Stanford Exploration Project, Report 79, November 16, 1997, pages 1–169

Stacking fundamentals

Jon F. Claerbout[1]

keywords: *modeling, amplitudes, velocity, approximation*

ABSTRACT

Velocity increasing with depth causes moveout curves to cross. Here we begin from the modeling operator and its adjoint, the stacking operator. Weighting functions can be used both before and after NMO. We find that weighting functions that make the operator quasi-unitary also sharpen the stack.

INTRODUCTION

The most important linear operator in exploration seismology is normal moveout and stack. Here we review this combined operator and its adjoint modeling. We particularly investigate velocity increasing with depth making moveout curves cross. This problem is traditionally solved by "muting", an ad hoc process of throwing away information at wide offset. To better understand crossing events and the stacking operator, I set up some synthetic data. Much to my surprise, I found that preconditioning transformations (scaling before and after transformation) give a theoretical basis for a process resembling muting. The preconditioning transformation differs from muting in that data is gained unexpectedly downward with time before stack and then upward after stack.

MOVEOUT AND STACK

We begin[2] from familiar normal moveout equations

$$v^2\, t^2 \;=\; z^2 \;+\; x^2 \tag{1}$$

$$t^2 \;=\; \tau^2 \;+\; \frac{x^2}{v^2} \tag{2}$$

[1]**email:** not available

[2]This material first appeared at the May 1993 SEP meeting at Fallen Leaf Lake in a preview of my textbook in preparation, "Basic Earth Imaging" page 44-48.

where τ is traveltime depth. A program for nearest-neighbor normal moveout as defined by equation (2) is `nmo0()` on page 2. Because of the limited alphabet of programming languages, I used the keystroke `z` to denote τ.

```
subroutine nmo0( adj, add, slow,    x, t0, dt, n,zz,  tt )
integer  it, iz, adj, add,                          n
real  xs, t , z,           slow(n), x, t0, dt, zz(n), tt(n)
call adjnull(    adj, add,                       zz,n,  tt,n)
do iz= 1, n {   z = t0 + dt*(iz-1)                # Travel-time depth
                xs= x * slow(iz)
                t = sqrt ( z * z + xs * xs)
                it= 1 + .5 + (t - t0) / dt        # Round to nearest neighbor.
                if( it <= n )
                        if( adj == 0 )
                                tt(it) = tt(it) + zz(iz)
                        else
                                zz(iz) = zz(iz) + tt(it)
        }
return; end
```

```
subroutine adjnull( adj, add, x, nx,  y, ny )
integer ix, iy,     adj, add,    nx,     ny
real                          x( nx), y( ny )
if( add == 0 )
        if( adj == 0 )
                do iy= 1, ny
                        y(iy) = 0.
        else
                do ix= 1, nx
                        x(ix) = 0.
return; end
```

The adjoint operation begins from a model like the zero-offset trace and sprays this trace to all offsets. The moveout and stack operations $\mathbf{S}'$ are done by subroutine `stack0()` with the `adj=1` argument and the modeling operation $\mathbf{S}$ is done by subroutine `stack0()` with the `adj=0` argument.

```
subroutine stack0( adj, add, slow,     t0,dt, x0,dx, nt,nx, stack, gather)
integer ix,        adj, add,                         nt,nx
real     x,                 slow(nt), t0,dt, x0,dx,   stack(nt), gather(nt,nx)
call adjnull(      adj, add,                          stack,nt,  gather,nt*nx)
do ix= 1, nx {
        x = x0 + dx * (ix-1)
        call nmo0( adj, 1, slow, x, t0,dt, nt, stack, gather(1,ix))
        }
return; end
```

Figure 1 illustrates the modeling operator and its adjoint. Notice the roughness on the waveforms caused by different numbers of points landing in one place. Notice also the increase of AVO (amplitude versus offset) as the waveform gets compressed into

