

Parallel GPU-based Implementation of One-Way Wave Equation Migration

*Alexander L. Pleshkevich*¹, *Vadim V. Lisitsa*², *Dmitry M. Vishnevsky*²,
*Vadim D. Levchenko*³, *Boris M. Moroz*¹

© The Authors 2018. This paper is published with open access at SuperFri.org

We present an original algorithm for seismic imaging, based on the depth wavefield extrapolation by the one-way wave equation. Parallel implementation of the algorithm is based on the several levels of parallelism. The input data parallelism allows processing full coverage for some area (up to one square km); thus, data are divided into several subsets and each subset is processed by a single MPI process. The mathematical approach allows dealing with each frequency independently and treating solution layer-by-layer; thus, a set of 2D cross-sections instead of the initial 3D common-offset vector gathers are processed simultaneously. This part of the algorithm is implemented using GPU. Next, each common-offset vector image can be stacked, processed and stored independently. As a result, we designed and implemented the parallel algorithm based on the use of CPU-GPU architecture which allows computing common-offset vector images using one-way wave equation-based amplitude preserving migration. The algorithm was used to compute seismic images from real seismic land data.

Keywords: GPU, nested OMP, MPI, seismic imaging.

Introduction

Seismic imaging is a procedure which allows construction of the subsurface images from seismic data. Eventually, a seismic image is a convolution of a recorded signal and two Green's functions (one for source position, the other one for the receiver position). This procedure should be applied to all possible source-receiver pairs. Thus, the most computationally intense part is the computation of the Green's functions. To simplify this step, depth extrapolation of the wavefield was suggested in [1], where the wavefield is computed by solving the initial value problem for the one-way wave equation (OWE). A pseudo-spectral method to solve one-way wave equation was suggested recently in [2, 3].

OWE-based migration has several computational features, which make it suitable for the processing of large common-offset datasets. First, Greens functions for different sources-receivers positions can be computed independently and then combined to construct the image in $(\omega-x)$ domain. Second, Greens functions and thus images are extrapolated layer-by-layer, thus at each step of the algorithm one deals with 2D cross-sections of the solution defined on plane $z = const$, and we can process a high number of solutions simultaneously. As a result, one can consider any combination of the Greens functions for a fixed frequency at a fixed depth and construct an image; i.e. common-shot, common-receiver, common-azimuth, etc.

1. Mathematical Background

Consider initial value problem for the one-way wave equation stated in $(\omega - x)$ domain:

$$\frac{\partial u}{\partial z} - i\sqrt{\frac{\omega^2}{c^2(x,y,z)} + \frac{\partial^2}{\partial y^2} + \frac{\partial^2}{\partial z^2}}[u] = 0, \quad u(\omega, x, y, 0) = u_0(\omega, x, y). \quad (1)$$

¹Central Geophysical Expedition JSC, Rosgeo, Moscow, Russian Federation

²Institute of Petroleum Geology and Geophysics SB RAS, Novosibirsk, Russian Federation

³Keldysh Institute of Applied Mathematics RAS, Moscow, Russian Federation

To solve equation (1), we suggest using the pseudo-spectral method presented in [2]. This approach allows to perform explicit stepping in vertical direction; i.e. assuming the solution to be known at a plane $z = z_0$, we can compute it at a plane $z = z_0 + \Delta z$ by the following rule:

$$u(\omega, x, y, z + \Delta z) = \sum_{j=n}^{n+1} \alpha_j F^{-1} \left[\exp \left(i\omega \Delta z \sqrt{\frac{1}{v_j^2} - \frac{k_x^2}{\omega^2} - \frac{k_y^2}{\omega^2}} \right) F[u(\omega, x, y, z)] \right] + \alpha_0 u(\omega, x, y, z), \quad (2)$$

where k_x and k_y are the spatial frequencies, $F[\]$ and $F^{-1}[\]$ denote the forward and inverse Fourier transforms with respect to the directions x and y , v_j are the nodal velocities, so that for each point $v(x, y, z) \in [v_n, v_{n+1}]$, parameters α_j are the coefficients of interpolation, as described in [2].

