

Reverse time migration of 3D vertical seismic profile data

Ying Shi¹ and Yanghua Wang²

ABSTRACT

Reverse time migration (RTM) has shown increasing advantages in handling seismic images of complex subsurface media, but it has not been used widely in 3D seismic data due to the large storage and computation requirements. Our prime objective was to develop an RTM strategy that was applicable to 3D vertical seismic profiling (VSP) data. The strategy consists of two aspects: storage saving and calculation acceleration. First, we determined the use of the random boundary condition (RBC) to save the storage in wavefield simulation. An absorbing boundary such as the perfect matching layer boundary is often used in RTM, but it has a high memory demand for storing the source wavefield. RBC is a nonabsorbing boundary and only stores the

source wavefield at the two maximum time steps, then repropagates the source wavefield backwards at every time step, and hence, it significantly reduces the memory requirement. Second, we examined the use of the graphic processing unit (GPU) parallelization technique to accelerate the computation. RBC needs to simulate the source wavefield twice and doubles the computation. Thus, it is very necessary to realize the RTM algorithm by GPU, especially for a 3D VSP data set. GPU and central processing unit (CPU) collaborated parallel implementation can greatly reduce the computation time, where the CPU performs serial code, and the GPU performs parallel code. Because RBC does not need the same huge amount of storage as an absorbing boundary, RTM becomes practically applicable for 3D VSP imaging.

INTRODUCTION

The imaging of vertical seismic profiling (VSP) can effectively identify small geologic objects and precisely describe reservoirs and hydrocarbon deposits. VSP often places receivers at various depths in a vertical borehole, to record seismic signals generated from a source at the earth's surface. Therefore, near-surface geology has less influence on the wavefield than a surface seismic survey. It usually leads to relatively high signal-to-noise ratio and high resolution of VSP data. For example, VSP acquisition can obtain sufficient subsalt information, by placing a development well in the vicinity of a subsalt area of interest, to receive the reflection waves of the target geologic bodies (Burch et al., 2010). In this area, a surface seismic survey usually cannot receive waves that have rays propagated through the target. The prime objective of this paper is to develop the reverse time migration (RTM) method to be applicable to 3D VSP data.

VSP migration algorithms include single-shot record inversion (Harwijanto et al., 1987), depth imaging by wavefield extrapolation (Amundsen, 1993), interferometric migration using extrapolated VSP Green's functions (Xiao and Schuster, 2009), and the use of image point transformation (Cosma et al., 2010), etc. These algorithms can image VSP data well, but the RTM method has its own advantage for complex structures with a strong lateral velocity variation. Because the input data volume for VSP is less than that for the surface seismic data, the RTM algorithm is especially attractive for VSP depth imaging in an area with complex geology (Neklyudov and Borodin, 2009).

RTM was proposed in 1983 (Baysal et al., 1983; McMechan, 1983; Whitmore, 1983) and has shown increasing advantages in handling seismic images of complex subsurface media in the recent decade. Developments include the extensions from isotropic models to anisotropic models (Zhang et al., 2011; Xie et al., 2014) and from the acoustic case (Zhang and Sun, 2009) to the elastic case (Yan and

Manuscript received by the Editor 12 May 2015; revised manuscript received 25 August 2015; published online 16 November 2015.

¹Northeast Petroleum University, School of Earth Science, Science and Technology Innovation Team on Fault Deformation, Sealing and Fluid Migration, Daping, China. E-mail: shiying@nepu.edu.cn.

²Imperial College London, Centre for Reservoir Geophysics, Department of Earth Science and Engineering, London, UK. E-mail: yanghua.wang@imperial.ac.uk.

© 2015 Society of Exploration Geophysicists. All rights reserved.

Xie, 2012; Nguyen and McMechan, 2015). Meanwhile, Q -compensated RTM also has been developed (Deng and McMechan, 2008; Zhu et al., 2014). However, it has not been used that widely in 3D seismic data because of the large storage and computation requirements.

In this paper, we attempt to develop a strategy so as to make the RTM scheme be applicable to 3D VSP data. We will investigate the following two aspects: storage saving and calculation acceleration.

First, we discuss the use of the random boundary condition (RBC) to save the storage needed in wavefield simulation. For 3D RTM, it is not feasible to save the source (or receiver) wavefields at all times. For example, an optimal checkpoint method proposed by Symes (2007) can reduce the storage needed in the 2D case, but the storage requirement is still a serious problem for 3D RTM. Storing fewer time samples, as long as they satisfy the sampling law, is practically an easy scheme (Sun and Fu, 2013). Data compression and reconstruction can also be a good approach to tackle the issue, whereby these two schemes can always be applied in conjunction with any other strategy, including the RBC strategy adopted in this paper. In the RBC proposed by Clapp (2009), storage is reduced at the expense of additional computation. It stores source wavefields only at the two maximum time steps and repropagates the source wavefield backward at every time step, whenever they are needed to correlate the receiver wavefield. When using RBC (in the model), random noise (in the data) will be suppressed using the nonlocal means (NLM) method (Buades et al., 2005; Bonar and Sacchi,

2012). NLM is basically a weighted summation based on the similarity of 2D pixel windows or 3D pixel cubes. It is in contrast to conventional noise attenuation methods based on the spatial predictability of seismic signals (Wang, 1999).

Second, we discuss the use of the graphic processing unit (GPU) parallelization technique to accelerate the computation. When using RBC, the computation demand is almost doubled because it needs to simulate the source (or receiver) wavefields twice. GPU parallelization technology can efficiently reduce the computation cost (Schiemenz and Igel, 2013; Shin et al., 2014). For any intense computing, GPU and central processing unit (CPU) collaborated parallel implementation can greatly reduce the computation time, where the CPU performs serial code, and the GPU performs parallel code. For 3D VSP RTM, especially when using RBC in the model, multiple GPUs are needed.

This two-step strategy will be demonstrated by first using a 2D version of RTM and then a 3D RTM. Because of the use of RBC, which does not need the same huge amount of storage as conventional absorbing boundary conditions, such as the perfectly matched layer (PML) boundary condition (Berenger, 1994), RTM becomes an effective imaging technology that is applicable to 3D VSP data.