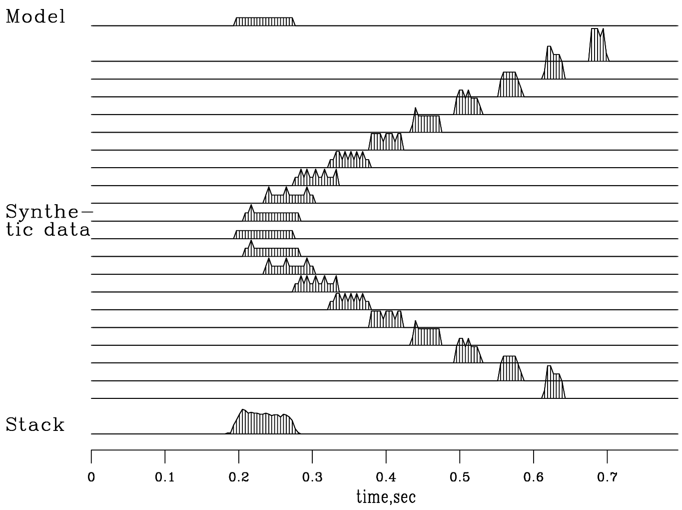

Figure 1: Top is a model trace $\mathbf{m}$. Center shows the spraying to synthetic traces, $\mathbf{S'm}$. Bottom is the stack of the synthetic data, $\mathbf{SS'm}$. jon1-stack [ER]

a smaller space. Finally, notice that the stack is a little rough, but the energy is all in the desired time window.

Parenthetically, we notice a contradiction of aspirations. On the one hand, an operator has smooth outputs if it "loops over output space" and finds its input where ever it may. On the other hand, it is nice to have modeling and processing be exact adjoints of each other. Unfortunately, we cannot have both. If you loop over the output space of an operator, then the adjoint operator has a loop over input space and a consequent roughness of its output.

CROSSING TRAVELTIME CURVES

Since velocity increases with depth, at wide enough offset a deep enough path will arrive sooner than a shallow path. In other words, traveltime curves for shallow events must cut across the curves of deeper events. Where traveltime curves cross, NMO is not a one-to-one transformation. To see what happens to the stacking process I prepared Figures 2-4 using a typical marine recording geometry (although for clarity I used larger $(\Delta t, \Delta x)$) and we will use a typical Texas gulf coast average velocity, $v(z) = 1.5 + \alpha z$ where $\alpha = .5$.

First we repeat the calculation of Figure 1 with constant velocity $\alpha = 0$ and more reflectors. We see in Figure 2 that the stack reconstructs the model except for two details: (1) the amplitude diminishes with time, and (2) the early waveforms have become rounded.

Then we repeat the calculation with the Gulf coast typical velocity gradient $\alpha = 1/2$. The polarity reversal on the first arrival of the wide offset trace in Figure 3 is evidence that in practice traveltime curves do cross. Comparing Figure 2 to Figure 3 we see that an effect of the velocity gradient is to degrade the stack's reconstruction of the model. The velocity gradient has ruined the waveform on the shallowest event,

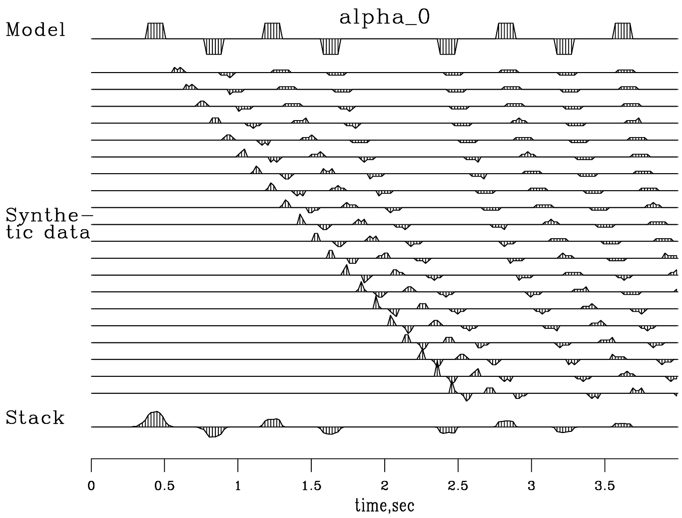

Figure 2: Synthetic CMP gather for constant velocity earth and reconstruction. jon1-nmo0alfa0 [ER]

at about 400ms. If the plot were made on a finer mesh with higher frequencies, we could expect ruined waveforms a little deeper too.

