International Journal of Engineering & Technology, *7 (3.31) (2018) 59-65*

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET

Research paper

A Study on Big Data Hadoop Map Reduce Job Scheduling

N. Deshai*, S. Venkataramana, I. Hemalatha, G. P. S. Varma

Department of Information Technology, S.R.K.R Engineering College, Bhimavaram, Andhra Pradesh, India-534201
**Corresponding author E-mail: desaiphd7@gmail.com; Tel: +91-9948572990*

Abstract

A latest tera to zeta era has been created during huge volume of data sets, which keep on collected from different social networks, machine to machine devices, google, yahoo, sensors etc. called as big data. Because day by day double the data storage size, data processing power, data availability and digital world data size in zeta bytes. Apache Hadoop is latest market weapon to handle huge volume of data sets by its most popular components like hdfs and mapreduce, to achieve an efficient storage ability and efficient processing on massive volume of data sets. To design an effective algorithm is a key factor for selecting nodes are important, to optimize and acquire high performance in Big data. An efficient and useful survey, overview, advantages and disadvantages of these scheduling algorithms provided also identified throughout this paper.

Keywords: *Big data; Hadoop; HDFS; Map Reduce; Scheduling;*

1. Introduction

We are working in the computerized advanced world information. The real world moving to big data day by day[1], where a massive data set from many of sources as Face book, Yahoo, Google, YouTube, Amazon, Microsoft and Twitter, eBay, diverse sensor systems, Airlines record, Global Position System, RFID per users, PC logs, Closed Circuit cameras, Internet of Things and bunches of other field sources. All these data makes complexities in dealing with and using. Lesser and lesser the comprehension of information is greater and greater it would progress to becoming. There are some planning viewpoints in huge information Hadoop and Map Reduce. Schedulers are in charge of errand task in Map Reduce where relegating the assignments to a specific information hub. The accessibility and the information area prompt the choice of the information hub which the preparing should be done to play out the Map and reduce. The design of an algorithm for a node to choosing is important to progress the Map Reduce working and optimize the performance [1]. This paper deeds to offer a more widespread meaning of huge information is Big Data that captures its exclusive and character. Dealing with Big information faces "V" challenges.

1.1. High Volume

It refers to the huge information measures created by an alternative association in an incoherent way. The most current information measure in the world is the best known for obtaining huge information as volume. The estimated volume of the information is expressed in terabytes, in pet-bytes, and in zettabytes.

1.2. High Variability

How to understand the distinctive words with the same meaning or a different meaning. How to understand different types of words with the same meaning or different meaning.

1.3. High Variety

It refers to the types of information that now can use, with certainty 80% of the world's information is not structured (content, image, sound, video, etc.) with a huge information innovation, we can dissect and gather information of different types, for example, messages, discussions on online networks, photographs, sensor information, video or voice accounts, recordings, clicks, information on machines and sensors.

1.4. High Visualization

Parallel, cones and circular network diagrams. Combine the multitude of different variables that are the result of the variety, speed and the very complex interrelationships between them, that not easy to develop meaningful visualization. Traditional images when we try to draw a trillion data points, we need different ways to display data, such as data grouping, treemaps, sunbeam.

1.5. High Speed

Many data require fast processing, so the speed is based on the velocity at which they are generated, created or updated and travels at high speed around the world. Sometimes 1 minute is too late, so big data is time sensitive. The Facebook data store stores over 300 bytes of data, but the speed will take at which new data is created.

1.6. High vulnerability

Huge information brings new security concerns. Insurance additionally implies building the notoriety and trust of the brand. Thorough security works on, including the utilization of cutting-edge diagnostic abilities to oversee protection and security issues, can separate organizations from the opposition and make solace and trust with the general population.

1.7. Value

The potential estimation of huge information is colossal. Esteem is the fundamental wellspring of enormous information since it is very important for organizations that the IT foundation framework stores a substantial number of qualities in the database. Paper [1] Numerous factors can influence the execution of the framework as the change in equipment assets and programming calculations, the system throughput, the rightness of information, area (locality), dealing with, investigation, stockpiling, the protection of information and job setting up. Each one of these issues makes challenges in huge data set as Big Data. Paper [4] Apache Hadoop Open Source is most popular tools to handle the giga to peta data, which is extensively used in a large application. Value accepts ordered data (table) or formless data. Hadoop open source can handle dublicates, scalable, parallel process and distributed things. Hadoop Map-Reduce is the best component of Apache Hadoop open source; it schedules the jobs by different scheduling algorithms. The purpose of Map Reduces programming model, which can run applications in a group (cluster) that comprises of countless. It too gives a solid (reliable), scalable and distributed services. In many systems the major issue exists is scheduling in all types of computation, because of best-handling of each task execution process in the system and the greatest allocation of resources. In Apache Hadoop open-source schedule the jobs is a most active area of Big Data. This research target many scheduling challenges in big data, and the design the algorithms to solve all issues.

