© 2016 IJSRSET | Volume 2 | Issue 4 | Print ISSN: 2395-1990 | Online ISSN : 2394-4099
Themed Section: Engineering and Technology

# A Survey on Hadoop-Mapreduce Environment with Scheduling Algorithms in Big Data

**Swathi Kiruthika V.*, Dr. Thiagarasu V.**

Department of Computer Science Gobi Arts & Science College, Gobichettipalayam, Erode, TamilNadu, India

## ABSTRACT

Hadoop and Map Reduce are the most efficient tools which are used to reduce the complexity of maintaining the big data set. MapReduce has been introduced by Google and it is an open source counterpart. Hadoop is focused for parallelizing computing in large distributed clusters of commodity machines. Thus the parallelizing data processing tool MapReduce has been gaining significance moment from both academy and industries. The objective of this survey is to study MapReduce with different algorithms to improve the performance in large dataset.

**Keywords:** Bigdata, Hadoop, MapReduce, Straggler, Data Skew and Job Scheduling.

## I. INTRODUCTION

Big data can be structured, unstructured or semi structured in its nature of formation. Traditional database management systems are not often acceptable for storing big data as it is difficult to process it. Map Reduce framework was introduced in 2004 by Google which is suitable for parallel data processing in the distributed computing environment and it handles processing of big data set in the distributed computing environment, proposed by Min Chen [2014].

### Hadoop

Hadoop is an APACHE open source framework tool which allows distributed processing of large datasets with stream run clusters, data access pattern on commodity hardware. Hadoop distributed file system (HDFS) has two components to measure the nodes such as NameNode and DataNode. NameNode is the master node also called as JobTracker is very expensive hardware used to maintain and manage the blocks on DataNodes. Across the Datanode, the Namenode splits and stores files in 64MB and 128MB data blocks. DataNode works as a slave actually stored the data and deployed on each machine and it is also responsible to read and write requests from client proposed by Poonam S. Patil [2014].

Figure 1 : shows the architecture of HDFS, which shows that rack is the storage area where multiple DataNode put together. To maintain the fault tolerance on Hadoop, multiple copies of data are maintained on multiple DataNode. This is called data replication and each copy of data is called replica proposed by Suryawanshi [2014].

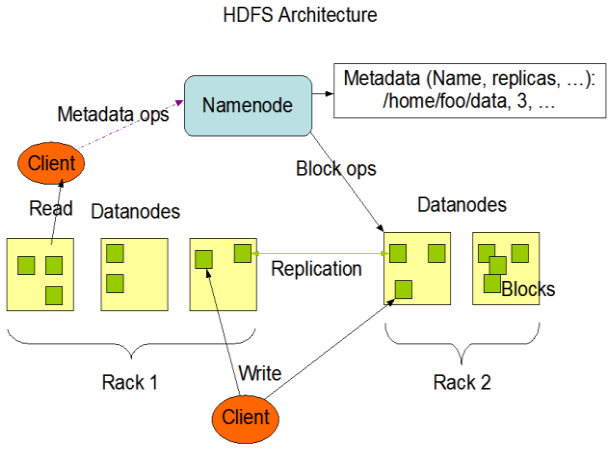


**Figure 1.** Architecture of HDFS (Hadoop Distributed File System)

### Map Reduce

MapReduce has two phases such as Map function and Reduce function. Map function takes the raw data like text file as an input then that input splits into several parts and each split is fed to separate map tasks.

Output is then fed to a combiner function which is a user defined function and it is used to reduce the tasks. The second phase Reduce phase fetches the intermediate results and carries out the computation process for the final result. The MapReduce paradigm is not an appropriate solution for the kind of low-latency processing because: MapReduce computations are batch processes that start and finish, while computations over streams are continuous tasks that only finish upon user request. The inputs of MapReduce computations are snapshots of data stored on files and the content of these files do not change during processing proposed by Mahesh Maurya [2011] and Ayma [2015].

**Straggler in MapReduce**

Straggler is a slowest running task which delays the execution of all the running jobs because of either internal or external factors. Straggler varies in two factors one is heterogeneous environment and second is homogeneous environment. In heterogeneous environment server OS and client OS are same so problems gets only because of different disk and computing capacity. But, in homogeneous straggler occurs because of data skew which is unable to resolve by simply transferring the task to other machine. Stragglers in heterogeneous environment are overcome by the method called speculative execution which can overcome it simply by transferring the overload to some other machine when the machine performing slowly or when some fault occurs in running task. The straggler caused by data skew issue and it occurs on both phases in MapReduce environment proposed by Qi Chen [2015].