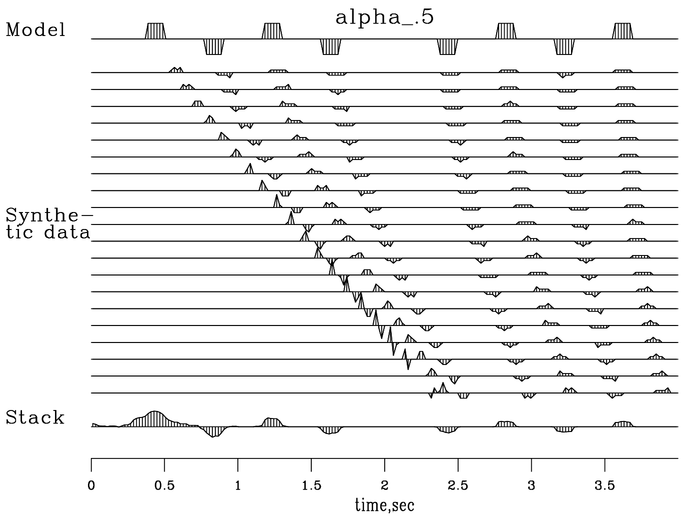

Figure 3: Synthetic CMP gather for velocity linearly increasing with depth (typical of Gulf of Mexico) and reconstruction. Notice that the `Stack` signal is not as sharp as in Figure 2. jon1-nmo0alfa.5 [ER]

In designing NMO and stack subroutines we gave no thought to signal amplitudes (although results showed an interesting AVO effect in Figure 1.) We could redesign the programs so that the modeling operator has the most realistic amplitude that we can devise. Alternately, we could design the amplitudes to get the best approximation to $\mathbf{S}'\mathbf{S} \approx \mathbf{I}$ which should result in the "`Stack`" signal being a good approximation to the "`Model`" signal. I experimented with various weighting functions until I came up with subroutine `nmo1()` on page 5 which embodies the weighting function $(\tau/t)(1/\sqrt{t})$ and which produces the result in Figure 4. This result is very pleasing. Not only is the amplitude as a function of time better preserved, more importantly, the shallow wavelets are less smeared and have recovered their rectangular shape. The reason the reconstruction is much better is the cosine weighting implicit in τ/t. It has muted away much of the energy in the shallow asymptote. I think this energy near the asymptote is harmful because the waveform stretch is so large there. Perhaps a

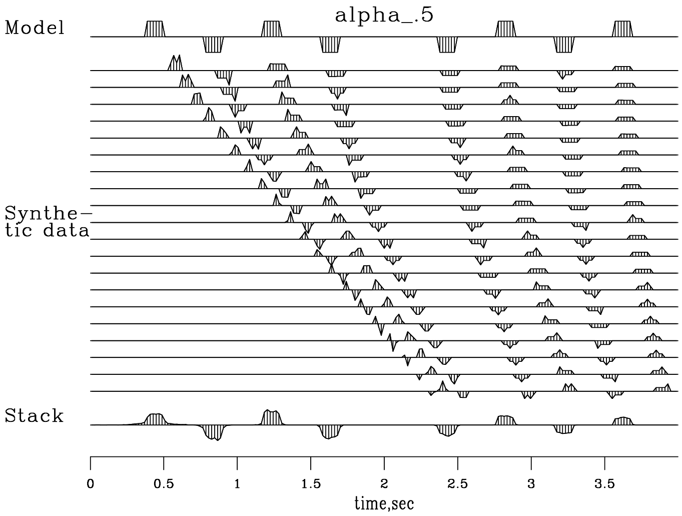

Figure 4: Synthetic CMP gather for velocity linearly increasing with depth and reconstruction with weighting functions in subroutine `nmo1()`. Notice that the `Stack` signal is much sharper than in Figure 3. (Should make this figure interactive). jon1-nmo1alfa.5 [ER]

