FWI with different boundary conditions

Xukai Shen, and Robert G. Clapp

ABSTRACT

We compare time-domain Full Waveform Inversion using different boundary schemes: absorbing boundary condition, random boundary condition and continuation of velocity. The absorbing boundary condition requires saving the wavefield, the other two do not, but require extra wavefield modeling. The random boundary condition results in a gradient calculation that is almost as good as the absorbing boundary condition, whereas the continuation of velocity result in a gradient that has strong artifacts. However, the final inversion results using different boundaries are similar where there is a lot of data to contrain the model. When there is not so much data to constrain the inversion, results from using the random boundary condition are similar to those resulting from the absorbing boundary condition, but the continuation of velocity boundary condition does not work as well. We demonstrate this with synthetic examples.

INTRODUCTION

Full Waveform Inversion (FWI) (Tarantola, 1984; Pratt et al., 1998; Mora, 1987) gives more accurate subsurface velocity estimation than conventional methods of velocity estimation, such as ray-based methods (Hampson and Russell, 1984; Olson, 1984; White, 1989), especially in geologically complex areas. Traditional time-domain FWI algorithm is very computationally expensive for two reasons: first, it involves iterative forward and backward two-way wavefield propagation; second, to compute FWI gradient, at least one of the wavefield needs to be saved. Because wavefields in large-scale 3D application can require terabytes of storage, I/O can be a non-trivial bottleneck. Computational time of two-way wavefield propagation can be dramatically reduced by using unconventional hardware such as GPUs (Micikevicius, 2008), leaving the I/O cost associated with the huge wavefield size the primary bottle neck. Time-reversible boundary conditions can almost eliminate the I/O cost entirely by propagating source and receiver wavefields in the same temporal direction (Clapp, 2009; Shen and Clapp, 2011). By doing this, gradient calculation can be done on the fly, and no wavefield saving is necessary. However, to achieve good results, boundaries must be carefully designed to avoid artifacts. For example, in Reverse Time Migration (RTM) applications, random boundary conditions (Clapp, 2009) result in good final images, whereas continuation of velocity gives rise to artifacts in the final image. RTM image calculation is similar to FWI gradient calculation. However, FWI performs those

calculations many more times, which makes it different from RTM as far as boundary artifacts are concerned. We first present the pseudo-code of gradient calculation using different boundary conditions; then we compare the results of a single gradient calculation and the whole inversion process for each boundary condition.

ALGORITHM DESCRIPTION

FWI as used here is a gradient-based time domain implementation (Shen, 2010). Different boundary conditions are used for the gradient calculation, leading to slight modifications of the gradient calculation algorithm. However, the steplength search uses absorbing boundary conditions in all cases. The pseudo-code of the gradient calculation using the absorbing boundary condition is as follows:

Algorithm 1 Pseudo code of gradient calculation using absorbing BC

for $is = 1, ns$ **do**
  Forward wavefield propagation, generate modeled data and record source wavefield;
  Calculate data residual using recorded data and modeled data;
  Reverse-time propagation of residual data, correlate residual wavefield with source wavefield to generate gradient;
end for

The pseudo-code of the gradient calculation using the time-reversible boundary condition (random boundary condition and continuation of velocity) is as follows:

Algorithm 2 Pseudo code of gradient calculation using time-reversible BC

for $is = 1, ns$ **do**
  forward wavefield propagation using absorbing boundary, generate modeled data, do not record wavefield;
  forward wavefield propagation using random boundary, record last two time slices of source wavefield;
  calculate data residual using recorded data and modeled data;
  reverse time propagation of residual data and source wavefield using random boundary, generate gradient on the fly;
end for

It is worth mentioning that for the MPI version of the code, wavefield propagation on different computational nodes uses different random boundary realizations, which further reduces artifacts.

GRADIENT COMPARISON

In this section and the next, we will use the 2D Marmousi velocity model to compare FWI results with different boundary conditions. The true velocity model is a modified version of the 2D Marmousi model, with 12-meter spacing in both x and z. The starting model is a smoothed version of the true model. The survey geometry simulates fixed spread land acquisition. A total of 60 shots are used with 84 m shot spacing. Figure 1 a shows the true model with continuation of velocity in the boundary region, figure 1 b shows the starting model with continuation of velocity in the boundary region, and figure 1 c shows the starting model with random velocity values in the boundary region.

