Image gather reconstruction using StOMP

Robert G. Clapp

ABSTRACT

Constructing 3-D angle gathers through cross-correlation poses a computational problem primarily due to the accompanying increase in volume size which forces the gathers to be stored in a computationally more expensive memory level. Compressive sensing can be used to mitigate this challenge. The correlation volume size can be reduced by both phase encoding and random subsampling. The full correlation gathers can then by reconstructed using an l_1 inversion scheme known as Stagewise Orthogonal Matching Pursuit. Preliminary results indicate that almost all angle gather information can be recovered.

INTRODUCTION

Reverse time migration (RTM) is quickly becoming the standard high-end seismic imaging technique. Significant work has been done on speeding up the kernel (Micikevicius, 2009; Nguyen et al., 2010; Nemeth et al., 2008; Clapp et al., 2010) but far less on constructing image gathers(Sava and Fomel, 2003, 2006) needed for rock property analysis or velocity updates. Image gather construction is a much less tractable problem because it is memory rather than compute intensive. The volume size of the domain increases by one to three orders of magnitude making the dominant cost reading/writing to distant memories (from main memory rather than a cache, across the PCI Bus, or from disk).

Donoho (2006) offers an approach termed *compressive sensing* potential solution to this computation and storage problem. In compressive sensing, a random subset of the desired measurements is made. An inversion problem is then set up to estimate in an ℓ_1, or preferably ℓ_0, sense, a sparse basis function that fully characterizes the desired signal. For compressive sensing to work, a signal must be highly compressible. For compressive sensing to be worthwhile, the cost of inverting for the basis function must be significantly less than the cost of acquiring the full signal. Clapp (2011) showed that correlation gather construction fit the first criteria for a successful compressive sensing problem. Multi-dimensional correlation gathers/angle gathers are compressible at nearly a 100:1 ratio. The challenge became finding an inversion scheme that could accurately enough recover the full model. Donoho et al. (2006) proposed a solution to the second problem, an ℓ_1 version methodology that works for a large number of unknowns. In this paper, I apply the Stagewise Orthogonal Matching Pursuit (StOMP) algorithm to correlation gather reconstruction. I show

that the angle domain representation of the sparsely acquired gathers is similar to the representation of the full data. I then apply a phase encoding technique, combining many different correlations to every data point to further improve the inverted model.

IMAGE GATHERS AND WAVELET COMPRESSION

In RTM downward continuation based wave-equation imaging we have a source $s(x, y, z, it, is)$ and receiver wavefield $g(x, y, it, is)$, where it is a given time and is is a given shot. We form our migrated image $i(x, y, z)$ as

$$I(x, y, z) = \sum_{is=0}^{ns} \sum_{it=0} nts(x, y, z, it, is) g(x, y, it, is). \tag{1}$$

The imaging condition for RTM just replaces the sum over time with a sum over frequency. This basic imaging condition hides our information about velocity and rock properties by stacking over all angles illuminated by our source/receiver geometry.

Several solutions to this problem have been proposed that attempt to extract information as a function of angle from the data. In this paper I focus on the family of methods that construct shift gathers by cross-correlation. The shift can be a function of space (Sava and Fomel, 2003) or time (Sava and Fomel, 2006), or both. In these cases the imaging condition takes the form of

$$I(x, y, z, h_x, h_y, h_z) = \sum_{\text{shot}} \sum_{t} s(x+h_x, y+h_y, z+h_z, it) g(x-h_x, y-h_y, z-h_z, it), \tag{2}$$

where h_x, h_y, h_z are how much the source and receiver wavefields have been shifted in a given direction. Introducing these shift gathers poses two problems. The first, and smaller of the problems, is that computational expense associated with constructing these gathers, when going beyond a single shift axis, is at least on the same order of magnitude as the propagation kernel. The larger problem is the expansion of the volume size anywhere from 20–1000 fold, means that the shift gathers must be stored a memory level further away from the processor, often on disk, which is often several orders of magnitude more expensive to access. Finding some way to reduce the volume size, and better still, the computational expense associated with shift gather construction, can be highly beneficial. In order for compressive sensing to be an appropriate way to reduce the volume size it is important to consider the compressibility of seismic data.

There is significant literature on compressing seismic data. Relatively low compression ratios are achievable by compressing a trace. Significantly higher compression ratios are achieved by multi-dimensional approaches. Generally, the best results have used either multi-dimensional wavelets (Mallat, 1999) or its successor curvelets (Cands and Donoho, 1999). Villasenor et al. (1996) showed that compression ratios of 100:1 were achievable by compressing a 4-D volume (t, h_x, h_y, s). Further, Villasenor