similar good result could be found by experimenting with muting programs. However, subroutine `nmo1()` on page 5 differs from muting in two significant respects: (1) `nmo1()` is based on a theoretical concept whereas muting requires observational parameters and (2) muting applies a weighting in the coordinates of the (t, x) input space, while `nmo1()` does that but also includes the coordinate τ of the the output space. With `nmo1()` events from different τ depths see different mutes which is good where a shallow event asympote crosses a deeper event far from its own asymptote.

```
subroutine nmo1( adj, add, slow,    x, t0, dt, n,zz,  tt )
integer  it, iz, adj, add,                        n
real  xs, t , z,          slow(n), x, t0, dt, zz(n), tt(n), wt
call adjnull(    adj, add,                    zz,n,  tt,n)
do iz= 1, n {   z = t0 + dt*(iz-1)
                xs = x * slow(iz)
                t = sqrt ( z * z + xs * xs) + 1.e-20
                wt = z/t * (1./sqrt(t))               #  weighting function
                it = 1 + .5 + (t - t0) / dt
                if( it <= n )
                        if( adj == 0 )
                                tt(it) = tt(it) + zz(iz) * wt
                        else
                                zz(iz) = zz(iz) + tt(it) * wt
           }
return; end
```

The most physically correct amplitudes are not likely to be those for which $\mathbf{S}'\mathbf{S} \approx \mathbf{I}$. Physically accurate amplitudes involve many theoretical and experimental issues not covered here. It is easy to include some effects (spherical divergence based on velocity depth variation) and harder to include others (surface ghosts and arrays). I will not set up the "inverse problem" here. It has many complicating pitfalls that you are welcome to work through.

Ideal weighting functions for stacking

The difference between stacking as defined by `nmo0()` on page 2 and by `nmo1()` on page 5 is in the weighting function $(\tau/t)(1/\sqrt{t})$. Notice that $(\tau/t)(1/\sqrt{t})$ can be factored into two weights, τ and $t^{-3/2}$. One weight could be applied before NMO and the other after. That would also be more efficient than weighting inside NMO, as does `nmo1()`. Additionally, it is likely that these weighting functions should take into account data truncation at the cable's end. Stacking is the most important operator in seismology. Perhaps some objective measure of quality can be defined and arbitrary powers of t, x, and τ can be adjusted until the optimum stack is defined. Likewise, we should consider weighting functions in the spectral domain. As the weights τ and $t^{-3/2}$ tend to cancel one another, perhaps we should filter with opposing filters before and after NMO and stack.

The preconditioners used here are related to wave theory in a constant velocity earth. We can expect improvement if we redesign them for a stratified earth. I expect the familiar velocity dependent divergence correction would appear.

CONCLUSION

Industrial standard muting accomplishes many goals and I do not claim the simple theory here forms an analytic basis for it all. Nevertheless, the concept that moveout and stack should be pre- and post- multiplied by weighting functions to make the composite operation quasi-unitary, provides a theoretical basis for what would otherwise be an entirely ad hoc process. Thus, quasi-unitary transformation (a form of operator preconditioning) is a mute-like process that should precede the human intervention of subsequent muting.

The example studied here of Texas gulf coast velocities is a mild form of the problem of traveltime cross overs, and this study needs elaboration for the much stronger velocity contrasts found in many other locations.

A FABLE OF PROCESSING VERSUS INVERSION

[3]Everybody knows about getting travel times correct. Let us define a data space $\mathbf{d}$, a model space $\mathbf{m}$, and an operator $\mathbf{K}$ called "Kinematics" that gets the travel times correct. The conjugate-transpose $\mathbf{K}'$ gives processing operators like "moveout and stack", and like the sequence "normal moveout, dip moveout, stack, zero-offset migration".

Different people have different ideas about amplitudes. We will have four different amplitudes to deal with here. Think of them as diagonal matrices $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$, and $\mathbf{D}$ that we all know how to invert.

[3]This section first appeared as a quiz.

