Fast Full Wave Seismic Inversion using Source Encoding

Jerome R. Krebs, John E. Anderson, David Hinkley, Anatoly Baumstein and, Sunwoong Lee, ExxonMobil Upstream Research Company, Ramesh Neelamani, ExxonMobil Exploration Company, Martin-Daniel Lacasse, ExxonMobil Research and Engineering Company*

Summary

Full Wavefield Seismic Inversion (FWI) estimates a subsurface elastic model by iteratively minimizing the difference between observed and simulated data. This process is extremely compute intensive, with a cost on the order of at least hundreds of prestack reverse time migrations. For time-domain and Krylov-based frequency-domain FWI, the cost of FWI is proportional to the number of inverted seismic source gathers. We have found that the cost of FWI can be significantly reduced by applying it to data processed by encoding and summing individual source gathers, and by changing the encoding functions between iterations. The encoding step forms a single gather from many input source gathers. This gather represents data that would have been acquired from a spatially distributed set of sources operating simultaneously with different source signatures. We demonstrate, using synthetic data, significant cost reduction by applying FWI to encoded simultaneous-source data.

Introduction

Due to the large number of model parameters estimated by FWI, the only practical optimization techniques are iterative gradient search methods (Nocedal et al., 2006). An iteration of gradient search requires a few evaluations of both the objective function and its gradient with respect to the model parameters. A majority of the ongoing FWI research is based on the work of Tarantola (1984), who showed that the gradient can be computed using the adjoint method for roughly the cost of $3N_s$ simulations.

Most FWI methods, to date, are based on finite-difference simulators. These simulations can be computed in either the time domain or frequency domain. For time-domain FWI, each source response must be simulated individually, and therefore the cost for time-domain FWI is proportional to the number of sources.

Frequency-domain FWI can potentially simulate all sources for the cost of one expensive matrix inversion followed by one inexpensive forward matrix multiply per source location (Marfurt, 1984). While this computational advantage can easily be realized in 2D, inversion of the 3D frequency-domain matrix is much more difficult. Krylov methods have been applied to 3D frequency domain simulation (Riyanti et al.,2006; Erlanga et al., 2006), but these methods must solve for each source individually, and thus the cost of FWI is proportional to the number of sources inverted.

Several authors have demonstrated methods for reducing the compute cost of FWI. Particularly noteworthy is work that demonstrates that frequency-domain FWI can yield very good results by inverting only a few frequencies (Sirgue et al., 2004). Other authors have suggested that efficiency of time-domain FWI could be improved by inverting coherent sums of sources (Berkhout, 1992), or by inverting sums of widely spaced sources (Mora, 1987; Capdeville et al., 2005).

Encoded source sums have also been used to speed up seismic acquisition (Neelamani, 2008a), in wave-equation migration (Romero et al., 2000), and in seismic simulation (Neelamani, 2008b; Ikelle, 2007). All of these methods suffer from cross-talk noise between the encoded sources, which limits the number of sources that can be summed. Popular encoding methods include phase reversal, phase shifting, time shifting and convolution with random sequences.

In this paper we demonstrate that the encoded summation method can achieve large efficiency gains for FWI, if the encoding of the sources is changed between iterations. In our implementation, this change is made by altering the random number seed used to generate the source encoding functions between iterations. The compute time needed to change coding is insignificant relative to the time needed for evaluating the objective function and its gradient. Changing the encoding between iterations changes the cross-talk noise between iterations in a manner that is incoherent from iteration to iteration. Thus, the cross-talk noise essentially stacks out of the inverted earth model, allowing summation of a large number of encoded sources. We call this method encoded simultaneous-source FWI (ESSFWI).

Theory

FWI attempts to minimize, with respect to subsurface model parameters, an objective function that measures the difference between measured and simulated data. We consider the simplest form of the FWI objective function:

$$h(c) = \frac{1}{2}\sum_{n=1}^{N_s} \left\| u(c, s_n) - d_n \right\|^2 , \qquad \text{Equation (1)}$$

where:

h = the objective function,
c = the subsurface property model (e.g., density, C_{33}…),
s_n = a source function,
$u(c,s_n)$ = the simulated wavefield,
d_n = measured seismic data for source s_n, and
N_s = number of source gathers in the seismic survey.

