International Journal of Engineering & Technology, 7 (2.21) (2018) 210-216

International Journal of Engineering & Technology

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

Research paper

Implementation of cost effective hierarchical Hadoop cluster–a case study for education

N.S. Kalyan Chakravarthy[1*], N. Sudhakar[2], E. Srinivasa Reddy[3]

[1]Research Scholar, Department of Computer Science, University College of Engineering & Technology, Acharya Nagarjuna University
[2]Professor, Department of Computer Science, Bapatla Engineering College.
[3]Principal, University College of Engineering & Technology, Acharya Nagarjuna University.
**Corresponding author E-mail: suryaetg@yahoo.com*

Abstract

To equip the younger generation of the province with computing skills and provide them access to a wide variety of modern educational resources such as multimedia based on educational content in government schools and colleges to set a strong foundation at an early stage of their education. Educational Media help to empower educational institutions by altering the way of using Information Communication Technology (ICT). To put this into practice, several challenges need to be addressed. It requires a scalable technological architecture and algorithms to form a cluster in which effective resource sharing like CPU, Memory among multiple nodes, tools to monitor, assess and evaluate data under hierarchical Hadoop cluster is needed. By analyzing text, audios, videos information, Periodic reports will be generated to assist the students, teachers and Government. In this case, a software framework is required to process big data stored in hierarchical nodes. As the architecture of Hadoop, an open source software framework doesn't support processing the data stored in hierarchical nodes. This case study proposes the Hierarchical Hadoop cluster to alter the way of using ICT. The proposed work helps in monitoring, reporting the usage of ICT and also acts a help desk to address the issues of the educational institutions. This establishes a novel communication media by generating reports on text, audio, video information based on analysis.

Keywords: *Hadoop, cluster, map reduce processing, Ed-Media.*

1. Introduction

Technology plays a prominent role in empowering the system of education even in rural India. Many of the government schools and colleges are partially equipped with Technological infrastructure like a Computer system, Projectors, Printers, Phones, Internet. Redundant data stored in multiple desktops may lead to confusion and memory wastage. These desktops have limited processing and storage capabilities. The communication with organizational chart is always done through traditional tools and technologies. Moreover, the Science laboratories are not utilized effectively.

Even though most of the countries are spending millions of rupees on education, still the final outcomes are not very effective [1]. Government is focusing on suitable pedagogical concerns for class-room teaching, virtual laboratories that enable students to perform experiments online. But, the main challenge lies in determining the impact of investments with respect to infrastructure, teacher training initiatives and resource usage [2].
A cost-effective energy efficient, scalable Linux based Hadoop cluster is required to address these issues and to keep track of the usage of the ICT services provided by the government to the schools and colleges in villages or talukas or mandals in India. In this model, all the government institutions desktops will act as data nodes and connected to a single Hadoop Cluster.

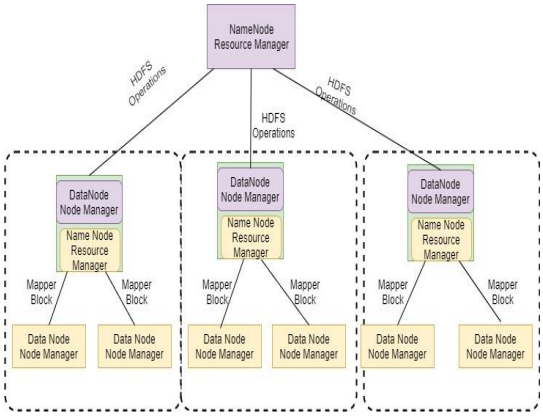


Fig. 1: Block diagram–hierarchical Hadoop Cluster

Copyright © 2018 Authors. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Hadoop hierarchical cluster helps in connecting schools under different mandals, districts and state clusters. To preserve data, these nodes are replicated with three-fold technique by default in this cluster. A reporting server should be installed, for establishing effective communication between these data nodes. Thus, the Indian educational system cans be empowered with the continuous monitoring and enhancing the usage of resources.

