Global Journals LaTeX JournalKaleidoscope™
Artificial Intelligence formulated this projection for compatibility purposes from the original article published at Global Journals. However, this technology is currently in beta. *Therefore, kindly ignore odd layouts, missed formulae, text, tables, or figures.*

Study of Effective Scheduling Algorithm for Application of Big Data

Tanmay Paul[1]

[1] Adamas Institute of Technology

Received: 10 December 2016 Accepted: 4 January 2017 Published: 15 January 2017

Abstract

In this new era with the advancement in the technological world the data storage, analysis becomes a major problem. Although the availability of different data storage component like electronic storage such as hard drive or virtual storage such as cloud still the problems remains. The major issue is processing the data because usually the data is in several format and size. Usually processing such huge amount of data with several formats can be time consuming. Using of application such as Hadoop can be beneficial but using of scheduling algorithm can be the best way to for data set analysis to make the process time efficient and analysis the requirement of different scheduling algorithm for the specific data set. In this paper we analysis different data set to explain the most effective scheduling algorithm for that specific data set and then store and execute data set after processing.

Index terms— big data, hadoop, scheduling algorithm, data analysis, HDFS, FCFS.

1 Introduction

n the data analysis the efficiency plays the most important factor and the development in the data storage, analysis efficiency in the stipulated time and the endeavor for the output of data analysis in the executional environment and storage of that data is defined as Hadoop distributed file system (HDFS) **??**1]. The application comprises of certain sub system application which reshape data in terms of times which are analyzed using scheduling algorithm MinMin [2], minimum completion time (MCT) [3]. In HDFS huge amount of data can be stored which provides cost effective and also reliability. In first come first serve (FCPS) [4] the big data changes dynamically for the application access which consists of different speed and size. In order to execute in an executional environment HDFS is implemented. HDFS allow large storage and data analysis but the problem is to process a large amount of data. In the computing environment HDFS gives efficient data analyzing, storage, execution. Scheduling algorithm administer data work flow within time constraints. Scheduling algorithm FCFS, Distributed heterogeneous earliest finish time(DHEFT) works unsurpassable for given set of data in cloud environment. Performance and data analysis is done by scheduling algorithm. For checking the performance of the specific data set the algorithm must be known to priori for ease in implementation and time effective manner. The task scheduling is executed single task at a time so that performance of the entire scheduling algorithm executed can be manifest. VM can execute single task at single time.

2 II.

3 System Architecture a) System Workflow

The data set is given as input for execution. Each data set is converted in smaller subtask and the entire subtasks are dependent on each other. The subtask is executed in sequential manner as every subtask execution is completed only after the execution of the previous subtask. The new subtask waits until the execution of the previous sub task gets completed. Below in fig 1 data execution work flow is given.

4 Fig.1: Data execution work flow b) Cloud Server Model

Cloud server [4] is the primary module which is the resource provider for performing the processing activity. Virtual machine (VM) constructed can be accessed only by the registered user. Once the file is received in the VM it is divided into the subtasks. VM executes single task and the remaining task is shared between VM through round [5] robin algorithm.

5 c) Scheduling Algorithm

Scheduling algorithm is implemented to VM to utilize the resources effectively so that no VM is ideal mode of operation. Initially during task assigning we have to assign proper VM for the specific task and also the resource mapping for execution. There are various scheduling algorithm which can be implemented for large data set to be examined their performance. The scheduling algorithms are MinMin algorithm, Data aware scheduling algorithm, MaxMin scheduling algorithm, first come first serve (FCFS) scheduling algorithm, MCT algorithm, and heterogeneous earliest finish time (HEFT) algorithm.

6 ? MinMin Algorithm

In MinMin algorithm [2] task is arranged in ascending order with least or minimum time of completion and the resources or VM are allocated to the fastest job and this process is looped until all the jobs are scheduled to the VM. ? Data Aware Scheduling Algorithm

In data aware scheduling algorithm [6] the data is stored in the VM which is vacant to the resources which are closest to be executed by the VM. It eliminate over utilization of time in scheduling the task one by one.

7 ? MaxMin Scheduling Algorithm

In the MaxMin [2] algorithm task is arranged in descending order with maximum time of completion for task allocation. It is in actually the opposite of the MinMin algorithm.

8 ? First Come First Serve (FCFS) Algorithm

In the first come first serve [7] algorithm the task scheduled in a queue and allocated according to first come first serve basis not according to the VM efficiency or maximum or minimum time completion. The only disadvantage of these is if the task which is longer executed in the VM then the smaller task has to wait for the longer task to be executed.

9 ? MCT Algorithm

In MCT algorithm [3] the task assigned to the resources or the available VM get executed with minimum time.