RTM OF VSP

For a standard VSP acquisition geometry, a source is located at the surface and a series of receivers is positioned in a vertical bore-

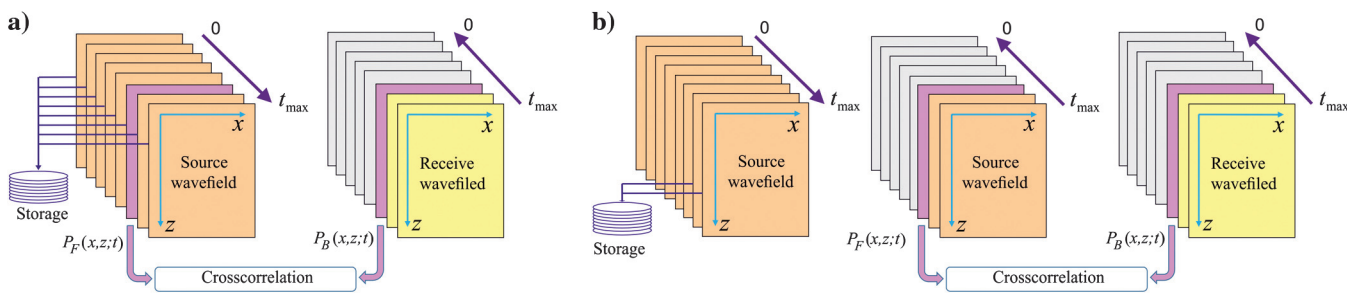
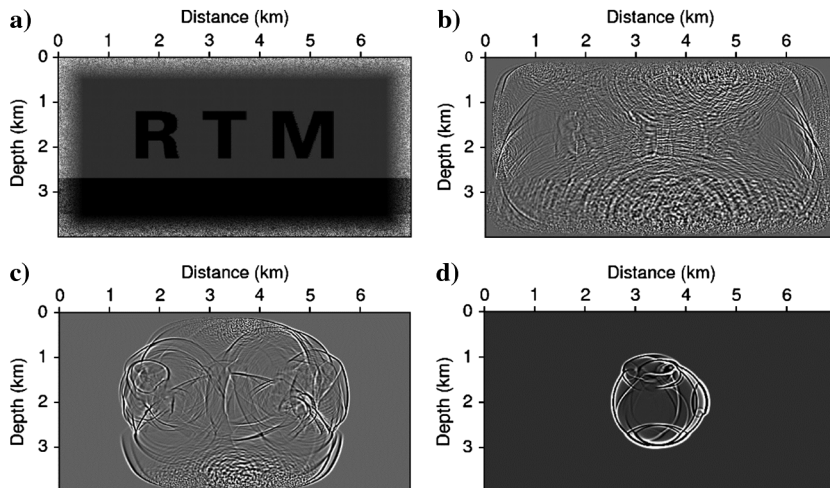


Figure 1. Schematic comparison between a standard RTM scheme and the one using the RBC in the model. (a) The standard RTM scheme, which needs to store the entire source wavefield for RTM imaging. (b) The RTM with RBC, in which only the wavefields at two time samples are stored. The source wavefield is then generated by backward propagation, using these two time samples, at the same pace as the back propagation of the receiver wavefield.

Figure 2. (a) A two-layer velocity model, surrounded by random boundaries. (b-d) Snapshots of the backward-propagated source wavefield at $t = 1.4, 0.85,$ and 0.4 s, respectively.



hole (Hardage, 2000). Because seismic receivers are placed in the subsurface, VSP has more of an abundance of rich wavefield information than does a surface seismic survey. Because VSP collects the information from the target layer, this geometry reduces the influence of the near-surface low-velocity area on the reflection signal.

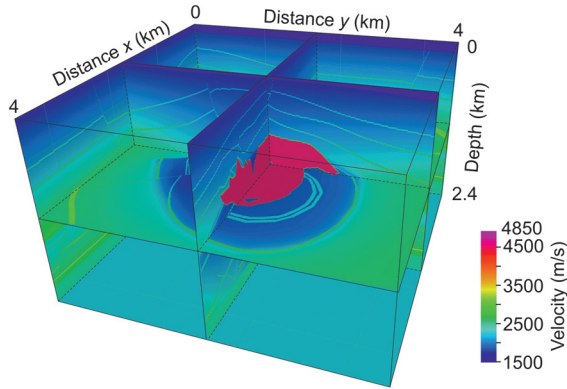


Figure 3. The 3D SEG/EAGE salt velocity model, for generating 3D VSP data. The presented vertical slices are across the center of the model, at $(x, y) = (2028, 2028)$ m.

In RTM, we use the following acoustic wave equation with constant density:

$$\frac{1}{v^2(\mathbf{x})} \frac{\partial^2 p(\mathbf{x}, t)}{\partial t^2} = \nabla^2 p(\mathbf{x}, t) + s(t)\delta(\mathbf{x} - \mathbf{x}_s), \quad (1)$$

where $v(\mathbf{x})$ denotes the velocity, ∇^2 denotes the Laplacian operator, $\mathbf{x} = (x, y, z)$ denotes the subsurface imaging location, $\mathbf{x}_s = (x_s, y_s, z_s = 0)$ is the shot position, and $s(t)$ is the source wavelet. We numerically generate two wavefields p_F and p_B (Figure 1a) using a finite-difference method, and we obtain the depth image using the following crosscorrelation imaging condition (Claerbout, 1971):

$$I(\mathbf{x}) = \int_0^T p_F(\mathbf{x}, t) p_B(\mathbf{x}, t) dt, \quad (2)$$

where T is the maximum time length of the data. A single shot gather will generate an RTM image. The stack of the images of all shots will produce the final RTM image. Thus, the computation time is strictly proportional to the number of shot gathers involved in the imaging.