This paper structure is as per the following. section II discus about associated work, section III examines Apache Hadoop system, section IV show the prerequisites from the compelling booking, section V examines various types of the scheduler in Hadoop-MapReduce. At long last, section VI presents conclusions.

2. Interrelated Work

In this section follows, we will show the interrelated work, This Research paper[1] considered some Apache Hadoop open source scheduler types and present a path on how to progress Map-Reduce job execution. The Hadoop schedulers focus on only homogeneous Hadoop clusters. The Hadoop, Map reduce schedulers are compared with some important features.

Furthermore, even traditional industries, such as banking and telecommunications, are adopting the usage of Hadoop in their environments, due to their demand of processing fast-growing complex of information [4].A complete review focus on processing a large dataset, it is put based on the Map-Reduce program framework. It also a set of recognized and designed systems covered, for to offer a good software interface on the Apache Hadoop Map Reduce programming framework. The large data processing systems researched and also examine some valuable views associated with many types of applications scenarios of most popular Map Reduce programming framework. In this survey to improve the query execution performance as parallel wise using more popular Map Reduce programming framework and presents the set of Map Reduce software framework limitations, and few techniques to solve this.

The job scheduling major issues and Apache Hadoop Map Reduce classification of scheduling algorithms presented in this paper. Also focused different resource monitoring tools and frameworks on how to help for getting best results from Hadoop Map Reduce programming model. The paper also covered useful examine of Map-reduce scheduling algorithms working, classification of a multidimensional framework, good quality requirements, types of scheduling, and work is done on the dynamic platform.

3. Apache Hadoop

Hadoop [7], which is MapReduce's most popular open-source implementation, is utilized by major companies including Twitter1 and Yahoo2 mainly due to its scalability features. Apache Hadoop open source is a software framework for handling huge data set based on distributed storage and efficient processing of computer clusters on commodity hardware. Hadoop framework can handle hardware failure automatically; it used in several applications such as Twitter, Facebook, Google, Yahoo, YouTube and Amazon etc. Apache open source Hadoop has more popular core components for efficient storing and processing large datasets on a distributed file system, the architecture of

Hadoop is shown in **Figure 1**. In Apache Hadoop splits the file into a set of blocks (pieces) and each block length is 64 megabytes or 128 megabytes, these blocks are circulated across multiple nodes of a cluster. To achieve efficient process the large data in the locality, open source Hadoop apply. Map Reduce framework process multiple nodes regarding parallel and distributes data processing. To manipulate a huge number of nodes, handle large data, to allow more speed processing of the large dataset in more efficiently, scalability than a conservative structural design with fast networking. Conventional architecture with fast. It has implemented by Google like most popular distributed file system can handle the large dataset using commodity hardware clusters as more efficient and reliable. GFS is developed to produce tremendously high throughput, small latency and reduce each hardware failure on the server.

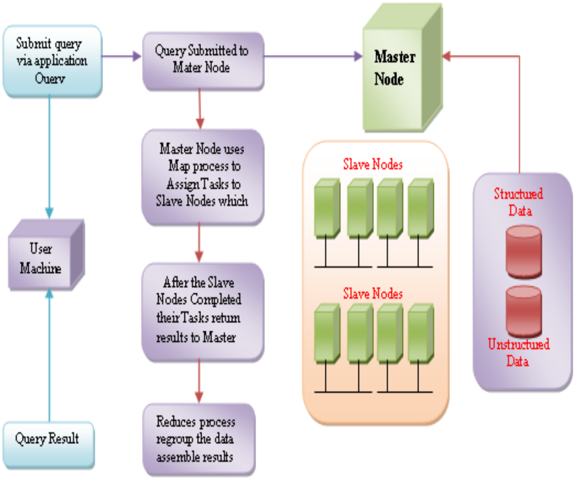
Submit query via application Query
Query Submitted to Mater Node
Master Node
User Machine
Master Node uses Map process to Assign Tasks to Slave Nodes which
Slave Nodes
Slave Nodes
Structured Data
Unstructured Data
After the Slave Nodes Completed their Tasks return results to Master
Query Result
Reduces process regroup the data assemble results

Fig.1: Sequence in Hadoop framework

3.1. Hadoop Distributed File System

Developed by Google as proprietary distributed file system (GFS) [9] Give efficient and reliable access to data using large commodity servers clusters. GFS is designed and make effective and efficient to offer an extremely high Google file system guide by, Hadoop open source Distributed File System use number of systems (machine) to keep any number of files. Through replicating factor on the number of servers we obtain good reliability [10]. Large data being stored on a number of multiple nodes, then we can gain fast computation and more reliable process. To place the large dataset through HTTP Protocol, then focus the client content accessing from web.