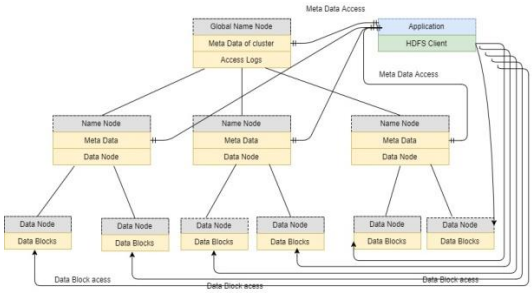


Fig. 2: Hierarchical hadoop distributed file system

2. Related work

Cheng-Zhong Xu et. al. proposed a structure called CoTuner, to coordinate the configurations of VMs and resident applications. It drives a virtual server cluster into the best possible or near-optimal configuration state on the fly and success in improving the throughput of the system by 30 percent compared to independent tuning strategies [3]. Georgios L. Stavrinides et. al. recommended two synchronization models to establish communication through the network and through files. The performance of these models were tested on dynamic multiple soft real-time workflow applications [4].

Yanfei Guo et.al. Suggested FT-MRMPI, fault tolerant MapReduce framework for HPC Clusters. This model is used to detect and recover from failures by using the MPI Semantics. This paper shows the experimental results generated by applying the proposed model on 256-node HPC cluster and determined that FT-MRMPI succeeded in detecting the faults and in reducing the job completion time by 39% [5].

Yanfei Guo et. al. presented a mechanism Flexslot, at echnique to identify the map stragglers automatically and alter the size of slots accordingly to accelerate task execution. It enables a flexible way to change the number of slots and memory size of the slot of a slave node. It achieved in reducing job completion time by 46% over stock Hadoop and 22%over Skew Tune techniques [6].

Dazhao Cheng et. al. stated an approach Ant, to search the optimal configurations of individual tasks running over different nodes automatically. This method achieves this by dividing nodes into a number of homogeneous sub-clusters and applies self-tuning algorithm on individual sub-clusters.

The experimental results showed that this technique reduces the job completion time by 31%, 20%, and 14% overstock Hadoop (Stock), customized Hadoop with industry recommendations (Heuristic)and a profiling-based configuration approach (Starfish) respectively [7].

Shanjiang Tanget. al. proposed a new fair resource allocation mechanism, Long-Term Resource Fairness(LTRF).LTRF allows to share the unused resources among other clients. It ensures that the client should pay only for the resources they used [8].

Jian Lin et. al. adopted aglobal persistent state model to solve the contradiction of dataset coupling relationships and the contradiction of state synchronization relationships. This technique achieves the objectives of transparency, efficiency, feasibility. For a usual 48GB TeraSort application, it introduces only 6.8% time overhead and 11% space overhead. For a failure-resume execution, the additional time overhead of local resume and remote resume are only 3.7%and 25%, which provides a 10% to 32% performance advantage compared with the –ve checkpoint solutions based on local or parallel storages [9].

Ioannis A. Moschakisa et.al., did research on optimization of interlinked cloud systems. In the scheduling of a dynamic multi-cloud system with virtual machines of heterogeneous performance serving Bag-of-Tasks applications, the author conducted Thermodynamic Simulated Annealing and Simulated Annealing. Experimental results showed that the use of these heuristics had a considerable impact in performance while maintaining a good cost-performance trade-off [10].

KeWang et. al. introduces distributed task execution framework, MATRIX to address YARN scaling issues. Matrix was compared with YARN in terms of Word Count, TeraSort, Grep and Random Writer, and the Lig and application in Bioinformatics on the Amazon Cloud. The simulation results demonstrated that MATRIX yield best results over YARN [11]. DjellelEddine Difallah et. al. proposed crowd sourcing system architecture to enhance the efficiency of the crowd while running different types of tasks in parallel [12].

GeorgiosL. Stavrinides et. al. examined the performance of different gang scheduling algorithms like EDF,EDF_RAC,EDF_FAC and demonstrated that EDF_ADC_RAC is yielding best results in terms of quality, whereas the EDF FAC policy generating great cost saving sand highest system performance [13]. Yi Yao et. al. in his paper mentioned that as current Hadoop is not suitable for dynamic slot configuration, he proposed new mechanism TuMM, to allocate slots for map dynamically and reduce tasks [14].