Wave simulation is the core in an RTM algorithm. When using RBC in the model, the energy traveling toward the boundaries will not be absorbed. Therefore, a reverse procedure will be able to reconstruct wavefields in the previous time steps. For RTM, RBC

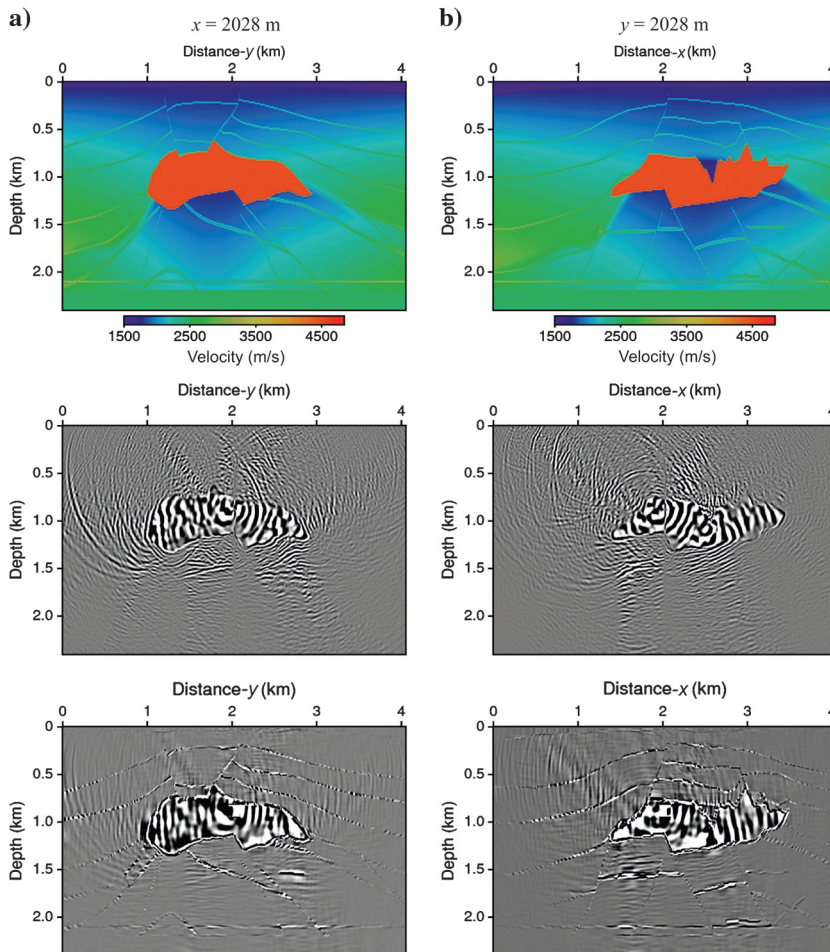


Figure 4. (a) The velocity profile cutting along the y -direction at $x = 2028$ m, RTM images of 2D walkaway VSP, before and after NLN noise attenuation. (b) The velocity profile cutting along the x -direction at $y = 2028$ m, RTM images of 2D walkaway VSP, before and after NLN noise attenuation.

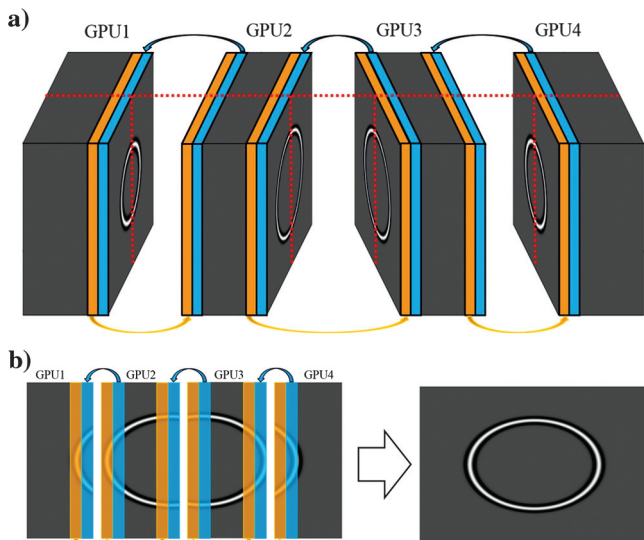
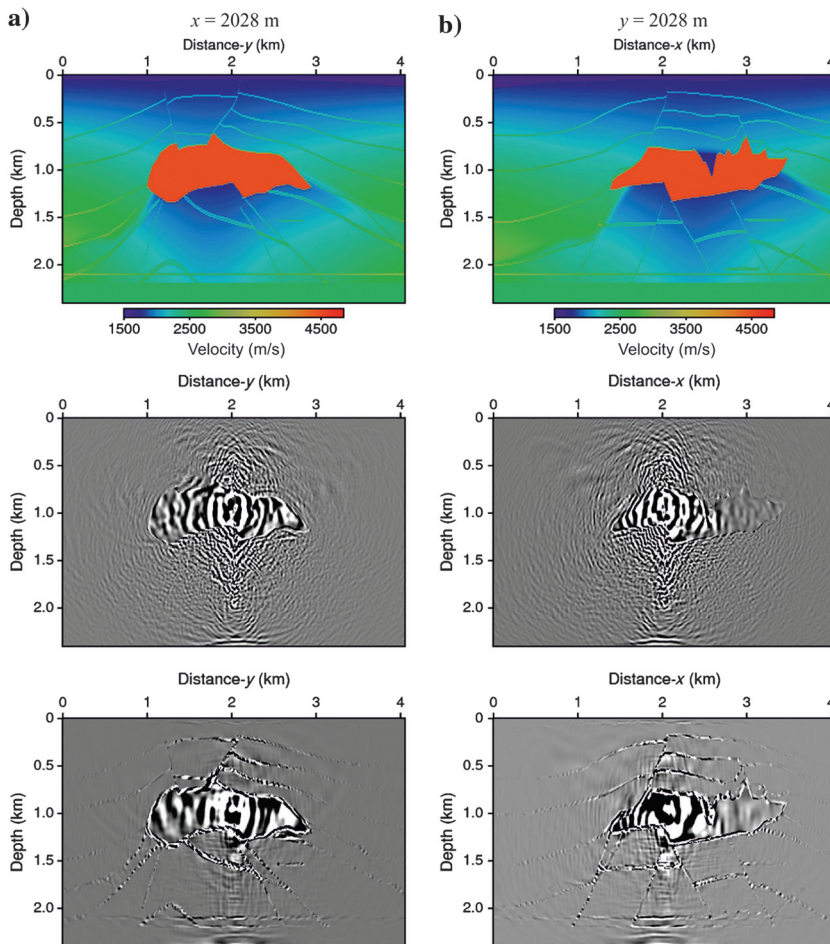


Figure 5. Wave simulation by multiple GPUs. (a) A 3D data set is cut into several parts. (b) A slice (cut along the red line in panel [a]) is partitioned to different GPUs and finally is merged to form an image. The two zones highlighted at an edge of a subblock are shared by neighboring subblocks, and the same color in the neighboring subblocks indicates the same data zone. Between these two zones, the outer zone is needed for the finite-difference calculation. This outer zone will simply be dropped off when the subblock is merged into the final image.

