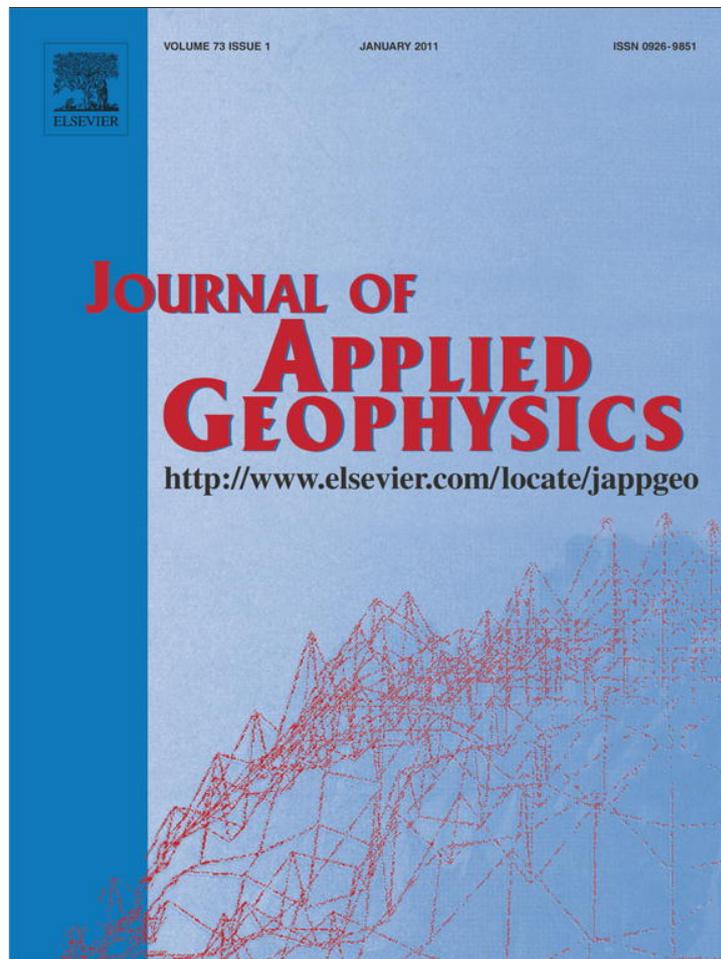


Provided for non-commercial research and education use.
Not for reproduction, distribution or commercial use.



(This is a sample cover image for this issue. The actual cover is not yet available at this time.)

This article appeared in a journal published by Elsevier. The attached copy is furnished to the author for internal non-commercial research and education use, including for instruction at the authors institution and sharing with colleagues.

Other uses, including reproduction and distribution, or selling or licensing copies, or posting to personal, institutional or third party websites are prohibited.

In most cases authors are permitted to post their version of the article (e.g. in Word or Tex form) to their personal website or institutional repository. Authors requiring further information regarding Elsevier's archiving and manuscript policies are encouraged to visit:

<http://www.elsevier.com/copyright>



Contents lists available at SciVerse ScienceDirect

Journal of Applied Geophysics

journal homepage: www.elsevier.com/locate/jappgeo

Seismic imaging based on spectral differentiation matrix and GPU implementation

Jing-Bo Chen*, Guo-Feng Liu, Hong Liu

Key Laboratory of Petroleum Resources Research, Institute of Geology and Geophysics, Chinese Academy of Sciences, P. O. Box 9825, Beijing 100029, China

ARTICLE INFO

Article history:

Received 16 July 2011

Accepted 5 January 2012

Available online 14 January 2012

Keywords:

Seismic migration

Imaging

Spectral differentiation

GPU

ABSTRACT

Finite-difference depth migration based on one-way wave equation uses second-order, fourth-order, or other finite-order approximations for spatial derivatives. These finite-order approximations often lead to spatial dispersion errors and low accuracy. To avoid these errors, smaller mesh spacings are used, which results in huge increase in computation cost. In this paper, we develop a new spectral differentiation matrix method for approximating spatial derivatives. The approximation of spectral differentiation matrix is of infinite-order, and therefore avoids dispersion errors and increases accuracy by using a smaller number of grid points. Recently-developed GPU computational technology is well suited to the seismic migration based on frequency-domain wavefield-continuation methods because such approaches can be sufficiently decoupled. We present a frequency-domain wavefield-continuation algorithm based on spectral differentiation matrix, and realize GPU implementation of this new algorithm on Marmousi model.

