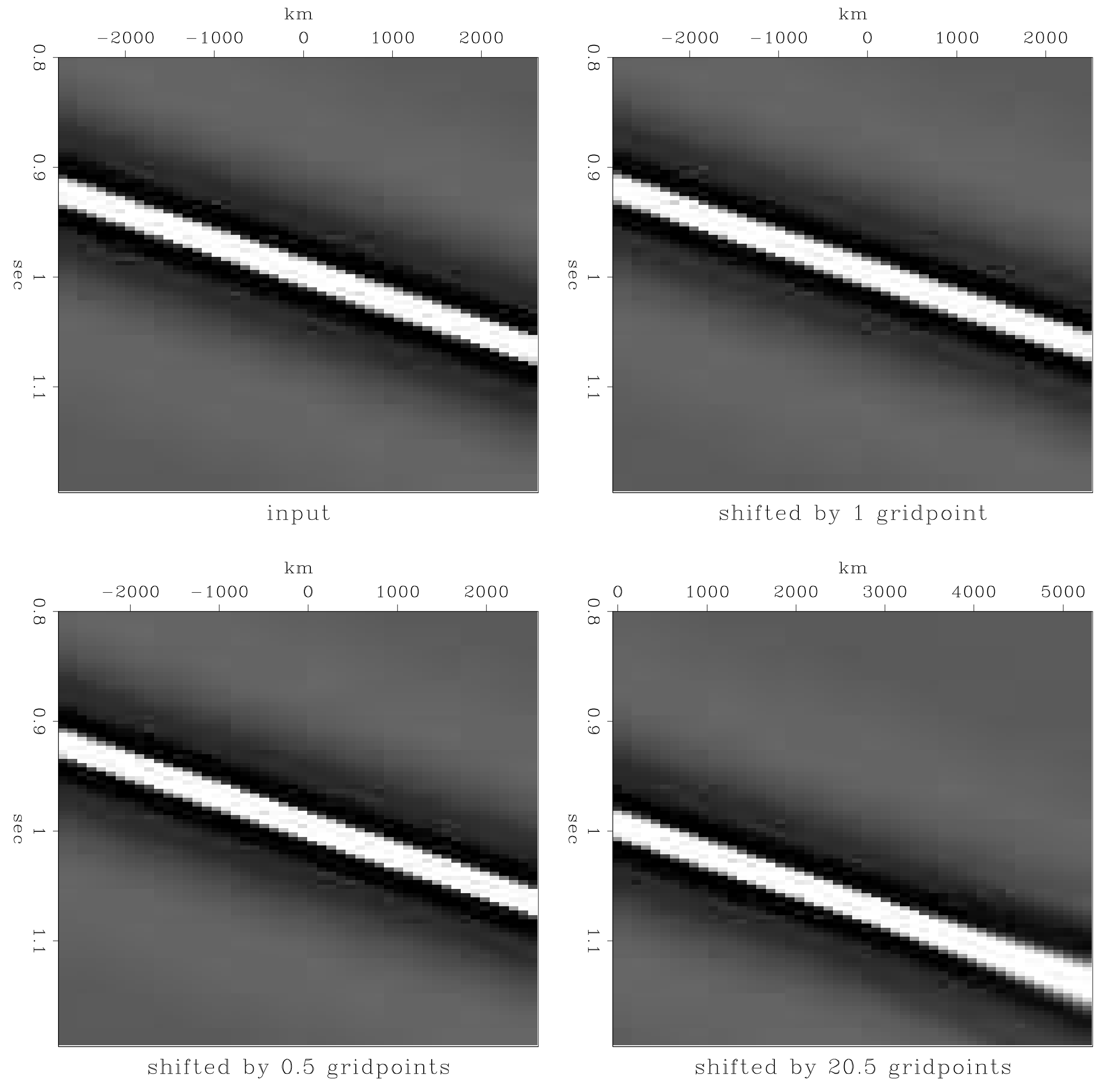


Figure 2: Linear event. Shifted sections by various numbers of gridpoints. Smallest shift 1/2 trace spacing, largest shift is 20.5 trace spacings. martin1-shift [ER]