For ESSFWI, we replace the objective function in Equation (1) with the following:

$$h(c) = \frac{1}{2}\left\| u\left(c, \sum_{n=1}^{N_s} e_n \otimes s_n \right) - \sum_{n=1}^{N_s} e_n \otimes d_n \right\|^2, \qquad \text{Equation (2)}$$

where the encoding functions e_n are functions of time, and $\otimes$ represents time-domain convolution. For simplicity, we assume that all encoding functions e_n have the same number of time samples T_c. Note that in general $e_n \neq e_m$ for $n \neq m$.

Note that Equation (2) is actually only appropriate if receivers are fixed, and if all receivers record data from all sources. While this is true of many types of land and ocean bottom acquisition, it is not true of marine streamer acquisition.

Equation (2) has the advantage that only one seismic simulator run is needed to compute u, as opposed to the N_s simulator runs needed for the conventional FWI objective function in Equation (1). On the other hand, the new encoded source signatures have $T_s + T_c - 1$ time samples. Therefore, the simulator must run for T_c more time steps to evaluate Equation (2) than are necessary in Equation (1). The compute efficiency gain for evaluating Equation (2) (η_{eval}) is then given by:

$$\eta_{eval} = N_s \frac{T_s}{(T_s + T_c - 1)}. \qquad \text{Equation (3)}$$

Importantly, since objective function gradients are computed using a few simulations, Equation (3) also gives the compute efficiency for evaluating gradients.

The ESSFWI objective function can be evaluated very efficiently. However, inversion using this objective function is more efficient only if this efficiency gain in objective function evaluation is not offset by a reduced convergence rate relative to the FWI objective function. We show in the following synthetic examples that the convergence rate of ESSFWI is close to that of conventional FWI, if the encoding functions are changed between iterations.

A high-level flow chart for our implementation of ESSFWI is shown in Figure 1. Note, the data and source signatures are re-encoded between update direction iterations.

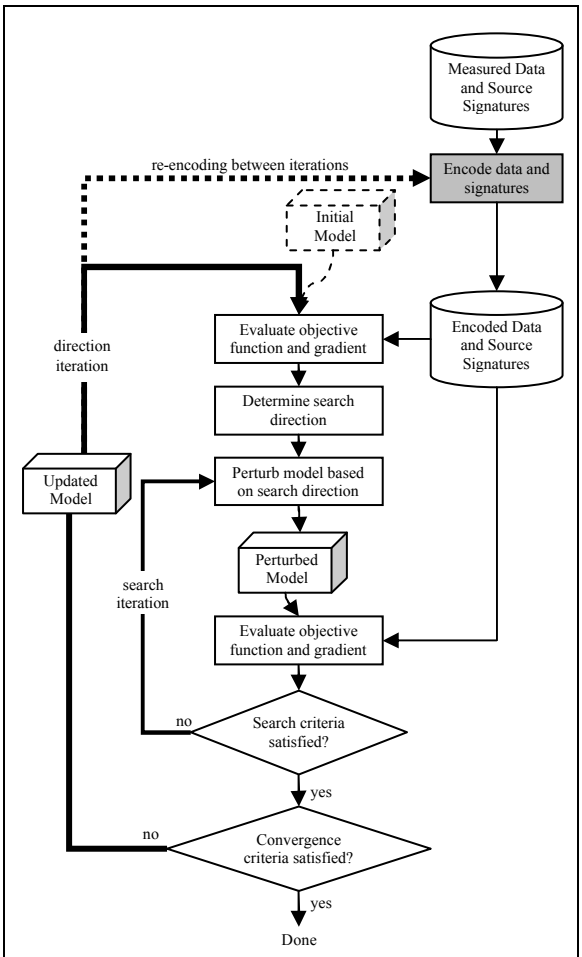

Figure 1: Typical flow for gradient search ESSFWI.

Method

For simplicity, all examples are 2D constant-density acoustic inversions of a modified version of the Marmousi II (Martin, 2004) model (see Figure 2a), with constant density and shear-wave velocity equal to zero. The model was first sub-sampled from 3400x700 (horizontal x vertical) cells at 5 m cell size to 850x175 cells at 20m cell size. The top 400 m of the model was then stripped off, corresponding to removal of most of the water layer. This left a model with 850x155 cells. Finally, a variable near-surface was added to the top 100 m of the model to

simulate an earth that would require application of static corrections. The near-surface perturbation was created by smoothing a random model using a smoother with dimensions 200m x 60m. The velocity perturbation range of this model was -500 m/s to 0 m/s. These modifications make the model more challenging for inversion, because there are significant errors in the starting model, especially near the sources and receivers.

