More fun with random boundaries

Robert G. Clapp

ABSTRACT

Transfering wave-field checkpoints from disk to the compute engine is often a bottleneck in Reverse Time Migration (RTM). The need to create wave-field checkpoints can be eliminated by running the wave equation backwards and choosing a time reversible boundary condition. In acoustic propagation, velocity can be made more and more random within a boundary region. Reflections from the incoherent boundaries are random in nature and produce minimal coherent correlations when applying the RTM imaging condition. In Vertical Transverse Isotropic (VTI) media, the horizontal and Normal MoveOut (NMO) velocities can be modified to cause wavefronts to propagate parallel to the boundary region causing further degradation of coherent signals. Coherent reflections can be further reduced by decreasing the randomized velocity within the boundary region.

INTRODUCTION

Time domain finite difference is the most computationally efficient algorithm for both RTM (Baysal et al., 1983) and waveform inversion (Woodward, 1990). The kernel's simplicity, Single Instruction Multiple Data (SIMD), and high level of data reuse has led to high-performance implementations on Field Programmable Gate Arrays (FPGA) (Nemeth et al., 2008) and General Purpose Graphics Processing Units (GPGPU) (Micikevicius, 2008) along with conventional CPUs.

Both waveform inversion and RTM require source and receiver wave-fields at equivalent time to be correlated. Unfortunately, the source wave-field is propagated forward in time while the receiver wave-field is propagated backwards in time. As a result, the wave-fields are computed sequentially and one of the two wave-fields must be saved to disk. To avoid having to save the entire wave-field to disk, checkpointing Symes (2007) or boundary reinjection (Clapp, 2008; Dussaud et al., 2008) have been suggested. These schemes result in lower IO requirements at the cost of 50% more computation. Given that disk IO bandwidth is increasing at a much slower rate than computational power, all of these approaches seem likely to face an IO bottleneck. Clapp (2009) proposes to change the boundary condition on the computation kernel from one that damps to one that randomizes the wave-field hitting the boundary. Unlike the damping boundary, a randomizing boundary, which only modifies the velocity within the boundary region, is time reversible. As a result the source wave-field can

be propagated from $t = 0$ to $t_{\max}$, and then the source and receiver wave-field can be propagated simultaneously backwards. This eliminates the need to write the entire wave-field to disk in RTM and requires only two time steps of the source wave-field to be saved to disk in the case of waveform inversion. Clapp (2009) showed that the RTM image produced using a random boundary was nearly identical to one using a damping boundary except at shallow depths.

In this paper, I propose several ways to improve the imaging result using a random boundary. I begin by proposing a modified random velocity boundary that not only becomes more random as energy propagates into it, but whose average velocity decreases, as a result causing larger time delays and less coherent reflections. In addition, I show for VTI propagation that if the horizontal and NMO velocities increase within the boundary region, energy can be turned parallel to the boundary, producing more randomness.

RANDOM BOUNDARY

Clapp (2009) used a simple scheme to introduce randomness into a wave-field hitting the boundary. Algorithm 1 summarizes the approach. The concept behind the increasing randomness was to draw in as much of the wavefront energy as possible into an ever-increasing chaotic system. Figure 1 shows several snapshots of the wave-field entering the randomized zone. Note how a very small portion of the wave-field is reflected from the beginning of the boundary and how chaotic the energy leaving the boundary ends up being. The end result is that energy reentering the non-boundary portion of the computational domain produces minimal coherent reflections with the receiver (or source) wave-field. Each point in the image is the result of stacking in multiple shot record migrations. We can think of the wave-field w at any time t as being the sum of the true wave-field signal s and the noise n introduced by energy leaving the random layer. By modifying the random layer at each shot, the noise changes from shot to shot (see Figure 2). By summing many shots, we significantly reduce the noise in the final image (see Figure 3).

Figures 1-3 are somewhat misleading because the source is in the middle of the computational domain. Figure 4 shows a more realistic scenario where the shot is at the edge of the computational domain. Note the similarity in the wave-fields at early times even with different randomized boundaries. As a result, reflections at these early times (at shallow depths) show many more artifacts than those at later times (Figure 3).