The Apache Hadoop open source Distributed File System[1] has two components are One and only One is Name Node uses about a block of data map to Data Nodes and operate files like file rename, manage file directories, file open and file close, also Name Node maintains namespace of files, store file operations by clients. Various large number of Data Nodes are primary process is to create, delete number of data blocks and also manage replication factor. This working process is shown in **Figure 2**.

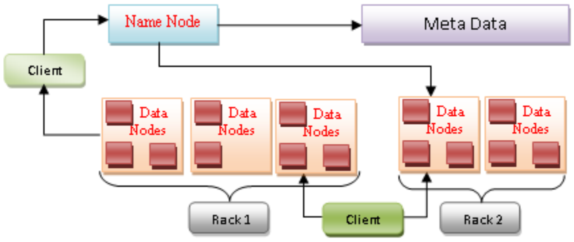

Fig .2: HDFS Architecture

3.2. Map Reduce

Google's MapReduce [5] features are more frequently used parallel and distributed programming model on clusters of commodity hardware. The Hadoop programming model as Map Reduce widely used to processing the large dataset in a cluster as parallel. It has been developed by Google Company for a large number of network search applications. The Hadoop Map-Reduce program framework is efficient can be used for data mining applications and machine learning tasks in different data centers. The main purposes of Map Reduce[1] is to give the permission to programmers for design and develop their own applications from the drawbacks of partitioning, parallel process and schedule the jobs. As shown in **Figure 3**, a map task takes input in a form of key/value pairs (k1, v1) and transforms them into another key/value pair (k2,v2).

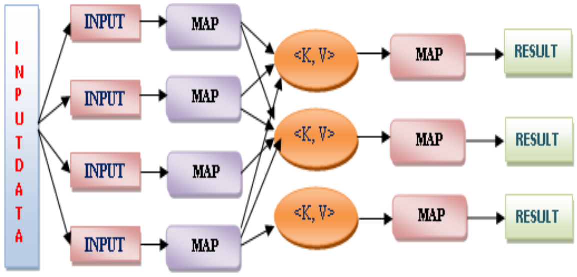

Fig .3: Map-Reduce Framework

Majorly all jobs are submitted to Hadoop Map Reduce[1], it splits into small divisions and Map Reduce design the tasks execute by Task Tracker its follow the computation in Map Reduce phase. In Map Reduce program to model the first phase is map it provides as Key k1 and values v1 pairs. The second phase is reducing it follows sort and shuffle process then give the result. Every each pair allows one key per time then process data per key k1 to yield key value (k1, v1) pairs. This programming model used different schedulers to assign jobs from queue designed by Map Reduce Job Tracker to Task Tracker. Each schedule will select the multiple nodes on what type of tasks are going to be executed. Also, reduce task carries summarized service, take care all types of communication services and to transfer large data across the system and one more thing it can reduce redundancy, handle fault tolerance situations. The scalability and system fault tolerance is a major responsibility of Hadoop Map Reduce framework; also these are obtaining to optimize the finishing time. Apache Hadoop Map-reduce distributing shuffle process can optimize the communication service cost. Through multithreaded implementation to handle the different datasets tasks as parallel. If reduce the total cost communication service on shuffle distributed process[1].
The first Step is Map: To execute the user (worker) nodes of local data by "map method" then the result (output) is written to temporary (private) storage.
The second Step is Shuffle: To reallocate the data of worker nodes regarding output keys, hence total data associated with few keys are placed on the same node.
Third Step is Reduce: To do every cluster output regarding keys in parallel by the worker node.

4. Issues of Scheduling in Mapreduce

To efficiently design set of schedule algorithms, to reach and progress some major requirements are High-quality service, fast response, data availability, efficient utilization of resources like communication, system hardware, fairness and good throughput[1]. So it is critical to design such scheduling algorithms to fulfilling the requirement because of high complexity. So Map Reduce scheduling algorithms are created by making the best required single attribute features. Also, it can be classified as adaptive means datasets, tasks, attributes, resources, and workloads and non-adaptive means fixed types [11].

