Scheduling to maximize the data transfer rate for big data Applications in Cloud System

D.Sugumaran, C. R. Bharathi

Abstract*: In cloud platform, parallel computing is precisely one of the methods to handle various computational tasks which need to perform fast on a large dataset. In a system each job was run by the respective processors. Jobs may need to be accompanying through nodes and it will share resources. So scheduling is important to share the resources and path diversity is very much of important in order to get the data within least retrieval time. The existing scheduling algorithms should not efficiently find the optimum solution. In this paper we make a survey to provide the better transfer scheduling algorithm for transfer the data within stipulated time, to maximize the data transfer rate and to choose cost effective paths.

***Keyword:* Data center networks, parallel computing, maximizing data transfer, Cloud System**

I. INTRODUCTION

The cloud is a web based data storage model where data is stored on several virtual servers. Big data is the term which gathers huge and difficult data sets and it's too hard to process by on-hand database management tools or conventional applications. In computing the processing of big data is a challenge mainly in cloud computing. Your data was stored in the remote server so that retrieving the data from the remote area is very difficult in the parallel processing. Parallel processing which means many cloud applications were request the data concurrently for simultaneous process. It's not easy to provide the data in stipulated time for the applications concurrently. Because uncountable number of users are utilizing the networks nowadays. All the applications are generating terabytes or petabytes of data per day. It's to

difficult to manage the data and also to transfer the data to the respective applications within time

Many big-data application was deployed in the cloud system all should need the data for parallel processing. So that data has to be transferred concurrently. Various open source frameworks deployed in cloud platform are facing lot of difficulties to provide scalable data for the big data applications. Parallel processing is nothing but splitting a single into multiple task and process it in available machines simultaneously.

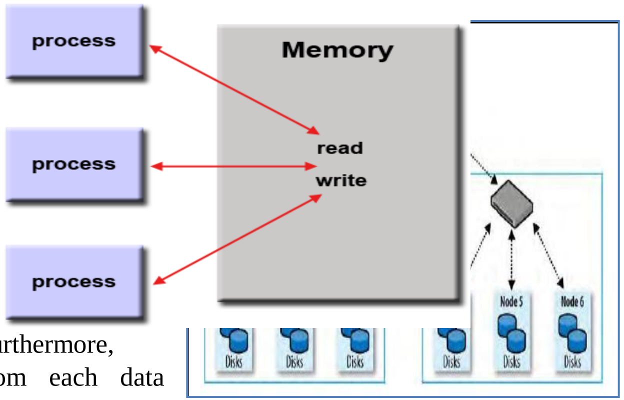
In the conventional system a program has to wait till the completion of the previous program. it increase the delay time for the process completion. So if time delay increased that reflects in overall delay of many task. Automatically path congestion will be occurred. To avoid this congestion multiprocessing technique was introduced in which the job was shared by more than two processors. This will reduce heavy workload in the data centre networks. It is called as symmetric multiprocessing system (SMP).

Here in SMP the handling of workflow that treats each and every processor as equally capable and responsible. It behaves like directly proportional model such as if there is increase in processors than there is increase in the time taken for propagating data in DCN.here it will transfer the new data to all over the system instead of the respective system. To overcome this, shared memory concept was introduced that shares the data, in which the respective system is utilize that and none by others. This is called as massively parallel processing (MPP) systems.

Whatever systems are exists nowadays still we are facing lot of problems in transferring concurrent information across the node. It's only possible if we have efficient path diversity and data replicas.

Data Center Networks

Data are generally replicated for redundancy and robustness.



Furthermore, from each data node in Data Center Networks (DCN), data transfer will be done by multiple paths; it is difficult to find the shortest paths, due to path redundancy in DCN. Although we are having many paths it is very much important to select the best node and the best path to retrieve a non-local data. Here we are facing the data retrieval problem. In the number of available paths, we have to select the perfect node and the path for data transfer in least retrieval time.

In present days many Big-data applications are retrieving data concurrently. This may cause path collision because of parallel transferring data to the big data applications. It ignores the bandwidth and seven the nodes and paths are

Revised Manuscript Received on July 22, 2019.