© 2012 Elsevier B.V. All rights reserved.

1. Introduction

Migration is the most widely used seismic imaging method. Migration methods can be classified into two categories: integral methods and wavefield-continuation methods (Biondi, 2006). The wavefield-continuation method consists of two steps: wavefield continuation and construction of the subsurface structure by using imaging conditions. The migration methods differ in their approach to wavefield continuation (Claerbout, 1985; Gazdag and Sguazzero, 1984).

Finite-difference wavefield-continuation is an important method because it can naturally account for both vertical and lateral velocity variations. Usually, finite-order finite-difference approximation is used to evaluate the spatial derivatives. These finite-order approximations often produce numerical dispersions and have limited accuracy. To reduce these numerical dispersions and increase accuracy, sufficiently small spatial samplings are needed, which significantly increases computational cost.

Spectral differentiation matrix method is an infinite-order approximation to the spatial derivatives (Chen, 2006; Chen and Liu, 2008; Fornberg, 1987). This method avoids numerical dispersion, and therefore relatively large spatial samplings can be used to reduce computational cost while maintaining high accuracy. The idea of this method was mentioned in (Chen et al., 2007), but was not discussed in detail. In this paper, we will explore this method more thoroughly.

In recent years, GPU computation based on CUDA programming has become a very promising computing technology. Due to its

inborn feature of parallel computing, GPU computation has great advantages over traditional parallel computing based on PC-clusters. Frequency-domain prestack depth migration methods based on one-way equations are well suited to GPU computation because of the shot and frequency decoupling properties possessed by these methods. Therefore, we can use GPU implementation to speed up the computation of the depth migration method based on spectral differentiation matrix.

2. One-way wave equation and its discretizations based on spectral differentiation matrix

Consider the acoustic one-way wave equation in frequency-wavenumber domain

$$\frac{\partial u}{\partial z} = i \sqrt{\frac{\omega^2}{c^2} - k_x^2} u, \quad (1)$$

where ω is the circular frequency, k_x is the wavenumber in the x -direction, i is the unit of imaginary numbers, and $c(x, z)$ is the velocity.

Expanding the square root in Eq. (1) by second-order continued fraction and transforming into space domain, we obtain the 45-degree equation

$$\left[1 + \left(\frac{c}{2\omega} \right)^2 \frac{\partial^2}{\partial x^2} \right] \frac{\partial}{\partial z} \bar{u} = \frac{ic}{2\omega} \frac{\partial^2}{\partial x^2} \bar{u}, \quad (2)$$

where $\bar{u} = \exp\{i \frac{\omega}{c} \Delta z\} u$. For details, see (Gazdag and Sguazzero, 1984; Lee and Suh, 1985).

* Corresponding author.

E-mail address: chenjb@mail.iggcas.ac.cn (J.-B. Chen).

Let $\bar{u}_{j,l}$ denote discrete values of the wavefield at grid points ($j\Delta x, l\Delta z$), i.e., $\bar{u}_{j,l} = \bar{u}(j\Delta x, l\Delta z)$, $j = 0, 1, 2, \dots, n_x$; $l = 0, 1, 2, \dots, n_z$. Here Δx and Δz are grid spacings, and n_x and n_z are the numbers of sampling along the x - and z -directions, respectively.

If we approximate the derivatives by the following second-order finite-difference methods:

$$\frac{\partial \bar{u}_{j,l+1/2}}{\partial z} \approx \frac{\bar{u}_{j,l+1} - \bar{u}_{j,l}}{\Delta z}, \quad (3)$$

$$\frac{\partial^2 \bar{u}_{j,l}}{\partial x^2} \approx \frac{\bar{u}_{j-1,l} - 2\bar{u}_{j,l} + \bar{u}_{j+1,l}}{\Delta x^2}, \quad (4)$$

then we obtain finite-difference discretization of Eq. (2)

$$\begin{aligned} \bar{u}_{j,l+1} + (\alpha - i\beta)(\bar{u}_{j-1,l+1} - 2\bar{u}_{j,l+1} + \bar{u}_{j+1,l+1}) \\ = \bar{u}_{j,l} + (\alpha + i\beta)(\bar{u}_{j-1,l} - 2\bar{u}_{j,l} + \bar{u}_{j+1,l}), \end{aligned} \quad (5)$$

where $\alpha = (\frac{c}{2\omega\Delta x})^2$ and $\beta = \frac{c\Delta z}{4\omega\Delta x^2}$.