All computations were performed using a 2D finite-difference time-domain acoustic simulator. The simulator is second order in time and 14^{th} order in space. Perfectly matched layer (PML) boundary conditions (Marcinkovich et al., 2003) were used at the sides and bottom of the model. A free-surface boundary condition was used at the top of the model so that our inversions use not just primary reflections and interbed multiples, but also surface-related multiples.

The data to be inverted in these examples were generated using the same simulator that was used to perform the inversion. A uniform fixed-spread geometry was simulated having the following parameters:

Number of receivers = 850 Receiver interval = 20 m
Number of sources = 850 Source interval = 20 m
Receiver depth = 50 m Source depth = 30 m
Source wavelet = 15 Hz Ricker wavelet
Trace length = 8 sec

Random noise with a uniform frequency spectrum was added to the simulated measured data before encoding or inversion. The noise level was chosen to mimic the S/N spectrum (with respect to ambient noise) of an actual OBC seismic survey from the Gulf of Mexico.

After adding random noise with a uniform spectrum to the data the S/N below 5 Hz and above 30 Hz is very low. We applied a Butterworth bandpass filter having both high and low cuts of 5 Hz and high and low slopes of 36 dB/octave to both the seismic data and to the source signatures. The 5 Hz low-cut frequency was chosen to reduce the weight of very noisy low-frequency components in the inversion. The 5 Hz high-cut frequency was chosen to mitigate the cycle-skipping problems in the inversion (Bunks et al., 1995); therefore, these examples actually only represent the initial phase of inversion, with later phases having increased values of the high-cut frequency. The only other pre-processing applied to the measured data was to mute the ambient noise above the first arrival.

Although 850 shots were acquired, not all shots are used in all the inversions, since these data are highly over-sampled for the purposes of inverting the low frequencies that are used in these examples. For conventional FWI, we found that only 50 sources (340 m source interval) were necessary for avoiding operator aliasing artifacts in the inverted model. Using more sources for FWI increases the cost of FWI while yielding no improvement in the inverted image. On the other hand, we found that it is slightly advantageous to use all sources for ESSFWI. Therefore, when comparing FWI to ESSFWI, we use 50 sources for FWI and all 850 sources for ESSFWI so that both methods were run in the most efficient manner possible.

The initial model (shown in Figure 2b) was created by smoothing the true model (Figure 2a), without the near-surface perturbation. The smoothing was performed in the slowness domain in an attempt to preserve traveltimes. The size of the smoothing operator increased from 0.5 km x 1.0 km (vertical x horizontal) at zero depth to 2.0 km x 4.0 km at the bottom of the model. The size of the smoothing operator was chosen to be the largest that would produce good convergence given the S/N of the low frequencies in the measured data.

The gradient of the objective function was computed using the adjoint-state method (Tarantola, 1984). The search direction was computed using the Hestenes-Stiefel conjugate gradient algorithm (Nocedal et al., 2006). We multiplied the search direction vector by the square root of depth plus 100m to approximately account for spherical divergence.

We used a line-search method to find an updated model. The line search evaluated the objective function for five models that were produced by adding uniformly scaled versions of the search direction to the model from the previous iteration. The velocities in these line search models were clipped to be between 1,000 m/s and 5,000 m/s. We then selected the model that gave the lowest objective function value as the current iteration's updated model. Finally, we adjusted the line-search scale such that the picked model would have been in the center of the line search. This updated scale was used to perform the line search during the next iteration.

We tested several different encoding schemes, but found that random phase encoding (Romero, 2000) provided the best convergence rates. We normalized the random phase codes so that they all had the same total power. We tested several different lengths for the encoding function, and found that a code having only one sample gave the most efficient inversion. Note that using a normalized random phase code having only one sample is equivalent to randomly multiplying the shot gathers by either one or minus one. We changed e_n's between iterations by simply changing the seed of the uniform random number generator used for generating the random phase code. Using short codes with T_c=1 maximizes the $T_s/(T_s+T_c-1)$ factor in

Equation (3) to a value of one, yielding larger overall efficiency gains.