D.SUGUMARAN, Department of Information Technology, Vel Tech Rangarajan Dr.Sagunthala R&d Institute Of Science And Technology, Chennai – 600062, Tamil Nadu, India

Dr. C. R. BHARATHI, Department of ECE, Vel Tech Rangarajan Dr.Sagunthala R&D Institute of Science And Technology, Avadi, Chennai – 600062, Tamil Nadu, India

Retrieval Number: B10530682S519/2019©BEIESP
DOI: 10.35940/ijrte.B1053.0782S519

Published By:
Blue Eyes Intelligence Engineering &
Sciences Publication

overlapped at the time of concurrent data transfer. Insight into cloud system is very much important to know the structure of nodes and paths. In cloud system the HADOOP is one of the most important open framework applications which is used to store massive data and also manage concurrent tasks.

The general structure of HADOOP consists of two-level network structure, which was shown in below figure. Approximately 30 to 40 servers will be there in each rack with 1GB switch. Another 1GB switch or uplink was used to connect the racks respectively. The important point is aggregate bandwidth is higher while transferring the data or information among the nodes in the same rack then the different rack.

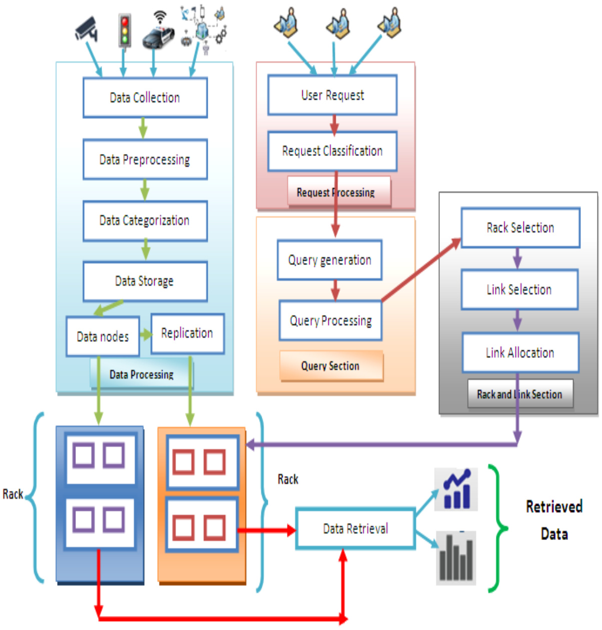
Configuring HADOOP is important to get maximum performance to know the structure of current network. If the cluster has multi rack then nodes mapping is essential otherwise you kept it as default.

Network locations are represented in the manner of hierarchical tree structure that mentions the locations distance. To decide for placing the block replicas the namenode get the information from network location and the job tracker used it for to find the closest replica, that is the input for to run on a task tracker.

For the network in Figure , the general rack structure is shown by two network locations, Such as, switch1/rack1 and /switch1/rack2. Here we are having one top-level switch in this cluster because

of two racks, further the locations can be changed as /rack1 and /rack2.

To overcome this problem in this paper we made a survey to find the optimum solution. Here we concentrate on to mapping the user request to the available rack by choosing the shortest path. In cloud, users may submit many requests, all request has to process concurrently without any delay. So we have to map the request to the respective rack for retrieving the data in minimum retrieval time.



The shortest path identification leads the performance problem because of multiple paths exists in the data center. Data retrieval problem arises during non local data with path selection. Heavy congestion on links because of overlapping of paths and nodes. So the perfect path selection and replica selection may increase the data transfer rate.

Problem Classification

The data are collected from various sources are identified and place over the data node either original or replicated data node. Then the data pre-processing method is used for collecting the information from multiple resources, transfer the data into required format, and stored it into the respective database is generally called as ETL in which 'E' stands for extraction, 'T' for transformation and 'L' stands for Loading. After the transformation we have to categorise it depends upon nature of the data.

Presently we are having data such as structured data, unstructured data and semi-structured data. Depends upon the categorization of data it will be separated and stored in data node and the replicated data node. In this survey we are going to concentrate on how to increase the data transfer rate across nodes in the cluster. First the user made a request for the retrieval of data, it should be analysed and fetch from the nearby replicate server with unique shortest path.

User raises a request for data retrieval. The HDFS has to classify the user request and it processes the request through a query engine, then the suitable link is selected from the rack and allocated for data retrieval process. The congestion in the link is eliminated by considering unused path to all other nodes.

To find out an optimum solution in the data transformation across the node in cloud system we need efficient scheduling algorithm. So that we undertake a survey on the existing scheduling algorithm and based on that to find a best scheduling algorithm.