The shallow depth image can be improved by modifying the random velocity boundary. Algorithm 2 introduces a decreasing maximum velocity towards the outer edge of the computational domain. Energy now takes longer to leave the boundary creating a larger time gap and more random energy pattern between the true wave-field and the noise introduced by the boundary. Figure 5 shows several snapshots with this new boundary condition. Note the increased separation in time and space (com-

---

Algorithm 1 Basic random bound construction

---

  for all boundary points **do**

    `found=false`

    Closest velocity within computational domain (*vel*)

    Percentage into boundary region (*pct*)

    while found **do**

      Select uniform normal (*rnd*)

      $dev = rnd * 2.5 * dist * val$

      if $vel + dev < v_{max}$ **then**

        `found=true`

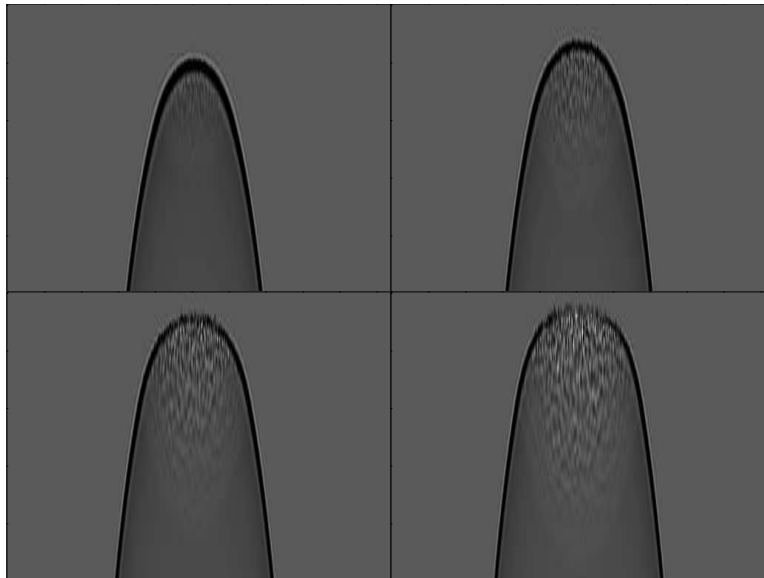 $vel = vel + dev$

      end if

    end while

  end for

Figure 1: A zoom-in on the wave-field as it propagates into the random boundary. The order in time is top-left, top-right, bottom-left, bottom-right. Note how very little of the wavefront is perturbed at early times. [**CR**]
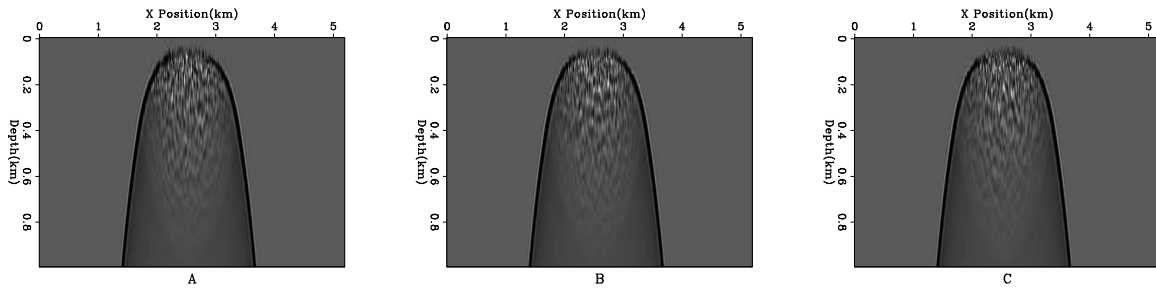
Figure 2: A zoom in on the wave-field as it passes into the random boundary. The three panels represent three different random boundaries. Note the differences in the wave-field leaving the boundary. [**CR**]

Figure 3: The result of stacking 16 different random realizations. Note how the noise introduced by the random boundary cancels out when summed. [**CR**]
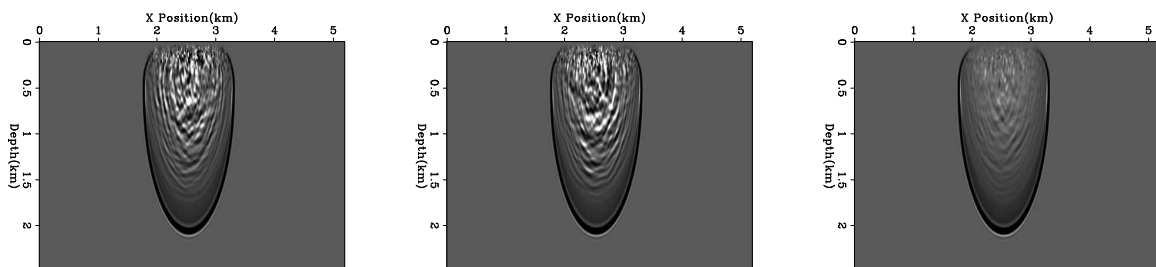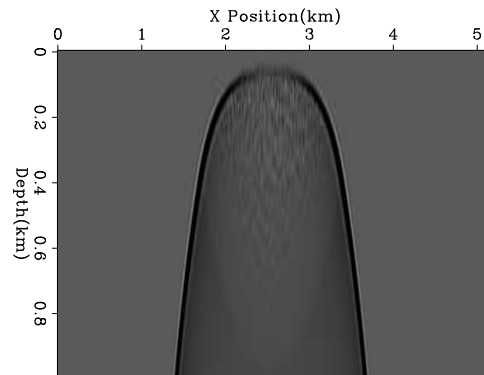

