Geophysical Research Abstracts
Vol. 13, EGU2011-5218, 2011
EGU General Assembly 2011

Speeding Up CRS-Stack Computation Using GPGPU

Alief Nugraha, Andri Hendriyana, and Rachmat Sule
Institut Teknologi Bandung, Geophysical Engineering, Bandung, Indonesia (alief@inet0.net, +62-22-253-4137)

Conventional seismic stack has difficulties in imaging complex subsurface structure. This could lead to produce bad seismic section, especially if the coverage of seismic data is low. Several methods have been introduced in the past years aiming to increase image quality of reflection seismic data. One of those methods is CRS-stack method. As has been already known, velocity analysis is a processing step that should be done carefully in the conventional stacking method. In contrary, the CRS-stack method does not need velocity function to obtain a stack section. Only estimation of near surface velocity model is needed. For 2D case, the CRS-stack method depends on three stacking operators, namely emergence angle, radius of NIP-waves and radius of normal waves. Velocity model is then estimated from the automatic CMP-stack. Since the stacking operator is determined from full coverage data, this method can map the reflectors in a better way, and the S/N-ratio is better than the results of conventional one, especially if the subsurface structure is complex. Several attributes are then produced as by-products, which can be used as interpretation tools.

One deficiency of this method is that this method needs longer processing time, especially if it is applied to a 3D seismic data. In order to overcome this problem, several efforts have been investigated, and one of them is by using GPGPU (General Purpose Graphics Processing Unit) as a form of parallel computation. The advancement of GPGPU for scientific computing is a viable solution for tackling different problem in geophysical computation.

This paper describes the development of new CRS-Stack computation software using NVIDIA CUDA library, which includes the workflow description, algorithm and problem solving strategy for optimizing parallel CRS-Stack computation in NVIDIA GTX 460 GPU. Several comparisons between original CRS-Stack (which only use single processing), CRS-Stack processing by using mpi parallel computing, and CRS-Stack processing by using GPGPU parallel computing are demonstrated in this paper. This study directs to a conclusion that the processing by using GPGPU parallel computing is a promising way in order to speed up the ZO CRS-Stack processing time, without neglecting the advantages of the CRS-Stack itself.