Survey on Existing methods

Multiple analytical jobs are assigned to multiple jobs over different data center using Max-Min algorithm. It is not suitable for large scale processing with real time data set [1].

Heuristic bandwidth-aware task scheduler for scheduling is combined with Hadoop in order to assign the task more efficiently. It suffers with scalability. Not suitable for larger network cluster in large scale data processing [2].

Storage-Tag-Aware Scheduler (STAS) uses the shared queue job based scheduling with tag synchronization. It doesn't support the scheduling with multi-homing nodes which are connected through Mesh topology with hadoop cluster [3].

Dynamic workload based execution model is considered for minimizing the energy. It doesn't use distributed Scheduler with multiple jobs. Disadvantage is it uses single Map-Reduce scheduler, multi-scheduling is not supported [4].

The number of jobs generated by the applications are arranged in sequence and connected to the available resources by the Cat Swarm Optimization (CSO) based heuristic scheduling algorithm. This algorithm selectively picks the unused energy and removes it. The main advantage is it will find the result in less number of iteration process. The drawback is it does not efficiently support to process number of task simultaneously. [5]

Round random partitioning method is used to manage the distributed data blocks by selecting samples of data from the overall dataset. Its time consuming method because it perform this partition one time on every input data. The drawback is it not efficient for huge data analysis in multiple processes such as real-time data and different data centres. [6]

Dynamic load balance scheduling method is used to manage data transmission dynamically for maximizing the network throughput. Two algorithms used for to manage the data flow in slot basis and also dynamically it will change the network states. The drawback is presently it support only specific open flow network. [7]

To manage the resource allocation and to enhance the job scheduling in the enormous range of big data applications a significant method named as Resource co-allocation method was used. At present its not supported to huge amount of real time data. [8]

Nowadays vast number of big data applications requests the data in parallel for the concurrent process. So there was imbalance in the input and output data which means lot of consequences are there at the time of reading and writing the data. To rectify this read/write problem a HM-LRU policy was designed to fetch the data from the local node itself. The Opass method monitor the node status of input/output and also it manage the disparity of the read/write data But still now we didn't achieve 100% of optimum solution. [9]

Iterative optimal optimization method was suitable for creating a real time workload schedule in a shorter time and uses a low memory for this processing. Most probably it used for to provide cloud services. But the schedule may be deteriorating in certain circumstances such as increase in noise. [10]

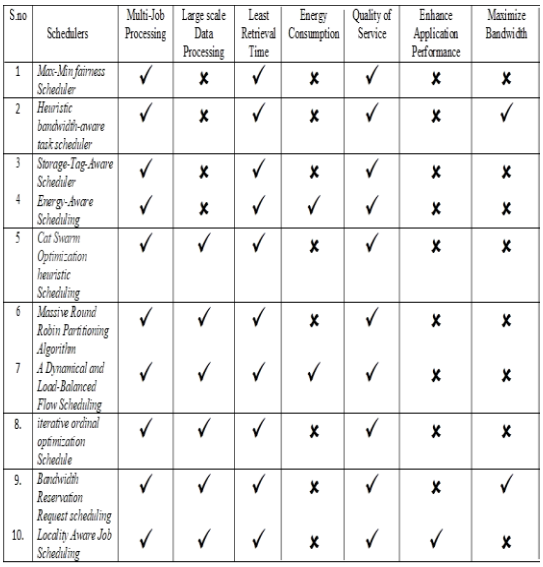
Many large-scale experimental and computational scientific applications are deployed in cloud computing in turns of big data generation is on daily basis. These kind of large volumes data are transferred to the remote applications for data analysis purpose. For lightning data transfer two efficient algorithm namely FBR-ECT (Fast Bandwidth Reservation algorithm for Earliest completion time) and FBR-SD(Fast Bandwidth Reservation algorithm for Shortest Duration) are used for to manage simultaneously schedule the multiple BRR's in one batch to achieve average ECT(earliest Completion time) and SD(Scheduled Duration) for Scheduled BRRs(bandwidth reservation requests).[11]

An efficient algorithm is used to schedule the job in minimum time for the virtual machine and by considering the job bandwidth and character, which access the job with status updation instead of Waiting for job finishing. . The drawback is not considering the dependency of each task to apply the task scheduling.[12]

The weight of the network path should not be same in the data centres, so considering the weight of the path to discover the efficient unequal path for splitting path to reduce path utilization in order to increase the network capacity by the Penalizing Exponential FlowspliTing (PEFT) algorithm.[13]