Dazhao Cheng et. al. developed and proposed a self-tuning request batching mechanism to enhance the performance of Internet applications and DVFS power management to diminish the energy consumption of server systems. [15]. Fang Zhou et. al. in his paper, stated that as Hadoop is not feasible for small files in terms of storage as well as processing. He adopted optimized Map Reduce framework for small files, SFMap Reduce. This paper states that SF Map Reduce achieves an improvement on MapReduce processing by 14.5 times and 20.8 times, over the original Hadoop and HAR layout [16].

Karthik Kambatla et. al. in this work, compared various MapReduce-based analytics of a cloud platform. This paper talks about the challenges facing by end-users in this environment and techniques to incur costs, while obtaining the bestperformance. The author presented a pilot approach to improve MapReduce provisioning by analyzing and comparing resource utilization over related resource utilization signatures of other applications [17].

Sun Shengtao et. al. stated that conventional MapReduce model and Hadoop framework bound themselves to work within single cluster. In this paper, A hierarchical distributed computing architecture of Hadoop is designed and the virtual Hadoop file system is proposed to provide global data view across multiple data centers. The user jobs will be automatically divided into several sub jobs which are then allocated and executed on corresponding multiple clusters [18]. Kannan Govindarajan et. al. studied different means to measure students learning efficiency and proposed a method to capture data from student learning interactions periodically. Then, this data is analyzed to discover their individual performances in terms of efficiency, accuracy and quality by using Particle Swarm Optimization (PSO) algorithm.

Copyright © 2018 Authors. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

S Nair et. al. recommended a data mining tool, The self-organizing map (SOM). This paper showed the experimental results of SOM and demonstrated that the tool can process more diverse data and enhances the performance of clustering [19].
Kiran Kumar Pulamolu et. al. proposed a novel priority based Weighted Arithmetic Mean resource sharing using YARN to overcome the challenges like resource fairness, energy, partition skew while running workloads on Heterogeneous clusters. This paper showed the experimental results stating that this approach helps in improving the resource sharing by 20% [20].
Kiran Kumar Pulamolu et. al. stated a new model to improve the efficiency of resource utilization. This paper stated Regression Ranking (CRRYARN).It predicts ranks of different workloads based on amount of resource utilization factors like CPU Intensive, Memory intensive etc.., Based on this ranking the resource sharing happened among the workload in Single Tenants of the same cloud. It works on loan lending with interest free, rank based discount share model. It improves performance ofthe existing model by 30% effectively [21].
Venkata et. al. proposed fuzzy based modeling to assess the effectiveness of any organizations' IT Security Policy. This model analyzes fuzzy rules having similar rule strengths by taking additional factors into consideration. The experimental results show that it accomplished in solving the vulnerabilities rose with high rule strength. [22] Ahmed Hosny Ibrahim et. al. proposed Resource sharing algorithm to share GPUs in Infrastructure as a Service model (IaaS) for the cloud users. This algorithm is scalable and extendable to meet Service Level agreements. In a cloud environment, GPUs are playing a vital role as processing accelerators in enhancing server CPU processing performance. This paper proposes a new algorithm, RAA to deal with the heterogeneous nature of the cloud environment by using GPU as a computing element. The proposed algorithm improves the scalability for cloud providers by enabling them to offer the GPUs in the SaaS context. [23]
Mahmoud Al-Ayyoub et. al. proposed multi-agent framework model to efficiently share and utilize resources in cloud environment. This paper presented dynamic resources provisioning and monitoring system (DRPM), a system to manage the cloud provider resources by satisfying the customers' QoS requirements as determined by the SLA. The adopted DRPM system is evaluated using the CloudSim tool and the results generated shows that the system allows the cloud providers in increasing resource utilization with less power consumption. [24]
Venkata et. al. proposed an approach to design a multidimensional Restful Resources and Web Services using mutli-dimensional metric model for KM systems. The author attempted to build a four dimensional model, the metric database and a statistical technique utilizing Weighted Average Mean to validate the effectiveness of the knowledge sharing in the KM systems. The experimental result proves that the combination of Metric Database with a Statistical technique such as WAM could be useful to predict the usefulness and effectiveness of the KM Systems. [25]

