Partial fast Fourier transform (PFFT) to improve the computer efficiency of hybrid-domain FWI

Ken Xu (SEIMAX Technologies) and George A. McMechan (UT Dallas)

Summary

To improve the signal to noise ratio (S/N) and the robustness of the inversion in the hybrid-domain (combined with frequency and time domains) FWI, generally a large number of frequencies are treated as a single frequency group before cycle skipping occurs. However, with this approach, discrete Fourier transforms (DFT) becomes costly for the extraction of multiple frequencies at every propagation time step. We propose a new partial fast Fourier transform (PFFT) to improve the computational efficiency of the multi-frequency extraction for inversion, which does not need the enormous requirements for disk storage and input/output (I/O) that the Fast Fourier Transform (FFT) method requires.

Introduction

Seismic full-waveform inversion (FWI) has becomes an important tool to recover high-resolution subsurface velocity models. FWI can be implemented in the time domain, as discussed by Tarantola (1984), or the frequency domain, as discussed by Pratt *et al.* (1998). Both time and frequency domains have inherent advantages and disadvantages for FWI as discussed by Virieux and Operto (2009), such as (1) time windowing can be easily implemented in the time domain; while discrete low-to-high frequency selection can be implemented naturally in the frequency domain, (2) time-domain modeling is very fast for a single shot, and is also easily parallelized by distributing shots across nodes, however large disk storage and input/output (I/O) is needed for the forward wavefields. While the memory storage of a single, or a few, complex-valued wavefield slices in the frequency domain is much smaller, the computational costs of frequency-domain modeling can be very high for large 2D or for 3D problems.

To combine the advantages in both domains for inversion, some researchers, including Sirgue *et al.* (2008), Etienne *et al.* (2010), and Xu and McMechan (2014) proposed a hybrid-domain FWI. In this hybrid algorithm, the wavefield extrapolations are performed using a time-domain numerical modeling engine (finite difference, finite element, or pseudo spectral), which is parallelized with MPI to distribute shot simulations over cluster nodes. The inversion is implemented in the frequency domain, and the discrete frequency wavefield slices are usually calculated at each extrapolation time step using a discrete Fourier Transform (DFT). Thus, the huge disk storage of the whole time-domain source forward-propagated wavefields, at all subsurface grid points for inversion, is greatly reduced to that for only a few frequency snapshots usually stored in memory space.

To improve the signal-to-noise ratio (S/N) and the robustness of FWI for complicated wavefields in the hybrid-domain (or frequency-domain) inversion, it is preferred to treat more frequencies as one frequency group before cycle skipping occurs, as discussed by Brossier *et al*. (2009), and Virieux and Operto (2009). However, to extract more frequency wavefield snapshots simultaneously in one modeling step in the hybrid-domain FWI, DFT becomes more expensive, since each frequency needs a separate DFT integral calculation at each time step. To extract multiple frequencies, Fast Fourier transformation (FFT) is a more efficient alternative for DFT. However, FFT requires that all time-domain wavefields be available before calculation, generating enormous requirements for disk storage and I/O, similar to that of time-domain FWI. Thus, FFT is generally impractical for the hybrid-domain FWI.

To improve the computational efficiency of the hybrid-domain FWI, we implement a new partial fast Fourier Transform (PFFT) to extract more frequency slices efficiently. PFFT inherits the computational efficiency of the FFT, but without the huge disk storage and I/O requirements.

Theory

In this paper, we propose PFFT as a new approach to substitute for DFT in the hybrid-domain FWI to extract any more frequency slices during the time-domain modeling.