The second-order finite-difference approximation in Eq. (5) usually results in dispersion errors. To reduce the dispersion errors, we can use smaller Δx , but this approach greatly increases the computational cost. Another approach to reducing dispersion errors is to use high-order approximations.

Note that the second-order approximation in Eq. (5) corresponding to a finite-difference matrix:

$$\frac{1}{\Delta x^2} \begin{pmatrix} -2 & 1 & 0 & \dots & 0 & 1 \\ 1 & -2 & 1 & \dots & 0 & 0 \\ 0 & 1 & -2 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & -2 & 1 \\ 1 & 0 & 0 & \dots & 1 & -2 \end{pmatrix}. \quad (6)$$

The second-order approximation in Eq. (5) can also be replaced by the following fourth-order approximation

$$\frac{\partial^2 \bar{u}_{j,l}}{\partial x^2} \approx \frac{-\bar{u}_{j-2,l} + 16\bar{u}_{j-1,l} - 30\bar{u}_{j,l} + 16\bar{u}_{j+1,l} - \bar{u}_{j+2,l}}{\Delta x^2}. \quad (7)$$

The fourth-order finite-difference approximation Eq. (7) corresponds to the following finite-difference matrix:

$$\frac{1}{12\Delta x^2} \begin{pmatrix} -30 & 16 & -1 & 0 & 0 & \dots & 0 & -1 & 16 \\ 16 & -30 & 16 & -1 & 0 & \dots & 0 & 0 & -1 \\ -1 & 16 & -30 & 16 & -1 & \dots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ -1 & 0 & 0 & 0 & 0 & \dots & 16 & -30 & 16 \\ 16 & -1 & 0 & 0 & 0 & \dots & -1 & 16 & -30 \end{pmatrix}. \quad (8)$$

We can continue to use higher finite-difference approximations and obtain the corresponding finite-difference matrices. But such finite-difference approximations only have finite-order of accuracy.

Now we present an infinite-order approximation by using the spectral differentiation matrix $D_{sm} = \{d_{p,q}\}$ ($p, q = 1, 2, \dots, n_x$) which is defined by

$$d_{p,q} = \begin{cases} \frac{1}{2} \left(\frac{2\pi}{n_x \Delta x} \right)^2 (-1)^{p+q+1} \frac{1}{\sin^2(\pi(p-q)/n_x)}, & p \neq q, \\ -\left(\frac{2\pi}{n_x \Delta x} \right)^2 \frac{2(n_x/2)^2 + 1}{6}, & p = q. \end{cases} \quad (9)$$

The spectral differentiation matrix is derived from trigonometric interpolation and exact calculation of derivatives, and possesses an accuracy of infinite order (Chen, 2006; Chen and Qin, 2001; Gottlieb et al., 1984). Fornberg (1987) proved that the spectral differentiation

matrix can be regarded as a limit of finite-difference approximations of derivatives as the order of accuracy tends to infinity. Therefore, in terms of accuracy, the spectral differentiation matrix approximation is the best choice. In terms of computational efficiency, second-order finite-difference approximation is the fastest because the resulting equation can be resolved by an efficient tri-diagonal solver. Since the spectral differentiation matrix is a full matrix, an algorithm based on LU decomposition is needed, which increases the computational cost. Fortunately, an LUGPU algorithm has been developed, which greatly improves the computational efficiency of LU decomposition by using GPU (Galoppo et al., 2005).

The spectral differentiation matrix corresponds to the following approximation to the second-order spatial derivative

$$\frac{\partial^2 \bar{u}_{j,l}}{\partial x^2} \approx (D_{sm} \bar{u}_l)_j, \quad (10)$$

where $\bar{u}_l = [\bar{u}_{1,l}, \dots, \bar{u}_{n_x,l}]^T$.