## II. MAPREDUCE SCHEDULING

### i) FIFO Scheduler

FIFO is the default Hadoop scheduler. The main objective of FIFO scheduler is to schedule jobs based on their priorities in first-come first-out of first serve order. The FIFO scheduler operates using a FIFO queue. Job is divided into independent tasks and then they are put into the queue and allotted to free slots as they get acquirable on TaskTracker nodes proposed by Pakize [2015].

### ii) Fair Scheduler

Users may assign jobs to pools, with each pool allocated a guaranteed minimum number of Map and Reduce slots. Free slots in ineffective pools may be allocated to new pools. Its preemptive technique that means the scheduler will kill tasks in pools running over capacity in order to give the slots to the pool running under capacity. Priority criteria are also assigned to various pools and tasks are scheduled in interleaved manner based on priority proposed by Pakize [2015].

### iii) Capacity Scheduler

Capacity Scheduler was developed by Yahoo and it addresses a usage scenario where the number of users is large, and there is a need to ensure a fair allocation of computation resources amongst users. This uses queues instead of pool, each queue is assigned to an organization, and resources are divided among these queues proposed by Pakize [2015].

### iv) Longest Approximate Time to End (LATE

Longest approximate time to end algorithm tries to improve Hadoop by attempting to find real slow tasks by computing remaining time of all the tasks it ranks tasks by estimated time remaining and starts a copy of the highest ranked task that has a progress rate lower than the Slow Task Threshold. LATE is based on three principles: prioritizing tasks to speculate, selecting fast nodes to run on, and capping speculative tasks to prevent thrashing proposed by Pakize [2015].

### v) Delay Scheduling

Delay scheduling method performs well in Hadoop workloads because Hadoop tasks are short relative to jobs, and because there are multiple locations where a task can run to access each data block. When a node requests a particular task, if the head of line job can't assign local task, scheduler skips that task and starts looking for next jobs proposed by Pakize [2015].

Following table shows relative comparison of all the scheduling algorithms:

| Algorithms | Implementation | Response Time | | Execution | | Load Balancing | | Data Locality | | Fairness | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Low | High | Serial | Parallel | Yes | No | Low | High | Yes | No |
| FIFO Scheduling | Simple | ✓ | - | ✓ | - | - | ✓ | ✓ | - | - | ✓ |
| Fair Scheduling | Less Complex | - | ✓ | - | ✓ | ✓ | - | ✓ | - | ✓ | - |
| Capacity Scheduling | Complex | - | ✓ | - | ✓ | ✓ | - | ✓ | - | ✓ | - |
| Late Scheduling | Complex | - | ✓ | - | ✓ | ✓ | - | - ✓ | | ✓ | - |
| Delay Scheduling | Complex | - | ✓ | - | ✓ | ✓ | - | ✓ | - | ✓ | - |

## III. LITERATURE SURVEY

Reddawayr [1991] expressed that, the retrieval text document based on word was been implemented by Active Memory Technology(AMT) on Distributed Array Processor for parallel computers just for a 20 giga byte database of 10million documents, which assist around 70 boolean queries per second. Compression method is used in the large data set which varies based on the parameters. A variety of compressed forms are there in an index hit data that expands to document bitmap form and it combines the data in different terms.

Biliris [1992] conferred the algorithm as well as structure for the large unstructured data sets. These large data set objects are stored in a disk blocks with sequence of variable size segments. Also, it is described that the structure of Enterprise Operating System (EOS) is a storage system for experimental database. The binary buddy system is used in the disk space management and the operations such as replace, insert and delete bytes are handled within the object. Following are the principle objectives of EOS:

1. Piece-wise operations are handled by EOS: such as append bytes in the end, read and replace a byte range, insert or delete bytes handled within the object.
2. Cost of the piece-wise operation is based on the number of bytes used in the operation not based on the size of the entire object.
3. Disk space is fast in the object allocation process.

Ghemawat and Dean [2004] represented that they have designed and implemented the Google File System, a scalable distributed file system for large distributed data-intensive applications and stated that MapReduce is a programming model which is used to generate the large dataset using Map and Reduce function. Map function is used to set a key/value pair to generate the data and Reduce function is used to merge the values which are associated with the same intermediate key.