4.1. Data Placement and Locality

The Hadoop Map-Reduce scheduler; the foremost decision is to arrange the computer methods close to the datasets [1].
For the reason that it can decrease the input-output of clusters in a disk, also finally it will raise the throughput. Paper [4] wished-for a k-means classical method of cluster analysis by putting the more relevant datasets in inside the data centre during the implementation and preparing stage**.** For analysis, the mutually dependent association between data sets and jobs follows the replication of multilevel tasks to reduce the size of intermediate data and send data centers and also increase the progress of job processing. Reduce the length of intermediate data to be converted between datasets and improve the performance of data streams. To overcome stagger issue by using replication factor until to complete the total tasks in Hadoop Map Reduce model [12].

4.2. Synchronization

To transfer Map function outputs (intermediate) submit to as Reducer input, also keys and values pairs are together as keys based on synchronization [1]. This synchronization service attain by Map Reduce shuffle and sort methods, also it needs circulating sorting service must involve every Map Reduce iterations. The problem becoming a major, in different datasets due to various capabilities and configuration of every node, also influence various map and reduce finishing time. In Map-Reduce framework the reduce process unable to terminate unless each map function has completed, a general optimization to reduce function to drag the data from map function if completed.

4.3. Throughput

It plays a major role in Map Reduce tasks because it is best features of Hadoop open source [1]. Throughput is number of Tasks completed in a unit of time regarding some characteristics and every Map Reduce scheduler must produce good throughput as well as high data location and usage.

4.4. Response Time

The interacting is that the total of the time needed to react to associate degree administration raise. The service here may be a task user should be operated among the reaction time is that the service time and waiting time the service time varies for a particular request very little if the work pressure will increase. The reaction time is a very important think about QOS (quality of service) for each planner in Map scale back Jobs and may be with some tolerance associated with the number of labor sent.

4.5. Availability

It is a type of reliability metrics, it can be defined as the average time between failures and repetitions of user requests, or it is the time when the system supports and continues to run to fulfill future requests. All work must be completed completely and the planner must check it. This is a type of reliability measure. Repeat the client request between failures, this is also the length of the system that keeps the query and continues to work to meet the user's future demands. The total tasks must be completed after this verification via the programmer.

4.6. Resource Utilization

In many scientific areas, the resource major feature is utilization. Majorly we maintain a maximum number of resources and efficient processing tools to handle massive data but in these is a lack in our process. The Map-Reduce planner must consume the resources efficiently and, in particular, the programmer must use his sources effectively, especially the closest system hardware closes the use of the communication. To optimize and manage resources to focus a number of tasks in clusters then achieving good results in our systems.

4.7. Fairness

It is a more significant feature which is through different types of Hadoop Schedulers. Fairness is to share the set of resources and task execution as poor fairly, uniformly, based on priority levels, size, and execution time between Map Reduce Jobs.

5. Map Reduce Schedulers

In this paper, we will show the work done by Map-Reduce planning with respect to the equity, priority, resources, time, utilize and deadline of the system [1]. If the scheduling process is static means offline since the allotments completed earlier than to start the execution or Dynamic means online since the allotments completed at the execution time regarding the lively atmosphere and desirable set of resources. Many times the scheduler is real-time computation system based on Storm distributed whenever manage the tasks regarding resource utilization.

5.1. FIFO Scheduler

FIFO scheduler have numerous confinements, for example, poor reaction times for short occupations contrasted with substantial employments low execution when run different sorts of employments and it focuses FIFO scheduling algorithm problems that’s why Fair and Capacity was introducing[13].

5.2. Fair Scheduler

It is developed by Facebook Company [4] to obtain good fairness among all jobs, allocates every desire resources to each job through equal fairness sharing. This scheduler design every job at minimum processing time to complete very quickly based on the equivalent distribution of resources. All users have pools along with at least desire sharing of resources and select the unused slot by assign the running tasks. This scheduler allocates the resources to every application on totally equivalent sharing [15]. This scheduler working on memory only by default and organize the work based on memory and CPU. It used the total cluster if the only single application is working. Idle or free resource assigned to the tasks of applications, the cluster should be arranged between fixed numbers of users. This schedule follows priority to assign weights for finding resources and allocate to the apps. It has a default queue is called as "default" [1] it is shared among total users and create each username on each queue along with minimum sharing ability, the minimum quota of slots, remaining slots means idle slots will be assigned to next pool. With Fair scheduler, each application can be run by default, but the total running apps are limited regarding configuration file of each user queue. It advantage whenever user presents more apps at a one time, also we can increase the scheduler performance by minimizing the intermediate data producing or switching. There are no chance applications failing because waiting for the scheduler queue up to complete the past user's request.

