Performance Improvement of MapReduce Applications using Flame-MR

Jorge Veiga, Roberto R. Expósito, Guillermo L. Taboada, Juan Touriño
{jorge.veiga, rreye, taboada, juan}@udc.es

January 10, 2017

Apache Hadoop is a popular MapReduce framework that has been widely adopted by many organizations. However, its performance limitations, mainly caused by redundant memory copies and high disk overhead, can hinder its usage for large scale data analytics. Although emerging alternatives like Spark and Flink are able to improve the performance of Hadoop, applications must be completely rewritten. This is not always feasible due to the high cost of rewriting existing applications or the impossibility to access unavailable source codes.

The proposal developed in this PhD Thesis is Flame-MR (http://flamemr.des.udc.es/), a new Java-based MapReduce framework that improves the performance of Hadoop while keeping compatibility with existing applications. Flame-MR is based on an event-driven architecture that leverages in-memory computing, avoiding redundant memory copies and using static memory allocation to reduce Java garbage collections. It makes use of efficient sort and merge algorithms that adjust the data being processed to the available memory space. Moreover, iterative workloads are also efficiently supported by reusing Java processes and caching intermediate results in memory, thus avoiding unnecessary read/write operations from/to HDFS.

Flame-MR has been evaluated comparatively with Hadoop on the Amazon EC2 cloud platform, using a 32-node cluster. The performance evaluations using representative workloads has shown significant performance increases, reducing Hadoop execution times by 55% on average. Future work will determine the performance benefits of Flame-MR in real-world use cases, while also studying further optimizations configurable by the user (e.g. automatic load balancing).

Keywords: Big Data; MapReduce; Hadoop; In-memory Computing; Cloud Computing