3. Proposed work

Many educational institutions have been set up all over the country. It is very difficult to monitor and keep track of resource utilization in hundreds of educational institutions continuously. There is no proper mechanism either to generate reports on this resource utilization or to store these reports. An online help desk is necessary to address the ICT issues of the schools and colleges. To address these challenges, this paper proposes a software framework called hierarchical Hadoop cluster to process big data stored at educational institutions. This mechanism will consider every school as a data node. We adopted and studied the architecture of Apache hadoop cluster. The structural design of the hierarchical methodology is as shown in fig.(1)
The architecture shows a master-slave relationship between global name node as master and data nodes as slaves. The intermediate node acts as name node. Global name node is the vital node which processes user requests. It splits the user tasks into sub-tasks and share out them to the corresponding name nodes. Global node also takes care about the metadata stored in global name node which is used to process user queries. The global name node is also responsible to manage the metadata of all the files available in every data node.
The data node is installed on each cluster and is enabled to run sub-jobs allocated by parent name node. Hierarchical Hadoop cluster provides a global view of all files stored at different clusters [26]. In Hierarchical Hadoop cluster, data must be managed in such a way that the information should be accessed by any master node. Global name node just records meta-data of all HDFS files obtained from name nodes, such as data size, file format, access path and so on. When a file is newly generated or existing file is modified, the respective name node sends the updated metadata to its parent name node. The architecture of Hierarchical Hadoop Distributed File System is shown in Fig. (2).
Global name node stands at the top of the hierarchy. It consists of metadata; data that holds the details of sub name nodes and access logs; used to access the data stored in name nodes. User can query and obtain the information based on metadata in global name node. The user query is further automatically converted into more detailed address for performing operations on files, which includes: cluster address, HDFS directory and block number. The location where the program is being executed is identified using cluster address, paths regarding data nodes are located using the HDFS directory, and data are identified with their block numbers [27].

Algorithm 1: HHC Resource Sharing Pseudo code

1: RA: Available Resources in Cluster.
2: RA'= (RA_1 ...RA_n) Resources Allocation. RA_i denotes resource allocation for client i.
3: RD =(RD_1...RD_n) resources demanded by tenants. rd_i denotes resources demanded by client i.
4: W=(W_1 ...W_n): Weight of workloads, W_i denotes the weight for client i.
5: if RD_i less than RA_i then
6: UA_i<- RDi#Allocate demanded resources
7: UP_i<- RA_i – RD_i #Resources Preempted
8: WAM <- $\sum W_i*RD_i / \sum W_i$
9: while : execute pending tasks
10: if WAM < RA && UP >0 then
11: UAi<- RDi
12: if WAM ≈ RA && UP >0 then
13: UAi<- RDi%50
14: if WAM > RA&& UP >0 then
15: UAi<- RDi%25
16: else
17: Wait until there is a released resource rifrom clienti
18: Update heuristic table for client i.

The major challenge of Hierarchical Hadoop cluster is to maintain fairness in resource allocation among all the tenants. To achieve this, a novel algorithm named HHC Resource Sharing is proposed in this paper. The steps involved in this algorithm are as shown inAlgorithm1. Let us consider n number of child clusters in Hierarchical Hadoop Cluster.

The Global Namenode chooses the child cluster based on the resources and data availability. In this scenario, the total number of resources available in cluster be RA. RA defines the set of individual resources allocated to each tenant. RD determines the total number of resources demanded by all the tenants at a particular instance of time where rdi represents the resources demanded by client i. W describes the set of weight of workloads of all the tenants and Wi represents the weight of individual tenant i. When a tenant sends a request for resources, the demand will be immediately granted if its demand for resources (RDi) is less than resources available(RAi) with it and user resource preempted (UO) will be updated.

If the tenant demands more resources than its availability, the request will be granted by taking WAM into consideration. Weighed arithmetic mean (WAM) is calculated by dividing the summation of product of each individual tenant's weights and resources demanded with summation of weights of all the tenants. If WAM is less than resources available and UP is greater than zero, then the total resources demanded will be allocated. If WAM is approximately equals to resources available and UP is greater than zero, then fifty percent of the available resources will be allocated.

