A Performance-Portable Implementation of the Finite Element Assembly in an Atmosphere and Land-Ice Code using the Kokkos Library

Irina Tezaur*, Jerry Watkins* and Irina Demeshko†

* Extreme Scale Data Science & Analytics Department
Sandia National Laboratories
P.O. Box 969, MS 9159
Livermore, CA 94551-0969 USA
e-mail: ikalash@sandia.gov, web page: www.sandia.gov/~ikalash

† Los Alamos National Laboratory
P.O. Box 1663
Los Alamos, NM 87545 USA

ABSTRACT

Performance portability to new and emerging architecture machines (e.g., multi-core, many-core and GPU systems) is becoming a requirement in many applications, climate modeling in particular. As high performance computing architectures become increasingly more heterogeneous, climate modeling tools must also adapt and be more efficient in taking advantage of potential performance capabilities.

Performance portability of a finite element code can be achieved using the Kokkos [1] library and programming model. Developed at Sandia National Laboratories, Kokkos is a library-based programming model that enables computational kernels to be performance portable across many-core architectures. This is accomplished by decoupling computational kernels from device-specific data access performance requirements (e.g., NVIDIA coalesced memory access) through an intuitive multidimensional array API. Specifically, the Kokkos library provides data abstractions to adjust (at compile time) the memory layout of basic data structures like 2D and 3D arrays and allow the transparent utilization of special hardware load and store operations. These features enable Kokkos to maximize the amount of user code that can be compiled for diverse devices and obtain similar performance as a variant of the code that is written specifically for that device.

This talk will detail our effort in developing a performance portable implementation of the finite element assembly in a multi-physics C++ code known as Albany [2], which houses several climate applications. We will describe the process of refactoring the finite element assembly process for two different climate simulation modules within Albany: the Aeras atmosphere solver and the FELIX land-ice dynamical core. We will then show results of strong as well as weak scalability studies on MPI+OpenMP and MPI+GPU frameworks for these two climate applications. We will conclude by discussing several methods for improving the performance of the code for future optimizations.

REFERENCES

[1] H. Carter Edwards, Christian R. Trott, and Daniel Sunderland. Kokkos: Enabling manycore performance portability through polymorphic memory access patterns. Journal of Parallel and Distributed Computing, **74**(12):3202–3216, 2014.

[2] I. Demeshko, W. Spotz, I. Tezaur, O. Guba, A. Salinger, R. Pawlowski, M. Heroux. "Towards performance-portability of the Albany finite element analysis code using the Kokkos library", submitted to *J. HPC Appl.*