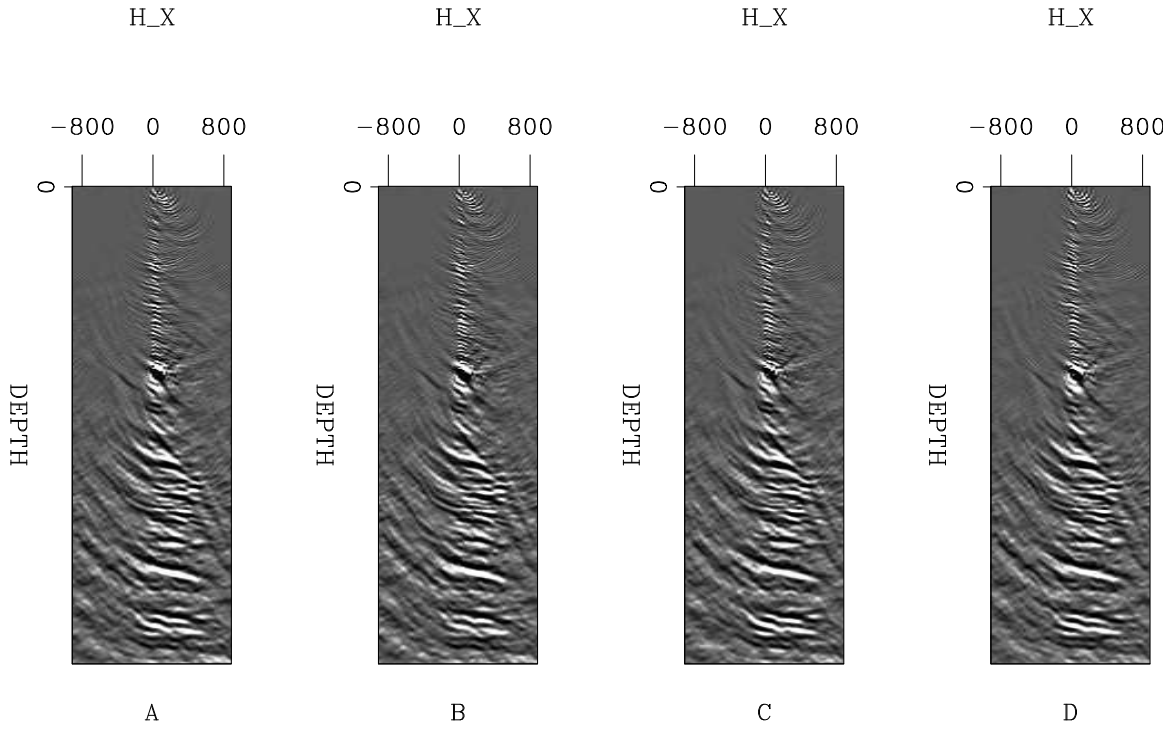


Figure 1: Five neighboring subsurface offset gathers. B and C are one midpoint in X before and after A. E is one midpoint in Y before A. Note the spatial similarity, which lends itself to compression. [**ER**]

et al. (1996) states that the header information was the limiting factor in achieving even higher ratios.

Subsurface offset gathers potentially represent even higher, up to six, dimensional data. To test compressibility, I used a 4-D volume (x, y, z, h_x) of dimensions (32,32,400,64). Figure 1 shows one of these subsurface offset gathers and its neighbors, note the similarlity. Following Villasenor et al. (1996), I chose the 9/7 bi-orthonormal transform (Antonini et al., 1992) used in JPEG compression. Figure 2 shows the resulting transform space and a histogram of the absolute values. I then used several different thresholds throwing away 90%, 95%, 98%, and 99% of the data in the wavelet domain respectively. Figure 3 shows the result of transforming these thresholded volumes back into the space-domain. The resulting images are near-perfect at 95% and potentially acceptable at 98%. This translates into an acceptable compression ratio of approximately 30:1.

COMPRESSIVE SENSING

Compressive sensing is a statistical technique attributed to Donoho (2006), but whose start could be placed as early as the basic pursuit work of Mallat and Zhang (1993). A compressive sensing problem at its heart is a special case of a missing data problem.