If WAM is greater than RA and UP is greater than zero, then only twenty five percent of the resources will be assigned. If there is no resource available to allot, then the tenant needs to wait until some other tenant preempts its resources. All the transitions regarding resource allocation and preemption of every tenant should be updated in heuristic table. An efficient Hierarchical Hadoop cluster can be designed using HCC resource sharing algorithm proposed in this paper. Hierarchical Hadoop cluster framework can be employed to empower educational institutions to alter the way of using Information Communication Technology (ICT). Central Online help desk will act as a global data node and is accountable to monitor and address the ICT issues of the schools and colleges. Every School acts as a data node. The online help desk at district will act as data node to central online helpdesk and as name node to all the schools under that district. This mechanism also aids in generating periodic reports on text, audios, and videos to assess resource utilization.

4. Results and Discussion

Hierarchical Hadoop Cluster setup with30node,each node is created with Intel dual core processor,2GB RAM, 50GB HDD specification. These nodes are installed with Linux/Cent OS 6.8 operating system, Java 1.8 and Hadoop 2.7.3 to establish hierarchical Hadoop Cluster. One node act as Global Name node, which runs Name Node and Resource Manager Services to communicate with slave clusters. In slave cluster one node is running with Name Node, Resource Manager Services and rests of the nodes are running with DataNode, Node manager services.

Benchmarks and applications

Word count –Word count is a map reduces application to count frequency of words in given dataset. The map program accepts key-value pair as input and produces key-value pair as output. In this phase each record tokenized into words. The reducer phase accepts key value pair as input and produces key-value pair as output. It counts the frequency of each word/key by accepting key-value pairs associated with the same key. This is the best way to understand map reduce programming. In this paper, five different Map Reduce workload shave submitted to seven Tenants to Test the performance of Capacity, Fair, LTYARN and HHCRS Algorithms.

The experimental results show that HHCRC Algorithm has succeeded in using the resources effectively and share resources with fairness.

Table 1: Benchmark Details

Benchmark	Input Size (GB)	Input Data
Tera-Sort	150	TeraGen
Tera-Validate	150	TeraSort
Wordcount	150	Wikipedia
Maxium Temp	80	NCDC
Grep	150	Wikipedia

Table 2: Performance Measure-CPU Utilization in Hierarchical Hadoop Cluster

CPU	T1	T2	T3	T4	T5	T6	T7	T8	Average CPU Time in ms
Capacity Scheduler	22972	5760	16852	34470	16853	34470	5760	19912	157047
Fair Scheduler	21555	4343	15435	33053	15435	33053	4343	18495	145710
LTYARN Scheduler	20835	3623	14715	32333	14715	32333	3623	17775	139950
WAMYARN Scheduler	20498	3285	14378	31995	14378	31995	3285	17438	137250

Table 3: Performance Measure-Memory Utilization in Hierarchical Hadoop Cluster

Memory in MB	T1	T2	T3	T4	T5	T6	T7	T8	Average Memory Usage in MB
Capacity Scheduler	8545	8545	3415	3415	3415	3415	8545	3483	42781
Fair Scheduler	8032	8032	2902	2902	2902	2902	8032	2970	38677
LTYARN Scheduler	7168	7168	2038	2038	2038	2038	7168	2106	31765
WAMYARN Scheduler	6223	6223	1093	1093	1093	1093	6223	1161	24205

