Iterative velocity model building using GPU based layer-stripping TTI RTM

Qien Ji, Sang Suh and Bin Wang, TGS*

Summary

The technology of using Tilted Transverse Isotropic Reverse Time Migration (TTI RTM) for subsalt velocity model building has been playing an important role in the seismic industry. Since TTI RTM is a computationally expensive technique, improving the program efficiency to meet the project turnaround schedule becomes a critical topic. In this paper, we will present an algorithm called "GPU based Layer Stripping TTI RTM". In layer stripping RTM, the model will be decomposed into two or more regions, horizontally. The wavefield redatuming for the top region wavefield will be saved as an input to the bottom region for RTM. In this approach, we do not need to repeat the shallow wavefield extrapolation; the grid size of the deeper migration can be increased and TTI RTM can be replaced by Vertically Transverse Isotropic (VTI) RTM, due to the fact that the dip field is not as sensitive in the deep region. This is a key to improving the efficiency of the RTM. The super parallelism of the GPU RTM plays another important role in the efficiency of the algorithm. In practice the application of GPU based layer stripping TTI RTM reduces the computation time by two orders of magnitude. In this paper, we will discuss some practical issues such as how to manage balancing the computer resources and the 3D data explosion of the redatumed wavefields.

Introduction

For proper subsalt imaging, an accurate velocity model is critical. Because of the large velocity contrast between the sediment's low velocity and the salt's high velocity, ray-based migration algorithms such as Kirchhoff and beam may not be sufficient to produce acceptable subsalt image quality. TTI RTM is a two-way wave-equation based algorithm which has no steep-dip limitation and is capable of handling multi-path energy; therefore it is capable of successfully imaging areas with sharp velocity contrasts. In order to produce superior image quality in a geologically complex area, the industry is trending towards using an unlimited number of iterations of TTI RTM runs in order to test different interpretation scenarios for the Base of Salt (BOS) before finalizing the BOS interpretation. To have the ability to run RTM to high frequencies is another demand we now see from the industry. The higher frequency the RTM the smaller grid size that will be needed to image the wavefield, in turn requiring more computer resources. The limitation of hardware can limit the capability of the RTM jobs.

Previously, we developed an interactive tool which uses post-stack wave equation migration (WEM) or beam migration to allow for quick salt geometry editing in areas of interpretation uncertainty (Wang et al., 2008; Wang et al., 2011). The tool makes for efficient testing of a large number of salt interpretation scenarios. To improve the image quality, using an efficient pre-stack TTI RTM to replace WEM or beam migration in the interactive image tool is desirable.

To improve the TTI RTM efficiency, we previously developed GPU TTI RTM (Sun and Suh, 2011) and Layer Stripping TTI RTM (Wang et al., 2011). Both algorithms dramatically improved the turnaround time for the RTM. Based on these two techniques, we recently implemented our GPU layer stripping TTI RTM. We found that using GPU TTI RTM as in the interactive tool had several benefits. 1) The super parallelism of the GPU technique makes the wave propagation (the kernel, the most computationally expensive part of the TTI RTM) super-efficient. GPU based TTI RTM can achieve an order of magnitude speed-up. 2) In layer stripping RTM, RTM is only performed below a subsurface datum using a redatumed wavefield. Therefore, the model size needed will be reduced. 3) The computational grid size can be greatly increased in the deeper region while still being able to avoid dispersion noise, because the minimum velocity will increase in deeper region. 4) The dip field in the TTI RTM is more sensitive in the shallow region because of the mini-basin. In the deeper region, TTI media can be replaced by VTI, therefore the efficiency can be improved further by three to five times.

The depth-variant grid size and depth-variant anisotropic media type for the RTM algorithm will make GPU based layer stripping RTM more efficient in its use of current hardware.

In this paper, we describe the methodology for the GPU-based layer stripping TTI RTM, the challenges in implementation and how the difficulties can be overcome. We will also show some real data examples.

Layer stripping RTM

RTM is a shot gather based migration method. In layer stripping RTM, the model is divided horizontally into a multiple regions with overlapping zones. The RTM will run sequentially layer by layer from top to bottom. When TTI RTM is run on the shallow region, the wavefield at the bottom of the region will be saved. The saved redatumed wavefield becomes the input for the subsequent, deeper

DOI http://dx.doi.org/10.1190/segam2012-1455.1

RTM run. This approach is well suited for velocity model building in production. Typically, there are a large number of iterations of RTM which will need to be run from top to bottom for building the velocity model in geologically complex areas. When the subsalt velocity model is used, RTM needs to be run only below the subsurface datum instead of over the entire region. The redatumed wavefield saved at the base of the top layer can be reused in all of the deeper iterations. The computational cost is greatly reduced for the numerous migration iterations by skipping the RTM run in the region above the subsurface datum. Figure 1 shows how the model has been decomposed into three regions and where the wavefield has been saved at the base of each region (top of the overlap zone).

