An Approach to Optimize QOS Scheduling for MapReduce in Big Data

Rama Satish K V
Research Scholar, R&D Centre
R N S Institute of Technology
Bangalore, INDIA
ramasatish.k.v@rnsit.ac.in

Dr N P Kavya
Professor and Head
Department of MCA, R N S Institute of Technology
Bangalore, INDIA
npkavya@rnsit.ac.in

***Abstract*— In present day scenario, the Cloud has become an inevitable need for majority of IT operational organizations. In the last few decades, the best platform to develop data-insensitive applications, the World Wide Web was most preferred as it posses the communication characteristics as more efficient and open in nature. The predominant techniques for data insensitive applications are various search engines, online retail operations, social media, web mails etc. These all data-insensitive applications are based on data mining and related indexing approach that require the spread out of data sizing from few gigabyte to multiple Petabytes. In order to gain a higher productivity and efficient data utilization, the analysis of Big Data is must. Apache Hadoop is one of the most emerging technologies being considered for QoS oriented data retrieval and cloud optimization.**

Keywords - Big Data, Data Mining, Hadoop, MapReduce, etc.,

I. Introduction (*Heading 1*)

The cloud computing is in general stated as an internet based computing/service where all the comprising functions such as data storage, servers and applications are facilitate to certain organizations using internet connectivity. Thus, comparing this technique with conventional approach of "Owe and Use" concept, the implementation of cloud technology might be a potential infrastructure that would eliminate the issue of purchasing and maintaining severs applications or infrastructure. The cloud infrastructure facilitates the allied users to access and employ the resources as per their requirement in real time applications. Thus, it can be stated here that this cloud computing system facilitates users to have a very expedient and of course on-demand resource access for the resources.

The emerging applications cause them to become more data-insensitive in behavior. In the last few decades, the best platform to develop data-insensitive applications, the World Wide Web was most preferred as it posses the communication characteristics as more efficient and open in nature[5]. The predominant techniques for data insensitive applications are various search engines, online retail operations, social media, web mails etc. These all data-insensitive applications are based on data mining and related indexing approach that require the spread out of data sizing from few gigabyte to multiple Petabytes. Considering an example of Google search engine, even if with MapReduce implementation in parallel processing, it has to process approximate 20 Petabytes of data every day. Certain survey states that in 2010, there were 1.2 Zeta bytes of data present as digital data and in the same way in 2011 there were 300 Quadrillion of file present as unstructured data [8]. Thus, considering such huge data collection and its efficient retrieval, a concept called Big Data came into existence.

The concept of present trends of Big Data is different from the traditional Business Intelligence approach. In present time, there are gigantic data collections in cloud which are required to be accessed in certain optimum way. A number of approaches have been developed for QoS oriented data placements and its quality utilization. And a number of issues are there which are required to be taken into consideration for optimizing the Big Data systems architecture. Few of the predominant issues are modeling of true risks factors, recommendation engines, threat analysis, Ad targeting, churn analysis of customers, trade surveillance, etc. In order to gain a higher productivity and efficient data utilization, the analysis of Big Data is must. Apache Hadoop is one of the most emerging technologies being considered for QoS oriented data retrieval and cloud optimization.

II. Literature Survey

In present day scenario, there has been a great demand of optimum data access and resource utilization with minimum latency and system complexity and for these all, there must be a fare management optimization in processing units of Hadoop. Considering the organizational details of Hadoop systems, there are two predominant parts of Hadoop. The first component presents Hadoop Distributed File systems (HDFS) and second components states for Map Reducing framework.

A. Hadoop Distributed File Systems (HDFS)

HDFS has been developed while keeping in mind of the function of Google File System (GFS). In the present day scenario, the HDFS system classifies whole files into certain defined block sets which are further replicated into numerous other nodes with no specific monitoring whether the blocks are being divided evenly. Whenever there is the initialization of certain Job the processor allied with individual node functions with its allied local disk. The specificity of Hadoop is that this system can perform efficiently with the structured as well as unstructured data.

In HDFS system, the overall data is at first organized into certain files and directories, where the files are divided into certain data blocks that are further distributed across cluster nodes. In order to handle the block there is always replication and for assuring the data security the checksum is taken into consideration. In Hadoop the system performance or the allied clusters can be effectively optimized because these multiple nodes do function concurrently so as to deliver higher throughput. This is the matter of fact that the Hadoop system is becoming a potential system for robust computing platform in data-insensitive applications, then while the rising up issues allied with performance due to the lack of high disk and network input-output resources and high paced increase in input output processing its efficiency is getting confined.

It can also be stated that in case of cluster environment for data-insensitive applications the disk and network input output related process causes higher bottlenecks in performance as compared to other factors like memory or CPU processing.

B. MapReduce Systems

The second and the most significant part of Hadoop is the MapReduce system that introduces itself as a framework that can effectively simplify the overall complexity of running parallel data process across the network of computing nodes. The system of MapReduce facilitates the programmer for distributing programs across the nodes in the cluster. In Hadoop system, MapReduce is a programming architecture and allied combined implementation of system for processing huge data sets and this scheme has been specially developed with the goal to process with unstructured data and thus it functions in such a way that it parallelizes and executes the huge data sets or jobs over a computing cluster. This system is sufficient for processing 100s of terabytes of data over 1000s of distributed nodes in certain defined clusters. This system handles chaotic information like dealing with failure, development of applications, duplications of task or jobs, result's aggregation, and thus this approach assists the developer or programmers to emphasize on the functional logic of applications. The framework MapReduce has been popularized by open-source Hadoop model because of its efficiency in exploiting huge data sets and in majority of the large scale Big Data applications, it has been evaluated for fulfilling growing business goals, and thus scheduling approaches for quality resource utilization and performance optimization is being advocated a lot to enhance its applications for Big Data era.

