

CSCI 5551 Parallel and Distributed Systems Term Project Proposal

**Parallel Processing with MapReduce in Big Data System**

Qian Mai, Bogki Yun

Computer Science and Engineering

University of Colorado at Denver

**Abstract**

With the fast development of data analysis, data structure and data processing, a popular term is widely used these days to describe the significant growth of data - Big data. It involves any collection of data sets which are large in size and complex in structure, thus becoming difficult for traditional data processing applications to handle. However, using a MapReduce programming can help simplify big data analysis [1][4].

MapReduce is a programming model and an associated implementation for processing and generating large data set with a parallel, distributed algorithm on a cluster. It is composed of two procedures: Map() and Reduce(). Map() procedure that performs filtering and sorting while Reduce() procedure performs a summary operation [3].

To storage and process large-scale data-set on clusters of commodity hardware, we can use an open-source software framework called Hadoop [6], which is an Apache top-level project being built and used by a global community of contributors and users. Hadoop Distributed File System (HDFS) is a distributed file system which stores data on commodity machines, providing very high aggregate bandwidth across the cluster [2][7].

Our goal, in this term project, is the comparison the MapReduce program performance at each node which is single, 2, 4, 8, and 16 of Hadoop system. We will install a Hadoop on Hydra and evaluate the runtime overheads using two benchmarks (namely, Sort, WordCount) in the HiBench Hadoop benchmark suite [8], as shown below Table, and tune the MapReduce configuration to get a better performance. And we will evaluate and analyze the results.

Table 1. Hadoop benchmark

|  |  |
| --- | --- |
| **Benchmark** | **Input Data** |
| Sort | 100 GB Amazon Public Data Sets |
| WordCount | 100 GB Amazon Public Data Sets |

**Motivation and Problem**

Currently, big data is popular and maybe as important to business and society in many aspects- as the Internet has become an indispensable part of our daily life. However, the challenges of handling big data usually include analysis, capture, curation, search, sharing, storage and transfer, visualization, and privacy violations. A MapReduce is known for a prominent parallel data processing tool and is gaining significant momentum from both industry and academia as the volume of data to analyze grows rapidly. As is known to all, MapReduce performance is highly related to Big Data system environment, for example, the number of nodes in Hadoop, some configuration such as Shuffle, Combiner class. That’s why we choose this topic as our term project.

**Solution**

To realize above goal, we will set a Big Data system on Hydra and experiment MapReduce program with 100GB data at a various nodes, single, 2, 4, 8, and 16. To implement it and evaluate the results, we will follow as:

1) Research the MapReduce, and Hadoop which are a basic of Big Data System.

2) Getting the Big Data (100GB) using by Amazon S3

3) Install Big Data system, Hadoop on Hydra and setup HiBench Hadoop benchmark suite

4) Evaluate runtime overheads using two benchmark MapReduce Program on the Big Data system while appending the nodes, 2, 4, 8, and 16 and changing the configuration variable of MapReduce such as Shuffle, Combiner class, etc.

5) Analyze the Results and make Final Report

**Project Milestone**

|  |  |
| --- | --- |
| 10/13 ~ 10-19 | Proposal of the Project |
| 10/20 ~ 10/27 | Research and Study Big Data System (MapReduce, Hadoop). |
| 10/28 ~ 11/10 | 1. Install Big Data System (Hadoop) on Hydra  2. Collecting Source Data (100GB) at Amazon S3 |
| 11/11 ~ 11/14 | Making MapReduce programming for analyzing the Big Data |
| 11/15 ~ 11/28 | Execute the MapReduce Program on different nodes (single, 2, 4, 8, 16) |
| 11/29 ~ 12/09 | Evaluate the results and complete the final project report |

**Application and Data Resources**

Hadoop, HDFS, Public Data Sets by Amazon, HiBench Hadoop benchmark suite, Java Programming, and MapReduce

**References**

[1] Kyong-Ha Lee, Yoon-Joon Lee, Hyunsik Choi, Yon Dohn Chung, Bongki Moon, “Parallel Data Processing with MapReduce: A Survey”, *In SIGMOD Record*, December 2011 (Vol. 40, No. 4)

[2] Konstantin Shvachko, Hairong Kuang, Sanjay Radia, Robert Chansler, “The Hadoop Distributed File System”, *In Proc, IEEE* 2010,

[3] MapReduce Wikipedia, http://en.wikipedia.org/wiki/MapReduce

[4] J. Dean, S. Ghemawat, “MapReduce: Simplified Data Processing on Large Clusters,” *In Proc. of the 6th Symposium on Operating Systems Design and Implementation*, San Francisco CA, Dec. 2004.

[5] Amazon Public data sets, common crawl corpus, http://aws.amazon.com/datasets/41740

[6] Apache Hadoop, [http://hadoop.apapche.org](http://hadoop.apapche.org/)

[7] The Hadoop Distributed File System: Architecture and Design,   
https://hadoop.apache.org/docs/r0.18.0/hdfs\_design.pdf

[8] S. Huang, J. Huang, J. Dai, T. Xie, B. Huang, “The HiBench Benchmark suite: Characterization of the Mapreduce-Based Data Analysis”, *In Proc. of IEEE 26th International Conference on Data Engineering Workshops*, 2010