PFFT to improve the computer efficiency of hybrid-domain FWI

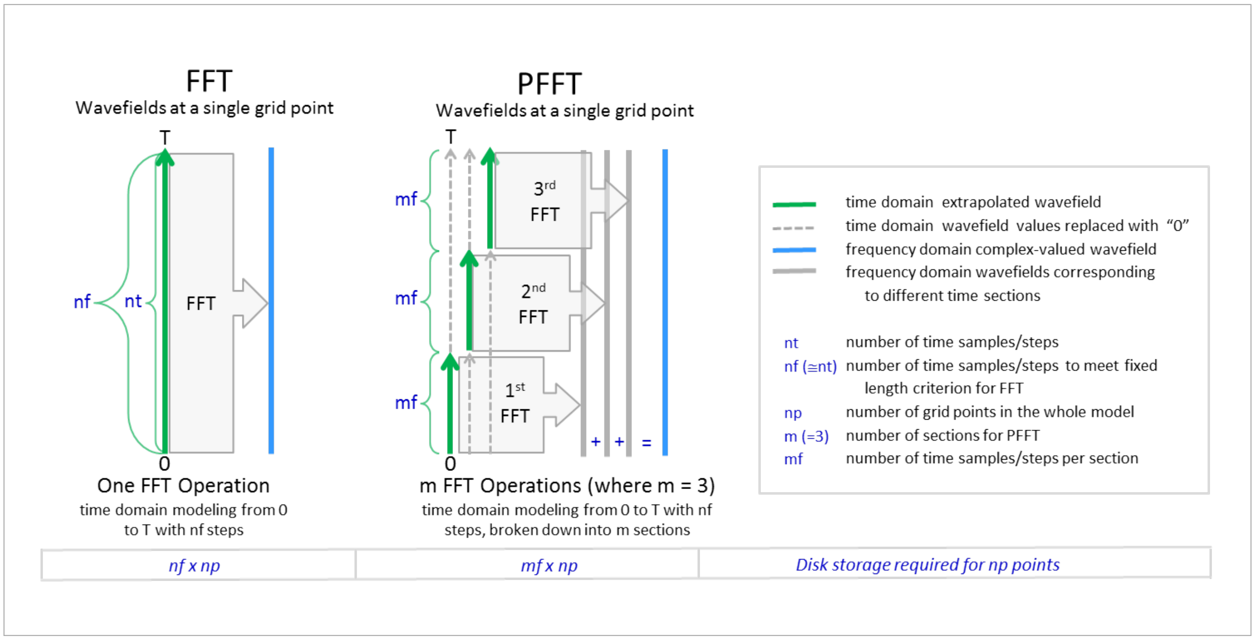

Figure 1: PFFT with multiple FFT operations is equivalent to one complete FFT.

PFFT breaks a time-domain wavefield into sections and performs a separate FFT operation corresponding to each section. To save the total storage of the extrapolated wavefields, only the extrapolated time steps within each time section is needed for each incremental FFT operation. Those time samples within the section are filled with the corresponding extrapolated wavefield values, while the other time samples are filled with zeros for each FFT operation. According to the linearity of Fourier transform, the sum of all the frequency results after multiple FFT operations for PFFT is equivalent to the corresponding frequency value after one FFT operation of the complete time-domain wavefield.

As illustrated in Figure 1, three FFT operations for PFFT get the same frequency values as a complete FFT does on the same time-domain wavefield. For PFFT, the whole wavefield is divided into m (=3) different sections. When the FFT operation is performed on each section, only that portion of the wavefield needs to be available. In comparison, for a complete FFT, the entire wavefield needs to be available in memory or disk storage. In Figure 1, since the wavefield has been divided into 3 sections for PFFT, its memory or disk storage is reduced by two thirds. And since the huge amount of disk storage for a complete FFT is not needed, the memory space is usually sufficient to contain the partial wavefields for PFFT. On the other hand, although the computational cost is increased for the multiple FFT operations for PFFT, it is still more efficient than DFT when more simultaneous frequencies as one group are involved in inversion.

To compare disk or memory storage space for PFFT and FFT, assume the entire time-domain wavefield at one model grid point has nt time samples, or time steps. To meet the fixed-length criterion of FFT (e.g., the power of 2), some zeros are added up in the end to make the length to be nf, where $nf \geq nt$. Since some advanced FFT software (Frigo and Johnson, 2005) relaxes the fixed-length criterion of FFT to be the power of the product of quite a few prime numbers (e.g., 2, 3, 5, 7, 11, and 13), nf can be set to be very close to nt (assume $nf \approx nt$). For PFFT, the nf ($\approx nt$) time samples are divided into m consecutive sections with the same time length $mf = nf/m$. Assume there are np model grid points to record all time-domain forward propagated wavefields for inversion, at which Fourier transformation is performed to extract frequency slices.