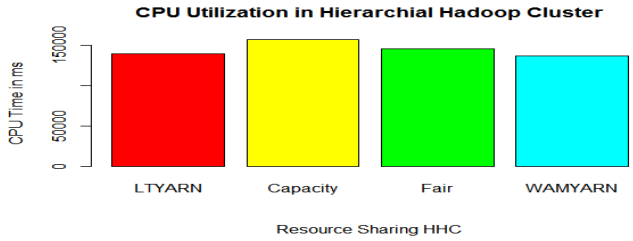


Fig. 3: Average CPU Utilization time in hierarchical hadoop cluster

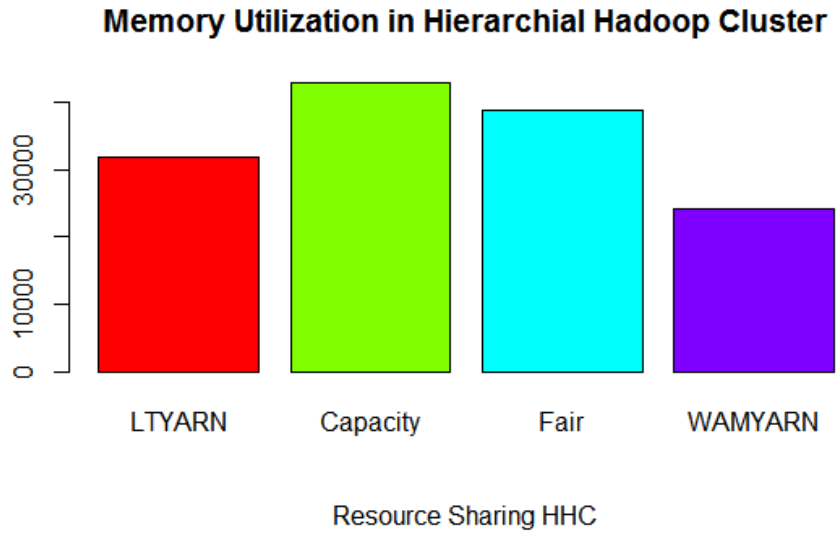


Fig. 4: Average memory utilization time in hierarchical hadoop cluster

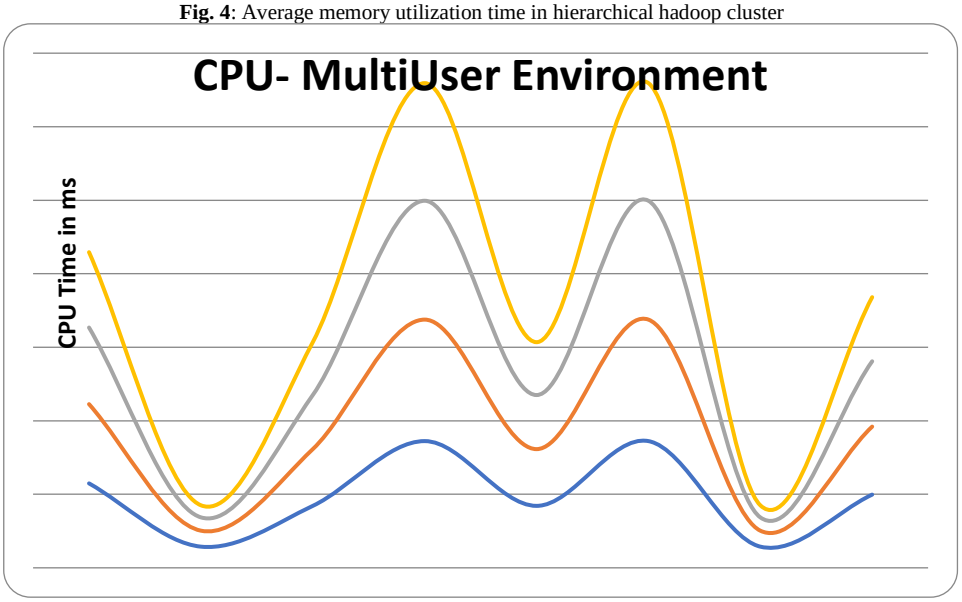


Fig. 5: Compare CPU utilization in hierarchical hadoop cluster

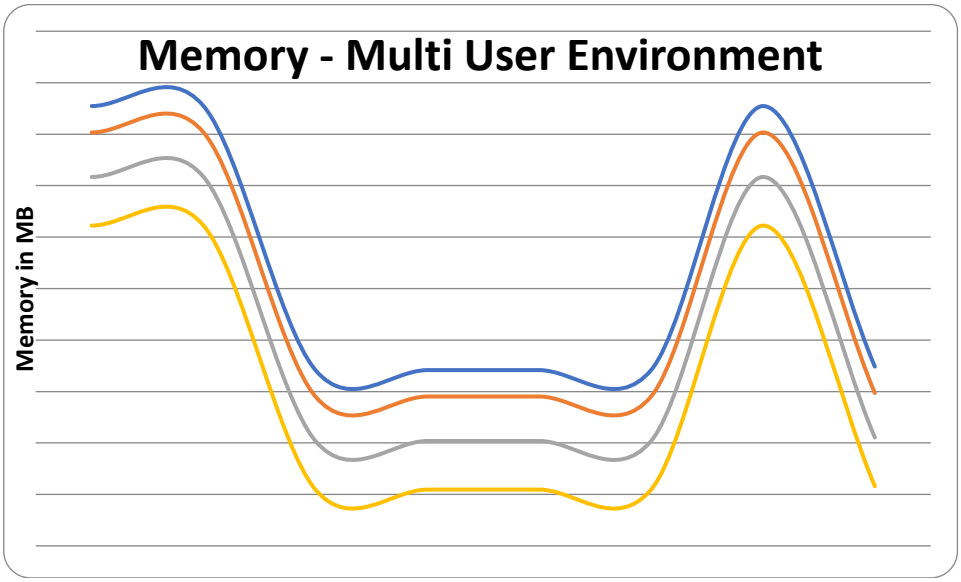


Fig. 6: Compare memory utilization in hierarchical hadoop cluster

5. Conclusion

In this paper, a Cost effective Hierarchical Hadoop Cluster is designed to process big data stored in hierarchical nodes. HHCRS algorithm is adopted in Hierarchical Hadoop Cluster to share resources effectively among multi tenants with resource fairness. This cost effective Hierarchical Hadoop framework can be employed to empower educational institutions to alter the way of using Information Communication Technology (ICT). The architecture sets up a novel communication media among stakeholders with the help of nodes in the cluster. It effectively collects and process data like text, audio, video information from all the nodes. The analytical reports generated can be aided in smooth functioning of the educational system. The experimental results demonstrates the performance of HHCRS algorithm over other scheduling algorithm with respective to CPU and memory utilization. It improves the resource utilization by 40% and succeeded in acquiring resource fairness in cloud based clusters.

References

[1] Govindarajan K, Somasundaram TS & Kumar VS, "Continuous clusteringin big data learning analytics", *IEEE Fifth International Conference on Technology for Education (T4E)*, (2013), pp.61–64.