Figure 6. (a) A vertical profile of the 3D velocity model, cutting along the y -direction across the VSP well located at $(x, y) = (2028, 2028)$ m, the corresponding profiles of 3D VSP RTM image, before and after NLM noise attenuation. (b) A vertical profile of the 3D velocity model, cutting along the x -direction across the VSP well, the corresponding RTM profiles before and after noise attenuation.



only adds a random-velocity layer outside the migration velocity field, that is, a random velocity model.

We construct the random boundary function as in the following:

$$v_n(\ell) = v_0(\ell) - r\ell, \quad (3)$$

where v_0 is the original velocity function at the boundary before adding a random velocity function, r is a random number, and ℓ is the distance between the random velocity point and the inner boundary of the random velocity. Equation 3 indicates that ℓ acts as a weighting factor to r , and the amplitude of the random velocity $v_n(\ell)$ varies according to the distance between the edge of the valid velocity area and the random velocity point.

The key difference between RBC and an absorbing boundary condition such as PML is that RBC is a nonabsorbing boundary. Because the source wavefield in the forward propagation is not absorbed by the random boundary, one can use the source wavefield at the two maximum times and repropagate them backward. In fact, one can use the wavefield at any two time samples $p_F(t + \Delta t)$ and $p_F(t)$ to reproduce $p_F(t - \Delta t)$ by backward propagation. Only the wavefields of two time samples, not the entire source wavefield, are needed at any time, and hence RBC greatly reduces the storage.

The source wavefield backward propagation is at the same pace as the receiver wavefield (Figure 1b). For the scheme above, alternatively, we can propagate the receiver wavefield first and store the wavefield at two minimum times. Then we simulate the forward source wavefield and repropagate the receiver wavefield simultane-

ously, and we conduct a correlation from the minimum time toward the maximum time.

Figure 2 demonstrates the backward-propagated source wavefield. Figure 2a is a two-layer velocity model, in which the constant velocity of the bottom layer is larger than the constant velocity of the top layer, and the letters “RTM” in the top layer have the same velocity as the bottom layer. This artificial velocity model is surrounded by four random boundaries. Figures 2b, 2c, and 2d are three snapshots of the source wavefield generated by backward propagation. We have compared these wavefield snapshots with those generated by forward propagation (not shown in this paper), and we found that the amplitude difference is less than 10^{-5} .

For a large data set, such as 3D VSP, in which we consider the high computation cost, it is preferable to use RBC rather than an absorbing boundary such as PML. After RTM with RBC in the model, low-frequency noise is suppressed by the Laplacian filter, and random noise is attenuated by the NLM method. The NLM process is a weighted average of all the pixels in the RTM image (Buades et al., 2005; Mahmoudi and Sapiro, 2005; Brox et al., 2008; Buades et al., 2010). Because it uses the redundancy of structures within a data set, it can be rather effective in noise attenuation. Summarized technical details, including development to the 3D case, are in Appendix A.

2D VSP RTM

The prime objective is to make RTM work for 3D VSP. To demonstrate the RTM strategy, we first simulate a 2D VSP acquisition across a SEG/EAGE 3D salt model (Aminzadeh et al., 1997). Figure 3 shows the 3D salt model, containing a complex salt body, and the velocity varies from 1500 to 4850 m/s. Source lines of 2D walkaway VSPs are positioned across the well, located at $(x, y) = (2028, 2028)$ m, at the center of the model. Note that the RTM image of this 2D VSP data set unavoidably will have the effect of side reflections, but it closely mimics 2D seismic acquisition in the real world.

Two walkaway VSP lines are perpendicular to each other. Each VSP source line has 11 shots in total, with a lateral interval of 360 m. The first source is 1800 m away from the borehole. There are 260 receivers in the borehole, with a depth interval of 6 m. Therefore, receivers are spanned over the range 420–1974 m in depth. The source is a Ricker wavelet with a peak frequency of 20 Hz. The recording time sample interval is 0.4 ms, and each seismic trace has 8000 samples.

For the 2D VSP RTM, although the input VSP data are generated by 3D modeling, we use a 2D finite-difference scheme to calculate the source wavefield and the receiver wavefield. The temporal derivative is approximated by a second order finite-differencing and the spatial derivative is

approximated by a 12th-order finite-difference scheme (Zhang et al., 2014). The grid numbers along two lateral directions are 676 each, and the grid number along the vertical direction is 401. The grid size is 6 m for all axes, so that there are five grid points per wavelength to avoid grid dispersion (Hall and Wang, 2009). In this case, the minimum velocity is 1500 m/s, the maximum frequency is 50 Hz, corresponding to the Ricker wavelet with the peak frequency of 20 Hz, and thus the minimum wavelength is 30 m.

