Random boundaries for elastic medium

Robert G. Clapp and Gustavo Alves

ABSTRACT

Elastic migration, due to both sampling requirements and the need for additional wavefields, requires significantly more storage per saved time step than acoustic migration. Using random boundaries that are time reversible is a solution to minimize storage requirements. We construct a random boundary for elastic modeling using randomly sized grains with increasingly random Lamé parameters away from the computational domain. We demonstrate that the elastic random boundary is more effective than an acoustic random boundary due to mode conversions.

INTRODUCTION

One of the computational challenges of Reverse Time Migration (RTM) is that the source wavefield is propagated forward in time while the receiver wavefield is propagated backward in time while our imaging condition requires both wavefields at the same time. For 3-D problems these wavefields are too large to store in memory. Some people solve this problem by writing the wavefield to disk, but disk I/O speeds have not kept up with computational speed increases over time, making this strategy time limited. Authors have suggested compressing the wavefield before writing to disk with some success but seismic data does not become highly compressible until dealing with 4-D volumes (Villasenor et al., 1996). Another strategy is to store a number of check-points in time (Symes, 2007) or along the boundary (Dussaud et al., 2008; Clapp, 2008). Using these checkpoints, a subset of the wavefield can be re-computed, thus allowing the imaging condition to be applied.

Clapp (2009) suggested replacing the conventional absorbing boundary with a random boundary. The random boundary attempts to produce incoherent energy rather than decaying the energy. With a random boundary term, the wave equation remains time reversible. By changing the random boundary as a function of shot, the incoherent energy produced by the boundary stacks out. Shen and Clapp (2011) noted that the boundary could be improved by introducing the concept of grain cells to the boundary, better attenuating lower frequencies.

In this paper we build a random boundary for elastic media. We build randomly sized grain cells in the boundary. We then fill these grain cells with random Lamé parameter values that increase in variability as we approach the outer boundary. We first discuss how we construct the random boundary. We then compare the elastic

random boundary to an acoustic random boundary. Finally, we show the result of elastic migration with the random boundary.

RANDOM BOUNDARY CONSTRUCTION

First, let us define some terminology for the discussion of random boundary construction. In Figure 1 we define the problem domain, boundary region, and computational domain. The information we care about is in the problem domain. The boundary region is how we are attempting to deal with the fact that we are imposing artificial limits to our simulation (we can't make a model that extends infinitely). The computational domain comprises both the boundary region and the problem domain.

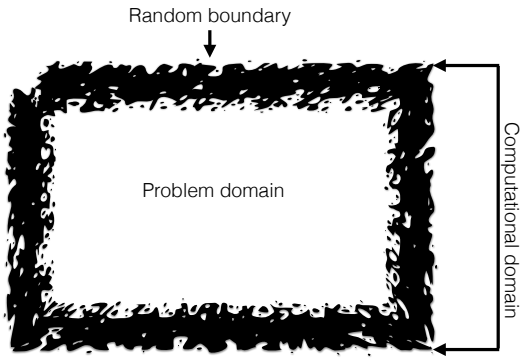

Figure 1: An explanation of the terminology used in this section. The problem domain refers to a region inside the boundary. The computational domain represents the problem domain plus the boundary region. [**NR**]

There are two main goals when constructing a random boundary. First, we want to maximize the chaotic nature of the wavefield as much as possible at a range of frequencies. Second, we want virtually all of the energy to be reflected back into the problem domain before the wavefield hits the edge of our computational domain (if significant energy hits the edge of the computational domain, it will produce a coherent, consistent artifact between shots.)

Clapp (2009) showed that low frequencies hitting the boundary region will reflect if the boundary is too random at the problem domain/random boundary region barrier. As a result, it is preferable to gradually increase the randomness towards the edge of the domain. To keep energy in the random domain as long as possible, and therefore allowing us to introduce more incoherence, we decrease the velocity in the case of acoustic propagation. For elastic propagation, this translates into decreasing λ and μ and increasing ρ. Shen and Clapp (2011) noted that introducing a randomness at the size of the grid cell was sub-optimal given the wavelengths that we were attenuating. Instead they used larger grid cells. In this paper we build on that concept but instead use randomly sized and shaped grid cells.