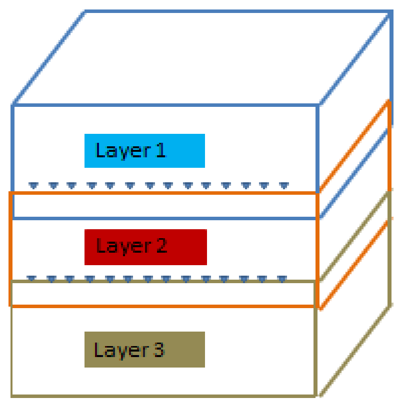

Figure 1: The diagram shows the decomposition of the model in layer stripping RTM.

The computational resources required for RTM are determined by the aperture of the migration and the grid size of computational wavefield. The smaller the grid size, the more computational resources will be required for the migration. The necessary grid size in the computational area is determined by the minimum velocity in this region. The velocities are typically smaller in shallow regions compared to velocities in deeper regions. Consequently, the number of computational grids can be reduced in the deeper layer for the RTM. For example we divided the computational region into two regions. The first region is from 0 km to 6 km and the second region is below 6 km. The minimum velocity is 1.5 km/s in the first region and 3.0 km/s for the second region. The grid size can be doubled and time step can be double either in this case for the deeper part. Theoretically, the computational cost can be reduced by a factor of sixteen in this example.

In the deeper region, the dip field which is required by TTI RTM, is not very accurate nor sensitive. We can run TTI RTM for the shallow region and VTI RTM in the deeper region. In other words, we can use different media types for the RTM algorithm in different layers. Because TTI is about three to five time slower than VTI, applying a different media type RTM to different layers will give additional savings on the computational cost.

GPU RTM

We deployed the CPU-based RTM to the latest GPU architecture from NVIDIA. As usual, there are two major steps in the RTM program. Step one is the forward modeling. Step two is the backward propagation. In step one, the program will save the wavefield on disk for each time step. In step two, the program will read in the wavefield which was saved in step one and used in step two. The GPU RTM involves the problem of how to balance of the usage of the GPU, CPU and disk input/output (I/O).

In our original multi-core parallelization of the CPU RTM, all of the resources are used on the wave propagation, and only one thread is used for writing out or reading in the wavefield as mentioned above. Since RTM, especially TTI RTM, is a computationally expensive algorithm, and kernel propagation is the most expensive part overall, this approach balances well with I/O in the system. GPU RTM breaks the balance of the system due to the speed-up from the super parallelism of the wave propagation. In order to speed up disk I/O, we implement the wavelet compression methodology. The wavefield will be compressed before it is written to disk. In this case, the disk I/O is reduced and the GPU, CPU and disk I/O are well balanced.

GPU based layer stripping RTM

To put it simply, the GPU based layer stripping RTM is a methodology which runs layer stripping RTM on GPU clusters. Layer stripping RTM makes the computational region smaller. The larger grid size allowed by the layer stripping RTM method makes the wave propagation faster. The necessary computational resources are reduced more on the GPU side than the CPU side. The large size of the input data in the layer stripping RTM method involves a well-known 3D input data explosion problem. In order to achieve the running speed which we expect, we must solve the balance problem in the system and the input data explosion problem as well.

The shot gather based input data is a point source and the size of the input data is limited by the cable size. For narrow azimuth data, there are typically six to ten cables for each shot. For wide azimuth data, there are up to one

DOI http://dx.doi.org/10.1190/segam2012-1455.1

hundred receiver cables for each super shot. In layer stripping RTM, every computational point will be sampled at the redatumed surface. Not only do the receiver side redatumed wavefields need to be saved, but also the source side redatumed wavefields need to be saved. Compared to regular RTM, the input data size of the layer stripping RTM will increase more than two orders of the magnitude. This large data set will impose problems in network bandwidth for data transfer and data storage capacity; and it will slow down the data input to the RTM. To solve these problems, a wavelet compression algorithm is used on the redatumed data (Figure 2). The algorithm we applied can give a compression ratio of up to two orders of magnitude without significantly damaging the quality of the redatumed data (i.e. the migration quality is not damaged from using the compressed redatumed data).