Based on Eq. (10), we obtain spectral-differentiation-matrix discretization of Eq. (2)

$$\bar{u}_{j,l+1} + (\bar{\alpha} - i\bar{\beta})(D_{sm} \bar{u}_{l+1})_j = \bar{u}_{j,l} + (\bar{\alpha} + i\bar{\beta})(D_{sm} \bar{u}_l)_j, \quad (11)$$

where $\bar{\alpha} = (\frac{c}{2\omega})^2$ and $\bar{\beta} = \frac{c\Delta z}{4\omega}$.

To obtain higher-degree equation, we can compose 45-degree equation with different coefficients. Using an optimization technique, Lee and Suh (1985) derived the coefficients up to 90-degree.

The 90-degree equation has the following form

$$\frac{\partial u}{\partial z} = i \frac{\omega}{c} \left(1 + \sum_{r=1}^4 \frac{a_r S}{1 + b_r S} \right) u, \quad (12)$$

where $S = -\frac{c^2 k_x^2}{\omega^2}$ and $a_1 = 0.000523275$, $b_1 = 0.994065088$, $a_2 = 0.014853510$, $b_2 = 0.919432661$, $a_3 = 0.117592008$, $b_3 = 0.614520676$, $a_4 = 0.367013245$, $b_4 = 0.105756624$.

Correspondingly, we can solve Eq. (12) by a splitting method. Eq. (12) can be decomposed into two successive equations:

$$\frac{\partial u}{\partial z} = i \frac{\omega}{c} u, \quad (13)$$

$$\frac{\partial u}{\partial z} = i \frac{\omega}{c} \sum_{r=1}^4 \frac{a_r S}{1 + b_r S} u. \quad (14)$$

Replacing ik_x by $\frac{\partial}{\partial x}$, Eq. (14) can be expressed in frequency-space domain as four successive equations:

$$\left[1 + \frac{b_r v^2}{\omega^2} \frac{\partial^2}{\partial x^2} \right] \frac{\partial}{\partial z} \bar{u} = \frac{ia_r c}{\omega} \frac{\partial^2}{\partial x^2} \bar{u}, \quad r = 1, 2, 3, 4. \quad (15)$$

Based on Eq. (10), we obtain spectral-differentiation-matrix discretization of Eq. (14)

$$\bar{u}_{j,l+1} + (\alpha_r - i\beta_r)(D_{sm} \bar{u}_{l+1})_j = \bar{u}_{j,l} + (\alpha_r + i\beta_r)(D_{sm} \bar{u}_l)_j, \quad (16)$$

where $r = 1, 2, 3, 4$, and $\alpha_r = \frac{b_r c^2}{\omega^2}$ and $\beta_r = \frac{a_r c \Delta z}{2\omega}$.

3. Numerical experiments

In this section, we compute the impulse responses for the two schemes (5) and (11), and make a comparison between them. We consider a medium with $v = 2500$ m/s. The spacings are taken as $\Delta x = 40$ m and $\Delta z = 20$ m, and the numbers of horizontal and vertical sampling are $n_x = 128$ and $n_z = 128$, respectively. A Ricker wavelet of peak frequency 25 Hz is placed at (0m, 0m).

Plots (a) and (b) in Fig. 1 show the impulse responses of Eqs. (5) and (11), respectively. For this large Δx of 40 m, the result of second-order finite-difference approximation to x-derivative shows severe dispersions, while the result of the spectral differentiation matrix approximation exhibits no dispersion within the propagation angle of 45° . However, the computation time of Eq. (11) is longer than that of Eq. (5). To achieve the effect of Eq. (11) by using Eq. (5), we have to use a much smaller sampling interval Δx , which leads to a greatly increased number of sampling. Plot (c) shows the impulse response of Eq. (5) with $\Delta x = 10$ m and $n_x = 512$, and the result is better than that of Eq. (5) with $\Delta x = 40$ m and $n_x = 128$, but the computation time is greatly increased, which is also much longer than that of Eq. (11) with $\Delta x = 40$ m and $n_x = 128$. The computation parameters for plots in Fig. 1 are summarized in Table 1.

4. GPU implementation

In this section, we apply scheme (16) to Marmousi model based on GPU. Fig. 2(a) shows the Marmousi model.