5.3. Capacity Scheduler

It is presented by Yahoo, which is developed [16] to handle complex clusters, also it can be applied to a number of applications when a number of clients are there, is designed especially to make efficient resource allotments of each computation between a number of clients and schedule the task allotment on domain resource. Also, this scheduler handles every job regarding user demands and memory utilization. Also, many queues are designed in the place of pools, to assign the tasks according to queues of each client and to make Map Reduce queue slot size. The remaining service of each queue is contributed to who queue needs. Once the task has submitted then get a priority order on each arrives task regarding time. The high priority level tasks will obtain the resources earlier than low priority tasks. Every cluster capacity is allocated among clients, not between tasks as like fair scheduler. We can manage the waiting time in capacity scheduler later than preemption applies to remaining queues job whenever fair sharing is low [1].

5.4. Delay Scheduler

It is designed by Face book [1], the delay occurs from an amount of time waiting to get the data on the local node skip if in case not available then subsequent jobs. The purpose is to overcome the efficient share locality drawback. If your task is scheduled and will not launch hey lack of job, lack of amount of time and reminding costs will launch its job instead [13], The relational intent is to multiplex hafiz clusters based on statistics and has minimal influence on equity very high data locality gain. The reallocation of resources is done by eliminating the remaining tasks of the previous work to organize a place for their work and waiting for a task to be completed in the assigned spaces for new work, but the disadvantage is the operational performance by eliminating the amount Tasks they do not arrive Useful because of waiting for jobs. From small jobs, many locality problems will occur because the file has low input data. So it contains a little number of blocks read, small drawback will happen whenever is the job follows ascending order list the head should be controlled one of the tasks is run on the remaining empty slot then whatever the node working is going on. The important problem with slot machines is that there is a possibility that a job is assigned to the same space that I will reuse. Finally, designers follow the schedule of delays because equity is free. Meanwhile, to increase the location of data in many jobs, they remain for the programming option with an efficient local data node. The Tosca assignment has a limitation on free job selection.

5.5. LATE Scheduler

This scheduler (the longest conjectured to finish) originally came from the optimizable runtime component of the speculative society, where singular jobs can be established on the option to allow or not, a breach is a license. It shows up [17] by making an effort for all identified tasks with some completed work when possible, since that the job will complete totally if the greater part has done. At the point when the undertaking runs gradually because of a few factors like lack of a number of resources, the programmer also finds this task, try sending another copy of the same task as a backup that can finish quickly and increase execution. The LATE programmer looks for a report of a problem because a performance problem, not from the point of view of the reliability of the view, does not look for the reasons why this report does not try to analyze and solve Moderate execution assignments and incorrect software configuration. On paper [17], the perception of Hadoop's homogeneity assumption leads to error and, occasionally, a more speculative implementation in the heterogeneous context may diminish the un-acquired performance with the injured speculation. Additionally, can guide serious execution corruption at the point when its understand assumption is broken down. It is strong for heterogeneity in data centers of various conditions. LATE is configured for three standards: the determination of fast nodes to execute, the speculation of priority level tasks and the overwriting of prevention actions leading to tentative tasks.

5.6. Deadline Constraint Scheduler

The schedule [18] planner works within the user-specified time limits, tries to meet work schedules, and increases system utilization. The planner will calculate the minimum number of Hadoop cards and reduce the spaces needed to complete the job (cost). If the planner ability test fails, the user will be prompted to enter another deadline value and, if approved, the jobs will be scheduled.

5.7. Resource-Aware Scheduler

Nowadays, many different heterogeneous nodes in the system that gives diversity in the allocation of nodes [1]. Other types of programmers did not consider the availability of resources in detail and assigned a fixed size of resources. The resource [19] conscious planner considers the availability of resources for planning tasks. It tries to minimize the consumption of these resources and increases the use. Configure each node to indicate the disk I / O throughput and the actual processing power available on each cluster machine and online modification of available slot machine capacity. Each Task Tracker node keeps track of the hardware resources. Virtual Machine Management Status Monitors Page error virtual machine-induced disk hit. There are two possible mechanisms in the conscious resources Job Tracker Programming:
1) The dynamic announcement of available free space calculation slots configured in each Task Tracker node. The improved response time of the job where no task will reach the bottleneck of resource consumption.
2) Free priority / filtering slots, where cluster administrators configure a maximum number of compute slots per node at the time of configuration.