Since ESSFWI involves random encoding, the inverted result can depend on the encoding functions chosen during an inversion run. In particular, since we generate our encoding functions using a random number generator, the inversion result can depend on the seed provided to the random number generator. We found that, when ESSFWI converges, there are only very subtle differences between inversions from different random number generator seeds.

Example

In this example we demonstrate ESSFWI's computational efficiency. FWI and ESSFWI were performed using the measured data and starting model discussed above. The only difference between these tests is whether the encoded simultaneous-source method was employed. The ESSFWI inversion was performed with a code length of one.

Comparing Figure 2c (FWI inverted model for iteration 300) with Figure 2d (ESSFWI inverted model for iteration 300) shows that ESSFWI yields an inverted image that is very close to that produced by FWI. Both images are close to the true model (Figure 2a), except that these inverted images do not resolve the smallest scale features of the true model due to the limited bandwidth of the measured data. Both inversions are considerably less accurate below 2.5 km. Some of this inaccuracy can be explained by poor illumination of the steep dips in this region. However, the inaccuracy of the inversion near the center of the model is more likely caused by inaccuracy of the starting model. Note that the ESSFWI inversion was computed with one-fiftieth the compute effort of the FWI inversion.

The main difference between the conventional FWI and ESSFWI images is the presence of some low-amplitude and short-wavelength noise in the ESSFWI images. This noise is caused by cross talk between the encoded sources and is much stronger in early iterations.

Conclusions

We have demonstrated, using 2D acoustic synthetic-data inversions, a factor of fifty increase in efficiency of full-wavefield inversion by using ESSFWI. Our testing indicates that both encoding the sources and changing the code between iterations are very important not only for increasing efficiency, but also for achieving a satisfactory inversion from ESSFWI.

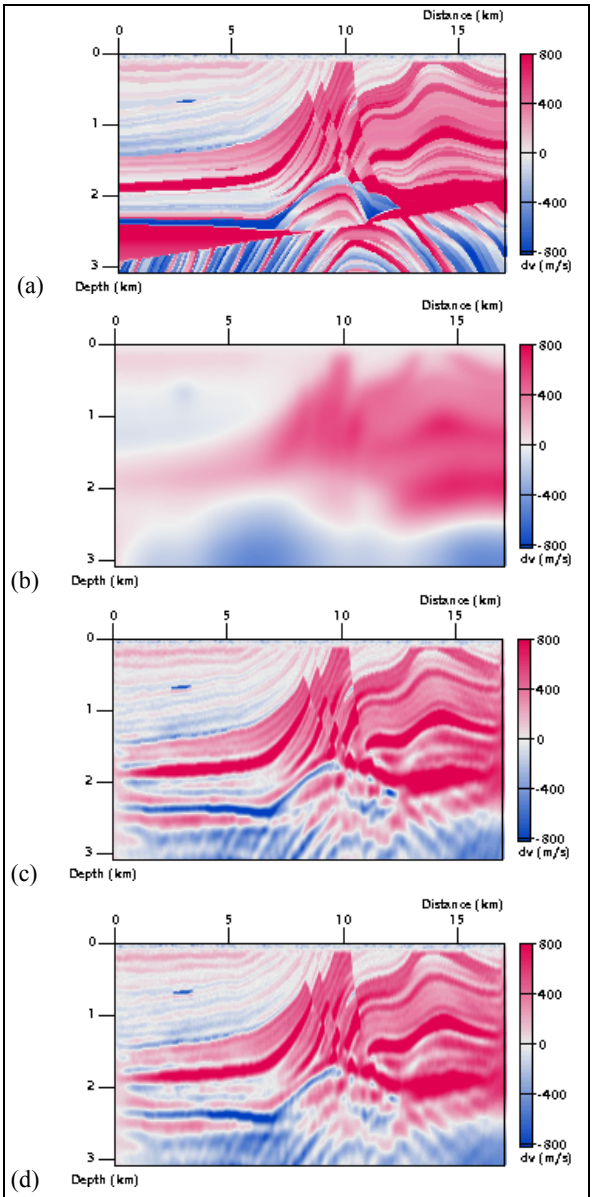

Figure 2: Comparison of FWI and ESSFWI. The true model (a), was smoothed to make the starting model (b). Conventional FWI resulted in (c) and ESSFWI is shown in (d). Result (d) was obtained with 50x less computational effort than the conventional result (c). The color scale is velocity relative to a very smooth background model.