GPU (Graphic Processing Unit) computation is a recently-developed computational technology based on CUDA programming (NVIDIA, 2008). CUDA provides a direction $\langle\langle Dg, Db \rangle\rangle$ to achieve parallel computation of a user-written kernel, where Dg determines the number of thread blocks $N(Dg)$ and Db determines the number of threads of each block $N(Db)$.

GPU computation is well suited to the seismic migration based on frequency domain wavefield-continuation methods because such approaches can be sufficiently decoupled.

First, we consider shot-decoupling. The synthetic data for Marmousi model consists of independent 240 shots. Therefore, the task is decoupled into 240 independent sub-tasks.

Second, we further consider frequency-decoupling. For each shot, the computation consists of M frequencies. Therefore, each shot is further decoupled into M independent sub-tasks.

In summary, the task of imaging Marmousi model using Eq. (16) can be decoupled into $240 \times N$ completely independent sub-tasks (threads).

Table 1
Computation parameters for plots in Fig. 1.

Plots in Fig. 1	a	b	c
Schemes	Eq. (5)	Eq. (11)	Eq. (5)
Sampling interval	$\Delta x = 40$ m	$\Delta x = 40$ m	$\Delta x = 10$ m
Number of sampling	$n_x = 128$	$n_x = 128$	$n_x = 512$
Computation time	$T = 10.8$ s	$T = 81.2$ s	$T = 194.3$ s

To achieve GPU implementation of our migration scheme (16), we write a program `Spectral-Diff-Matrix-Migration` (parameters) to compute Eq. (16) for each shot data and each frequency, and this program is called a kernel.

Then the following direction can achieve GPU implementation:

```
intmain()
{ Spectral-Diff-Matrix-Migration<<<Dg, Db >>> (parameters)}
```

Here $N(Dg) = 240$ and $N(Db) = M$. The above direction will run the kernel

`Spectral-Diff-Matrix-Migration` (parameters)

$240 \times M$ times in a parallel way, in which each thread block computes one shot data and each tread of the block computes one frequency.

Fig. 2(b) shows the migration result of Eq. (16) on Marmousi model based on GPU implementation. The migration result is satisfactory, and the faults, anticlines, and high-velocity salt body are both clearly imaged. Our GPU implementation is carried out on NVIDIA Tesla C 1060. Because of memory limitation, we only use frequency parallelism. Compared to CPU with peak frequency 2.5 G, Tesla C 1060 can achieve an improvement of computational efficiency by a factor of 30 times.

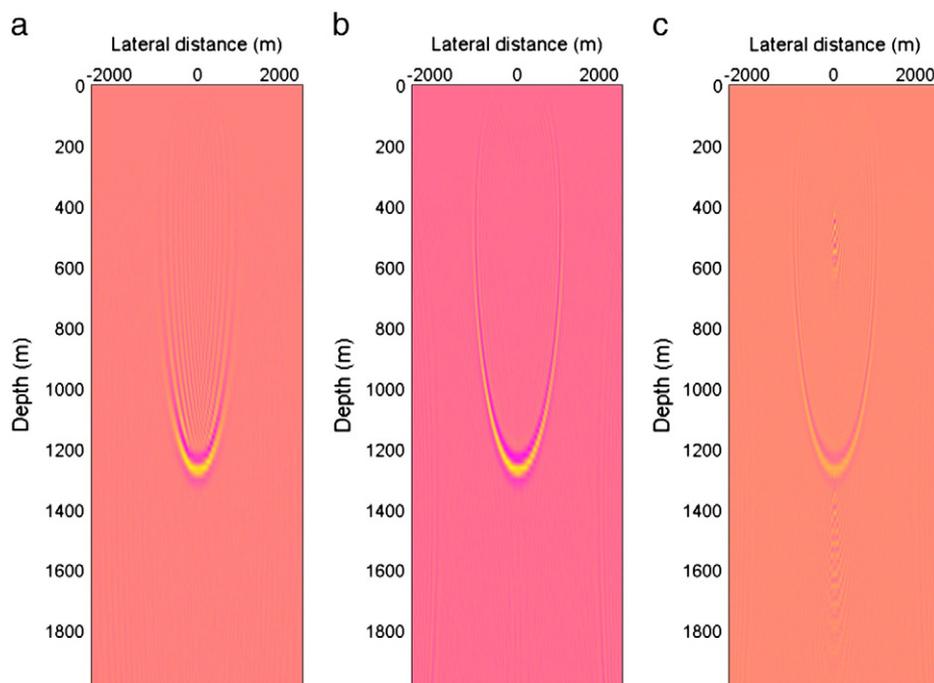


Fig. 1. Impulse responses of schemes (5) and (11). (a) Scheme (5) with $\Delta x = 40$ m and $n_x = 128$. (b) Scheme (11) with $\Delta x = 40$ m and $n_x = 128$. (c) Scheme (5) with $\Delta x = 10$ m and $n_x = 512$.