Figure 4: The left and center panels show the wave-field at the same time with two random boundaries generated using algorithm 1. The right panel shows the result of adding 16 realizations together. Note how some energy was stacked coherently. [**CR**]

pared to the wave-fields seen in Figure 1). The increased time gap and randomness produces less noise at shallow depths (Figure 6).

Algorithm 2 Basic random bound construction

    for all boundary points **do**

      `found=false`

      Closest velocity within computational domain (*vel*)

      Percentage into boundary region (*pct*)

      while found **do**

        Select uniform normal (*rnd*)

        $dev = rnd * 1.3 * dist * val$

        if $vel + dev < v_{max} * (1. - pct) * .4$ **then**

          `found=true`

          $vel = vel + dev$

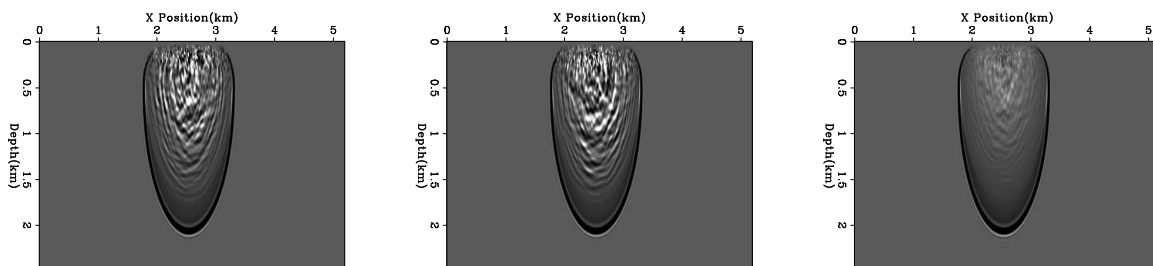   **end if**

      end while

    end for

Figure 5: The left and center panels show the wave-field at the same time with two random boundaries generated using algorithm 2. The right panel shows the result of adding 16 realizations together. Note how some energy was stacked coherently. Note the larger gap in time between the main wavefront and the energy generated from the random boundary. Further note the decreased energy in the result of stacking multiple realizations. [**CR**]

VTI RANDOM BOUNDARIES

Both Clapp (2009) and the previous section describe how to create a boundary for acoustic propagation. VTI propagation is slightly more complicated. We start by defining the horizontal velocity v_h, the vertical velocity v_v, and the the Normal Move-Out velocity v_n, defined in terms of Thompson parameters as

$$v_n = v_v(1 + 2\delta)^{\frac{1}{2}} \tag{1}$$
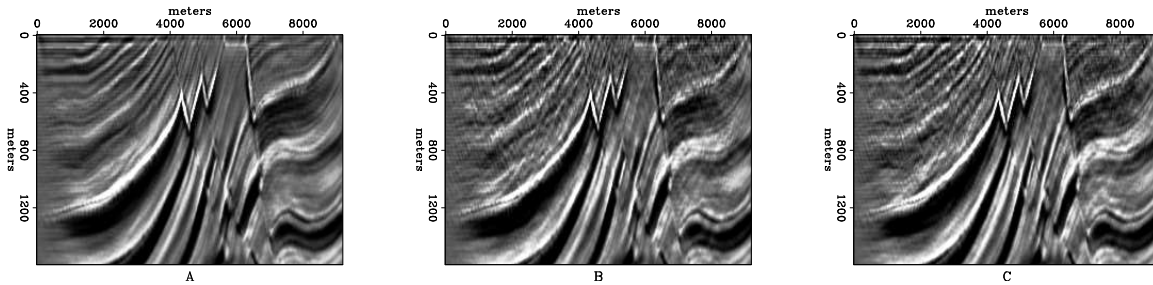
and

$$v_h = v_v(1 + 2\epsilon)^{\frac{1}{2}}. \tag{2}$$

Figure 6: The result of migrating using a damped boundary condition(A), the boundary condition described in algorithm 1 (B), and the boundary condition described in algorithm 2 (C). Note how the shallow reflectors are preserved in (C). [**CR**]