The fact that we are dealing with elastic medium gives another level to introduce incoherency into our wavefield. By randomly choosing λ, μ, and ρ parameters we are

continually converting between modes. Later in the paper, we will show how this significantly increases our ability to introduce incoherence.

Algorithm 1 describes how we build the boundary with arbitrarily-sized grid cells. The basic idea is to choose random points within the boundary for a new grain. We then randomly decide whether or not to enlarge the grain by choosing a random direction to expand the grain. Once the boundary is completely composed of grains, we use the starting point's distance from the problem domain and randomly assign it λ, μ, and ρ properties. Algorithm 2 outlines the procedure to assign properties to each grain.

Algorithm 1 Randomized block algorithm

```
1: while any(b==-1) do  ▷ While any boundary element have not been assigned
2:     c=b[rand()*b.size()]  ▷ Randomly choose an unassigned element of the boundary
3:     grow=true  ▷ Whether to continue growing the cell
4:     while grow do
5:         if rand() <s then  ▷ to whether or not to continue to grow cell
6:             found=false  ▷ Whether or not we have a found a direction to move
7:             d=true  ▷ Create a vector of all possible directions we could move
8:             while !found && !any(d) do  ▷ Search until we find a direction
9:                 r=rand(dir)  ▷ Randomly chose a direction to try to grow the cell
10:                d(r)=false
11:                if valid(c+r) then  ▷ If unassigned cell in boundary
12:                    c=c+r
13:                    found=.true.
14:                end if
15:            end while
16:            if !found then
17:                grow=false  ▷ We could not move
18:            end if
19:        else
20:            grow=false
21:        end if
22:    end while
23: end while
```

Figure 2 shows a closeup of a random boundary. Note the arbitrarily-sized grain cells and how the property, in this case λ, decreases and becomes more random as we move away from the problem domain. We found that it was important that the grain cell pattern be the same for all three parameters. Otherwise the boundary acts too much as a reflecting surface for low frequency components.

Figure 3 shows nine different snapshots of a wavefield as it expands and eventually hits the boundary. Note how little coherent energy comes off the random boundary

Algorithm 2 Grid cell assignment

```
b=1          ▷ We don't want to change the property much at the edge of problem domain
if λ —— μ then
    e=.1666          ▷ Low value at the edge of computational domain
else
    e=6          ▷ in the Case of density we want a high value at the boundary
end if
pct=d/size          ▷ What percentage are we into the boundary
base=(1.-pct)*b+pct*e          ▷ average value of property at this distance
found=false
while !found do
    ptest=base+(rand()-.5)*pct
    if (λ —— μ) && (ptest ¿ e/3. && ptest¡ 1.) then
        found=true
    end if
    if !λ && !μ && ptest ¿ e/3. && ptest¡ 1. then
        found=true
    end if
end while
prop=ptest*val
Property was its original value scaled by ptest
```

and how little coherent energy of large amplitude remains once the reflected and transmitted wave have hit the boundary.

Another way to assess how well the boundary works is to sum up the results of several different simulations with different random boundaries. Figures 4 and 5 show respectively σ_{zz} and v_z gathers. The left two panels show two different simulations with different random boundaries. The right panel shows the result of summing together 18 different simulations with different random boundaries. Note how different the artifacts caused by the boundary are in the left two figures. Also note how virtually no boundary-related energy remains after summing the 18 experiments.

To test how much more effective an attenuator of coherent noise an elastic random boundary is, we perform two different modeling experiments with an acoustic media. In one case we only allowed random variations in the boundary in λ. In the other case we also allowed variation in μ and ρ. Figure 6 shows a snapshot from the same time using the two different approaches. The left panels show the result of using an acoustic random boundary, the right an elastic. Note how much more coherent, high amplitude noise energy is seen in the strictly acoustic random boundary. Figure 7 shows σ_{zz} gathers using the two different methods. Again, note how much coherent noise can be seen in the acoustic boundary case.