[2] Pulamolu MKK, "A novel resource allocation using dynamic heterogeneity priority based flow shop algorithm in yarn", *Alexandria Engineering Journal*, (2017).

[3] Bu X, Rao J & Xu CZ, "Coordinated self-configuration of virtual machines and appliances using a model-free learning approach", *IEEE transactions on parallel and distributed systems*, Vol.24, No.4, (2013), pp.681–690.

[4] Stavrinides GL, Duro FR, Karatza HD, Blas JG & Carretero J, Different aspects of workflow scheduling in large-scale distributed systems", *Simulation Modelling Practice and Theory*, Vol.70 (2017), pp.120–134.

[5] Guo Y, Bland W, Balaji P & Zhou X, "Fault tolerant mapreduce-mpi for hpc clusters", *Proceedings of the International Conference for High Performance Computing Networking, Storage and Analysis*, (2015).

[6] Guo Y, Rao J, Jiang C & Zhou X, "Flexslot: Moving hadoop into the cloud withflexible slot management", *SC14: International Conference for High Performance Computing, Networking, Storage and Analysis*, (2014), pp. 959–969.

[7] Cheng D, Rao J, Guo Y, Jiang C & Zhou X, "Improving performance of heterogeneous map reduce clusters with adaptive task tuning", *IEEE Transactions on Parallel and Distributed Systems*, Vol.28, No.3, (2017), pp.774–786.

[8] Tang S, Lee BS, He B & Liu H, "Long-term resource fairness: Towards economic fairness on pay-as-you-use computing systems", *Proceedings of the 28th ACM international conference on Supercomputing*, (2014), pp. 251–260.

[9] Lin J, Liang F, Lu X, Zha L & Xu Z, "Modeling and designing fault-tolerance mechanisms for mpi-based map reduce data computing framework", *IEEE First International Conference on Big Data Computing Service and Applications (Big Data Service)*, (2015), pp. 176–183.