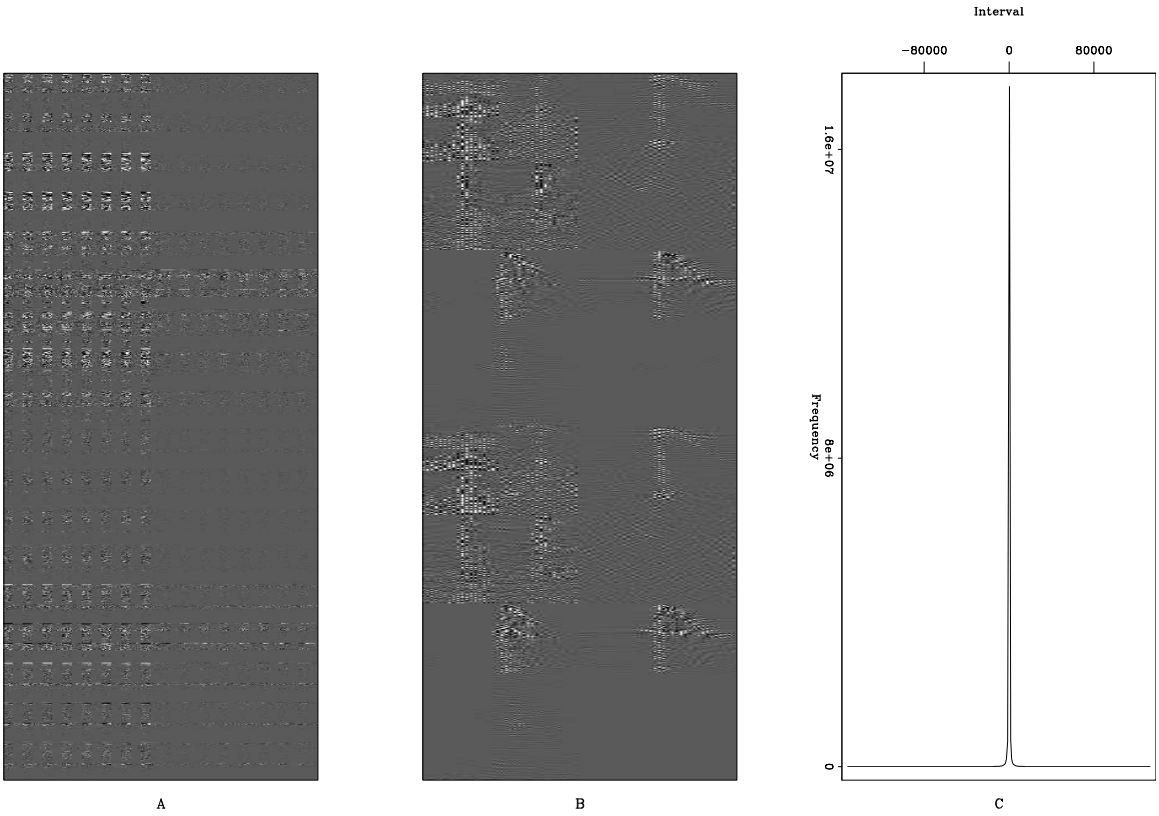


Figure 2: Panel A shows the wavelet domain representation of the 4-D volume used in this experiment. Panel B shows a zoom into a portion of the wavelet domain. Panel C shows a histogram of the wavelet domain values. Note how the vast majority of the values are nearly zero. [**ER**]

In geophysics, we often think of a missing data problem as solving for a model $\mathbf{m}$ given some data $\mathbf{d}$ which exist in the same vector space. We have a masking operator $\mathbf{R}$ (1 where the data is known, 0 elsewhere). We add in some knowledge of the covariance of the model through a regularization operator $\mathbf{A}$. We then estimate the best model from the following system of equations in a ℓ_2 sense,

$$\begin{aligned} \mathbf{0} \approx \mathbf{r_d} &= \ell_\mathbf{2}(\mathbf{d} - \mathbf{Rm}) \\ \mathbf{0} \approx \mathbf{r_m} &= \ell_\mathbf{2}(\mathbf{Am}), \end{aligned} \tag{3}$$

where $\mathbf{r_d}$ and $\mathbf{r_m}$ are the result of taking the ℓ_2 norm of the first and second equations. The success of this approach relies on the accuracy of $\mathbf{A}$ to describe the covariance of the model.

Compressive sensing approaches the problem from a different perspective. It starts from the notion that there exists a basis function that $\mathbf{d}$ can be transformed into through the linear operator $\mathbf{L^T}$ in which very few non-zero elements are needed to represent the signal. The compressive sensing approach is then to set up the missing data problem in two phases. First, estimate the elements of the sparse basis function

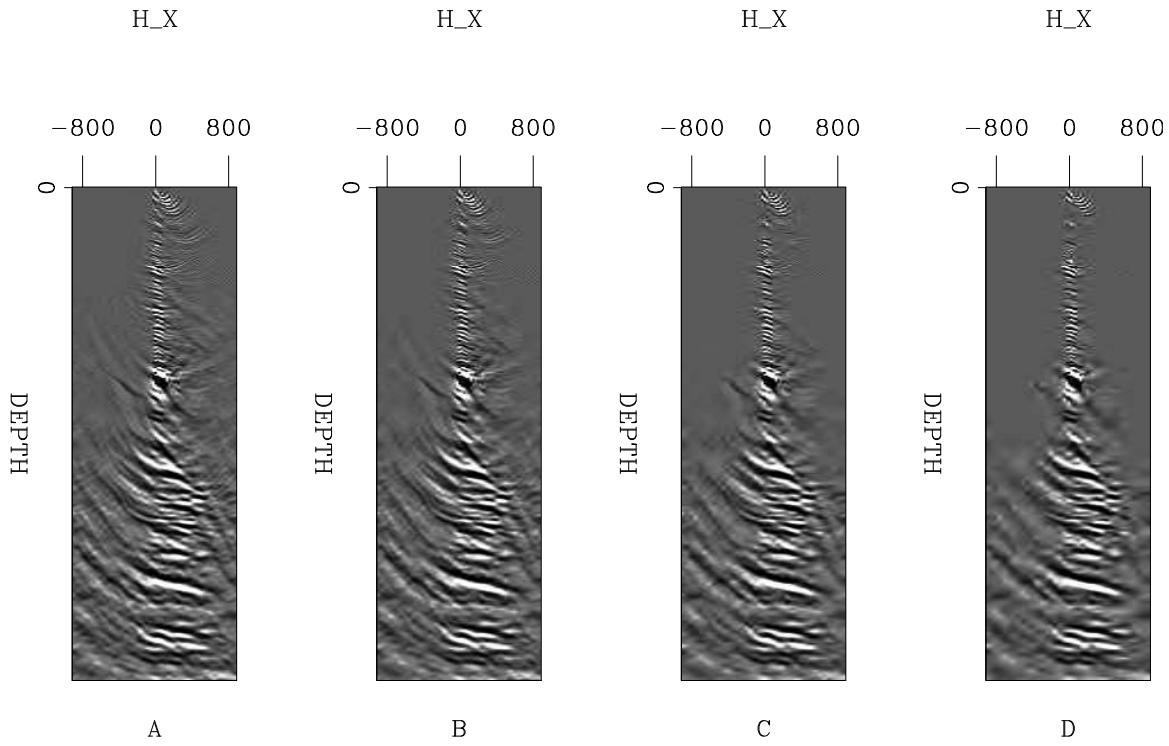


Figure 3: The result of zeroing the smallest values of the wavelet domain representation shown in Figure 2A. All four panels show the same subsurface offset gather shown in Figure 1. A shows the result of clipping 90% of the values; B, 95%; C, 98%; and D,99%. Note how the reconstructed gather is nearly identical up to a 98% clip. [**ER**]