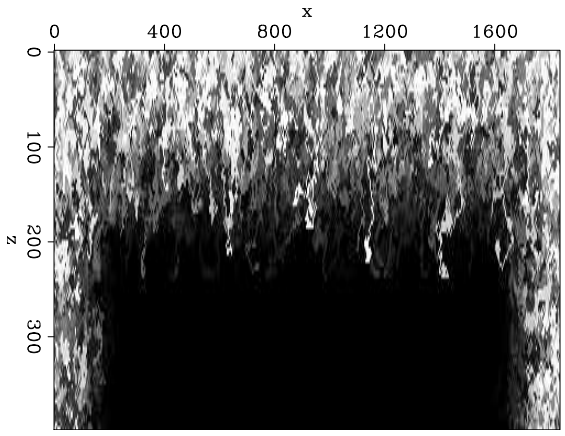

Figure 2: An example of a random boundary with random sized grains produced using the algorithm 1. [**ER**]

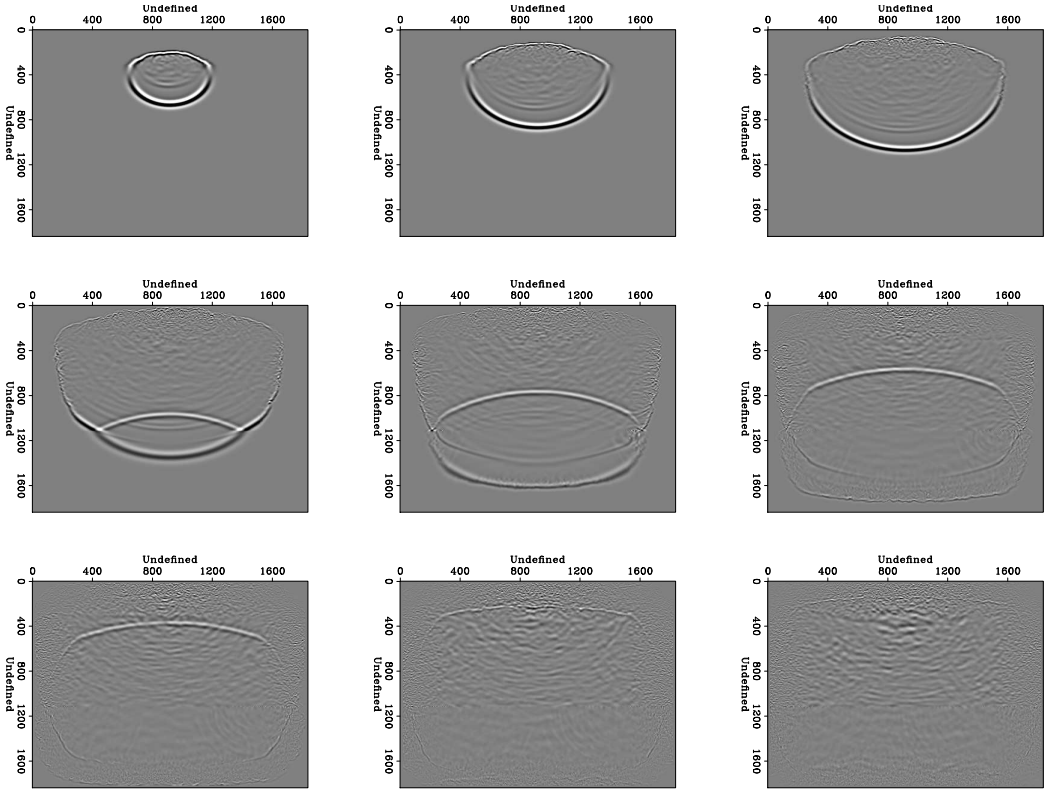

Figure 3: Several different frames of v_z using an elastic random boundary. Note the incoherent nature of the energy coming off the random boundary. [**ER**]