[10] Moschakis IA & Karatza HD, "Multi-criteria scheduling of bag-of-tasks applications on heterogeneous interlinked clouds with simulated annealing", *Journal of Systems and Software*, Vol.101, (2015), pp.1–14.

[11] Wang K, Liu N, Sadooghi I, Yang X, Zhou X, Li T, Lang M, Sun XH & Raicu I, "Overcoming hadoop scaling limitations through distributed task execution", *IEEE International Conference on Cluster Computing (CLUSTER)*, (2015), pp.236–245.

[12] Difallah DE, Demartini G & Cudré-Mauroux P, "Scheduling human intelligence tasksin multi-tenant crowd-powered systems", *Proceedings of the 25th International Conference on World Wide Web, International World Wide Web Conferences Steering Committee*, (2016), pp.855–865.

[13] Stavrinides GL & Karatza HD, "Scheduling real-time parallel applications in saas clouds in the presence of transient software failures", *International Symposium on Performance Evaluation of Computer and Telecommunication Systems (SPECTS)*, (2016), pp. 1–8.

[14] Yao Y, Wang J, Sheng B, Tan CC & Mi N, "Self-adjusting slot configurations for homogeneous and heterogeneous hadoop clusters", *IEEE Transactions on Cloud Computing*, Vol.5, No.2, (2017), pp.344–357.

[15] Cheng D, Guo Y & Zhou X, "Self-tuning batching with dvfs for improving performance and energy efficiency in servers", IEEE *21st International Symposium on Modeling, Analysis & Simulation of Computer and Telecommunication Systems (MASCOTS)*, (2013), pp. 40–49.

[16] Zhou F, Pham H, Yue J, Zou H & Yu W, "Sfmapreduce: An optimized map reduce framework for small files", IEEE *International Conference on Networking, Architecture and Storage (NAS)*, (2015), pp. 23–32.

[17] Kambatla K, Pathak A & Pucha H, "Towards optimizing hadoop provisioning in the cloud", *HotCloud*, Vol.9, (2009).

[18] Sharma B, Wood T & Das CR, "Hybridmr: A hierarchical mapreduce scheduler forhybrid data centers", *IEEE 33rdInternational Conference on Distributed Computing Systems (ICDCS)*, (2013), pp.102–111.

[19] Nair S & Mehta J, "Clustering with apache hadoop", *Proceedings of the International Conference & Workshop on Emerging Trends in Technology*, (2011), pp. 505–509.

[20] Pulamolu KK, Bhavani T & Subramanian DV, "Intra-Tenant resource sharing in yarn based on weighted arithmetic mean", *International Conference on Networks & Advances in Computational Technologies (NetACT)*, (2017), pp.262-265.

[21] Pulamolu MKK, "An efficient resource optimization in intra-tenant heterogeneous hadoop cluster", *IEEE Conference, International Conference on Intelligent Computing and Control Systems–ICCS*, (2017).

[22] Subramanian DV & Kumar KP, "Fuzzy based modeling for an effective it security policy management", *SAI Computing Conference (SAI)*, (2016), pp.173–181.
[23] Ibrahim AH, Faheem HEDM, Mahdy YB & Hedar AR, "Resource allocation algorithm for gpus in a private cloud", *International Journal of Cloud Computing*, Vol.5, No.1-2, (2016), pp.45–56.
[24] Al-Ayyoub M, Daraghmeh M, Jararweh Y & Althebyan Q, "Towards improving resource management in cloud systems using a multi-agent framework", *International Journal of Cloud Computing*, Vol.5, No.1-2, (2016), pp.112–133.
[25] Subramanian DV, Geetha A, Mehata K & Hussain KM, "Kmsystem evaluation using four dimensional metric model, database and restful resources", *International Journal on Web Service Computing*, Vol.3, No.3, (2012).
[26] Stavrinides GL & Karatza HD, "Scheduling different types of applications in a saas cloud", *Proceedings of the 6th International Symposium on Business Modeling and Software Design (BMSDâŁ™16),* (2016), pp. 144–151.
[27] Li B, Zhao H & Lv Z, "Parallel isodata clustering of remote sensing images based on map reduce", *International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery(CyberC)*, (2010), pp.380–383.