In this exercise, we mimic real-world seismic data processing by using a synthetic data set generated from a 3D model with a point source. However, in 2D RTM, wavefield simulation is conducted

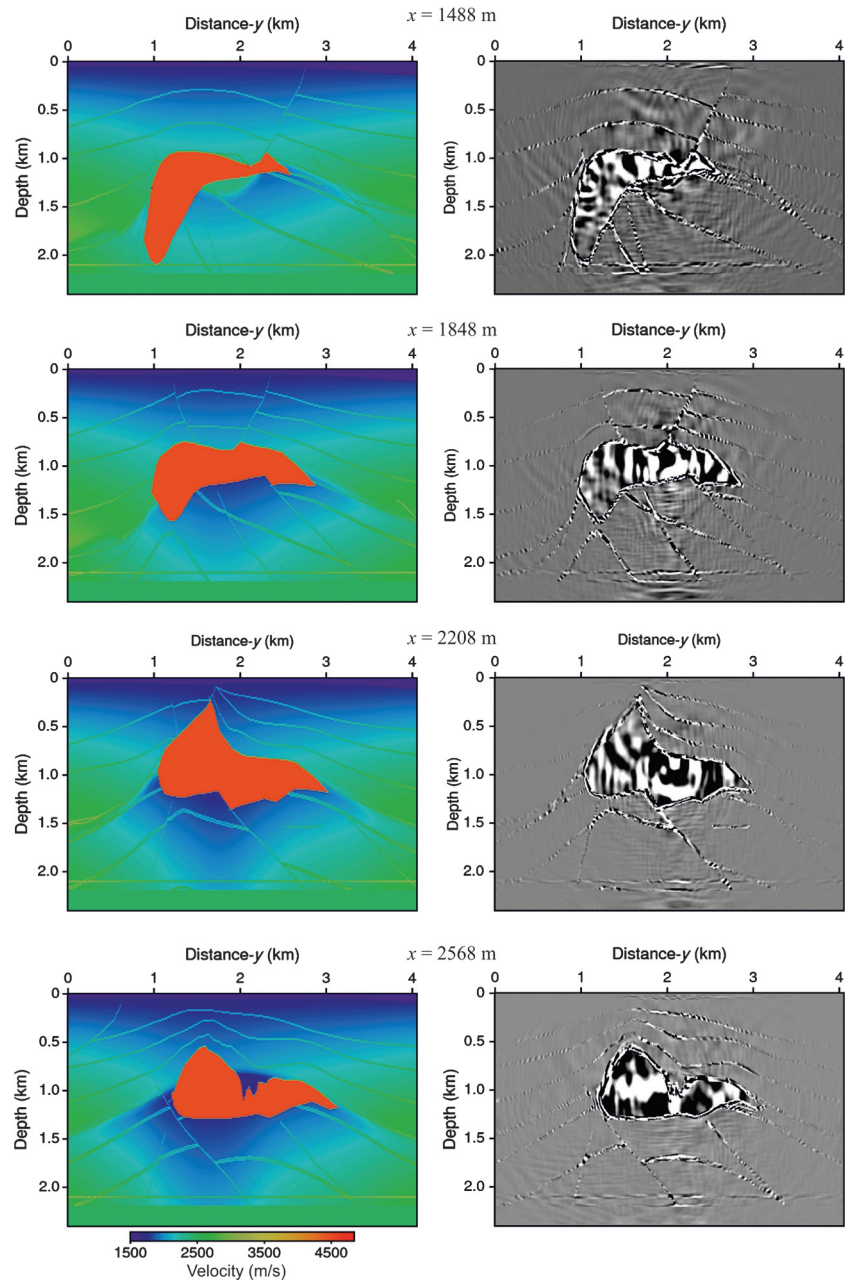


Figure 7. A series of profiles cutting along the y -direction, with an interval of 360 m in the x -direction. Each row is the velocity profile and the corresponding final seismic image of 3D VSP RTM.

based on a 2D model and line sources that are perpendicular to the 2D section. Therefore, before RTM imaging, the data set needs to be transformed from a point source to a line source. According to Wang and Rao (2009), this preprocessing consists of the following two steps: time-domain amplitude correction and frequency-domain phase change.

RTM images of 2D VSP data are displayed in Figure 4. Figure 4a is a walkaway VSP image along the y -direction, across the well at $x = 2028$ m. Figure 4b is a walkaway VSP image along the

x -direction, across the well at $y = 2028$ m. In each panel, the top is the velocity profile, and the middle is the RTM image before noise attenuation. In these images, a Laplacian filter has been used, as always in RTM, to attenuate the low-frequency noise. At the bottom is the RTM image after NLM noise attenuation. This example demonstrates that NLM can effectively remove random noise in the RTM image. The effectiveness will be more obvious in the following 3D case.

3D VSP RTM

The RTM approach is now applicable to 3D VSP data. For the 3D salt model, shown in Figure 3, there are 11 shot lines and 11 source points along each shot line. Thus, there are 121 shots in total. For RTM, the model parameters and the seismic geometry are identical to that used in the synthetic wave simulation stage.

In the example, NLM denoising is performed within the cubes (see Appendix A). The decaying parameter $h = 300$ is set for the exponential function. A large value for h will provide a similar weight for all pixels in the image (equation A-3 in Appendix A). The standard deviation of the noise $a = \sqrt{5}$ is given heuristically for the Gaussian kernel (equation A-6 in Appendix A) and is kept constant for processing. The pixel neighborhood is chosen to be a $3 \times 3 \times 3$ cube. The search neighborhood size is chosen to be a $10 \times 10 \times 10$ cube.

For 3D RTM imaging, we use GPU acceleration technology to reduce the computation cost. We partition the 3D VSP wavefield into several parts and assign different parts to different GPUs for the calculation (Figure 5a). We use a synchronous order to guarantee a synchronous calculation process on different GPUs. In Figure 5a and 5b, the same color in the neighboring subblocks indicates the same data zone. The two zones highlighted at an edge of a subblock are shared by neighboring subblocks. Between these two zones, the outer zone is needed for the finite-difference calculation. In the case of 12th-order spatial finite differencing, the outer data exchange zone is six grids in size. This outer zone will simply be dropped off when the subblock is merged into the final image.

We perform data partitioning in the CPU, but wave simulation and crosscorrelation imaging are performed within multiple GPUs at each of the RTM time samples. Because of the use of RBC, wave simulation, including forward and backward calculation, and imaging can be undertaken on a GPU, and there is no need to transfer data between the CPU host and the GPU device. This saves a good amount of the time needed for data communication between the CPU and GPU.