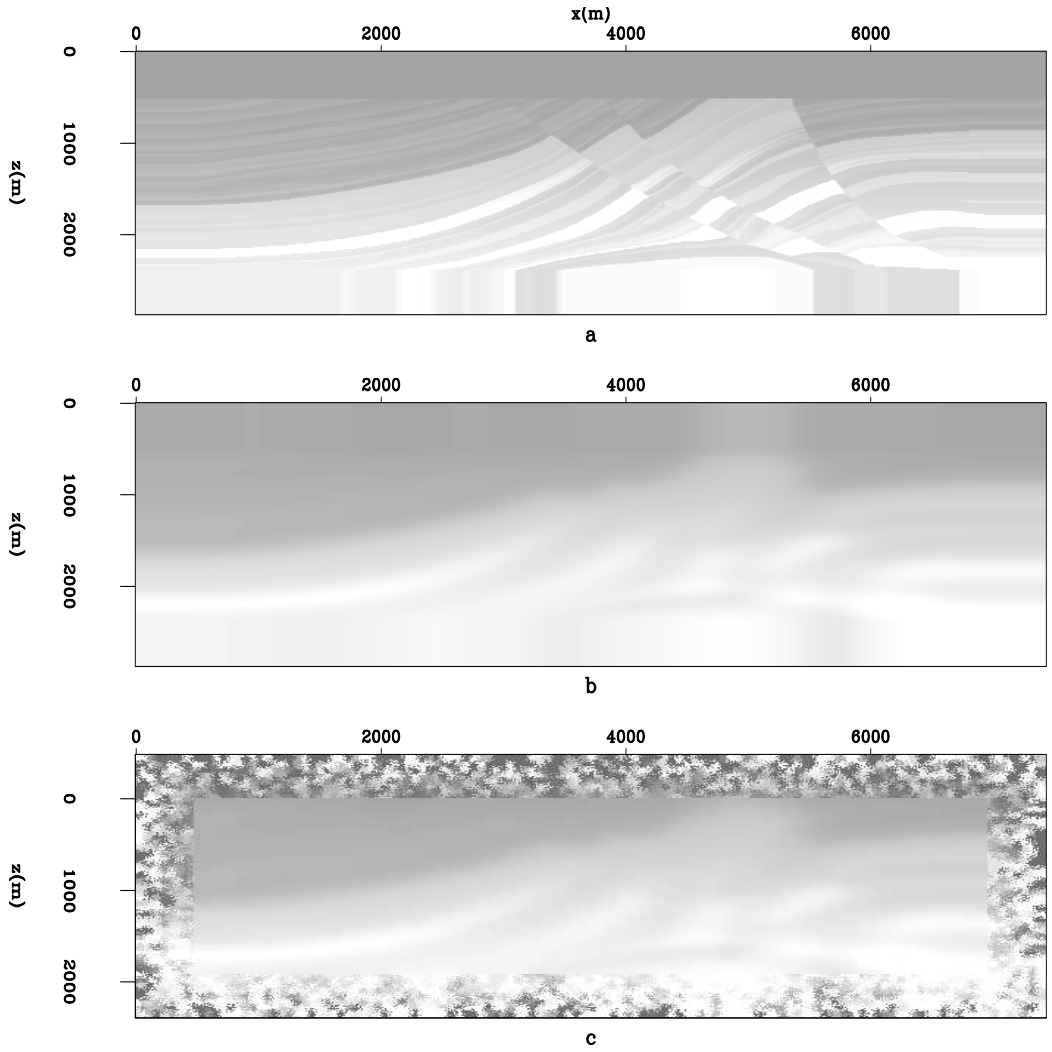

Figure 1: Different velocity models with different boundary regions: a) true velocity with continuation of velocity in the boundary region; b) starting velocity with continuation of velocity in the boundary region; c) starting velocity with random velocity values in the boundary region. [**ER**]

Figure 2 shows gradients of the first iteration using different boundary conditions. Due to the similarity between gradient calculation and RTM image calculation, it can be seen that using a random boundary condition gives a good gradient that is very similar to the ones from using an absorbing boundary condition, except for the usual strong amplitude near source locations. The gradient calculated using continuation of velocity in the boundary region, on the other hand, has some artifacts from reflections in the boundary region. This is particularly obvious in the entire shallow depth region.