Professor Fussy Logic, a deep thinker from The Continent, is not satisfied with kinematics because he likes to get amplitudes *correct.* This gives him lots of weighting functions. Some of his weights are in model space and some in data space. Some of his weights are best represented in Fourier space, such as $\sqrt{-i\omega}$. We cannot fully decipher all his thoughts about amplitudes but to some (unknown) level of approximation we can think about his work as

$$\mathbf{AKBm} \quad = \quad \mathbf{d} \tag{3}$$

Professor Fussy Logic's view of seismic data processing is that $(\mathbf{AKB})$ is a linear operator that his students can pass to the conjugate-gradient solver which must provide *the correct answer.* Since the data space and model space have about a million pixels, the CG solver needs about a million iterative applications of the seismic operator $\mathbf{AKB}$ and its adjoint. We hope his students can get good results after a dozen CG iterations (which is computationally feasible) but we are fearful that Professor Fussy Logic's modeling might not include all the messy aspects of real data and that any near null space could lead to disaster.

Dr. Mucho Moola, is founder-owner of a successful geophysical contracting company. (It would be impolitic of me to reveal his exact identity but you can think of him as an MIT PhD of the '60s or an SEP PhD of the '70s.) Dr. Mucho Moola likes to get amplitudes correct too, but he could tell you a dozen reasons why that has nothing to do with a correct modeling theory. Starting from his notion that "all data is important" and from the kinematic operator $\mathbf{K}$, by various schemes that are "company confidential" he finds weighting functions $\mathbf{C}$ and $\mathbf{D}$ such that

$$(\mathbf{CKD})'(\mathbf{CKD}) \quad \approx \quad \mathbf{I} \tag{4}$$

In competitive bidding situations, he offers oil companies the simple process

$$\hat{\mathbf{m}} \quad = \quad (\mathbf{CKD})'\mathbf{d} \tag{5}$$

For very important surveys he has a special secret nonlinear process $\mathbf{Q}$ that is really just a combination of AGC and decon which he uses to supplement equation (5). For lots of extra bucks, oil companies can purchase his custom result

$$\hat{\hat{\mathbf{m}}} \quad = \quad Q((\mathbf{CKD})'Q(\mathbf{d})) \tag{6}$$

Ms. Sepper, a new student at SEP, wants to reconcile logic with success. She figures that Professor Fussy Logic has the problem formulated right but his solution is impractical whereas Dr. Mucho Moola is successful (despite ignoring the physical theory) because he has a good approximation to the inverse of a matrix with the same kinematics as Professor Fussy Logic's matrix. She sets out to apply Dr. Mucho Moola's inverse method to Professor Fussy Logic's problem. Starting with equation (3), she premultiplies by $\mathbf{CA}^{-1}$ and she also inserts $\mathbf{I} = \mathbf{DD}^{-1}$ in a strategic location getting

$$(\mathbf{CKD})\mathbf{D}^{-1}\mathbf{Bm} \quad = \quad \mathbf{CA}^{-1}\mathbf{d} \tag{7}$$

Then she premultiplies by $\mathbf{B}^{-1}\mathbf{D}(\mathbf{CKD})'$, and invokes Dr. Mucho Moola's approximation (4) getting

$$\mathbf{m}_{\text{sep}} \quad = \quad \mathbf{B}^{-1}\mathbf{D}\mathbf{D}'\mathbf{K}'\mathbf{C}'\mathbf{C}\mathbf{A}^{-1}\mathbf{d} \tag{8}$$

Recognizing the importance of her result, Ms. Sepper plans to start a new geophysical contracting company specializing in the high end of the market.

1. According to Ms. Sepper's formula, whose weights are more important, Professor Fussy Logic's **A** and **B** or Dr. Mucho Moola's **C** and **D**?
2. In view of competition from Dr. Mucho Moola's secret Q process, how should Ms. Sepper advertise her product?
3. How much would you like to invest in the new company? Why?

Epilogue

A couple years later, while revising PVI which had gone out of print, Professor Claerbout notices that equation (3) is an oversimplification. Actually, all the quantities in it should be partitioned matrices. He tries in vain to convince the SEP students to update Dr. Mucho Moola's approximation (4) and apply it routinely using C++. Professor Wily Dymes at Rice University in Houston figures this out independently, however, and as a consequence SEP funding declines rapidly. Claerbout retires to putter with earthquake seismograms and all the SEP students have to move to Houston to complete their degrees.