In the computation example, we use four GPU kernels in total (Figure 5). The GPU accelerator is an Nvidia Tesla K10, and each Tesla K10 card

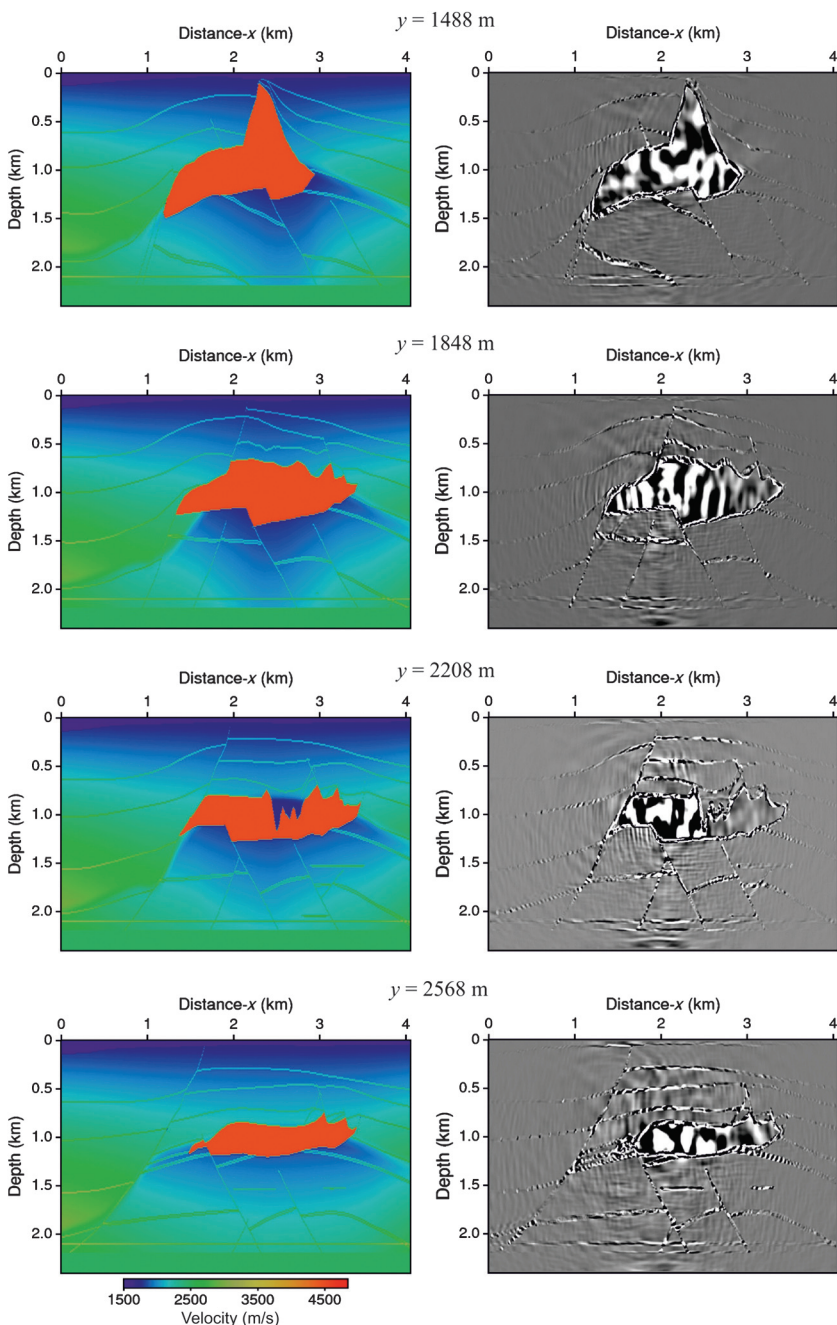


Figure 8. A series of profiles cutting along the x -direction, with an interval of 360 m in the y -direction. Each row is the velocity profile and the corresponding final seismic image of 3D VSP RTM.

consists of two GPU kernels. That is, we use two GPU cards in total. Each GPU kernel has 4 GB graphic memory, and its single-precision performance is 2288 GFlops. For the model (Figure 3) with a size of $676 \times 676 \times 401$ grids, the actual computation size is $776 \times 776 \times 501$ grids, which includes 50 grids of a random boundary in each side. The run time for the 3D RTM of a single shot record is about 4000 s.

Figure 6 shows the 3D RTM profiles, directly comparable with the 2D RTM results shown in Figure 4. Some near-vertical faults, either on the top or beneath the salt body, are clearly presented in the 3D RTM image. Because 3D migration is obtained from sources around all azimuths, the complex boundary of the salt body should be better positioned in the 3D image than in the 2D counterpart.

These features can be observed from a series of vertical profiles cutting through the 3D model, either parallel to the y -direction (Figure 7) or to the x -direction (Figure 8).

CONCLUSIONS

We have presented an effective RTM method applicable to 3D VSP data. To save the storage of the source wavefield for RTM imaging, we have used random boundaries so that we can generate any needed source wavefield by backward propagation. Because of the use of random boundaries, there are random noises in the wavefield. We have used the NLM method to suppress any of these random noises.

This method needs to generate the source wavefield twice, one of which is forward simulation and another of which is back propagation. We have used the GPU parallelization technique and multiple GPUs simultaneously to accelerate the computation. Therefore, RTM is applicable now for 3D VSP imaging, and it is potentially affordable for the imaging of 3D surface seismic data.

ACKNOWLEDGMENTS

The authors are grateful to the National Natural Science Foundation of China (grant no. 41574117) and the sponsors of the Centre for Reservoir Geophysics, Imperial College London, for supporting this research.

APPENDIX A

THE NONLOCAL MEANS METHOD

In VSP RTM, we use the NLM algorithm to attenuate random noise in the image. The NLM process (Buades et al., 2005; Mahmoudi and Sapiro, 2005; Brox et al., 2008; Buades et al., 2010) is a weighted average of all the pixels in the RTM image:

$$\tilde{u}(i) = \sum_j w(i, j) u(j), \quad (\text{A-1})$$

where $u(j)$ is a pixel of the RTM image; $\tilde{u}(i)$ denotes the image pixel after denoising; and $w(i, j)$ is a weight coefficient, depending upon the similarity between the pixels $u(i)$ and $u(j)$. The weight must satisfy the following two conditions:

$$0 \leq w(i, j) \leq 1, \quad \sum_j w(i, j) = 1. \quad (\text{A-2})$$

To quantify the similarity between the pixels $u(i)$ and $u(j)$ (Figure A-1a), we define a window N_i around the pixel $u(i)$ and a window N_j around the pixel $u(j)$. Window N_i and window N_j are centered upon the pixels $u(i)$ and $u(j)$, respectively. Using windows, instead of individual pixels allows us to incorporate the neighborhood information into the evaluation.