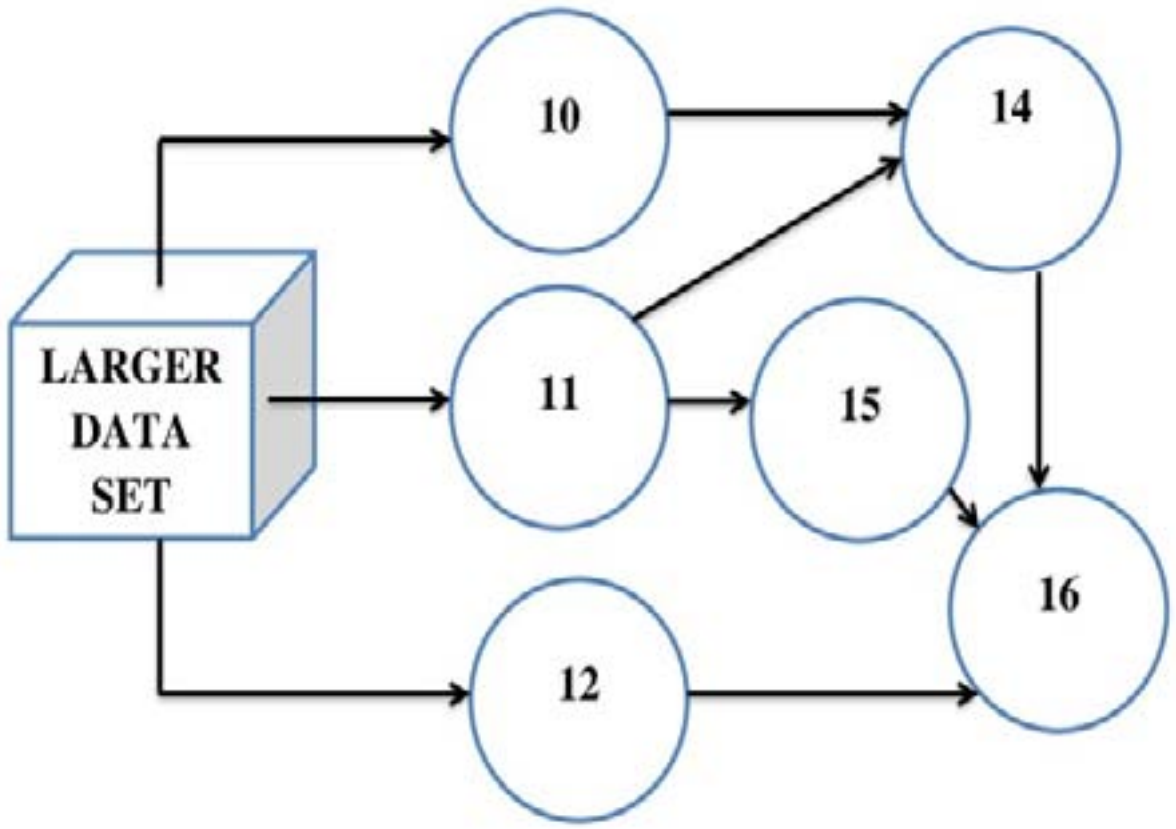
10 III. Experimental Analysis

In this paper the performance of various scheduling algorithm is analyzed on big data. Cloudera [8] has been used as a platform for analyzing in eclipse [9] environment. Three VM has been created for user registration. The task available is allocated to VM by the server and all tasks which are in queue are allocated simultaneously to all available VM at same for the execution purpose. Dynamic data set has been used to performance evaluation analysis which composed of data of different size and set using various scheduling algorithm. In the evaluation of data set two parameters has been considered firstly the delay and secondly the task span. Delay may occur due to two major causes firstly system failure secondly due to low system memory in comparison to the task allotted because every time there is input in the data set there is change in size due to big data which can be sometimes non compatible. We have considered three cases 12Kb, 22Kb and 55Kb of data set. In the first case 12Kb data set we implemented all the scheduling algorithm where x-axis defines the scheduling algorithm and y-axis defines the time. The entire scheduling algorithm differs with each other. Make span comprises of addition of data processing time, time taken for data transfer from storage to execution, waiting time and time of computation.