5.8. Node Selection scheduling Algorithm in HPCA

Table 1: Hadoop Schedulers Features

Name of the Scheduler	Speculative Execution	Head of line problem	Sticky Slots	Locality Problem
Default FIFO Scheduler	NA	NA	NA	YES
Fair Scheduler	NA	YES	YES	YES for small jobs
Capacity Scheduler	NA	NA	NA	YES
LATE Scheduler	YES	NA	NA	YES
Delay Scheduler	NA	NA	NA	Improved compared fair scheduler
Dynamic Priority Scheduler	NA	NA	NA	NO
Resource-Aware Scheduler	NA	NA	NA	YES
Deadline Constraint Scheduler	NA	NA	NA	YES for small jobs
Learning Scheduler	Enabled	NA	NA	YES
COSHH	NA	YES	YES	NO
N/WAware Scheduler	NA	NA	NA	NO

The health nodes, priority, capacity, and availability (HPCA) based selection algorithm [20] provides the creation of a node queue available to accept new tasks with a node selection task optimized to provide performance node execution. Redundancy to overcome the failure of the execution task. Consider how to select the resource node based on the success rate at work, the priority of the job and the predetermined priority of the resource chosen to complete the job, the capacity of the selected resource, and the availability of resources.

5.9. Round Robin Scheduler

The round robin with a varied feedback algorithm [21] tries to solve the problem of the FIFO scheduler, where the work starts in the order of presentation and this reason the work present later to start later. Even if the schedule of the preliminary round, the last submitted work, you will receive a quick response and it will start with a small delay. This planning can minimize the normal response time. As an insider, in [22] task scheduling algorithm based on Hadoop rounded tap-weighted enhancements lays the weight update rules by analyzing tasks to allocate weight to each queue and schedule assignments. Different sub-queues according to the weight.

5.10. Other Scheduler

Table 2: Hadoop Schedulers Properties

Name of the Scheduler	Job Allocation	**Imple melted**	**Remarks**
Default FIFO Scheduler	Statically	**YES**	**Homogenous system, Static allocation, no**
Fair Scheduler	Statically	**YES**	**Homogenous Static**
Capacity Scheduler	Statically	**YES**	**Homogenous Static, Primitive**
LATE Scheduler	Statically	**YES**	**Homogenous Heterogeneous, static**
Delay Scheduler	Statically	**YES**	**Homogenous Static**
Dynamic Priority Scheduler	Dynamic	**YES**	**Homogenous Heterogeneous, Dynamic**
Resource-Aware Scheduler	Both	YES	**Homogenous Heterogeneous, Dynamic**
Deadline Constraint Scheduler	Dynamic	**Not in real time**	**Homogenous Static**
Learning Scheduler	Both	**YES**	**Homogenous Heterogeneous, Dynamic**
COSHH	Dynamic	**Not in real time**	**Homogenous Static**
N/WAware Scheduler	Dynamic	**YES**	**Homogenous Static**

There are many other improvements in the programming algorithms of the Hadoop works; we are going to demonstrate some of them: paper [23] proposed a programming algorithm of Bayesian classifier for the programmer's learning when executing the operative state and performing the self-classification. In [24] two efficient planners, it is proposed: the first proposes to act as an interactive service of high priority at the level of virtualization, and the second as part of Hadoop to help batch processing works to reach the deadline. The method finds the most excellent servers that will provide the excellent performance of Hadoop tasks and schedule the jobs that arrive efficiently by scheduling with the deadline. The combination of these types of programmers provides a kind of expectation and security in the use of intensive resource tasks in Hadoop with intensive latency applications on the web [25].

When considering the organization of the slot configuration, it is initially fixed and static. This proposes a better Hadoop system called Fair-efficient Slot.

Table 3: Hadoop Schedulers Properties

Name of the Scheduler	Preem option	Priority in job queue	Fairness/ Fair sharing of Resources	Better Working **With**
Default FIFO Scheduler	NO	No by Default	NO	small clusters
Fair Scheduler	YES	YES	A fair share of the cluster capacity over time	small clusters
Capacity Scheduler	No	No by Default	YES	large clusters
LATE Scheduler	YES	YES	YES	large clusters
Delay Scheduler	YES	YES	Less than Fair	small clusters
Dynamic Priority Scheduler	YES	YES	YES	large clusters
Resource-Aware Scheduler	YES	YES	YES	large clusters
Deadline Constraint Scheduler	YES	YES	YES	large clusters
Learning Scheduler	YES	YES	YES	small clusters
COSHH	YES	YES	YES	large clusters
N/W Aware Scheduler	YES	YES	YES	large clusters

In the proposed research work [25], a workflow that requires a large amount of Data Optimization Algorithm (DOA) used for heterogeneous, which calculates that the systems are processed according to the data flows. The algorithm provides reduced latency with an increase in the performance of the application's task graph that is divided so that the inter-partition minimizes the movement of the data. Each partition is assigned to the execution node that performs a minimum run time for that partition. DOA also promotes duplication of partial tasks to reduce latency. The document [26] is different from the equality between user restrictions, focusing on the goal of balancing the workload and satisfying the delay of Map Reduce users' requests. Proposed programming algorithms with a fixed approach associated to balance the server workload and also meet the response time requirements related to card downsizing work. The [27] implemented algorithms are designed with a resource manager as an open implementation of the Hadoop source. Configuration Planner for Apache Hadoop Clusters to improve configuration of the dynamic location configuration and appropriately assign tasks to available spaces.