To increase the performance of cloud application and to reduce the local input/output dispute a heuristic algorithms was developed. A new archetype was developed to schedule the cloud services. At present it not supported to providing a GPFS interface. [14]

A load-balanced scheduler GLOBE is used in data center which manage the traffic between static and dynamic components. It's the best scheduler to reduce the traffics in dynamic. The drawback is the time consuming is lit bit high when compared to the other schedulers. [15]



S.no	Schedulers	Multi-Job Processing	Large scale Data Processing	Least Retrieval Time	Energy Consumption	Quality of Service	Enhance Application Performance	Maximize Bandwidth
1	*Max-Min fairness Scheduler*	✓	✗	✓	✗	✓	✗	✗
2	*Heuristic bandwidth-aware task scheduler*	✓	✗	✓	✗	✓	✗	✓
3	*Storage-Tag-Aware Scheduler*	✓	✗	✓	✗	✓	✗	✗
4	*Energy-Aware Scheduling*	✓	✗	✓	✓	✓	✗	✗
5	*Cat Swarm Optimization heuristic Scheduling*	✓	✓	✓	✗	✓	✗	✗
6	*Massive Round Robin Partitioning Algorithm*	✓	✓	✓	✗	✓	✗	✗
7	*A Dynamical and Load-Balanced Flow Scheduling*	✓	✓	✓	✓	✓	✗	✗
8.	*iterative ordinal optimization Schedule*	✓	✓	✓	✗	✓	✗	✗
9.	*Bandwidth Reservation Request scheduling*	✓	✓	✓	✗	✓	✗	✓
10.	*Locality Aware Job Scheduling*	✓	✓	✓	✗	✓	✓	✗

Table 1- Characteristics of various Schedulers based on Analysis

Conclusion

Based on the analysis finally we conclude that for the parallel computing still now we didn't have efficient scheduling algorithm to get the optimum solution. Because of the vast number of big data application in the cloud system we are in necessity of providing concurrent data transfer for all the applications. The data are collected from various sources are identified and place over the data node either original or replicated data node. The links should be analyzed and group those links for further data movement. The suitable link is selected and allocated for data retrieval process. The main objective is avoiding congestion at the time of link selection and have to select the nearest replica node to fetch the data in least retrieval time.To obtain this objective i have planned to develop a data transfer scheduling scheme based on hybrid algorithm. The objective of this paper is to develop an optimal data transfer scheduling using a hybrid approach combining monarch butterfly and fruit fly algorithm (HMBFF) in the cloud computing environment which will find, best scheduled path leading to the least data transfer time. The

MB is one of the recently proposed algorithms. It has solving global optimization problems fast and this algorithm ideally suited for parallel processing and well capable of making trade-off between intensification and diversification. FF can reach the global optimum easily, it has Solves the problems fast, easily adaptable to the applications and it has few parameters. In proposed work, FF is used in MB for migration operation, and this incorporated strategy can only accept the monarch butterfly individuals that have better fitness than their parents. This will improve the performance and to speed up the optimization process in data transfer scheduling. Finally, the performance of data transfer scheduling in terms of different evaluation metrics is analyzed.

