AAPG Datapages/Search and Discovery Article #90255 © 2016 AAPG/SEG International Conference & Exhibition, Barcelona, Spain, April 3-6, 2016

Using Second-Order Adjoint State Methods in GPUS to Quantify Resolution on Full Waveform Inversions

Sergio Abreo[1], Ana Ramirez[1], and Oscar Mauricio Reyes Torres[1]

[1]Universidad Industrial de Santander, Santander, Colombia

Abstract

The Hessian matrix in Full Waveform Inversion (FWI) allows quantifying resolution of the velocity model obtained. Although there are different ways to compute approximations of the Hessian matrix such as BFGS, Newton, Gauss-Newton and Levenberg-Marquardt; our interest is to obtain the exact Hessian matrix. Particularly, we use the Second Order Adjoint State Method (SOASM) to obtain the Hessian matrix-vector products. This work make use of Graphical Processing Units (GPUs) giving the inherent parallelism of the algorithm. In order to obtain the Hessian matrix-Vector products, it is necessary to perform four wave propagations using a finite differences scheme in time. In such scheme, the next layer is computed using information from the current layer and the previous layer. Furthermore, every spatial point of the next layer can be computed independently. In this work, this independence is exploited by an architecture with a high degree of parallelism such as the GPU. This work presents detailed interpretation and implementation of the SOASM theory to take advantage of GPU's architectures. We use a section of the Marmousi velocity model in all the tests. Specifically, the size of the section is 5.25 km x 1.7 km (a grid of 210 x 68 points and spatial resolution of 25 m), which produces a Hessian matrix of 14280x14280 points. We compare the performance of two implementations: Intel Core I7, and Geforce GTX 860M. The CPU and GPU implementations compute a column of the Hessian matrix in 28.04 and 0.05956 seconds, respectively. It means a speedup factor of 470x by using the GPU. In our experiment, it is necessary to compute 14280 columns to obtain the complete Hessian matrix for one shot and one iteration. Assuming a linear performance of both implementations and extrapolating the measured times, we find a lower bound for both implementations using five shots per iteration. For the CPU implementation, the lower bound is 23,17 days whereas the GPU implementation has a lower bound of 1.18 hours.