11 IV. Conclusion

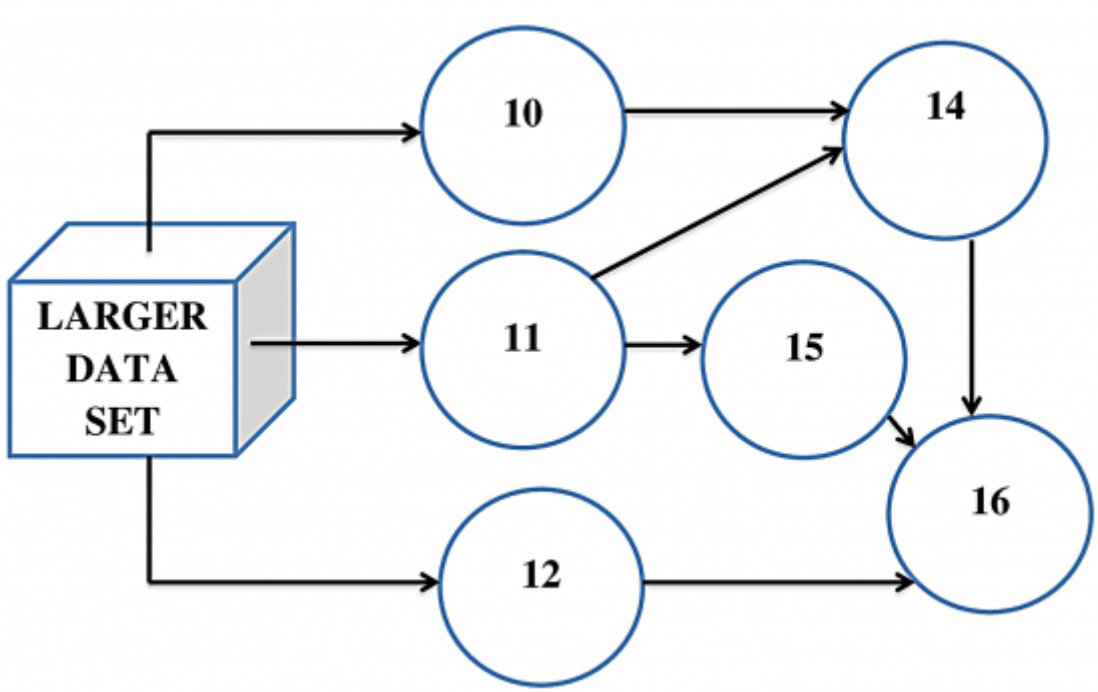
The scheduling algorithm on data set of big data comprising FCFS algorithm, MCT algorithm, MinMin algorithm, DAS algorithm, HEFT algorithm is performed for analyzing. The result of performance analysis varies differently with dynamic dataset. After the data analysis the data is stored in the form of HDFS in encrypted. For the future work various data set of different data size can be used for performance analyzing and assessment. [1]

[1]( ) C © 2017 Global Journals Inc. (US) Study of Effective Scheduling Algorithm for Application of Big Data



2

Figure 1: Fig. 2 :



3

Figure 2: Fig. 3 :

