Performance Analysis of Machine Learning Algorithms for Phylanx: An Asynchronous Array Processing Toolkit

Weile Wei, Rod Tohid, Bibek Wagle, Shahrzad Shirzad, Parsa Amini, Bita Hasheminezhad, Katy Williams, Adrian Serio, Hartmut Kaiser
Louisiana State University

Abstract
Phylanx is an asynchronous array processing toolkit which transforms Python and NumPy operations into code which can be executed in parallel on HPC resources by mapping Python and NumPy functions and variables into a dependency tree executed by HPX, a general purpose, parallel, task-based runtime system written in C++. In this poster, we present early results that compare our implementation of widely used machine learning algorithms to accepted NumPy standards.

Performance Results

Phylanx program flow: Phylanx frontend generates an AST (PhySL) of the decorated Python code. The AST could be directly passed to the compiler to generate the execution tree or, optionally, fed to the optimizer first and then the compiler. Once the Kernel is invoked, Phylanx triggers the evaluation of the execution tree on HPX. After finishing the evaluation, the result is returned in Python.

Visualization of AST using Traveler Tool

References

Acknowledgments
This material is based upon work supported by the National Science Foundation under Grant No. 1737785. Any opinions, findings, conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

This work is supported by The Defense Technical Information Center under the contract: DTIC Contract FA8075-14-D-0002/0007.

https://github.com/STEllAR-GROUP/phylanx wwei9@lsu.edu