m through,

$$\mathbf{0} \approx \mathbf{r} = \ell_{\mathbf{1}}(\mathbf{d} - \mathbf{RLm}), \tag{4}$$

where we are now estimating **m** in the $\ell_{\mathbf{1}}$ sense. We can then apply **L** to recover the full model. The magic of compressive sensing is that you only need to collect a small multiple, typically 4-5, more data points than the number of non-zero basis elements. In the case of correlation gather compression this would indicate collecting in the range of 5% of the correlations should be sufficient to recover the entire model, much smaller than what the Nyquist-Shannon (Nyquist, 1928) criteria would suggest.

STOMP ALGORITHM

The most difficult challenge in solving a compressive sensing problem is often finding an effective ℓ_1 (preferably ℓ_0) solver. These solvers generally have difficulty converging and are notoriously slow. The large size of any cross-correlation volume (10e7 to 10e8 elements) or, with a compressed basis still 10e5-10e6, make most ℓ_1 solver approaches impractical. Donoho et al. (2006) noted this problem with many ℓ_1 approaches and suggested an alternative, Stagewise Orthogonal Matched Pursuit (StOMP).

StOMP attempts to leverage the power of ℓ_2 solver with the ℓ_1 properties of approaches such as basic pursuit (Chen et al., 1998). The basic idea is to iteratively select the most important model elements by mapping into the model space and selecting model locations with the largest amplitudes. An ℓ_2 inversion problem is setup allowing only the locations selected in the previous step to be non-zero. The problem is mapped back into data space, the residual is recalculated, and the process is repeated. Algorithm 1 describes the approach in more detail.

Algorithm 1 StOMP algorithm

residual $\mathbf{r} = \mathbf{d}$
non-zero elements $\mathbf{M}$ = EmptySet
model $\mathbf{m} = \mathbf{0}$
operator $\mathbf{L}$
for StOMP iterations **do**
  $\mathbf{g} = \mathbf{L^T r}$
  Meaningful non-zero elements $\mathbf{N}$ = Clip($\mathbf{g}$)
  $\mathbf{M} = \mathbf{M} \cup \mathbf{N}$
  $\mathbf{m} = (\mathbf{M^T L^T L M})^{\mathbf{2}} \mathbf{M L^T r}$
  $\mathbf{r} = \mathbf{r} + \mathbf{L M m}$
end for

As a first attempt to use this method, I randomly subsampled a 4-D cr-correlation space (z, h_x, cmp_x, cmp_y) by a factor of 25 to form $\mathbf{d}$. I used the multi-dimensional wavelet compression operator $\mathbf{W}$ described in Clapp (2011), choosing 4 levels in z, 3 levels in h_x, 2 levels in cmp_x and cmp_y as $\mathbf{L}$. Figure 4 shows two Common Reflection Point (CRP) gathers in the subsurface offset and angle domain. Figure 5 shows a portion of the wavelet domain representation of the cross-correlation gathers, note how few samples are non-zero.

The "art" of StOMP is choosing the number of iterations and the clipping scheme to use to select important basis elements. In this example, I took a rather crude approach. I did StOMP iterations increasing the number selected non-zero elements to approximately match the level of sparsity observed in the wavelet domain representation. Figure 6 shows the same two subsurface offset gathers and their corresponding angle gathers. I used a large value clipping scheme that guaranteed 4% non-zero basis elements after 10 iterations Clapp (2011) showed that subsurface offset gathers could be compressed by a factor of 100 and still retain most relevant information. Note that though there is visible noise in the subsurface offset gathers the angle domain representation is quite similar. Figure 7 shows the resulting wavelet domain which appears to be significantly dissimilar to Figure 5. From C and D of Figure 6 however it appears that most of this difference doesn't map into coherent events in the angle domain.