Using delta-functions as initial data one can compute Green's functions, corresponding to the positions of the sources and the receivers, and construct an image:

$$I(x, y, z) = \int_{\vec{x}_s \in D_S} \int_{\vec{x}_r \in D_R} \int_{\omega_{min}}^{\omega_{max}} G(\omega, x, y, z, \vec{x}_s) G(\omega, x, y, z, \vec{x}_r) \bar{f}(\omega, \vec{x}_s, \vec{x}_r) d\vec{x}_s d\vec{x}_r d\omega, \quad (3)$$

where $\vec{x}_s = (x_s, y_s)$ is a vector of sources coordinates at the daylight, and $G(\omega, x, y, z, \vec{x}_s)$ is the Green's function corresponding to the source position \vec{x}_s . Same notations are used for the receivers. Function f is the impulse recorded by the receiver at point \vec{x}_r and emitted by the source at point \vec{x}_s . Choice of the domains of integration D_s and D_r define the type of constructed images (common-shot, common-receiver, common-offset etc.).

Note, that a coarse grid (50 m) in a vertical direction is used to solve OWE, after that the solution is interpolated to the grid with the step of 5 m to construct an image. In this work, we use a linear frequency-dependent interpolation, which can be applied directly to the image, rather than to the Green's function, which drastically reduces the number of computations.

2. Parallel Implementation

To account for data parallelism, we divide the input data into subsets, to be processed by a single MPI process. We use the coordinates of the mid-points to parametrize the imaging domain, so we apply the 2D domain decomposition and consider all the sources and receivers corresponding to the mid-points from a subdomain as a single dataset or "stencil". Data division leads to the loss of the algorithm scaling because some of the sources/receivers belong to several datasets. Thus, we need to compute the Green's functions for these sources/receivers several times; however, we achieve the number of the Green's functions recomputations as low as 2.3 on average.

Parallel implementation of the computational part of the algorithms includes two aspects. First, for each time-frequency, we use a loop with respect to depth; i.e. we compute the Green's functions for all sources and receivers positions, combine them and multiply by the corresponding signal. Thus, at fixed depth level, we construct all possible images $I^k(\omega_m x, y, z_l)$. The GPUs do the computation of the Green's functions and construction of images. After that, a set of 2D cross-sections of the images $I^k(\omega_m x, y, z_l)$ are uploaded to RAM. Next, we apply all-to-all MPI communications to exchange the images, so that each MPI process accumulates single common-offset images for all stencils. Second, we sum up images from different stencils (integrate over sources/receivers positions) and interpolate the result within a slab $[z, z + \Delta z]$. CPU performs these operations in parallel with the computation of the Green's functions described in the

previous paragraph. The output data of each MPI process is a single common-offset vector image for all stencils. The block-scheme of the algorithm is presented in Fig. 1.