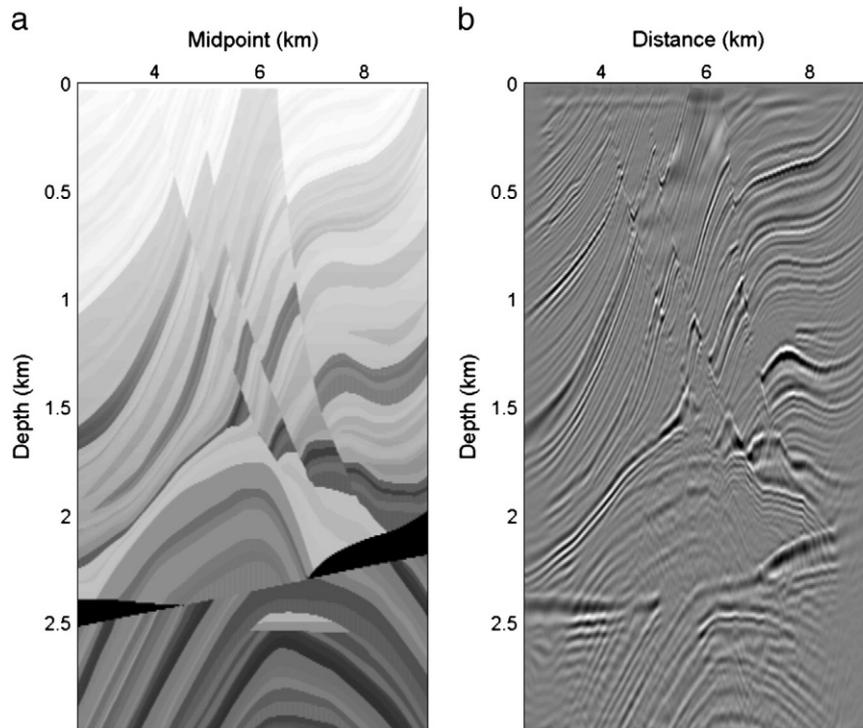


Fig. 2. (a) Marmousi model. (b) Migration result of scheme (16).

5. Discussions on 3D implementation

With respect to 3D implementation of finite-difference migration, there are two choices. One choice is to solve the 3D equation directly. However, this choice leads to prohibitively high computational cost (Claerbout, 1985). The second choice is to use a directional-splitting strategy. Instead of solving the 3D equation directly, this method solves two or more 2D equations and uses some kind of correction

technique to compensate for the directional-splitting errors (Li, 1991).

We use the second choice to implement 3D spectral differentiation matrix migration based on GPU. To this aim, our work consists of two steps. First, we implement 3D second-order finite-difference migration based on GPU. Second, we try to incorporate the result in the first step into LUGPU (Galoppo et al., 2005) and develop an efficient algorithm to implement 3D spectral differentiation matrix migration based on GPU. At present, we have finished the task in the first step. Implementation of 3D second-order finite-difference migration based on GPU has been carried out on SEG/EAGE slat model. Fig. 3 shows two crossline sections. The work in the second step is currently underway. We feel that the key is how to make a combination of migration-GPU and LUGPU. A lot of work remains to be done in this regard.

