Exploring Performance of GeoCAT data analysis routines on GPUs

Haniye Kashgarani¹, Cena Miller², Supreeth Suresh², and Anissa Zacharias²

¹University of Wyoming, ²National Center for Atmospheric Research

GeoCAT

The GeoCAT-comp program is one of the GeoCAT repositories, including previous NCAR Command Language (NCL)'s non-WRF (Weather Research and Forecasting model) computational routines and other geoscientific analysis functions in Python.

GeoCAT-comp is built on the Pangeo software ecosystem. The routines in GeoCAT-comp are either sequential or take advantage of Dask for parallelization on the CPU.

CPU nodes: 2 18-core 2.3-GHz Intel Xeon Gold 6140 (Skylake) processors per node
8 NVIDIA Tesla V100 32GB SXM2 GPUs with NVLink

Challenges

- Adapting Xarray and Dask with CuPy
- Inability to get performance improvements with some GPU tasks, e.g., Search functions: xarray.where()
- Numba JIT compiler auto-parallelizes NumPy arrays on CPU, but it is not adapted to CuPy arrays
- Correct way for benchmarking and gathering data:
  - Setting the correct chunksize

Performance Comparison (Only Computation Time for GPUs):

Scalability: Strong and Weak Scaling

Conclusion and Future Work

- Explored ways to port GeoCAT-comp to run on GPUs
- Provided a template to port other GeoCAT-comp routines to GPU
- Ported some serial and CPU parallelized GeoCAT-comp routines to GPU, and analyzed the performance
- Validated the results of NumPy and CuPy to a precision of 10⁻⁷

Future Work:

- Port other GeoCAT-comp routines
- Push the ported code to production
- Investigate writing kernel functions with Numba, and cuNumeric

Acknowledgement

Thanks to Cena Miller, Supreeth Suresh, and Anissa Zacharias for their mentorship.

Special thanks to ASAP, GeoCAT, CSG, and SIParCS Team!

Ported Branch:

GeoCAT Github:

hkashgar@uwyo.edu