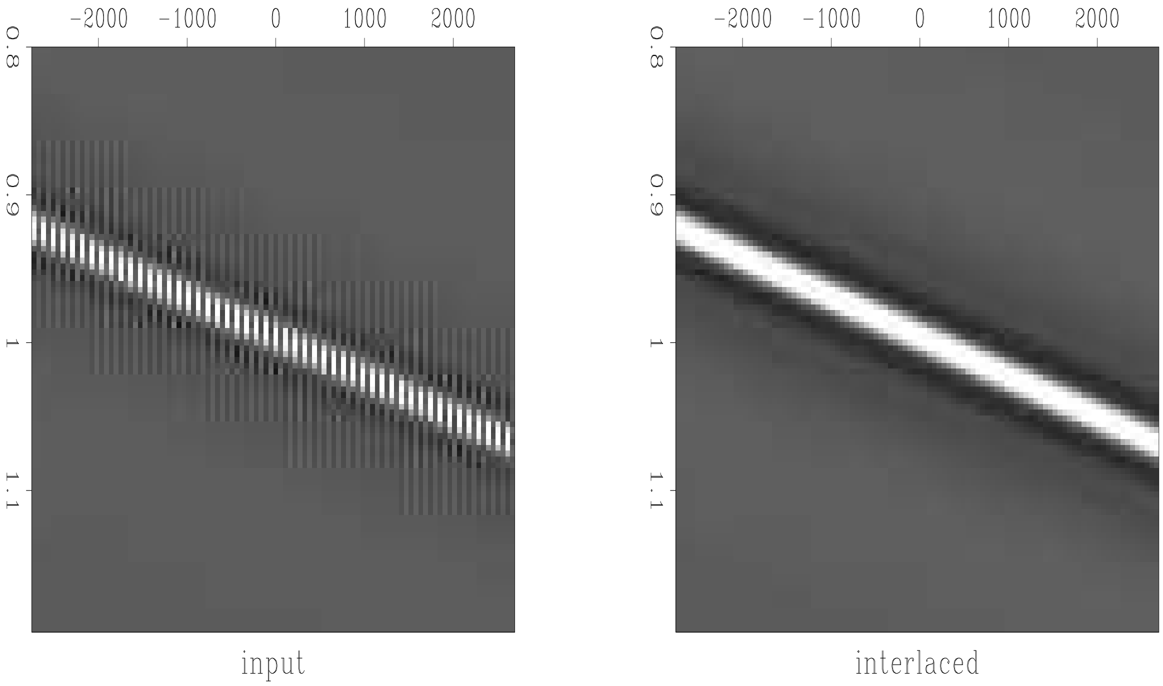


Figure 3: The $p$-$\tau$ domain is estimated in a least squares sense for the given geometry. The best fitting model is projected back onto the new geometry (missing trace construction). martin1-interlace [ER]

## SEMBLANCE WEIGHTING

All the examples I showed above use linear events just to make the concept clear. In reality there are large amplitude variations across a section. Effects of these large amplitudes can be equalized by using some coherence measure and deriving a weighting function to be applied in the slant stack domain.

Taner et al. (1971) define semblance in general terms. I rewrite their definition in terms of slant stacks:

$$\mathbf{semb}(p,\tau) = \frac{< (\mathbf{S^t}\ \mathbf{d}(x,t))^2 >_w}{N < \mathbf{S^t}\ (\mathbf{d}(x,t))^2 >_w}, \tag{6}$$

where $N$ is the number of "live " traces used in the slant stack and $w$ is a smoothing window. $< ... >$ denotes simple averaging in that window. The weighting function I applied to the $p$-$\tau$ domain is a simple step function of the semblance coefficient, killing everything below a certain threshold (let's say 10% of the maximum semblance coefficient). Thus, weak, but spatially coherent events still have a good chance of being recovered, while large but incoherent events are suppressed. The data estimation is now nonlinear (since semb is a nonlinear function of the data) and is described by

$$\mathbf{d}'(x,t) = \mathbf{S_2}\ \mathbf{semb_1}\ (\mathbf{S_1^t}\ \mathbf{S_1})^{-\mathbf{1}}\ \mathbf{S_1^t}\ \mathbf{d}(\mathbf{x},\mathbf{t}). \tag{7}$$

I tried several different weighting functions, such as weighting by powers of energy, but a semblance based weighting was most versatile and effective in my tests.