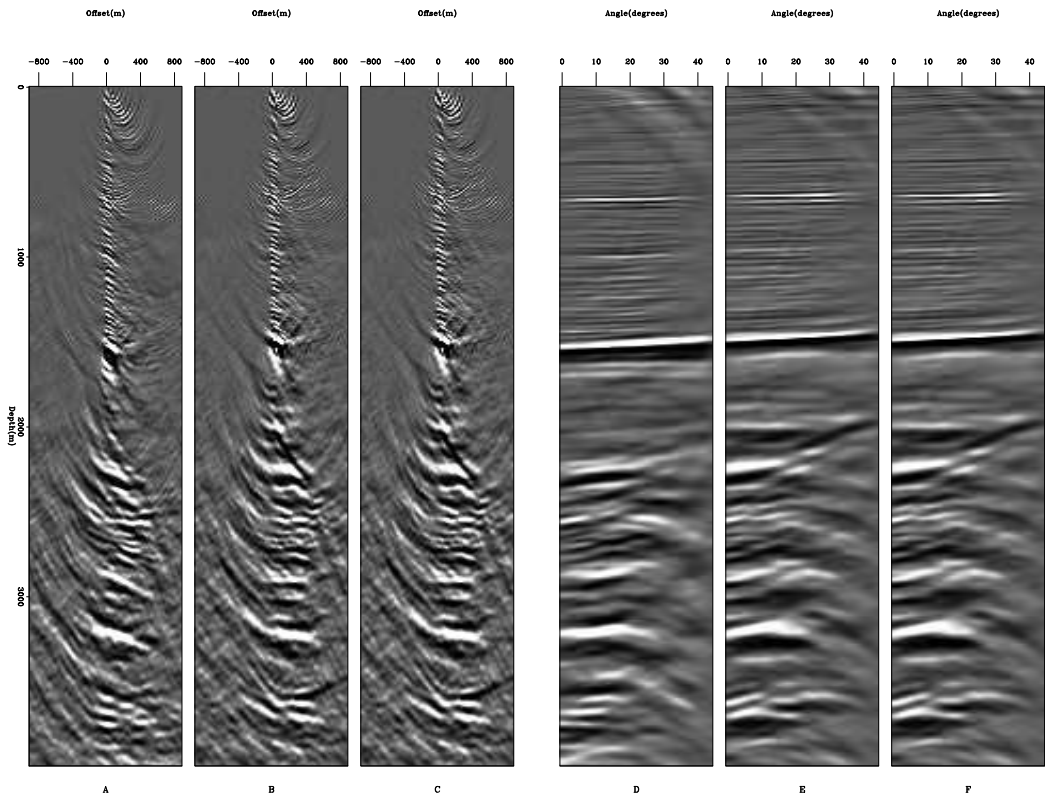


Figure 4: A, B,C and show two subsurface offset gathers from a 4-D volume. D, E, and F show the same to CRP gathers in the angle domain. [**ER**]

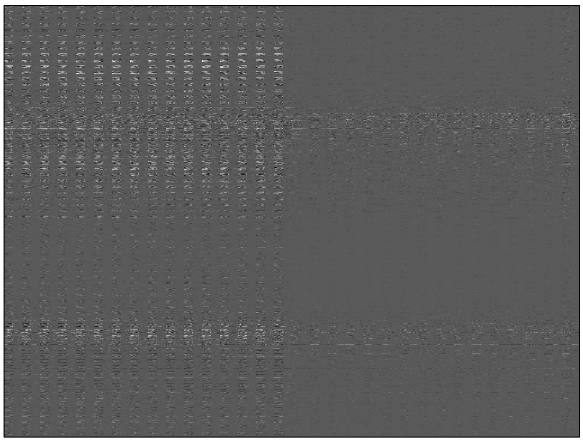


Figure 5: The wavelet domain representation of the subsurface offset domain 4-D volume partially shown in A and B of Figure 4. [**ER**]

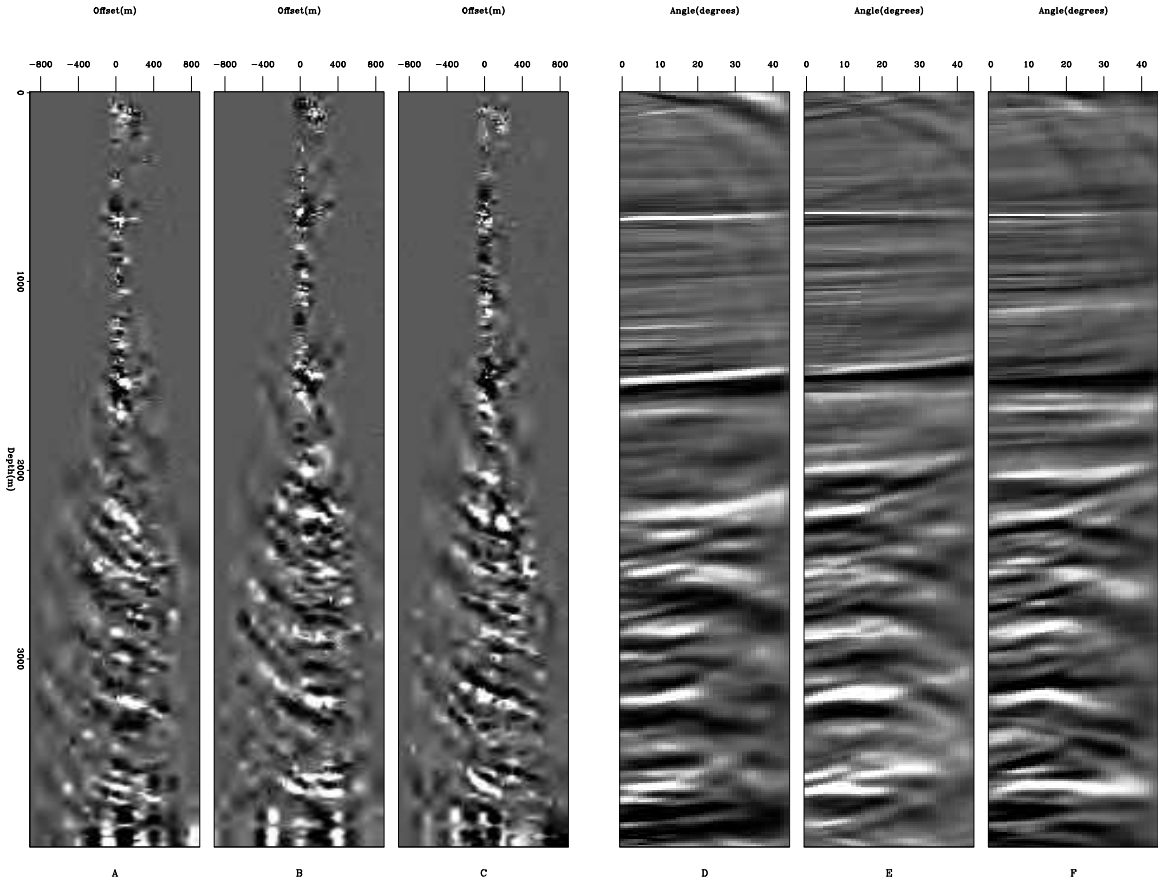


Figure 6: A, B, and C show the recovered subsurface offset gathers using compressive sensing as shown in plot A, B, and C of Figure 4. D, E, and F show the corresponding angle gathers. Note how even though A, B, and C are significantly different than A, B, and C of Figure 4, D, E, and F contain virtually the same information. [**ER**]

