MPC in Practice Research Plan

Tianxing Wu

January 21, 2024

1 Research Purpose

The purpose of this research is to explore and understand the compatibility between various message-passing frameworks (such as the k-machine model, congested clique, and BSP) and different distributed computing frameworks, including MPI, OpenMP, Pytorch, and TensorFlow.

Evaluation of Compatibility/Incompatibility:

1. Evaluating Performance:
    - How does this framework perform in comparison to another framework with the same message-passing model and problem, in terms of communication rounds and memory usage?
    - How does this framework's performance change if the problem scales, in terms of strong scaling and weak scaling?
2. System Requirements from the Programming Framework:
    - Memory usage
    - CPU usage
    - Network bandwidth
    - Disk I/O

2 Objective

For a specific framework and model, the objective is to demonstrate one of the following:

- The framework and model are perfectly compatible.
- The framework and model are compatible with some additional effort, enabling them to work together efficiently to solve problems.
- The framework and model are completely incompatible.

To demonstrate compatibility, algorithms will be presented with accompanying proofs. To indicate incompatibility, proofs will also be required.

3 Method

- Reviewing and understanding the documentation of programming frameworks.
- Analyzing the compatibility of frameworks and models:
    1. Selecting classical problems that require High-Performance Computing (HPC), such as graph problems.
    2. Designing algorithms using the selected frameworks and models.
    3. Analyzing performance in terms of time and memory usage, considering scalability.
    4. Concluding the compatibility between a specific model and a framework.
    5. (Optional) Applying the algorithm on real machines to verify the results.

4 Plan

The duration of this research is planned for 6 months.

Months 1 - 2:

- Literature Review. Conduct a comprehensive review of existing studies on MPI, OpenMP, PyTorch, TensorFlow, and message-passing models like the k-machine model, congested clique, and BSP.
- Selecting problems for analysis.
- Developing metrics for evaluating performance.

Months 3 - 4:

- Design algorithms tailored to the frameworks and models, analyze their performance, and evaluate them using the developed metrics.

Months 5 - 6:

- Writing the research report.

Optional:

- Applying the algorithms on real machines to verify the results.

References

[1] Leonardo Dagum and Ramesh Menon. "OpenMP: an industry standard API for shared-memory programming". In: *IEEE Computational Science Engineering* 5.1 (1998), pp. 46–55.

[2] Arvind Krishnamurthy and John Zahorjan. *Parallel Algorithms*. University of Washington. 2002.

[3] Martín Abadi et al. *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems*. Software available from tensorflow.org. https://www.tensorflow.org/. 2015.

[4] *MPI: A Message-Passing Interface Standard Version 3.1*. Message Passing Interface Forum. 2015. URL: https://www.mpi-forum.org/docs/.

[5] John Augustine et al. "Distributed Computation in the Node-Congested Clique". In: *arXiv preprint arXiv:1805.07294v1* (2018).

[6] J. Augustine, K. Kothapalli, and G. Pandurangan. "Efficient Distributed Algorithms in the k-machine model via PRAM Simulations". In: *2021 IEEE International Parallel and Distributed Processing Symposium (IPDPS)*. 2021, pp. 223–232. DOI: 10.1109/IPDPS49936.2021.00031.