Again Ghemawat [2006] described about a distributed storage system Bigtable, which manages the structured data and also bigtable is designed to scale a very large scale petabytes data around thousands of commodity servers. In this period, many Google projects have stored the data in Bigtable. B.Patel [2012] addressed about bigdata using Hadoop and MapReduce: it stated that, bigdata describes the optimal solutions using hadoop cluster, hadoop distributed file system (HDFS) for storage and mapreduce framework for parallel processing to access the large datasets.

Puneet Singh Duggal [2013] described on bigdata analysis tools, hadoop, HDFS and MapReduce which are used to store and manages bigdata and helps organizations to understand better customers and market. MapReduce techniques implemented using HDFS for the bigdata analysis. Sagiroglu, S. and Sinanc D. in 2013 describes about the content, scope, methods, samples, advantages and challenges of bigdata and states that privacy and security are the main issues caused in bigdata environment. Min Chen [2014] stated on cloud computing, hadoop and focuses on 4 phases they are data generation, data acquisition, data storage and data analysis which are the value chains of bigdata. P.Sarada Devi [2014] introduced about ETL (Extra Transform Load) which introduces for taking business intelligence decisions in hadoop.

Poonam S.Patil [2014] describes on RDBMS, NoSQL, hadoop, MapReduce are the challenges to deal analysis of bigdata and it is very flexible to use any other languages to write algorithm and the database of bigdata is splits among three dimensions: volume, variety and velocity. K.Arun [2014] states mining techniques like association rule learning, clustering classification that it helps in decision making in business environment by implementing data mining techniques. Mukherji.A [2014] describes that bigdata analysis the large amount of data to get the useful information and to uncover the hidden data patterns and it is refers to the mapreduce framework, which is developed by the google.

## III. CONCLUSION

As a volume of database increases day by day traditional frequent itemset mining algorithms becomes inefficient. This survey stated that straggler is the main issue that causes the system running slow. To improve the data processing method in the MapReduce, it is important to overcome the dataskew in MapReduce applications. It is observed that dataskew can be overcome by using MR-DBSCAN (MapReduce-Density Based Spatial Clustering of Application with Noise) method can be applied with frequent itemset algorithm and all the critical sub-procedures are completely parallelized.

## IV. REFERENCES


[1] Min Chen, Shiwen Mao and Yunhao Liu, "Big Data: A Survey", Springer, New York, Volume 19, Issue 2, April 2014.

[2] Poonam S. Patil and Rajesh N. Phursule, "Survey Paper on Big Data Processing and Hadoop Components", International Journal of Science and Research (IJSR), Volume 3, Issue 10, October 2014.

[3] Suryawanshi, Shital, and V. S. Wadne, "Big Data Mining using Map Reduce: A Survey Paper", IOSR Journals (IOSR Journal of Computer Engineering), Volume 16, Issue 6, December 2014.

[4] Ayma, "Classification Algorithms for Big Data Analysis, a MapReduce Approach", The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences, Volume XL-3/W2, March 2015.

[5] Mahesh Maurya and Subita Mahajab, "Comparative analysis of MapReduce job by keeping data constant & varying cluster size technique", Elsevier, Volume 5, Issue 11, May 2011.

[6] Qi Chen, Jinyu Yao and Zhen Xiao, Senior Member, IEEE, "LIBRA: LightWeight DataSkew Mitigation in MapReduce", IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS, Volume 26, Issue 9, September 2015.

[7] Pakize, Seyed Reza, "A comprehensive view of Hadoop MapReduce scheduling algorithms", " International Journal of Computer Networks & Communications Security ", Volume 2, Issue 9, September 2014.

[8] Reddaway.S, "Information Processing & Management", Springer, New York, Volume 27, Issue 4, 1991.

[9] Biliris A, "An Efficient Database Storage Structure for Large Dynamic Object", IEEE Data Engineering Conference, Phoenix, Arizona, February 1992.

[10] Ghemawat S. and Dean.J, "MapReduce: Simplified Data Processing on Large Clusters", OSDI '04: 6th Symposium on Operating Systems Design and Implementation, August 2004.

[11] Puneet Singh Duggal and Sanchita Paul, "Big Data Analysis: Challenges and Solutions", International Conference on Cloud, Big Data and Trust, RGPV, 2013.

[12] K.Arun and Dr. L. Jabasheela, "Big Data: Review, Classification and Analysis Survey", International Journal of Innovative Research in Information Security (IJIRIS), Volume 1, Issue 3, 2014.

[13] Mukherjee A., Datta J., Jorapur R., Singhvi R., Haloi S., Akram W., "Shared disk big data analytics with Apache Hadoop", "Institute of Electrical and Electronics Engineers (IEEE)", Volume 12, December 2012.