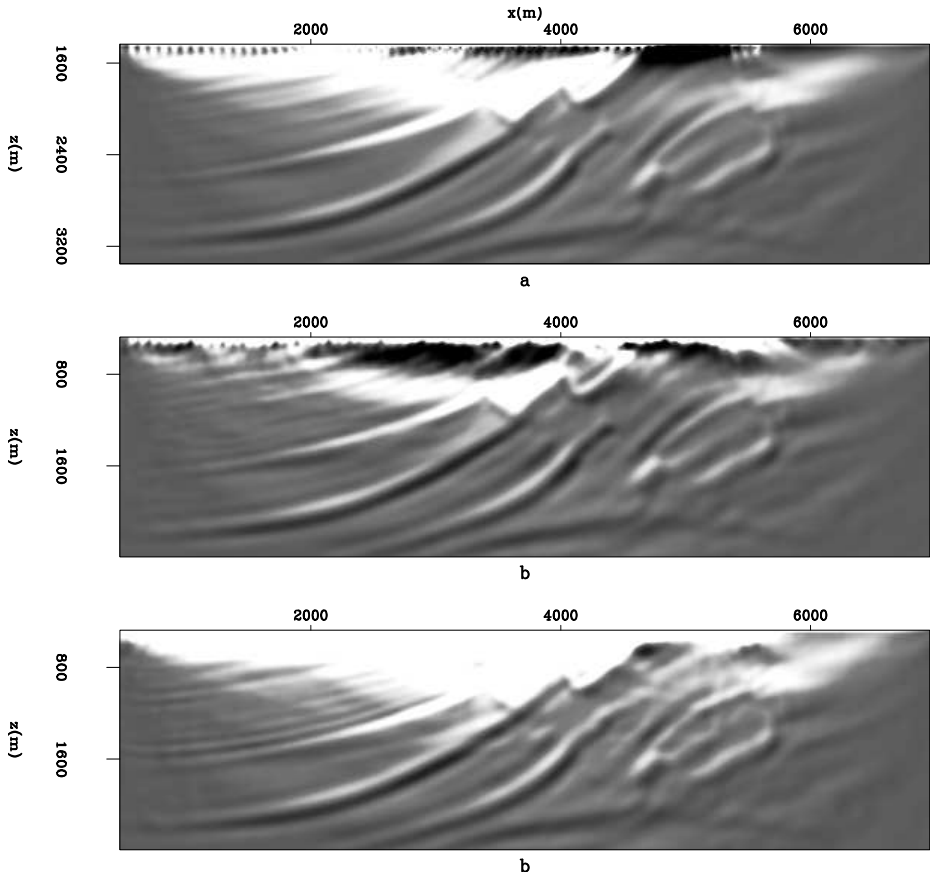

Figure 2: First iteration gradient with different boundary conditions:a) gradient with absorbing boundary condition;b) gradient with random boundary condition;c) gradient with continuation of velocity in the boundary region. [**CR**]

INVERSION COMPARISON

For inversion, a total of 160 iterations are run for each boundary condition. Figure 3 shows inversion results using different boundary conditions. Inversion results from the random boundary condition and the absorbing boundary condition are very similar. Results from using continuation of velocity have more artifacts, in the side and bottom parts of the model where data fitting constraints are relatively weak. In other words,

strong artifacts in the shallow part in the early iterations are eliminated by data fitting constraints as the inversion proceeds. So the artifacts pattern in the inversion results is different from the one in the first gradient.