Achieve the correct and effective configurations of the slots and the programming of Hadoop clusters. The proposal attempts to perform two main tasks: deciding on the slot configuration when deciding how many cards / aisle reduction are appropriate and assigning assignment / reduction tasks to the available spaces.

This document verifies application level indications with different priorities of flow processing tasks that allow programming of application-specific [28] Distributed Flow Processing System (DSPS) resources. Suggest a general method for improving priority-based resource planning in DSPS diagrams, providing application developers with the ability to enhance DSPS graphics with priority metadata and present an extensible set of prioritized schemes to manage using DSPS extended automatically Place the Met Scheduler before the FIFO Scheduler to improve its performance. I'm also [29] trying to create a real-time type planner to reduce the map, to reduce the limitation of other schedules. Avoid accepting the tasks that will guide the deadline and try to improve the uses of the cluster. When a new card reduction job arrives, the admission controller determines whether it is possible to schedule the new job without compromising the security of the previously accepted job. A return controller is developed to keep the admission controller updated. The main goal of this research paper is to study and analysis on different types of hadoop and map reduce algorithms and techniques which are useful to increase the Hadoop working performance in comming large application .these hadoop and mapreduce features, properties and advantages and disadvantages and challenges and issues as shown in **Table 1,2, and 3**.

6. Conclusion

In today market the big data, hadoop, Mapreduce Algorithms are in big demand. In every industry has huge volume of data being is stored but there is not tool to handle such complex data, hadoop was introduced by Doug cutting to hadle complex data at low cost and good speed, performance. We can see that most of these mapreduce schedulers are discussed in this paper also addresses major problems**.** In this paper we studied, analyzed and described the overview of various hadoop and mapreduce algorithms for making efficient scheduler for speed up our system and make the scheduler effective for the fast processing.

References

[1] Ehab Mohamed Zheng Hong, “Hadoop-MapReduce Job Scheduling Algorithms Survey”, 7th International confrence on Cloud Computing and Big Data, (2016), 237– 242.

[2] Abhishek Verma, Ludmila Cherkasova, Roy H. Campbell, "ARIA: Automatic Resource Inference And Allocation for MapReduce environments", *8th Autonomic computing ACM*, IEEE, (2011), 235 – 244.

[3] S. Bardhan, D. A. Menasce, "Queuing Network Models to Predict the Completion Time of the Map Phase of Map reduce Jobs", *ICMG*, IEEE, (2012).

[4] J.V.Gautam, Harshadkumar, Vipul K Dabhi, Sanjay Chaud hary B,"A survey on job scheduling Algrithms in Big data processing”, *(ICECCT), IEEE*, (2015), 1 – 11.

[5] Nikos Zacheilas, Vana Kalogeraki, “Pareto-Based Scheduling of Map Reduce Workloads” *19th International Symposium on Real-Time Distributed Computing (ISORC), IEEE,* (2016), 174 – 181.

[6] Mark Yong, Nitin Garegrat, Shiwali Mohan, “Towards a Resource Aware Scheduler in Hadoop”, *in Proc. ICWS*, (2009), 102– 109.

[7] A. Thusoo, J. S. Sarma, N. Jain, Z. Shao, P. Chakka, S. A nthon, H. Liu, P. Wyckoff, R. Murthy, “Hive - A W rehousing Solution over a Map-Reduce Framework”, *PV LDB*, (2009), 1626 – 1629.

[8] J. K. Laurila, D. Gatica-Perez, I. Aad, O. Bornet, T.-M.T.D O. Dousse, J. Eberle, and M. Miettinen, “The Mobile Data Challenge: Big Data for Mobile Computing Research”, *Nokia Mobile Data Challenge Workshop, Newcastle, U K,* (2012), 321 – 330.

[9] Sanjay G, Howard G, S.T.Leung, “The Google file system” , *19th Symposium Op. Sys. Principle, New York,* (2003), 29 – 43.

[10] Casavant et al, “Taxonomy of scheduling in general purpose Distributed computing systems", *IEEE Transactions*, (1988), 141 – 154.