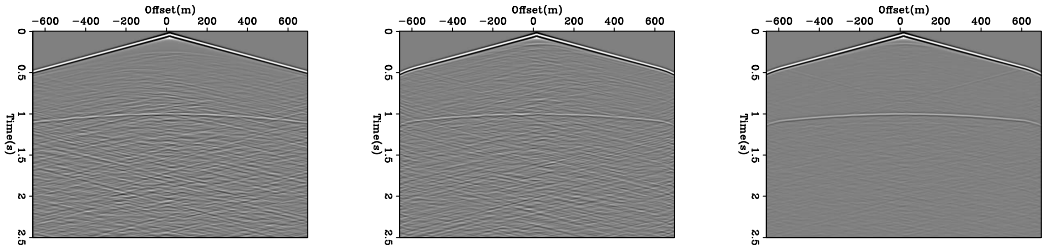

Figure 4: The left two panels are two different σ_{zz} gathers with different random boundaries. The right panel shows the result of summing together 18 simulations. Note the significant reduction in the noise associated with the random boundary. [**ER**]

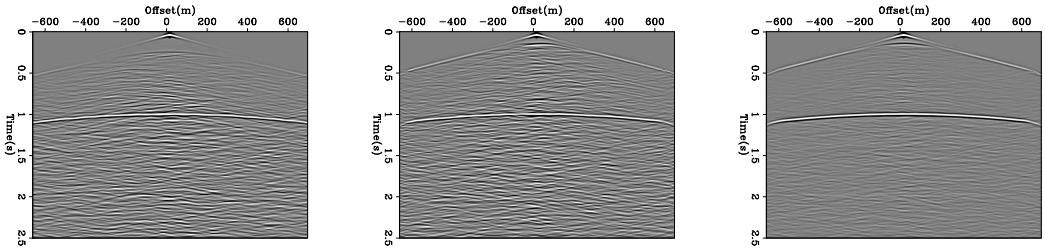

Figure 5: The left two panels are two different v_z gathers with different random boundaries. The right panel shows the result of summing together 18 simulations. Note the significant reduction in the noise associated with the random boundary. [**ER**]

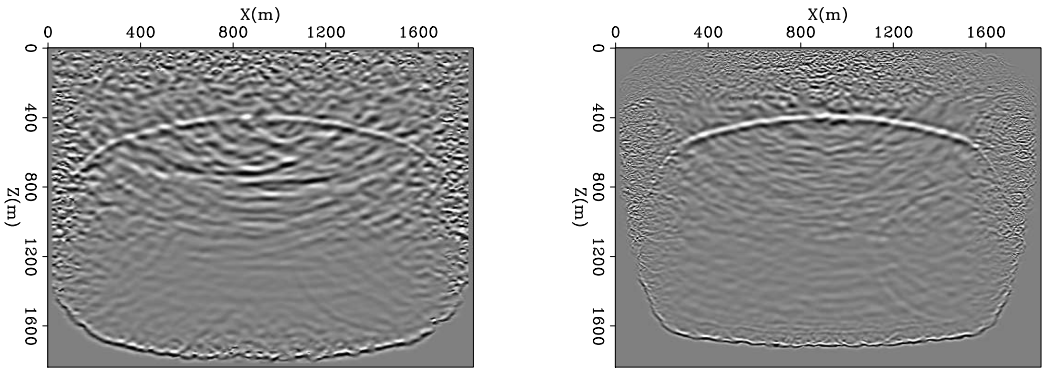

Figure 6: Snapshots from an acoustic modeling experiment using an acoustic random boundary (left) and an elastic random boundary (right). Note how much more coherent noise energy can be seen in the acoustic boundary case. [**ER**]