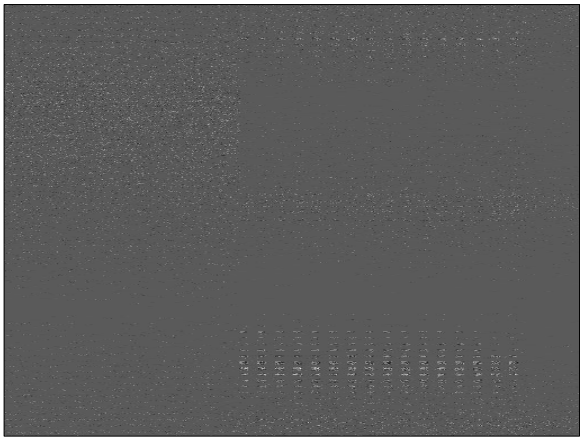


Figure 7: The wavelet domain signal after 10 iterations of StOMP. Note the dissimilarity with the correct wavelet domain show in Figure 5. [**ER**]

PHASE ENCODING

Phase encoding is another powerful technique to reduce data size. The basic idea of phase encoding is to sum several independent experiments together each scaled by a different multiplier. This multiplier could be something as sophisticated as a Gold code (Gold, 1967) or as simple as scaling the different experiments by -1, 0, or 1. The idea is that the combining of the different experiments will not, or will minimally, affect the final model. This "magic" is achieved because the encoded experiments only interact slightly when applying the operator to produce the model or through inversion techniques that attempt to undo the encoding. Examples of the first approach in geophysics can be seen in phase encoded migrations, such as plane wave migration (Whitmore, 1995), or more general phase encoded migrations(Shan, 2008). Examples of using phase encoding in inversion can be seen velocity estimation (Guerra, 2010) did inverse imaging (Tang, 2011; Leader and Almomin, 2012).

For the correlation gather construction problem, I attempted to encode multiple different correlations into each data sample. In terms of operators, we can think of the subsampling of correlations as applying a subsampling operator **S** to all possible correlations **d**. In the phase encoded case, we are going to add another operator **P** that first sums a number of different correlations together and then subsamples them leaving a new dataset **SRd**. This in turn changes the operator **L** in algorithm 1 to **PW**. For this test, I combined 20 different correlations in a random pattern to form each data point. Figure 8 shows the same three offset and angle gathers seen in Figures 4 and 6. Note the noticeably better job recovering the deeper portion of subsurface offsets. Figure 9 shows the angle gathers from the fully sampled correlation gathers and the subsampled, phase encoded gathers muted to the believable angle range. The gathers with the notable exception of more lower frequency noise in the recovered gathers.

DISCUSSION AND CONCLUSIONS

Creating full 3-D angle gathers through cross-correlation gathers is currently computationally impractical given the volume size that needs to be read/written for every shot. Compressive sensing offers a potential solution to this problem by collecting a subset of the correlation gathers and then forming the entire volume after all shot contributions have been stacked. The StOMP algorithm appears to be an effective method to obtain a sparse basis function necessary for a successful compressive sensing effort. Phase encoding to encode multiple correlations into every data point appears to offer an improved result. Further work on full 3-D shift gathers is needed to prove the feasibility of the method.

