Geophysical Research Abstracts
Vol. 16, EGU2014-10751, 2014
EGU General Assembly 2014

Fast 2D FWI on a multi and many-cores workstation.

Philippe Thierry (1), Daniela Donno (2), and Mark Noble (2)
(1) Intel Corp., France, (2) MINES ParisTech, France

Following the introduction of x86 co-processors (Xeon Phi) and the performance increase of standard 2-socket workstations using the latest 12 cores E5-v2 x86-64 CPU, we present here a MPI + OpenMP implementation of an acoustic 2D FWI (full waveform inversion) code which simultaneously runs on the CPUs and on the co-processors installed in a workstation.

The main advantage of running a 2D FWI on a workstation is to be able to quickly evaluate new features such as more complicated wave equations, new cost functions, finite-difference stencils or boundary conditions. Since the co-processor is made of 61 in-order x86 cores, each of them having up to 4 threads, this many-core can be seen as a shared memory SMP (symmetric multiprocessing) machine with its own IP address. Depending on the vendor, a single workstation can handle several co-processors making the workstation as a personal cluster under the desk.

The original Fortran 90 CPU version of the 2D FWI code is just recompiled to get a Xeon Phi x86 binary. This multi and many-core configuration uses standard compilers and associated MPI as well as math libraries under Linux; therefore, the cost of code development remains constant, while improving computation time. We choose to implement the code with the so-called symmetric mode to fully use the capacity of the workstation, but we also evaluate the scalability of the code in native mode (i.e running only on the co-processor) thanks to the Linux ssh and NFS capabilities. Usual care of optimization and SIMD vectorization is used to ensure optimal performances, and to analyze the application performances and bottlenecks on both platforms.

The 2D FWI implementation uses finite-difference time-domain forward modeling and a quasi-Newton (with L-BFGS algorithm) optimization scheme for the model parameters update. Parallelization is achieved through standard MPI shot gathers distribution and OpenMP for domain decomposition within the co-processor. Taking advantage of the 16 GB of memory available on the co-processor we are able to keep wavefields in memory to achieve the gradient computation by cross-correlation of forward and back-propagated wavefields needed by our time-domain FWI scheme, without heavy traffic on the i/o subsystem and PCIe bus.

In this presentation we will also review some simple methodologies to determine performance expectation compared to real performances in order to get optimization effort estimation before starting any huge modification or rewriting of research codes. The key message is the ease of use and development of this hybrid configuration to reach not the absolute peak performance value but the optimal one that ensures the best balance between geophysical and computer developments.