Geophysical Research Abstracts
Vol. 18, EGU2016-17098-1, 2016
EGU General Assembly 2016

Exploiting on-node heterogeneity for in-situ analytics of climate simulations via a functional partitioning framework

Karan Sapra (1), Saurabh Gupta (2), Scott Atchley (1), Valentine Anantharaj (1), Ross Miller (1), and Sudharshan Vazhkudai (1)

(1) Oak Ridge National Laboratory, National Center for Computational Sciences, Oak Ridge, United States (anantharajvg@ornl.gov), (2) Clemson University, Clemson SC, United States of America

Efficient resource utilization is critical for improved end-to-end computing and workflow of scientific applications. Heterogeneous node architectures, such as the GPU-enabled Titan supercomputer at the Oak Ridge Leadership Computing Facility (OLCF), present us with further challenges. In many HPC applications on Titan, the accelerators are the primary compute engines while the CPUs orchestrate the offloading of work onto the accelerators, and moving the output back to the main memory. On the other hand, applications that do not exploit GPUs, the CPU usage is dominant while the GPUs idle.

We utilized Heterogenous Functional Partitioning (HFP) runtime framework that can optimize usage of resources on a compute node to expedite an application's end-to-end workflow. This approach is different from existing techniques for in-situ analyses in that it provides a framework for on-the-fly analysis on-node by dynamically exploiting under-utilized resources therein.

We have implemented in the Community Earth System Model (CESM) a new concurrent diagnostic processing capability enabled by the HFP framework. Various single variate statistics, such as means and distributions, are computed in-situ by launching HFP tasks on the GPU via the node local HFP daemon. Since our current configuration of CESM does not use GPU resources heavily, we can move these tasks to GPU using the HFP framework. Each rank running the atmospheric model in CESM pushes the variables of of interest via HFP function calls to the HFP daemon. This node local daemon is responsible for receiving the data from main program and launching the designated analytics tasks on the GPU.

We have implemented these analytics tasks in C and use OpenACC directives to enable GPU acceleration. This methodology is also advantageous while executing GPU-enabled configurations of CESM when the CPUs will be idle during portions of the runtime. In our implementation results, we demonstrate that it is more efficient to use HFP framework to offload the tasks to GPUs instead of doing it in the main application. We observe increased resource utilization and overall productivity in this approach by using HFP framework for end-to-end workflow.