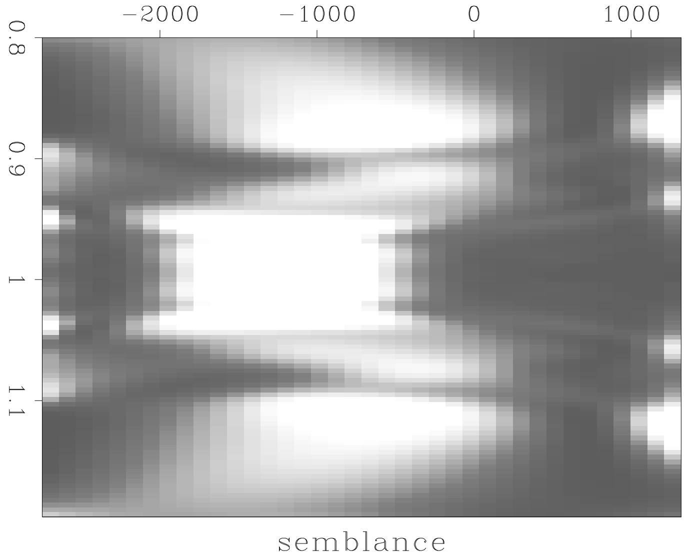


Figure 4: Semblance computed from the stack of the data squared and the squared stack of the data. martin1-semblance [ER]

## CONCLUSIONS

Slant stacks are a convenient way to shift source and receiver coordinates. Taking advantage of the least squares forward stack and the straight transpose stack properties, one can describe those two processes as first finding the best available data model in $p$-$\tau$ space under a given geometry, then projecting that best fitting model into a given geometrically constrained data space.

## ACKNOWLEDGMENTS

Thanks to Dave Nichols for suggesting the idea and for implementing the operator with "dead trace detection".

## REFERENCES

Kostov, C., 1989, Finite aperture slant stack transforms: SEP–**61**, 261–280.

Neidell, N.S., and Taner, T., 1971, Semblance and other coherency measures for multichannel data: Geophysics, **36**, 482–497.

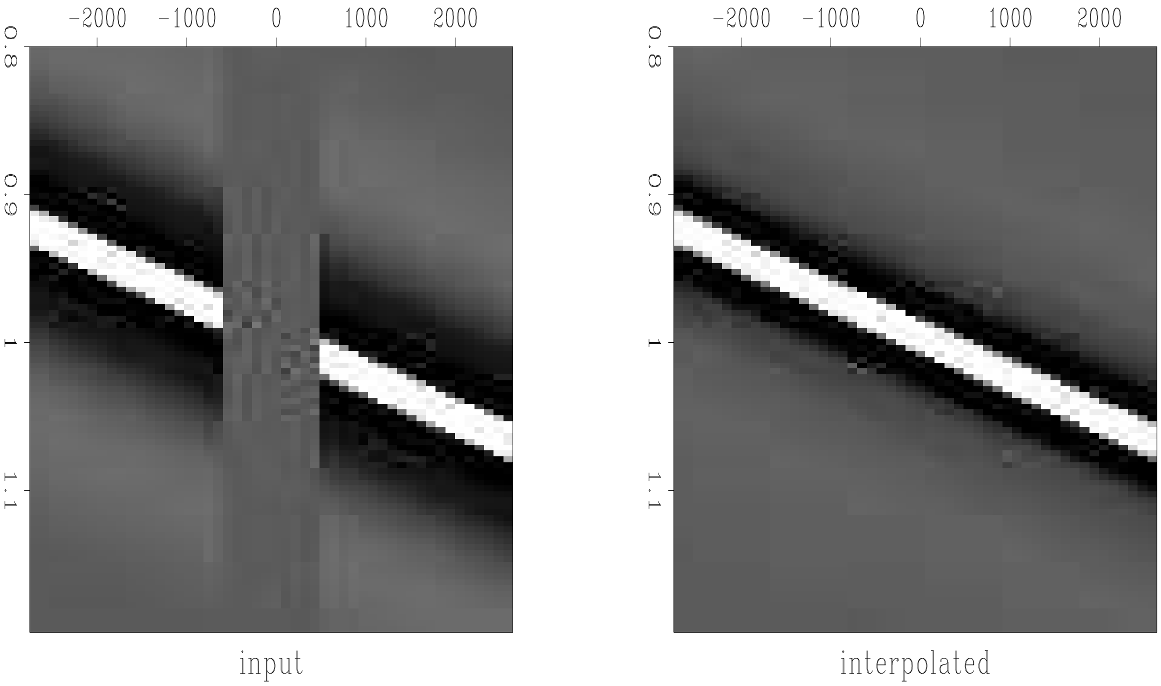


Figure 5: Interpolate dead traces and shift using a semblance weighted least squares stack. martin1-interpolw [ER]