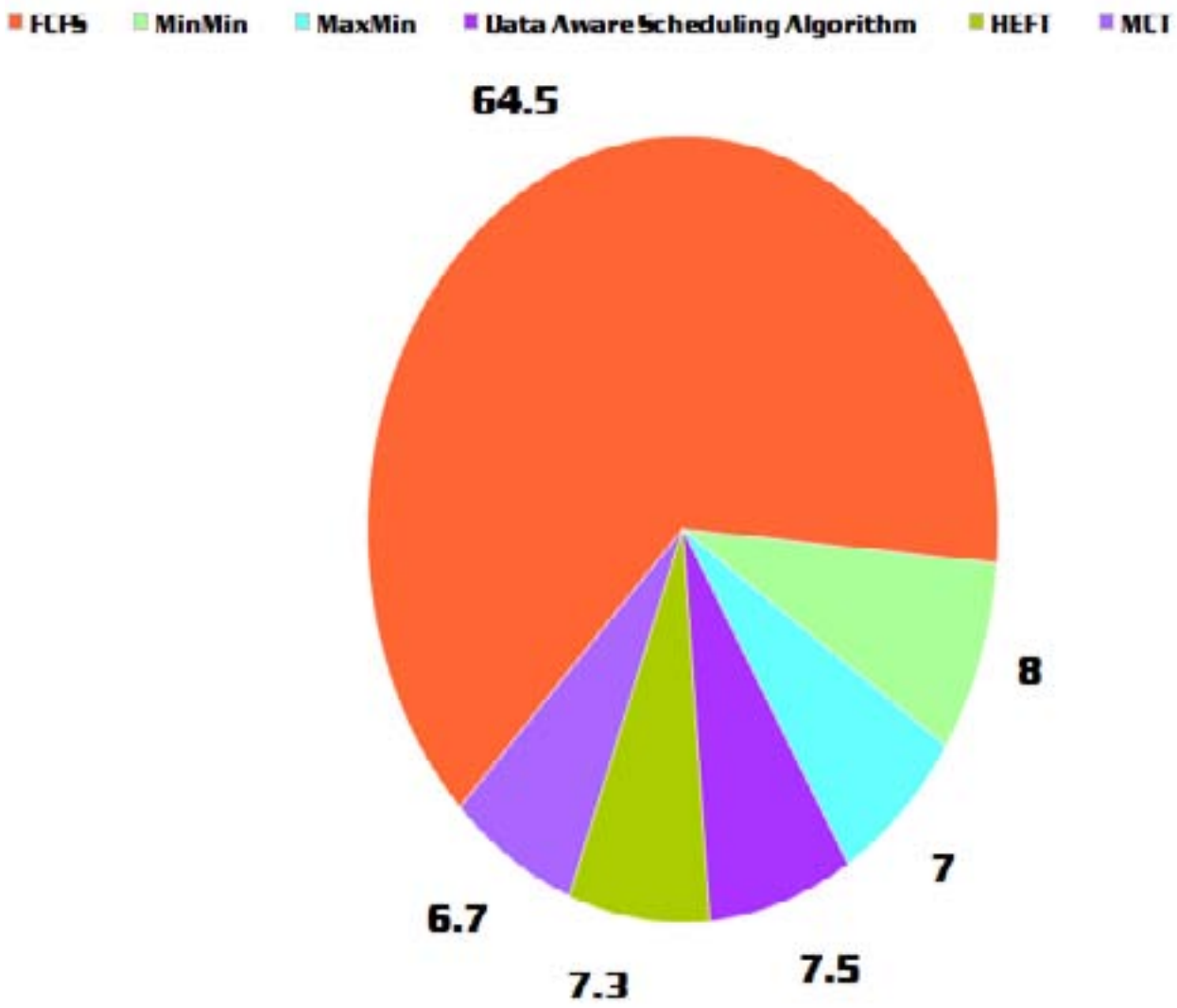


Figure 3: Fig4:

[Tabak] , E Tabak .

[Guia and Espírito-Santo] , S S Guia , ; A Espírito-Santo .

[Jiang] , Jianhua Jiang .

[Xu] , Gaochao Xu .

[Taylor-Fuller] , David Taylor-Fuller .

[Paciello; F. Abate and Pietrosanto ()] *A comparison between FFT and MCT for period measurement with an ARM microcontroller*, V Paciello; F. Abate , ; A Pietrosanto . 2015.

[Lincke] 'A QoS comparison of 4G first-come-first-serve load sharing algorithms involving speech & packet data'. Susan J Lincke . *2007 IEEE International Conference on Electro/Information Technology*,

[Bhowmik et al. (2016)] 'An Efficient Load Balancing Approach in a Cloud Computing Platform'. Saptarshi Bhowmik , Sudipa Biswas , Karan Vishwakarma , Subhankar Chattoraj . *IOSR Journal of Computer Engineering (IOSR-JCE)* Ver. VI (ed.) Nov.-Dec. 2016. 18 (6) .

[Wei] 'An Enhanced Data-aware Scheduling Algorithm for Batch-mode Data intensive Jobs on Data Grid'. Xiaohui Wei . *2006 International Conference on Hybrid Information Technology*,

[Chang] 'Design and implementation of HDFS data encryption scheme using ARIA algorithm on Hadoop'. Jae-Woo Chang . *2017 IEEE International Conference on Big Data and Smart Computing (BigComp)*,

[Yadav et al.] 'Efficient & Accurate Scheduling Algorithm for Cloudera Hadoop'. Swati Yadav , ' Santoshvishwakarma , Ashok Verma . *2015 International Conference on Computational Intelligence and Communication Networks (CICN)*,

[IEEE International Instrumentation and Measurement Technology Conference (I2MTC) Proceedings] *IEEE International Instrumentation and Measurement Technology Conference (I2MTC) Proceedings*,

[Barla Cambazoglu; Cevdet and Aykanat ()] 'Improving the Performance of Independent Task Assignment Heuristics MinMin, MaxMin and Sufferage'. B Barla Cambazoglu; Cevdet , Aykanat . *IEEE Transactions on Parallel and Distributed Systems Year* 2014. p. 5.

[Laverman et al. ()] 'Integrating Vehicular Data into Smart Home IoT Systems Using Eclipse Vorto'. Jeroen Laverman , Dennis Grewe , Olaf Weinmann , Marco Wagner , Sebastian Schildt . *IEEE 84th Vehicular Technology Conference*, 2016.

[Tang et al.] 'On First Fit Bin Packing for Online Cloud Server Allocation'. Xueyan Tang , Yusen Li , Runtian Ren , Wentong Cai . *2016 IEEE International Parallel and Distributed Processing Symposium (IPDPS)*,

[References Références Referencias 1. Youngho Song; Young-Sung Shin] *References Références Referencias 1. Youngho Song; Young-Sung Shin*, (Miyoung Jang)

[Monte and Pattipati ()] 'Scheduling parallelizable tasks to minimize make-span and weighted response time'. J D Monte , K R Pattipati . *IEEE Transactions on Systems, Man, and Cybernetics -Part A: Systems and Humans Year*, 2002. p. 3.