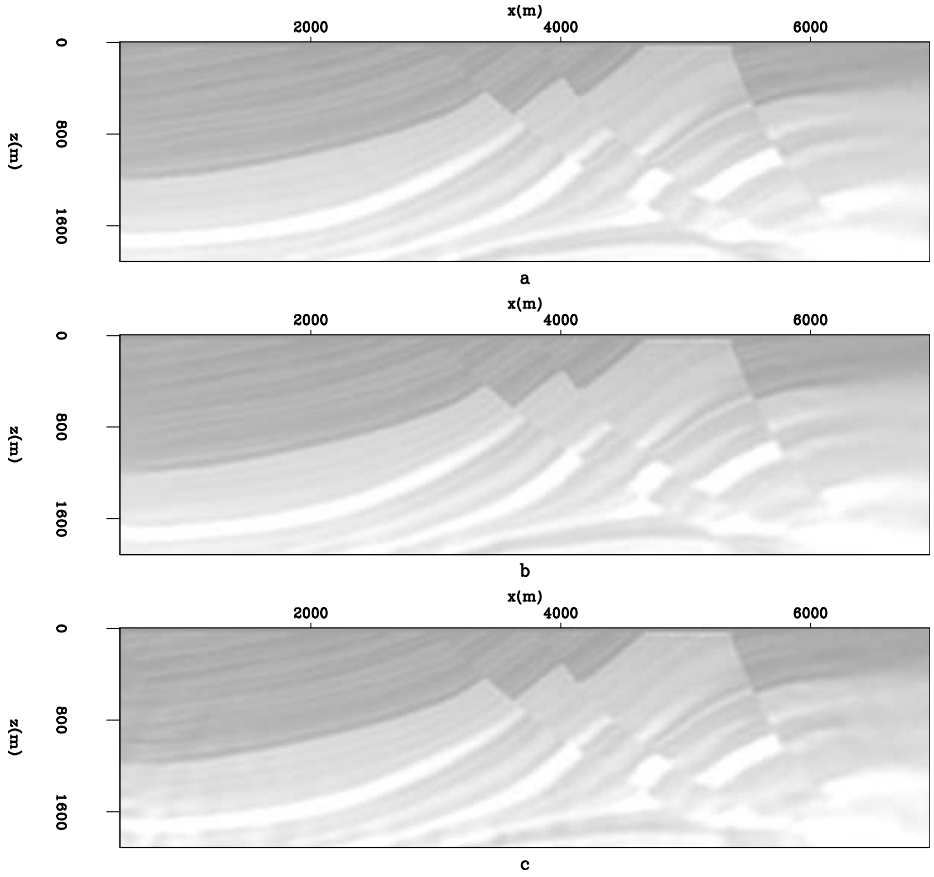

Figure 3: Inversion results with different boundary conditions: a) with an absorbing boundary condition; b) with a random boundary condition; c) with continuation of velocity in the boundary region. [**CR**]

CONCLUSION

We compared FWI using three different kinds of boundary conditions: absorbing, random and continuation of velocity. The random and continuation-of-velocity boundary conditions eliminate almost all I/O cost associated with wavefield storage and transfer at the expense of two extra wavefield propagations per iteration. Random boundary inversion results are almost as good as those from the absorbing boundary condition, which is current industry practice. The continuation-of-velocity boundary condition, on the other hand, works quite well in well-constrained areas, and less so in poorly constrained areas This is particularly attractive on unconventional architecture where computational cost is much less than memory access cost (i.e. GPUs).

REFERENCES

Clapp, R. G., 2009, Reverse time migration with random boundaries: SEG Expanded Abstracts, **28**, 2809–2813.

Hampson, D. and B. Russell, 1984, First-break interpretation using generalized linear inversion: Journal of Canadian Society of Exploration Geophysicists, **20**, 40–54.

Micikevicius, P., 2008, 3D finite difference computation on GPUs using CUDA: 2nd Workshop on General Purpose Processing on Graphics Processing Units, Expanded Abstracts.

Mora, P., 1987, Elastic wavefield inversion: Stanford Exploration Project Ph.D. Thesis.

Olson, K. B., 1984, A stable and flexible procedure for the inverse modelling of seismic first arrivals: Geophysical Prospecting, **37**, 455–465.

Pratt, R. G., C. Shin, and G. Hicks, 1998, Gauss-Newton and full Newton methods in frequency domain seismic waveform inversion: Geophysical Journal International, **133**, 341–362.

Shen, X., 2010, Near-surface velocity estimation by weighted early-arrival waveform inversion: SEG Expanded Abstracts.

Shen, X. and R. G. Clapp, 2011, Random boundary condition for low-frequency wave propagation: SEG Expanded Abstracts.

Tarantola, A., 1984, Inversion of seismic reflection data in the acoustic approximation: Geophysics, **49**, 1259–1266.

White, D. J., 1989, Two-dimensional seismic refraction tomography: Geophysical Journal International, **97**, 223–245.