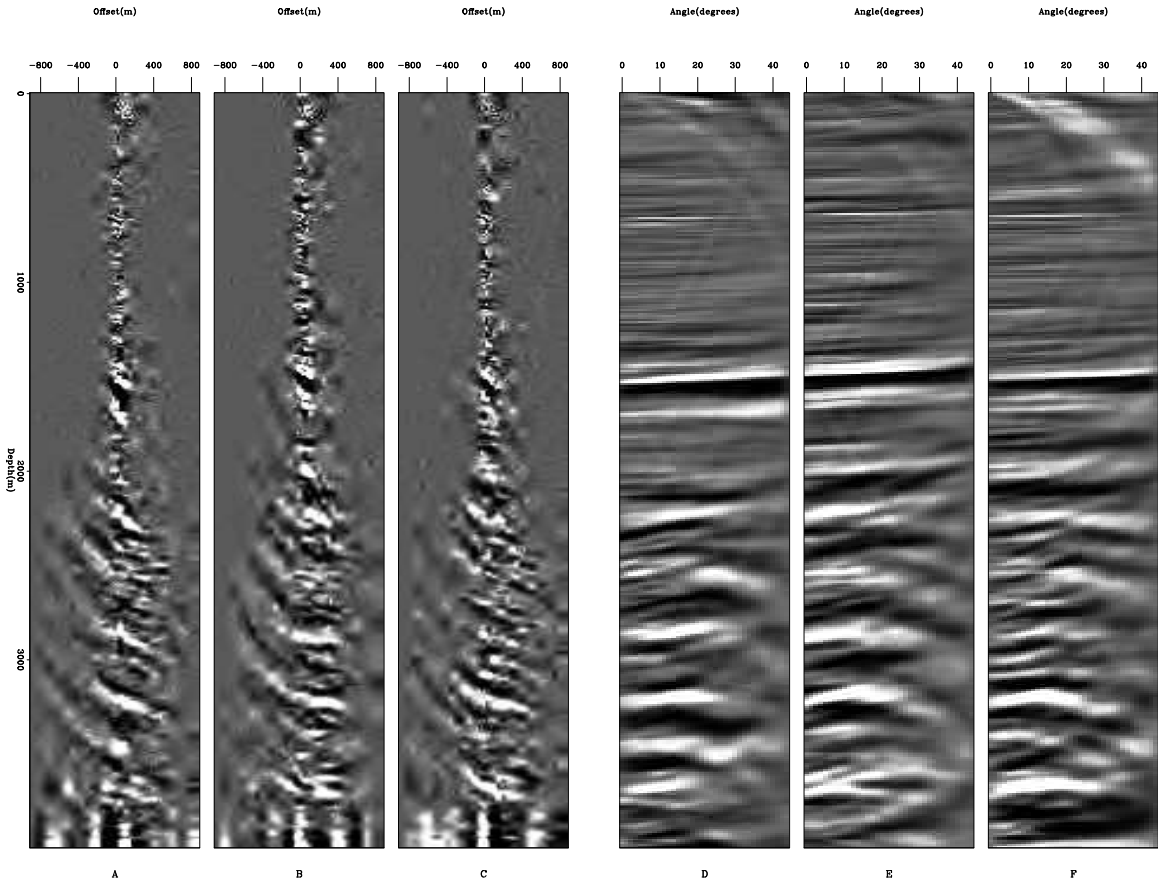


Figure 8: A, B, and C show the recovered subsurface offset gathers using compressive sensing as shown in plot A, B and C of Figure 4 and 6. D, E, and F show the corresponding angle gathers. Note the noticeable improvement in A, B, and C compared to the data shown in Figure 6. [**ER**]

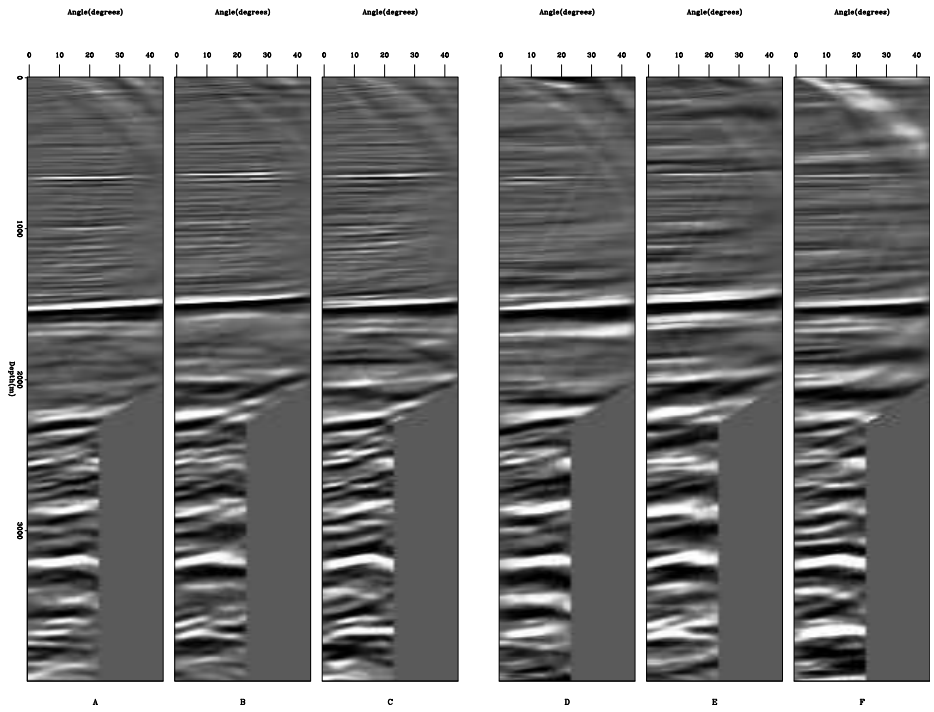


Figure 9: A comparison of the angle gathers after muting of the fully sampled correlation gathers (A, B, and C) and the phase encoded, subsampled correlation gathers (D, E, and F). [**ER**]

REFERENCES

Antonini, M., M. . Barlaud, P. Mathieu, and I. Daubechies, 1992, Image coding using wavelet transform: IEEE Transactions on Image Processing, 205–220.

Cands, E. J. and D. L. Donoho, 1999, Curvelets — a surprisingly effective nonadaptive representation for objects with edges: Curves and Surfaces, 105–120, Vanderbilt University Press.

Chen, S. S., D. L. Donoho, Michael, and A. Saunders, 1998, Atomic decomposition by basis pursuit: SIAM Journal on Scientific Computing, **20**, 33–61.

Clapp, R. G., 2011, Imaging using compressive sensing: SEP-Report, **143**, 149–158.

Clapp, R. G., H. Fu, and O. Lindtjørn, 2010, Selecting the right hardware for reverse time migration: Leading Edge, **29**, 48–58.

Donoho, D. L., 2006, Compressed sensing: IEEE Transactions on Information Theory, **52**, 1289–1306.

Donoho, D. L., Y. Tsaig, I. Drori, and J.-L. Starck, 2006, Sparse solution of underdetermined linear equations by stagewise orthogonal matching pursuit: Technical report.

