Kumpula Computational Chemistry and Physics 7.12.2022

GPU programming using C++, Fortran and Python

Jeff Hammond, NVIDIA

Abstract

I will describe the current state of parallel programming models for GPUs for users of C++, Fortran and Python, with examples relevant to computational chemistry and physics. The newest features here are support for ISO language parallelism, based on C++17 and Fortran 2008, along with multi-GPU support for Numpy. The other options include directives (OpenMP and OpenACC), which support Fortran, C and C++, and CUDA, which has support in Fortran, C, C++ and Python. Because the range of choices can be overwhelming, I'll compare all of them for simple operations to demonstrate the tradeoffs in performance and portability/productivity. I'll conclude with some results with the results for the quantum many-body methods known as coupled-cluster theory found in NWChem.

Biography

Jeff Hammond is a Principal Engineer at NVIDIA, based in Helsinki. He works on open standards for parallel programming, including the MPI, OpenMP, Fortran and C++. Jeff received his PhD in Chemistry from the University of Chicago and has been a developer of computational chemistry software (NWChem, mostly) for more than 15 years. See https://github.com/jeffhammond