1 Abstract

To explore the relative merits between single-node computer power in depth versus multi-node compute power in breadth, I propose an experiment comparing the performance and efficiency of Cannon's algorithm[2, 1] for the matrix multiplication problem $C = A \times B$ between a single-node OpenMP implementation utilizing accelerators and a multi-node MPI implementation spread across a cluster.

1.1 Data

For the sake of time, I will only consider the C-stationary case for a limited number of matrix sizes (256x256, 1024x1024, 4096x4096, 16384x16384, 256x1024, 256x4096, 256x16384, 16384x256, 16384x256, 4096x256, 1024x256). The matrices will be filled with random numbers for the performance trials. Validation trials will be made through specially devised matrices A and B such that each cell of the product matrix C will have a unique value.

1.2 Resources

The OpenMP trials will be executed on the GPU nodes on BRIDGES[1] and/or the KNL nodes on STAMPEDE2, whereas the MPI trials will be executed on the appropriate non-accelerated nodes on BRIDGES.

1.3 Efficiency

The efficiency of the MPI trials will be gauged against a strong-scaling roofline model based on the physical characteristics of a single non-GPU node on BRIDGES. The processor in this case is the 28-core, 2.30 GHz Intel E5-2695, which is theoretically capable of 2.30 GHz * 28 cores * 4 SIMD instructions/cycle (AVX256) = 257.6 GFLOPs/s/node.

The efficiency of the OpenMP trials will be gauged against a roofline model based on the physical characteristics of the accelerator. The NVIDIA P100 is theoretically capable of 9.3 SP TFLOPs/s/card[3], whereas the NVIDIA K80 is theoretically capable of 8.74 TFLOPs/s/card[4]. In the event that KNL nodes on STAMPEDE2 are utilized, the theoretical roofline will be considered as 1.4 GHz * 68 cores * 8 SIMD instructions/cycle (AVX512) = 761.6 GFLOPs/s/node[5][2].

[1] If the GPU nodes on BRIDGES do not support OpenMP offloading to the P100 and K80 GPUs, the acceleration will be done via OpenACC, or, as a last resort, CUDA.

[2] While KNL supports 4 threads/core, only 1 is considered here as performance may degrade over shared resources.

The matrix multiplication functions provided with the Intel Math Kernel Library (MKL) will be consulted as a reference for empirically achievable single-node performance either without GPU acceleration or in the case of KNL nodes while the matrix multiplication sample provided by the CUDA toolkit utilizing CUBLAS functions will be consulted as a reference for empirically achievable single-node performance with GPU acceleration.

1.4 Hypotheses

I predict that the accelerated OpenMP solution will outperform the MPI solution for smaller matrix sizes until reaching an inflection point where the contention between resources within the node is a greater bottleneck than the communication overhead across multiple nodes. This inflection point will largely depend on the architecture and layout of the accelerator used, i.e., the dimensions of the various hierarchical groupings of its compute resources.

References

[1] H. Gupta and P. Sadayappan. *Communication Efficient Matrix-Multiplication on Hypercubes*. 1994. URL: `http://ilpubs.stanford.edu:8090/59/1/1994-25.pdf` (visited on 11/13/2017).

[2] Lennart Johnsson. *Introduction to HPC Lecture 17. Parallel Algorithms: Matrix Multiplication*. Department of Computer Science, University of Houston. Oct. 24, 2017.

[3] *NVIDIA® Tesla® P100 GPU Accelerator*. NVIDIA Corporation. Oct. 6, 2016. URL: `https://images.nvidia.com/content/tesla/pdf/nvidia-tesla-p100-PCIe-datasheet.pdf` (visited on 11/13/2017).

[4] Ryan Smith. *NVIDIA Launches Tesla K80, GK210 GPU*. Nov. 14, 2017. URL: `https://www.anandtech.com/show/8729/nvidia-launches-tesla-k80-gk210-gpu` (visited on 11/13/2017).

[5] *Stampede2 User Guide*. Texas Advanced Supercomputing Center, University of Texas at Austin. Oct. 22, 2017. URL: `https://portal.tacc.utexas.edu/user-guides/stampede2` (visited on 11/13/2017).