**An asynchronous GPU algorithm for extreme scale fourier pseudo-spectral simulation of turbulence**
Kiran Ravikumar, David Appelhans, P. K. Yeung, Oscar Hernandez

Georgia Institute of Technology

kravikumar7@gatech.edu

**Abstract**

An asynchronous GPU-driven Fourier pseudo-spectral algorithm has been developed on Summit to enable turbulence simulations at a problem size exceeding the prevailing state of the art. The primary question is how to re-design communication-intensive applications for maximum benefit from emerging pre-Exascale machine architecture, which is characterized by a shift from massive CPU-based parallelism to heterogeneous platforms with fat nodes and fast accelerators.  In this poster we present the basic elements of such an algorithm.  First, communication performance is improved by using fewer MPI processes each with multiple cores and multiple GPUs, as well as coalescing multiple smaller messages into fewer but larger messages.  Second, we overcome problem size limitations associated with the relatively small GPU memory by processing data from host memory asynchronously in smaller batches on GPUs. Third, we have used unique CUDA Fortran features to optimize data transfer between the CPU and the GPU, and to overlap these transfers with highly-optimized computations on the GPUs.  Favorable performance is obtained for our new turbulence code up to a $18432^3$ problem size on 3072 nodes of Summit, with a GPU to CPU speedup of 4.7 at $12288^3$ (the largest problem size previously published in turbulence literature). OpenMP implementation of this algorithm is also under development. OpenMP and cudaFFT library interoperability has proven to be challenging especially when asynchronous execution is introduced on both programming models. We are also exploring how to manage buffers on the accelerator for copying a large host resident memory in batches to device memory without allocating and deallocating memory on the device and we are exploring solutions in the context of OpenMP 5.0 or extensions for OpenMP 5.1/6.0.