In acoustic finite difference, a second derivative in x,y, and z ($p_{d2x}, p_{d2y}, p_{d2z}$) is calculated at all ix, iy, and iz at a given time step it of the wave-field p. The wave-field at the next time step is then calculated using these derivatives and the sampling in time dt, along with a source term s and the previous value of the wave-field. The computational kernel becomes

$$
\begin{aligned}
p(iz, ix, iy, it+1) & = p(iz, ix, iy, it+1) - p(iz, ix, iy, it-1) + 2p(iz, ix, iy, it) \\
& + v(iz, ix, iy)dt^2(p_{d2x} + p_{d2y} + p_{d2z}).
\end{aligned}
\tag{3}
$$

Following the approach in Alkhalifah (2000) involves an auxiliary wave-field q.; Derivatives in x and y are calculated on p and derivatives in z are calculated on q (q_{d2z}). The computation kernel then becomes

$$
\begin{aligned}
p(iz, ix, iy, it+1) & = p(iz, ix, iy, it+1) - p(iz, ix, iy, it-1) + 2p(iz, ix, iy, it) \\
& + dt^2(v_h(iz, ix, iy)(p_{d2x} + p_{d2y}) + v_v(iz, ix, iy)q_{d2z}
\end{aligned}
\tag{4}
$$

and

$$
\begin{aligned}
q(iz, ix, iy, it+1) & = s(iz, ix, iy, it+1) - q(iz, ix, iy, it-1) + 2q(iz, ix, iy, it) \\
& + dt^2(v_n(iz, ix, iy) * (p_{d2x} + p_{d2y}) + v_v(iz, ix, iy)q_{d2z}.
\end{aligned}
\tag{5}
$$

Randomizing v_v, v_h, and v_n independently creates an unstable system. The most straightforward way to add stable random boundaries to the VTI problem is to create an acoustic random boundary layer by setting ϵ and δ to 0 (therefore $v_v = v_h = v_n$). A better strategy is to take advantage of the extra flexibility of having three parameters describing moveout. The longer a wavefront travels through the randomized layer, the more chaotic the resulting wave-field and the longer the delay between the true signal and the beginning of noise. By increasing ϵ and δ while decreasing v_v we can cause the wave-field to turn parallel to the random boundary. To see this effect, I have removed the random component of the boundary layer while still decreasing v_v and increasing ϵ and δ. Figure 7 shows the result of overlaying two wave-fields, one using an isotropic and one using anisotropic boundary. The anisotropic boundary results

energy traveling longer in the boundary region. Figure 8 shows the randomized wave-field at several different time steps using both an isotropic and anisotropic boundary condition. Note how the noise pattern using the anisotropic boundary condition is much less regular.
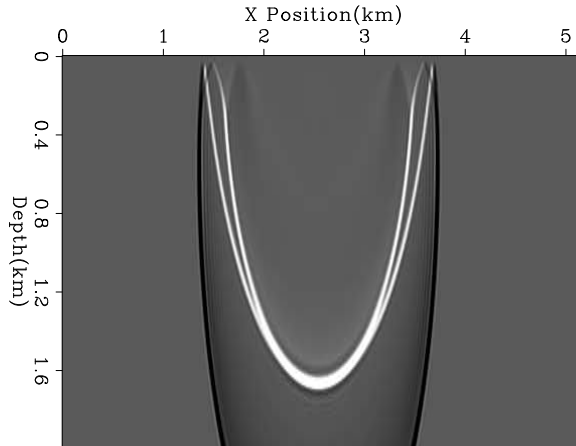

Figure 7: The result of over-laying the wave-fields using an isotropic and anisotropic boundary. The anisotropic boundary results in energy traveling longer in the boundary region.

CONCLUSIONS

By reducing IO requirements, random velocity boundaries have computational advantages for correlation-based imaging conditions, compared to damping boundary conditions. They can be further improved by systematically decreasing the mean velocity in the boundary. In VTI modeling/migration, the boundaries' effectiveness can be improved by systematically modifying δ and ϵ.

