Seismic wave propagation on heterogeneous systems with CHAPEL

Alexey Gokhberg (1,2) and Andreas Fichtner (3)

(1) Department of Computer Science, ETH Zurich, Zurich, Switzerland (alexey.gokhberg@inf.ethz.ch), (2) Unicorn Enterprises SA, Schwyz, Switzerland, (3) Department of Earth Sciences, ETH Zurich, Zurich, Switzerland (andreas.fichtner@erdw.ethz.ch)

Simulations of seismic wave propagation play a key role in the exploration of the Earth's internal structure, the prediction of earthquake-induced ground motion, and numerous other applications. In order to harness modern heterogeneous HPC systems, we implement a spectral-element discretization of the seismic wave equation using the emerging parallel programming language Chapel.

High-performance massively parallel computing systems are widely used for solving seismological problems. A recent trend in the evolution of such systems is a transition from homogeneous architectures based on the conventional CPU to faster and more energy-efficient heterogeneous architectures that combine CPU with the special purpose GPU accelerators. These new heterogeneous architectures have much higher hardware complexity and are thus more difficult to program. Therefore transition to heterogeneous computing systems widens the well known gap between the performance of the new hardware and the programmers' productivity.

In particular, programming heterogeneous systems typically involves a mix of various programming technologies like MPI, CUDA, or OpenACC. This conventional approach increases complexity of application code, limits its portability and reduces the programmers' productivity.

We are approaching this problem by introducing a unified high-level programming model suitable for both conventional and hybrid architectures. Our model is based on the Partitioned Global Address Space (PGAS) paradigm used by several modern parallel programming languages. We implemented this model by extending Chapel, the emerging parallel programming language created at Cray Inc. In particular, we introduced the language abstractions for GPU-based domain mapping and extended the open source Chapel compiler (version 1.8.0) with facilities designed to translate Chapel high-level parallel programming constructs into CUDA kernels.

We used this extended Chapel implementation to re-program the package for the simulation of elastic wave propagation and waveform inversion in a spherical section (SES3D). We have tested the resulting package on both conventional CPU-based and heterogeneous GPU-enabled parallel systems. We used the same code base on both architecture classes. To select a desired target architecture for each run we set a few configuration parameters that control the data distribution. The satisfactory performance has been achieved in both homogeneous and heterogeneous cases. This work was supported by a grant from the Swiss National Supercomputing Centre (CSCS) under the project ID d22.