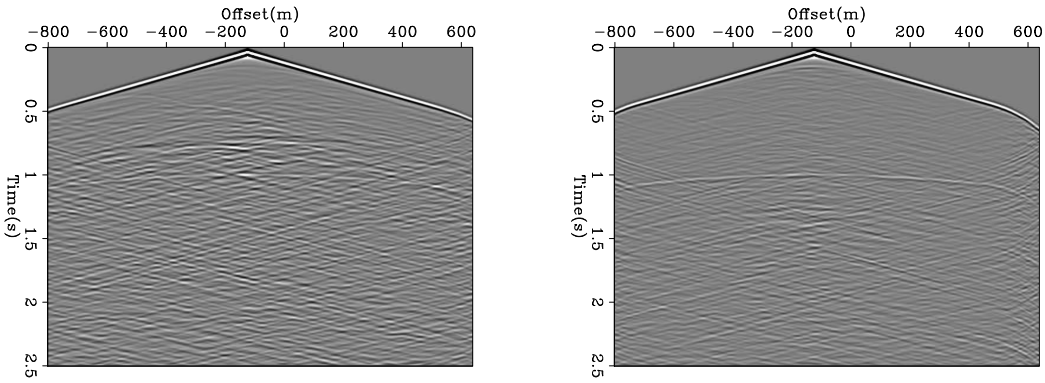

Figure 7: Gathers showing σ_{zz} in the case of an acoustic (left) and elastic (right) random boundary. Note how much less coherent noise energy can be seen in the elastic case. [**ER**]

MIGRATION TEST

Next, we perform an elastic Reverse Time Migration using random boundaries. Our model is a simple single layer reflector. First, we generate residual data by forward propagating elastic wavefields for the true model and a constant properties model and taking the difference of the two. This is done to remove the direct arrivals in the residual data. At this point, we also store the last two time steps of the propagation using the constant properties model. Figure 8 shows what the σ_{zz} wavefield looks like at the last time step. Even though the wavefield looks completely incoherent, there are no absorbing boundaries, so all the initial energy is still present and the time propagation is completely reversible.

We migrate the data by simultaneously back-propagating the receiver data injected at the receiver locations and the source data that is reconstructed from the two last time steps stored when forward propagating. Figure 9 shows density, bulk and shear moduli for a single shot migration of the single reflector model. Note how we are able to image the reflector clearly, even without stacking many shots.

CONCLUSION

We have shown how to construct a random boundary for elastic propagation. We see that the boundary is more effective than its acoustic counterpart due to mode conversions. Finally, we see that a random boundary for elastic migration produces minimal artifacts.