REFERENCES

1. "Scheduling Jobs across Geo-Distributed Datacenters with Max-Min Fairness", IEEE Transactions on Network Science and Engineering,2018, Chen, L., Liu, S., Li, B., Li, B.
2. "Bandwidth-Aware Scheduling With SDN in Hadoop: A New Trend for Big Data", IEEE SYSTEMS JOURNAL, VOL. 11, NO. 4, DECEMBER 2017, Peng Qin, Bin Dai, Benxiong Huang, and Guan Xu
3. "Storage-Tag-Aware Scheduler for Hadoop Cluster",IEEE Access,2017, Qureshi, N.M.F., Shin, D.R., Siddiqui, I.F., Chowdhry, B.S.
4. "Energy-Aware Scheduling of MapReduce Jobs for Big Data Applications", IEEE TRANSACTIONS ONPARALLEL AND DISTRIBUTED SYSTEMS, VOL. 26, NO. 10, OCTOBER 2015, Mashayekhy, L., Nejad, M.M., Grosu, D., Zhang, Q., Shi, W.
5. "Workflow scheduling in cloud computing environment using Cat Swarm Optimization', Souvenir of the 2014 IEEE International Advance Computing Conference, IACC 2014, Saurabh Bilgaiyan, Santwana Sagnika, Madhabananda Das
6. A distributed data management system to support large-scale data anal ysis", Journal of Systems and Software,148, pp. 105-115, February 2019, Emara, T.Z.aEmail Author,Huang, J.Z.a,b
7. "A Dynamical and Load-Balanced Flow Scheduling Approach for Big Data Centers in Clouds" IEEE TRANSACTIONS ON CLOUD COMPUTING 2016, Feilong Tang Member, IEEE, Laurence T. Yang Senior Member, IEEE, Can Tang, Jie Li Senior Member, IEEE, and Minyi Guo Senior Member, IEEE
8. "A Resource Co-Allocation method for load-balance scheduling over big data platforms", Future Generation Computer Systems Volume 86, September 2018, Wanchun Dou a, *, Xiaolong Xu b , Xiang Liu a , Laurence T. Yang c,d , Yiping Wen a
9. "Achieving Load Balance for Parallel Data Access on Distributed File Systems", IEEE Transactions on ComputersVolume 67, Issue 3, 1 March 2018, Dan Huang†, Dezhi Han∗ ‡, Jun Wang†, Jiangling Yin †, Xuhong Zhang†, Xunchao Chen†, Jian Zhou†, Mao Ye†,
10. "Adaptive Workflow Scheduling on Cloud Computing Platforms with Iterative Ordinal Optimization", IEEE Transactions on Cloud Computing, 2015, Zhang, F., Cao, J., Hwang, K., Li, K., Khan, S.U.
11. "Concurrent Bandwidth Reservation Strategies for Big Data Transfers in High-Performance Networks", IEEE TRANSACTIONS ON NETWORK AND SERVICE MANAGEMENT, VOL. 12, NO. 2, JUNE 2015, Liudong Zuo, Student Member, IEEE, and Michelle M. Zhu, Member, IEEE
12. "Enhancement of Task Scheduling Technique of Big Data Cloud Computing',. 2018 International Conference on Advances in Big Data, Computing and Data Communication Systems, icABCD 20188465422, Abed, S., Shubair, D.S.
13. "Improving Data Center Network Utilization Using Near-Optimal Traffic Engineering", IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS, VOL. 24, NO. 6, JUNE 2013, Tso, F.P., Pezaros, D.P.
14. "Locality-aware Scheduling for Containers in Cloud Computing", IEEE Transactions on Cloud Computing,2018, Zhao, D., Mohamed, M., Ludwig, H.
15. "Profile-based power-aware workflow scheduling frame work for energy-efficient data centers", Future Generation Computer Systems,2019, Qureshi, B.
16. "objective Cat Swarm Optimization Algorithm for Workflow Scheduling in Cloud Computing Environment", Advances in Intelligent Systems and Computing Volume 308 AISC, Issue VOLUME 1, 2015, Saurabh Bilgaiyan, Santwana Sagnika and Madhabananda Das
17. "PRISM: Fine-Grained Resource-Aware Scheduling for Map Reduce", IEEE Transactions on Cloud Computing,2015, Zhang, Q., Zhani, M.F., Yang, Y., Boutaba, R., Wong, B.
18. "Joint Static and Dynamic Traffic Scheduling in Data Center Networks", IEEE/ACMTRANSACTIONSONNETWORKING,2016, Cao, Z., Kodialam, M., Lakshman, T.V.
19. "Intelligent Scheduling for Parallel Jobs in Big Data Processing Systems", 2019 International Conference on Computing, Networking and Communications, 2019, Xu.,M, Wu, C.Q., Hou, A., Wang, Y.
20. "Cross-platform Resource Scheduling for Spark and Map Reduce on YARN", IEEE TRANSACTIONS ON COMPUTERS, VOL. , NO. , NOVEMBER 2016, Dazhao Cheng, Xiaobo Zhou, Palden Lama, Jun Wu and Changjun Jiang
21. "DyScale: a MapReduce Job Scheduler for Heterogeneous Multicore Processors", IEEE Transactions on Cloud Computing 5(2), pp. 317-330, 2017, Yan, F., Cherkasova, L., Zhang, Z., Smirni, E.
22. "Flutter: Scheduling Tasks Closer to Data Across Geo-Distributed Datacenters", IEEE INFOCOM 2016 - The 35th Annual IEEE International Conference on Computer Communications, Zhiming Hu1, Baochun Li2, and Jun Luo1