The similarity between N_i and N_j is expressed as

$$w(i, j) = \frac{1}{Z(i)} \exp\left(\frac{-D^2(i, j)}{h^2}\right), \quad (\text{A-3})$$

where $D(i, j)$ is the Euclidean distance between pixels $u(i)$ and $u(j)$; h is a constant, which ensures the decay of the exponential function; and $Z(i)$ is a normalization factor, which can be expressed as

$$Z(i) = \sum_j \exp\left(\frac{-D^2(i, j)}{h^2}\right) \quad (\text{A-4})$$

to ensure $\sum_j w(i, j) = 1$.

The Gaussian weighted euclidean distance $D(i, j)$, between the neighborhood around the pixel $u(i)$ and the neighborhood around the pixel $u(j)$, is

$$D(i, j) = \sqrt{\sum_{k=1}^n (G_a(k) [u(N_i(k)) - u(N_j(k))])^2}, \quad (\text{A-5})$$

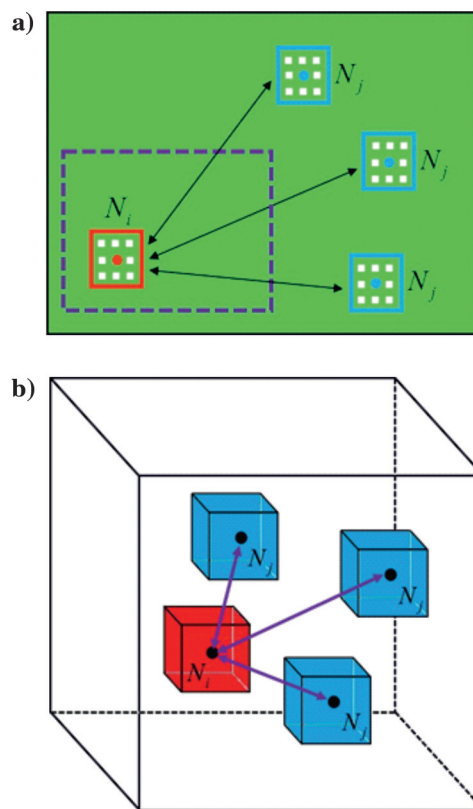


Figure A-1 (a) The similarity between a 2D window (N_i centered at index i) and any 2D window (N_j centered at index j). (b) Similarity between a 3D cube (N_i) and any 3D cube (N_j).

where G_a represents the Gaussian kernel with standard deviation a , $u(N_i)$ is the neighborhood around the pixel $u(i)$, $u(N_j)$ is the neighborhood around the pixel $u(j)$, k is an element within a neighborhood, and n is the total number of elements within a window.

The Gaussian kernel G_a can be expressed as

$$G_a(x, y) = \exp\left(-\frac{(x-x_0)^2 + (y-y_0)^2}{2a^2}\right), \quad (\text{A-6})$$

where x_0 and y_0 denote the center of the Gaussian kernel and x and y correspond to the coordinates of the element k in equation A-5.

For a 3D data set, the neighborhood is a cube (Figure A-1b) and the nonlocal means denoised image value “image(i)” is estimated as a weighted average of all the pixels within a search cube. It can be expressed as

$$\text{image}(i) = \sum_j w(i, j) \text{original}(j), \quad (\text{A-7})$$

where “original(j)” denotes the pixel corresponding to point j . The weights $w(i, j)$ depend upon the similarity between the two cubes centered at the pixel i and j . This extension from 2D to 3D is straightforward.