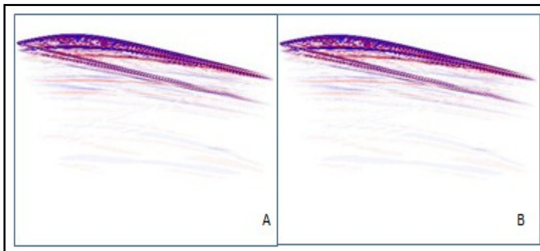

Figure 2: Comparison of the quality of the compressed and uncompressed redatumed data. A) uncompressed redatumed data. B) compressed redatumed data.

In layer stripping RTM, the model of the region being imaged is relatively smaller than if the entire region were being imaged, and the computational grid size is larger for deeper layers. The experience of running the CPU layer stripping RTM has shown that multiple or unlimited numbers of iterations of layer stripping RTM is desired for production processing jobs, in order to achieve outstanding imaging results. Because the kernel of propagation is running very fast in the GPU environment, a speed-up in the CPU and disk I/O are needed to keep the system balanced. The multi-thread wavelet compression in the CPU is applied to compress the wavefield before writing the wavefield to disk. The multi-thread wavelet decompression in CPU is applied to decompress the wavefield after reading the wavefield from disk. The parallelism of the CPU compression algorithm is applied when the parallelized wave propagation is running on the GPU. The well synchronized GPU based cluster makes our current TTI RTM two orders magnitude faster compared to the CPU based RTM program. Figure 3 shows the steps necessary to balance the computer system in GPU based RTM.

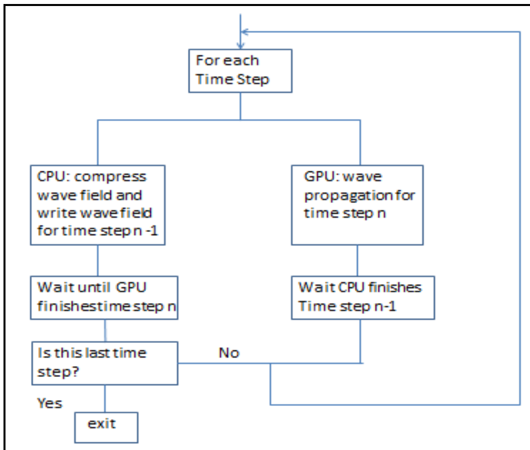

Figure 3: Steps for balancing the computer system in GPU based RTM

Examples

Subsalt velocity scanning is a routine practice in the seismic industry for updating subsalt velocity models. Using TTI RTM for these scans is demanded by the industry. Following is an example of a velocity scan job which uses the GPU based layer stripping TTI RTM. A subsalt velocity update is implemented in this example. The first layer is from 0 km to 6 km in depth. TTI RTM with a fine-grid migration is implemented in the first layer. The second layer is from 5 km to 16 km in depth. There is a 1 km overlap zone between the first and second layers. Because of the flatness of the region below salt, VTI RTM is used in the second layer. Due to the minimum velocity increasing with depth, a coarse grid sample migration is used in the second layer as well. Velocity scans ranging from 85% of the migration velocity to 115% of the migration velocity are implemented in this job. The migration time for each velocity model is less than 10 minutes. It will take about 18 hours to finish one migration in CPU based TTI RTM. The image qualities from GPU-based layer stripping TTI RTM are similar to the CPU-based TTI RTM. After the migrations using the different velocity models are finished, the best focused image is picked on the different migration volumes using the graphical picking tool which is shown in Figure 4. These picks are translated into velocity updates which are applied to the migration velocity model. After the subsalt velocities are updated, the second layer RTM is run again and the subsalt image using the updated velocity model is generated.

DOI http://dx.doi.org/10.1190/segam2012-1455.1

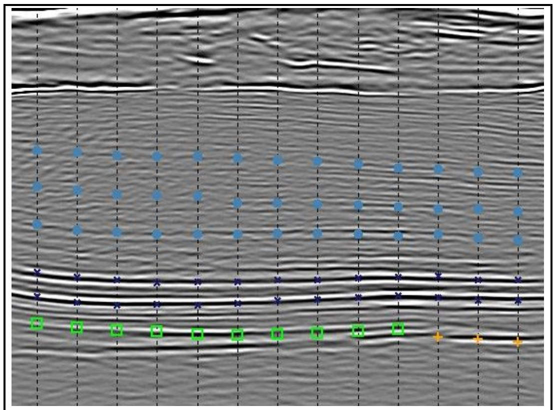

Figure 4: Graphical tool for picking the best focused image in the velocity scan. There are eleven 3D velocity models which are scanned in this example.