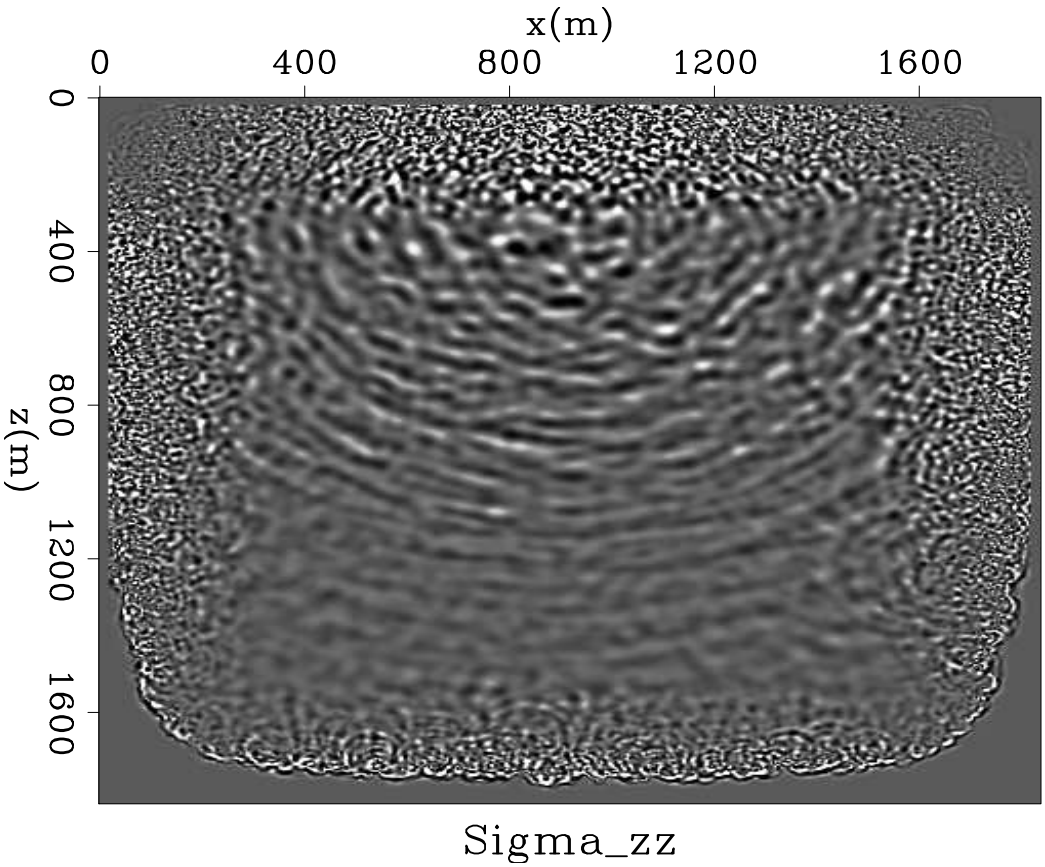

Figure 8: Last time step of the $sigma_{zz}$ wavefield for the constant property model. The random boundary conserves the total energy in the model, so the time propagation is completely reversible. [**ER**]

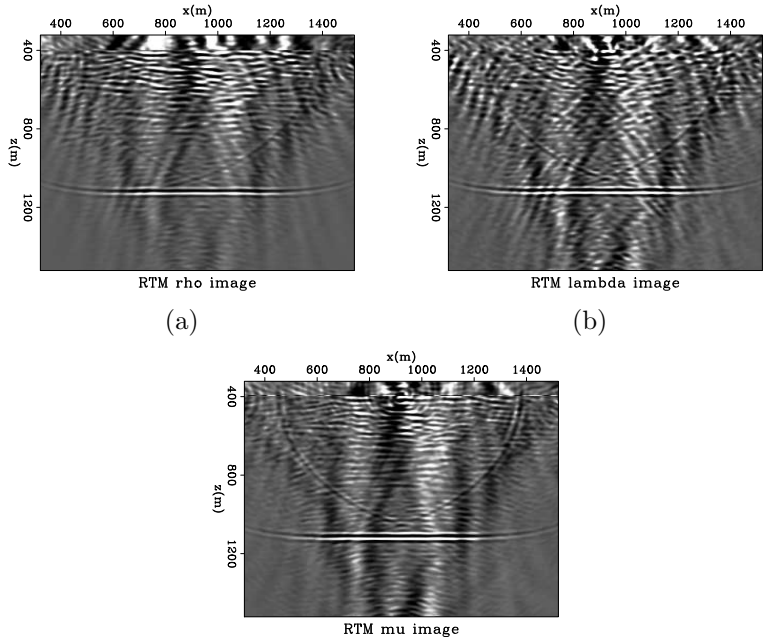

Figure 9: Final elastic RTM images for density, λ and μ for the single reflector example. [**ER**]

REFERENCES

Clapp, R. G., 2008, Reverse time migration: Saving the boundaries: SEP-Report, **136**, 137–144.

———, 2009, Reverse time migration with random boundaries: SEP-Report, **138**, 29–38.

Dussaud, E., W. W. Symes, L. Lemaistre, P. Singer, B. Denel, and A. Cherrett, 2008, Computational strategies for reverse-time migration: 78th Annual Internat. Mtg., Soc. Expl. Geophys., Expanded Abstracts, SPMI 3.3.

Shen, X. and R. G. Clapp, 2011, Random boundary condition for low-frequency wave propagation: SEP-Report, **143**, 85–92.

Symes, W. W., 2007, Reverse time migration with optimal checkpointing: Geophysics, **72**, SM213–SM221.

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.