REFERENCES

- Aminzadeh, F., J. Brac, and T. Kunz, 1997, 3-D salt and overthrust models: SEG/EAGE 3-D Modeling Series 1, SEG.
- Amundsen, L., 1993, Depth imaging of offset vertical seismic profile data: *Geophysical Prospecting*, **41**, 1009–1031, doi: [10.1111/j.1365-2478.1993.tb00896.x](https://doi.org/10.1111/j.1365-2478.1993.tb00896.x).
- Baysal, E., D. Kosloff, and J. Sherwood, 1983, Reverse time migration: *Geophysics*, **48**, 1514–1524, doi: [10.1190/1.1441434](https://doi.org/10.1190/1.1441434).
- Berenger, J., 1994, A perfectly matched layer for the absorption of electromagnetic waves: *Journal of Computational Physics*, **114**, 185–200. doi: [10.1006/jcph.1994.1159](https://doi.org/10.1006/jcph.1994.1159).
- Bonar, D., and M. Sacchi, 2012, Denoising seismic data using the nonlocal means algorithm: *Geophysics*, **77**, no. 1, A5–A8, doi: [10.1190/geo2011-0235.1](https://doi.org/10.1190/geo2011-0235.1).
- Brox, T., O. Kleinschmidt, and D. Cremers, 2008, Efficient nonlocal means for denoising of textural patterns: *IEEE Transactions on Image Processing*, **17**, 1083–1092, doi: [10.1109/TIP.2008.924281](https://doi.org/10.1109/TIP.2008.924281).
- Buades, A., B. Coll, and J. M. Morel, 2005, A review of image denoising algorithms, with a new one: *Multiscale Modeling and Simulation*, **4**, 490–530, doi: [10.1137/040616024](https://doi.org/10.1137/040616024).
- Buades, A., B. Coll, and J. M. Morel, 2010, Image denoising methods. A new nonlocal principle: *SIAM Review*, **52**, 113–147, doi: [10.1137/090773908](https://doi.org/10.1137/090773908).
- Burch, T., B. Hornby, H. Sugianto, and B. Nolte, 2010, Subsalt 3D VSP imaging at Deimos Field in the deep water Gulf of Mexico: *The Leading Edge*, **29**, 680–685, doi: [10.1190/1.3447781](https://doi.org/10.1190/1.3447781).
- Claerbout, J. F., 1971, Toward a unified theory reflector mapping: *Geophysics*, **36**, 467–481, doi: [10.1190/1.1440185](https://doi.org/10.1190/1.1440185).
- Clapp, R. G., 2009, Reverse time migration with random boundaries: 79th Annual International Meeting SEG Expanded Abstracts, 2809–2813.
- Cosma, C., L. Balu, and N. Enescu, 2010, 3D VSP migration by image point transform: *Geophysics*, **75**, no. 3, S121–S130, doi: [10.1190/1.3396316](https://doi.org/10.1190/1.3396316).
- Deng, F., and G. A. McMechan, 2008, Viscoelastic true-amplitude prestack reverse-time depth migration: *Geophysics*, **73**, no. 4, S143–S155, doi: [10.1190/1.2938083](https://doi.org/10.1190/1.2938083).
- Hall, F., and Y. Wang, 2009, Elastic wave modelling by an integrated finite difference method: *Geophysical Journal International*, **177**, 104–114, doi: [10.1111/j.1365-246X.2008.04065.x](https://doi.org/10.1111/j.1365-246X.2008.04065.x).
- Hardage, B. A., 2000, *Vertical seismic profiling: Principles*: Pergamon.
- Harwijanto, J. A., C. P. A. Wapenaar, and A. J. Berkhout, 1987, VSP migration by single shot record inversion: *First Break*, **5**, 247–255, doi: [10.3997/1365-2397.1987012](https://doi.org/10.3997/1365-2397.1987012).
- Mahmoudi, M., and G. Sapiro, 2005, Fast image and video denoising via nonlocal means of similar neighborhoods: *IEEE Signal Processing Letters*, **12**, 839–842, doi: [10.1109/LSP.2005.859509](https://doi.org/10.1109/LSP.2005.859509).
- McMechan, G., 1983, Migration by extrapolation of time-dependent boundary values: *Geophysical Prospecting*, **31**, 413–420, doi: [10.1111/j.1365-2478.1983.tb01060.x](https://doi.org/10.1111/j.1365-2478.1983.tb01060.x).
- Neklyudov, D., and I. Borodin, 2009, Imaging of offset VSP data acquired in complex areas with modified reverse-time migration: *Geophysical Prospecting*, **57**, 379–391, doi: [10.1111/j.1365-2478.2008.00745.x](https://doi.org/10.1111/j.1365-2478.2008.00745.x).
- Nguyen, B. D., and G. A. McMechan, 2015, Five ways to avoid storing source wavefield snapshots in 2D elastic prestack reverse time migration: *Geophysics*, **80**, no. 1, S1–S18, doi: [10.1190/geo2014-0014.1](https://doi.org/10.1190/geo2014-0014.1).
- Schiemanz, A., and H. Igel, 2013, Accelerated 3-D full-waveform inversion using simultaneously encoded sources in the time domain: Application of Valhall ocean-bottom cable data: *Geophysical Journal International*, **195**, 1970–1988, doi: [10.1093/gji/ggt362](https://doi.org/10.1093/gji/ggt362).
- Shin, J., W. Ha, H. Jun, D. J. Min, and C. Shin, 2014, 3D Laplace-domain full waveform inversion using a single GPU card: *Computers & Geosciences*, **67**, 1–13, doi: [10.1016/j.cageo.2014.02.006](https://doi.org/10.1016/j.cageo.2014.02.006).
- Sun, W., and L. Y. Fu, 2013, Two effective approaches to reduce data storage in reverse time migration: *Computers & Geosciences*, **56**, 69–75, doi: [10.1016/j.cageo.2013.03.013](https://doi.org/10.1016/j.cageo.2013.03.013).
- Symes, W. W., 2007, Reverse time migration with optimal checkpointing: *Geophysics*, **72**, no. 5, SM213–SM221, doi: [10.1190/1.2742686](https://doi.org/10.1190/1.2742686).
- Wang, Y., 1999, Random noise attenuation using forward-backward linear prediction: *Journal of Seismic Exploration*, **8**, 133–142.
- Wang, Y., and Y. Rao, 2009, Reflection seismic waveform tomography: *Journal of Geophysical Research*, **114**, B03304, doi: [10.1029/2008JB005916](https://doi.org/10.1029/2008JB005916).
- Whitmore, D., 1983, Iterative depth migration by backward time propagation: 53rd Annual International Meeting, SEG, Expanded Abstracts, 382–385.
- Xiao, X., and G. T. Schuster, 2009, Local migration with extrapolated VSP Green's functions: *Geophysics*, **74**, no. 1, S115–S126, doi: [10.1190/1.3026619](https://doi.org/10.1190/1.3026619).
- Xie, W., D. H. Yang, F. Q. Liu, and J. S. Li, 2014, Reverse-time migration in acoustic VTI media using a high-order stereo operator: *Geophysics*, **79**, no. 3, WA3–WA11, doi: [10.1190/geo2013-0224.1](https://doi.org/10.1190/geo2013-0224.1).
- Yan, R., and X. B. Xie, 2012, An angle-domain imaging condition for elastic reverse time migration and its application to angle gather extraction: *Geophysics*, **77**, no. 5, S105–S115, doi: [10.1190/geo2011-0455.1](https://doi.org/10.1190/geo2011-0455.1).
- Zhang, S., F. Yang, D. Xu, and Y. Wang, 2014, Seismic wave simulation by velocity-stress wave equations in two-phase anisotropic media: *Journal of Geophysics and Engineering*, **11**, 015008, doi: [10.1088/1742-2132/11/1/015008](https://doi.org/10.1088/1742-2132/11/1/015008).
- Zhang, Y., and J. Sun, 2009, Practical issues in reverse time migration: True amplitude gathers, noise removal and harmonic source encoding: *First Break*, **26**, 29–35.
- Zhang, Y., H. Zhang, and G. Zhang, 2011, A stable TTI reverse time migration and its implementation: *Geophysics*, **76**, no. 3, WA3–WA11, doi: [10.1190/1.3554411](https://doi.org/10.1190/1.3554411).
- Zhu, T. Y., J. M. Harris, and B. Biondi, 2014, Q-compensated reverse-time migration: *Geophysics*, **79**, no. 3, S77–S87, doi: [10.1190/geo2013-0344.1](https://doi.org/10.1190/geo2013-0344.1).