[11] N. Tiwari, "Classification Framework of Map Reduce Scheduling Algorithms", *ACM Computing Surveys*, 47, 3, (2015), 49.

[12] Quan Chen, Daqiang Zhang, Minyi Guo, Qianni Deng, Song Guo, "SAMR: A Self-adaptive MapReduce Scheduling Algorithm in Heterogeneous Environment", *IEEE 10th International Conference*, (2010), 2736 – 2743.

[13] M.Zaharia, "Delay scheduling: a simple technique for achieving locality and fairness in cluster scheduling”, 5th European conference on computers, New York, (2010), 265-278.

[14] PEI Shu-jun, Zheng Xi-min, Hu Da-ming, Lou Shu-hui, Zhang Yuan-xu, “Optimization and Research of Hadoop Platform Based on FIFO Scheduler”, *Seventh Internation al Conference on Measuring Technology and Mechatroni cs Automation, IEEE*, (2015), 727 – 730

[15] Kc K, Anyanwu K, ”Scheduling Hadoop Jobs to Meet Deadlines Cloud Computing Technology and Science (C1oudCom)” *2nd International Conference, IEEE*, (2010), 388-392.

[16] J. Chen et al, "A Task Scheduling Algorithm for Hadoop Platform", *journal of Computers*, 8, 4, (2013), 29 – 936.

[17] Matei Zaharia, Andy Konwinski, Anthony D. Joseph, Randy Katz, Ion Stoica, "Improving Map Reduce Performance in Heterogeneous environments", *8th USENIX Symposium*, (2008), 26 – 33.

[18] Geetha J., N. Uday Bhaskar, P. Chenna Reddy, Neha Sniha,” Hadoop Scheduler with Deadline Costraint ", (*IJCCSA*), 4, 5, (2014), 1– 7.

[19] Mark Yong, Nitin Garegrat, Shiwali Mohan, “Towards a Resource Aware Scheduler in Hadoop”, *Proc. ICWS*, (2009), 102–109.

[20] Archana G.K, V. Deeban, "HPCA: A Node Selection and Scheduling Method for Hadoop MapReduce", *(ICCCT’15) , IEEE*, (2015), 368 – 372.

[21] Y. Wang, Ruonan Rao, Yinglin Wang, “A Round Robin with Multiple Feedback Job Scheduler in Hadoop", *Progress in Informatics and Computing (PIC) International Conference, IEEE*, (2014), 471 – 475.

[22] J. Chen, Dan Wang, Wenbing Zhao," A Task Scheduling Algorithm for Hadoop Platform ", *Journal of Computers*, 8, 4, (2013), 929– 936.

[23] Yingjie Guoa, Linzhi Wub, Wei Yuc, Bin Wud, Xiaotian Wang, “Research and Improvement of Job Scheduling Algorithms in Hadoop Platform”, *IEEE*, (2010), 15– 21.

[24] Wei Zhang, Sundaresan R., “Timothy Wood, Min gfa Zhu " MIMP: Deadline and Interference Aware Sched-uling of Hadoop Virtual Machines",*14th IEEE/ACM I international Symposium on Cluster, Cloud and Grid Com putting, IEEE*, (2014), 394 – 403.

[25] Saima Gulzar Ahmad, Chee Sun Liew, M. Mustafa Rafique, Ehsan Ullah Munir, Samee U. Khan " Data-Intensive Work flow Optimiztion based on Application Task Graph Partitioning in Heterogeneous Computing Systems ", *Fourth International Conference on Big Data and Cloud Computing, IEEE*, (2014), 129 – 136.

[26] Zhao-Rong Lai1, Che-Wei Chang, Xue Liu, Tei-Wei Kuo, Pi-Cheng Hsiu, “Deadline-Aware Load Balancing for MapReduce ", *20th (RTCSA), IEEE*, (2014), 1–10.

[27] J. Wang, Jiayin Wang, Yi Yao, Ying Mao, Bo Sheng, Ning fang Mi, "FRESH: Fair and Efficient Slot Configuration and Scheduling for Hadoop Clusters", *International Conference on Cloud Computing, IEEE*, (2014), 761– 768.

[28] Paolo Bellavista, Antonio Corradi, Andrea Reale, and Nicola Ticca, " Priority-based Resource Scheduling in Distributed Stream Processing Systems for Big Data Applications”, *ACM 7th International Conference on Utility and Cloud Co mputing, IEEE*, (2014), 363 – 370.

[29] C. He, Y. Lu and D. Swanson," Real-Time Scheduling in MapReduce Clusters ", doi: 10.1109/HPCC.and.EUC. 216, 2013.