The complete FFT requires about $nf*np$ disk or memory storage space for all time domain wavefields at all grid points. By comparison, disk or memory storage for PFFT is greatly reduced to be approximately $1/m$ of that for the complete FFT, as $mf*np$, which can be adjusted small enough to be contained in memory. The $mf*np$ memory space is repeatedly filled up for one by one section through the all *m* consecutive sections, along the time steps of the wavefield propagation. After the memory space is filled up with one section, the FFT operation for PFFT is performed across all grid points, then the *mf**np* space is dumped and will be refilled for the next section. Note, even parts of time-domain values at a single grid point are filed with zeros, each FFT operation for PFFT still needs the same memory space as the complete FFT does. However, the memory space for each FFT operation is much small or negligible compared to the total disk or memory storage for all the wavefields, and can be used repeatedly for every grid point.

In the other side, to compare the computational costs for PFFT and DFT approaches, assume we want extract *ni* frequency wavefield slices at one grid point in the hybrid-domain FWI, the DFT needs about $O(ni*nf)$ arithmetical operations. Because the time-domain wavefields are real valued, the complete FFT needs $O(0.5*nf*\log(nf))$ arithmetical operations, and the PFFT needs $O(0.5*m*nf*\log(nf))$ arithmetical operations. From the above expressions for the arithmetical operations, as long as $m < 2*ni/\log(nf)$, PFFT is always more efficient than DFT. Table 1 gives some numerical comparisons of the computational costs between PFFT and DFT. It also demonstrates that because the cost of the PFFT is not related to the number of frequencies *ni* for inversion, when more frequencies are involved into the inversion, PFFT is more efficient than DFT.

nf (≈ nt)	ni	m (for PFFT)	$2*ni/\log(nf)$	PFFT's arithmetic operations	DFT's arithmetic operations
4000	5	5	2.78	36.0 k	20 k
4000	5	2	2.78	14.4 k	20 k
4000	10	5	5.55	36.0 k	40 k
4000	10	2	5.55	14.4 k	40 k
4000	20	5	11.10	36.0 k	80 k
4000	20	2	11.10	14.4 k	80 k

Table 1. The computational costs for PFTT and DFT

Conclusion

For the hybrid-domain FWI, more simultaneous frequencies extracted as one frequency group before cycle skipping occurs, are usually beneficial for the stability and robustness of the inversion. However, current Fourier transform methods DFT and FFT have their inherent limitations or disadvantages for the extraction of frequency slices in the hybrid-domain FWI. Here, we propose a new Partial Fast Fourier Transform method (PFFT) to inherit the computational efficiency of the FFT, but without the requirement of the huge disk storage or I/O. It needs only a fraction of the storage of the whole wavefields, compared with FFT. Since the computational cost of PFFT is not related to the number of frequencies extracted for inversion, it has an obvious speed advantage over DFT, when more simultaneous frequencies are involved in inversion.

Acknowledgements

Xu. K appreciates the colleague: Kim Human to polish the paper. Xu K. also thanks SEIMAX Technologies LP for permission to publish this abstract.

References

*Brossier, R., S. Operto, and J. Virieux, 2009, Seismic imaging of complex onshore structures by 2D elastic frequency-domain full-waveform inversion: Geophysics, **74**, no. 6, WCC105–WCC118.*

Etienne, V., J. Virieux, S. Operto, and Y. Jia, 2010, Computational issues and strategies related to full waveform inversion in 3D elastic media: Methodological developments: 81st Annual International Meeting, SEG, Expanded Abstracts, 1050–1054.

*Matteo Frigo and Steven G. Johnson, 2005, The Design and Implementation of FFTW3: Proceedings of the IEEE, **93**, 216–231.*

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

Sirgue, L., J. T. Etgen, and U. Albertin, 2008, 3D frequency domain wave- form inversion using time domain finite difference methods: 70th Annual International Conference and Exhibition, EAGE, Extended Abstracts, F022.

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

*Virieux, J., and Operto, S., 2009, An overview of full-waveform inversion in exploration geophysics: Geophysics, **74**, no. 6, WCC127-WCC152.*

*Xu, K., and McMechan, G., 2014,2D frequency-domain elastic full-waveform inversion using time-domain modeling and a multistep-length gradient approach: Geophysics, **79**, no. 2, R41-R53.*