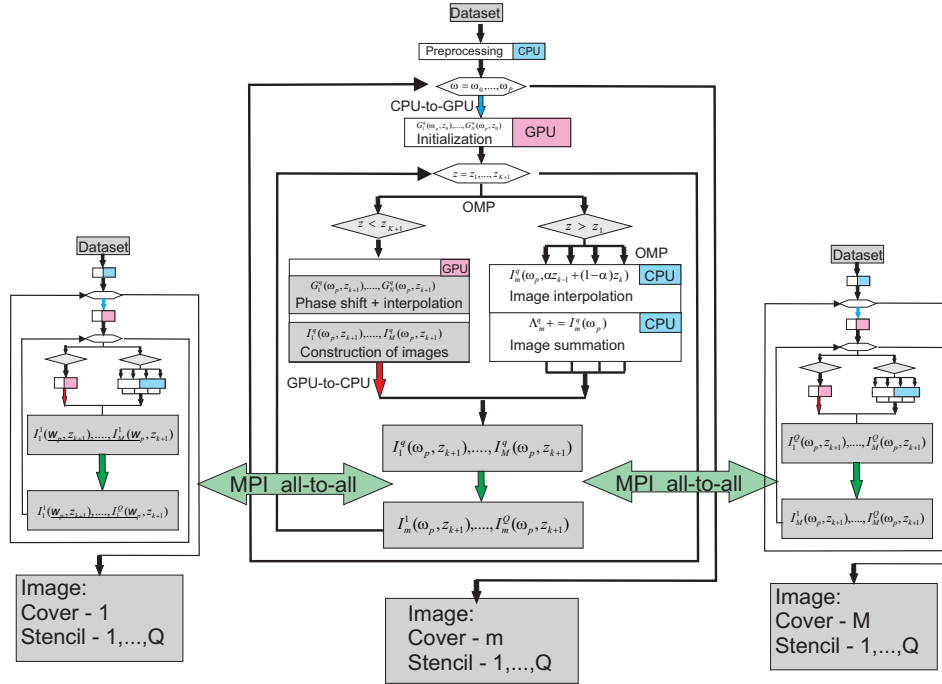


Figure 1. The block-scheme of the parallel algorithm

3. Numerical Experiments

3.1. Weak Scaling

To estimate a weak scaling of the algorithm, we consider the SEG Salt model (open-source model), for which we compute 17 images, having 2600 datasets. We perform simulations using 17, 34, 51, and 68 MPI processes, so, that each node deals with one dataset at a time. We provide the times needed for simulation in Tab. 1. The loss of the efficiency is caused by the MPI exchanges, which are implemented with enforced synchronization.

Table 1. Computation time for weak scaling estimation

	17 proc	34 proc	51 proc	68 proc
Time (hours)	1.91	2.22	2.64	3.02
W scaling (%)	-	86	72	63

3.2. Real-Data Example

We use the algorithm to construct the seismic image using real onshore seismic data. The size of the model is 15525 m InLine (x-direction) and 11250 m CrLine (y direction), depth is 5000 m. We compute 40 common-offset vector images, using 320 datasets, and 40 computational nodes. The total wall-clock time is about 320 hours; thus, the computational time is 11520 node-hours to construct the set of 40 images. A slalom-line section of the obtained 3D seismic image is presented in Fig. 2.

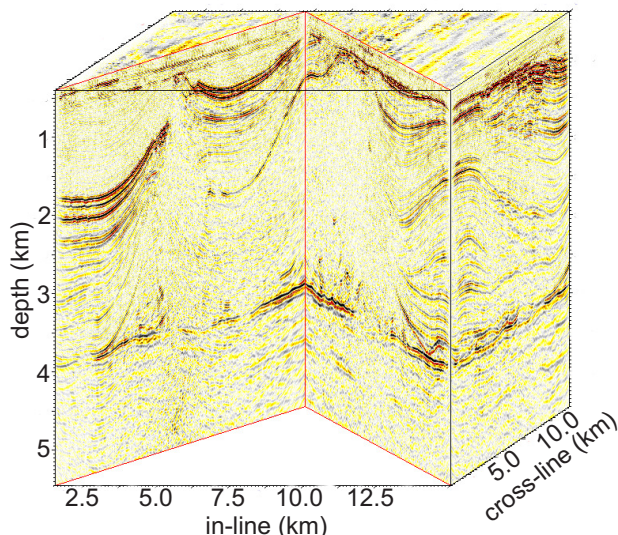


Figure 2. Slalom-line section of the obtained 3D seismic image

Conclusions

We presented an original algorithm of seismic migration, based on the solution of the one-way wave equation. The algorithm combines MPI, OMP, and CUDA technologies. Dataflow is parallelized via MPI so that each node deals with a single dataset. Computations of the Green's functions and the images are performed by GPU. After that, the images are passed between the nodes using MPI. Additional, computations and I/O are implemented via OMP technology.

Acknowledgments

This research was initiated and sponsored by Central Geophysical Expedition JSC of Rosgeo. V. Lisitsa and D. Vishnevsky are also thankful to Russian Foundation for Basic Research for partial financial support of this work, grants No. 18-05-00031, 18-01-00579, 16-05-00800. The research is carried out using the equipment of the shared research facilities of HPC computing resources at Lomonosov Moscow State University supported by the project RFMEFI62117X0011 and cluster NKS-30T+GPU of the Siberian Supercomputer Center.

This paper is distributed under the terms of the Creative Commons Attribution-Non Commercial 3.0 License which permits non-commercial use, reproduction and distribution of the work without further permission provided the original work is properly cited.

References

1. Gazdag, J.: Wave equation migration with the phaseshift method. *Geophysics* 43(7), 1342–1351 (1978), DOI: 10.1190/1.1440899
2. Pleshkevich, A., Vishnevskiy, D., Lisitsa, V.: Development of pseudospectral amplitude-preserving 3d depth migration. *Russian Geophysics (S)* 0, 94–101 (2017)
3. Pleshkevich, A., Vishnevskiy, D., Lisitsa, V.: Explicit additive pseudospectral schemes of wavefield continuation with high-order approximation, pp. 5546–5550 (2017), DOI: 10.1190/segam2017-17633779.1