Gold, R., 1967, Optimal binary sequences for spread spectrum multiplexing (Corresp.): IEEE Transactions on Information Theory,, **13**, 619–621.

Guerra, C., 2010, Migration-velocity analysis using image-space generalized wavefields: PhD thesis, Stanford University.

Leader, C. and A. Almomin, 2012, How incoherent can we be? Phase encoded linearised inversion with random boundaries: SEP-Report, **147**, 149–158.

Mallat, S. G., 1999, A wavelet tour of signal processing: Academic Press.

Mallat, S. G. and Z. Zhang, 1993, Matching pursuits with time-frequency dictionaries: IEEE Transactions on Signal Processing, **41**, 3397–3415.

Micikevicius, P., 2009, 3D Finite Difference Computation on GPUs using CUDA: GPGPU, **2**.

Nemeth, T., J. Stefani, W. Liu, R. Dimond, O. Pell, and R. Ergas, 2008, An implementation of the acoustic wave equation on FPGAs: SEG Expanded Abstracts, **27**, 2874–2878.

Nguyen, A., N. Satish, J. Chhugani, C. Kim, and P. Dubey, 2010, 3.5-D Blocking Optimization for stencil computations on modern CPUs and GPUs: Super Computing 2010, 1–13, IEEE Computer Society.

Nyquist, H. L., 1928, Certain topics in telegraph transmission theory: Transactions American Institute of Electrical Engineer, **47**, 617–644.

Sava, P. and S. Fomel, 2006, Time-shift imaging condition in seismic migration: Geophysics, **71**, 5209–5217.

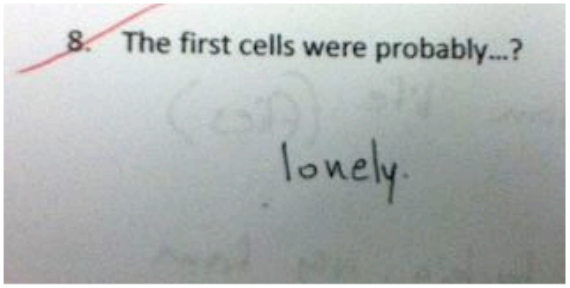
Sava, P. C. and S. Fomel, 2003, Angle-domain common-image gathers by wavefield continuation methods: Geophysics, **68**, 1065–1074.

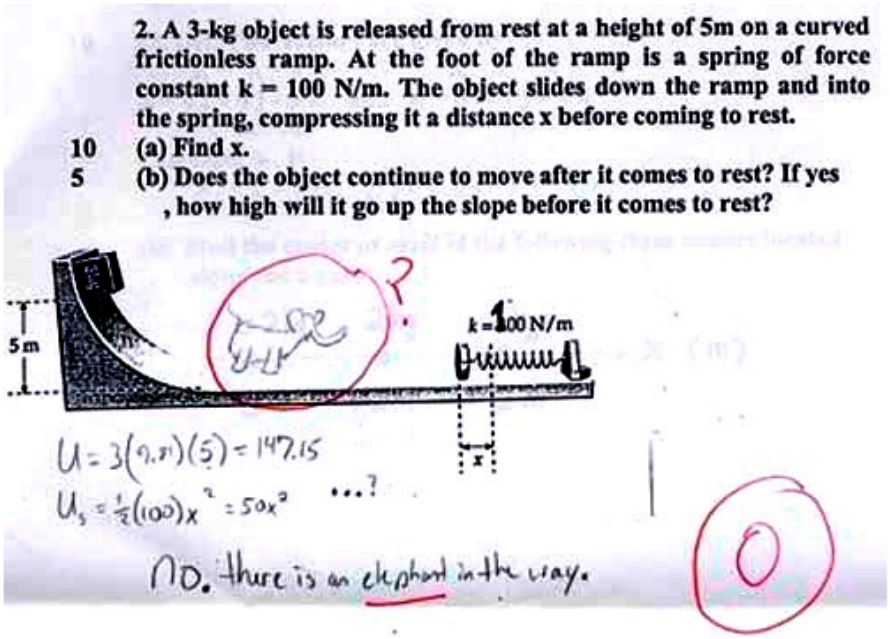
Shan, G., 2008, Imaging of steep reflectors in anisotropic media by wavefield extrapolation: PhD thesis, Stanford University.

Tang, Y., 2011, Imaging and velocity analysis by target-oriented wavefield inversion: PhD thesis, Stanford University.

Villasenor, J. P., R. A. Ergas, and P. L. Donoho, 1996, Seismic data compression using high-dimensional wavelet transforms: Snowbird, UT, USA, Expanded Abstracts, 396–405, IEEE Computer Society Press.

Whitmore, N. D., 1995, An imaging hierarchy for common angle plane wave seismogram: PhD thesis, University of Tulsa.


8. The first cells were probably...?
lonely.


2. A 3-kg object is released from rest at a height of 5m on a curved frictionless ramp. At the foot of the ramp is a spring of force constant k = 100 N/m. The object slides down the ramp and into the spring, compressing it a distance x before coming to rest.
10 (a) Find x.
5 (b) Does the object continue to move after it comes to rest? If yes, how high will it go up the slope before it comes to rest?
5m
k=100 N/m
x
U = 3(9.81)(5) = 147.15
$U_s = \frac{1}{2}(100)x^2 = 50x^2$ ...?
NO. there is an elephant in the way.
0