REFERENCES

Alkhalifah, T., 2000, An acoustic wave equation for anisotropic media: Geophysics, **65**, 1239–1250.

Baysal, E., D. D. Kosloff, and J. W. C. Sherwood, 1983, Reverse time migration: Geophysics, **48**, 1514–1524.

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

——, 2009, 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.

Micikevicius, P., 2008, 3d finite difference computation on gpus using cuda: 2nd Workshop on General Purpose Processing on Graphics Processing Units, Expanded Abstracts, 79–84.

Nemeth, T., J. Stefani, W. Liu, R. Dimond, O. Pell, , and R. Ergas, 2008, An implementation of the acoustic wave equation on fpgas: 78th Ann. Internat. Meeting, Expanded Abstracts, 2874–2877, Soc. Expl. Geophys.
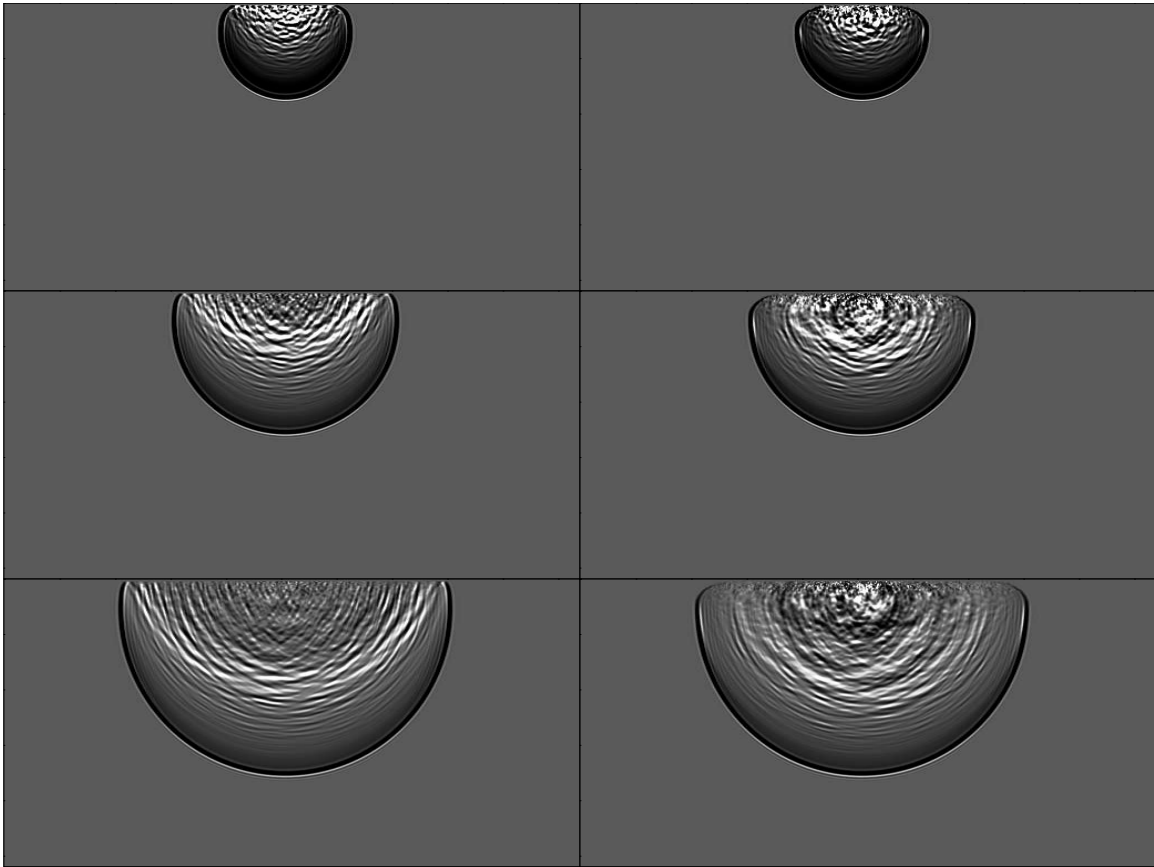
Figure 8: The left panels show a wave-field at several different time steps using an isotropic boundary condition. The right panel shows the wave-field using an anisotropic boundary condition. [**CR**]

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

Woodward, M., 1990, Wave equation tomography: **62**.