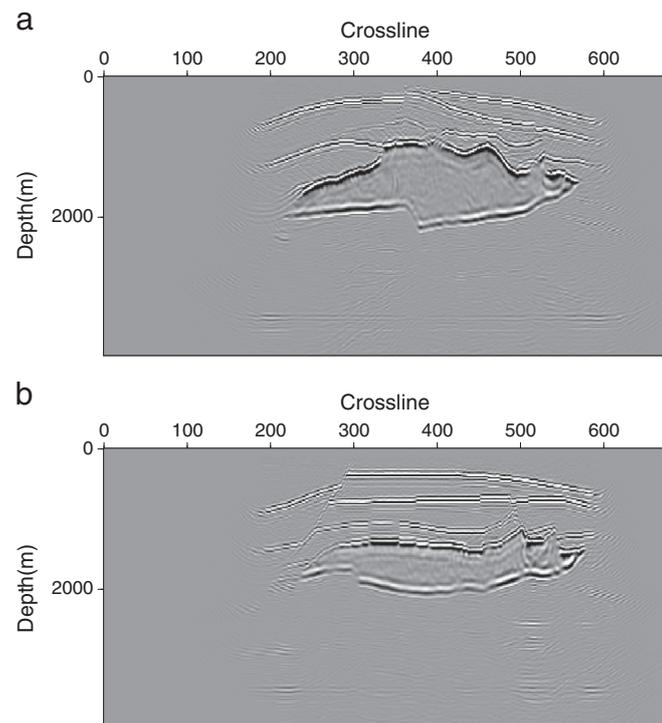


Fig. 3. Migration results of 3D salt model for inline 300 (left) and inline 400 (right).

6. Conclusions

In this paper, we develop a new migration method based on spectral differentiation matrix. This method suffers no dispersion errors resulting from the approximation of horizontal spatial derivatives, and therefore, large horizontal spatial intervals can be used. This method with large horizontal spatial interval is more efficient than the second-order finite-difference approximation with a much smaller horizontal spatial interval. We also achieve GPU implementation of this new migration method based on NVIDIA Tesla C 1060 GPU on Marmousi model, and this implementation realizes a 30 times of improvement of computational efficiency in comparison with CPU with a peak frequency of 2.5G.

Acknowledgements

We thank the anonymous reviewers for valuable suggestions. This work is supported by the National Natural Science Foundation of China under grant Nos. 40830424 (key project), 40974074 and 40774069.

References

- Biondi, B.L., 2006. 3-D Seismic imaging. Society of Exploration Geophysicists.
- Chen, J.B., 2006. Symplectic and multisymplectic Fourier pseudospectral discretizations for the Klein-Gordon equation. *Letters in Mathematical Physics* 75, 293–305.
- Chen, J.B., Liu, H., 2008. Modal expansion of one-way operators based on spectral differentiation matrix. *Applied Mathematics and Computation* 206, 193–197.
- Chen, J.B., Qin, M.Z., 2001. Multisymplectic Fourier pseudospectral method for the nonlinear Schrödinger equation. *Electronic Transactions on Numerical Analysis* 12, 193–204.
- Chen, J.B., Liu, H., Zhou, D.G., 2007. A Hamiltonian framework for wavefield depth continuation in seismic imaging. *Wave Motion* 44, 385–394.
- Claerbout, 1985. *Imaging the Earth's interior*. Blackwell Scientific Publications Inc.
- Fornberg, B., 1987. The pseudospectral method: comparisons with finite differences for the elastic wave equation. *Geophysics* 52, 483–501.
- Galoppo, N., Naga, K., Govindaraju, N.K., Henson, M., Manocha, D., 2005. LU-GPU: Efficient Algorithms for Solving Dense Linear Systems on Graphics Hardware. <http://gamma.cs.unc.edu/LU-GPU/lugpu05.pdf>2005.
- Gazdag, J., Sguazzero, P., 1984. Migration of seismic data by phase shift plus interpolation. *Geophysics* 49, 124–131.
- Gottlieb, M.Y., Hussaini, M., Orszag, S.A., 1984. Theory and applications of spectral methods, in *Spectral methods for partial differential equations*. In: Voigt, R.G., Gottlieb, D., Hussaini, M.Y. (Eds.), SIAM-CBMS, Philadelphia, pp. 1–54.
- Lee, M.W., Suh, S.Y., 1985. Optimization of one-way wave equations. *Geophysics* 50, 1634–1637.
- Li, Z., 1991. Compensating finite-difference errors in 3-D migration and modeling. *Geophysics* 56, 1650–1660.
- NVIDIA, 2008. *A guide to programming with CUDA (version, 2.0)*.