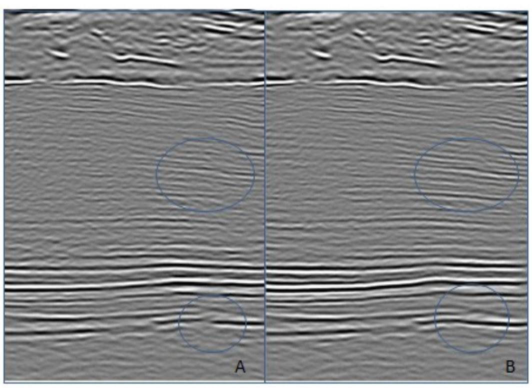

Figure 5: Comparison of the velocity scan. A) subimage before velocity update B) subimage after velocity update.

Another use of layer-stripping RTM is to test different salt velocity models. Figure 6 shows an example of several possible salt velocity models and their corresponding migration images for a Gulf of Mexico project. The shape and depth of the salt keel are not clear. More than 20 different salt models were tested. Shown are three of the more accurate salt models (Figure 6, left column) that were interpreted. Based on the RTM image, the bottom salt model makes the most sense and was chosen to be the final model.

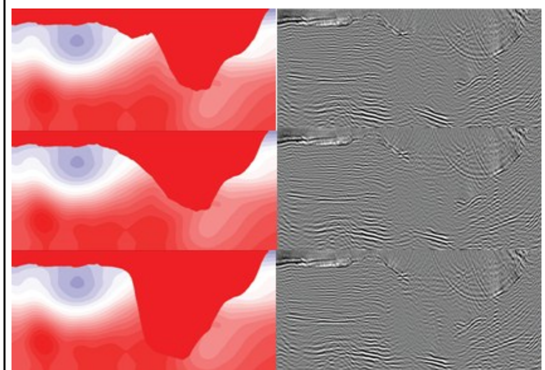

Figure 6: Three interpreted velocity models (left column). The right column shows the corresponding layer stripping RTM images

Conclusion

We have developed a GPU based layer stripping TTI RTM. To keep the balance of the computer system, wavelet compression plays an important role in the algorithm. The run time for the TTI RTM is dramatically improved, and required computer resources are reduced by using the GPU based layer stripping RTM. Due to the depth-variant grid size and the depth-variant anisotropic media type used in the algorithm, we can dramatically reduce the computational cost. A decrease in the computation time by two orders of magnitudes can be gained. The demand can be met for overnight turnaround for one velocity building iteration in a large scale project, while interactively testing unlimited numbers of salt interpretation scenarios. Higher frequency RTM migrations are also affordable to run, since the layer stripping method allows for using smaller grid sizes shallow, while using larger grid sizes in the deeper regions.

Acknowledgments

The authors would like to thank the following TGS colleagues for the contributions and helpful discussions: Kwangjin Yoon, Chuck Mason, Alex Yeh, James Cai and Zhiming Li. We also thank Laurie Geiger and Simon Baldock for reviewing and proof-reading this paper. The Freedom WAZ survey in the real data example is a cooperative effort between TGS and WesternGeco. Finally, we thank TGS management for permission to publish this paper.

DOI http://dx.doi.org/10.1190/segam2012-1455.1

http://dx.doi.org/10.1190/segam2012-1455.1

EDITED REFERENCES

Note: This reference list is a copy-edited version of the reference list submitted by the author. Reference lists for the 2012 SEG Technical Program Expanded Abstracts have been copy edited so that references provided with the online metadata for each paper will achieve a high degree of linking to cited sources that appear on the Web.

REFERENCES

Sun, X. and S. Suh, 2010, Maximizing throughput for high performance TTI-RTM: From CPU-RTM to GPU-RTM: 81st Annual International Meeting, SEG, Expanded Abstracts, 3179–3183.

Wang, B., J. Ji, K. Yoon, J. Cai, W. Whiteside, C. Mason, and Z. Li, 2011, Layer-stripping RTM based on wavefield redatuming: 81st Annual International Meeting, SEG, Expanded Abstracts, 3280–3284.

Wang, B., Y. Kim, C. Mason and X. Zeng, 2008, Advances in velocity model-building technology for subsalt imaging: Geophysics, **73**, no. 3, VE173–VE181.

Wang, B., C. Mason, K. Yoon, J. Ji, J. Cai, S. Suh, and Z. Li, 2011, Complex salt model building using a combination of interactive imaging and layer-striping RTM: First Break, **29**, no. 7, 47–54.