The uniqueness of MapReduce framework is that this approach takes into consideration of handling of data collections across the multiple nodes and then returns back the single entity results or data sets. On the other hand this is the factor that ascertains the fault-tolerance which is visible to the developers or programmers. Thus, considering the architectural view of Hadoop application for Big Data scenario, the performance enhancement can be obtained efficiently when focused on MapReduce optimization. The proper load distribution across the clusters and the allied nodes might be a significant step to ensure the Quality of Service (QoS).

III. Performance Optimisation

Here in this paper, we have focused on the highly robust factor that could efficiently optimize the system performance and it is MapReduce with certain scheduling approach to optimize QoS. In order to develop or optimize the existing Hadoop system's MapReduce model, it is better to understand the real operations to be executed in MapReduce model. Herewith the processing of MapReduce Model has been discussed so as to come with an objective, where the possibility of further optimization can be explored.

The individual of the MapReduce application possesses two predominant kinds of operations. The first is Map operation while second one signifies for Reduce operation where the Map as well as Reduce operation takes place in individual application. As the Mapping process is independent of the others so it can be executed in parallel with multiple virtual machines. In the same way the process of Reduce can be exhibited while performing reduction phase. The keys being used for Map process is shared in the Reduction step also. The key based process makes this system highly robust in case of fault-tolerability. These all characteristics make the MapReduce system to operate efficiently as compared to commodity servers. In Map phase the system introduces user-oriented logics to the input data while in reduce phase the intermediate results are processed and it results as the final reduced data in the form of key/value pair presentation. In this overall system the execution of Map process is further categorized into two phase. In first phase achieves the task and then organize the tasks into records which are further processed for further Map process. And then as discussed earlier the Reduce process comes into existence that Reduce the task with the help of key pairs. The optimum space for QoS optimization in cloud environment and in Big Data environment, the Load Distribution could be a potential factor to be explored for enhancements.

Load balancing can be considered as one of the predominant factor allied with cloud computation. Here the load factor could be anything like memory, CPU capacity; delay factors etc. so in order to have an optimum QoS it is always needed to share or distribute the load across numerous nodes so as to enhance the overall resource utilization and performance enhancement. This might ensure the prevention of situation where certain nodes could be overloaded and some would remain idle. The overall objective of the Load balancing approach could be to:

- Optimize the overall system performance,
- Maintaining stability of the system,
- Developing a fault tolerant system model,
- Accommodating future modifications to ensure higher performance,

Considering the above mentioned brief of MapReduce mechanism in Hadoop and the requirement of Load balancing in Hadoop while exhibiting MapReduce process, it can be found that there could be a huge possibility for further system development and optimization in terms of efficient load distribution across the comprising nodes and it can be a significant step to optimize the overall system performance. The scheduling approach might be a significant factor for ensuring the higher quality of service and here in this research domain it would be a potential segment to be explored. Developing a robust scheduling scheme for optimizing overall MapReduce operation while taking into account of dominant factors like capacity, delay/time, resource allocation, even load distributions among all comprising clusters or nodes, would be a noble step to optimize overall Hadoop framework for Big Data applications.

Thus getting motivated form this possibility and scope for further enhancement here in this research proposal the author has proposed a system model called, SBLsM: QoS oriented Scheduling Based Load Distribution Scheme for MapReduce applications which is in fact a load distribution based system model for MapReduce in Hadoop system functional on Microsoft Azure platform where the Hadoop would be installed in HD INSIGHT. Here in this proposed system, the Virtual Machines (VMs) would be generated using CENT OS running on VMs. In this research work, the consideration of HD Insight has been emphasized because it encompasses numerous grids and for accessing these grids in real time applications it is required to have Hadoop system that is ultimately a cluster of Virtual Machines in cloud environment. Thus, the research proposal would be emphasizing its root point on the optimization of MapReduce scheme while taking into account of efficient scheduling process to ensure optimum load distribution across the nodes in the cloud network.

IV. Related Research Work

In order to define an issue and research scope in cloud network and specially Hadoop Model for Big Data applications, here we have conducted a review process where all the related and existing systems were discussed and analyzed for their strength and weaknesses. And thus considering those all factors, we have defined certain objectives in this research work and scope for further development.

In conventional approach, the MapReduce framework considers a tightly tied homogeneous cluster employed with an applications of data-intensive kinds. Few other work such as [19] illustrated that in case of computational heterogeneous environment, the systems developed in Hadoop for recognizing the straggler jobs gets break down and thus the developed LATE scheduling scheme facilitates better measure for performing identification, prioritizing and scheduling.

The reviews conducted are as follows:

TABLE I. Related Work on Optimization in Big Data

Author	*Approach used*	*Application*
Radu Tudoran et al [1]	Leveraging data locality	Optimum selection of transfer protocol and then exposing data layout across the virtual machines
Cairong Yan et. al [2]	Virtual machine (VM) placement	Global resource optimization.
Manjula L et al [3]	Software load balancer scheme and rebalancing using Bulk Synchronous Parallel (BSP)	Cloud file transfer system possessing less file movement costs and algorithmic overheads for data clustering.
Kurazumi, S. et al [4]	I/O intensive jobs of Hadoop	Reducing I/O wait while performing certain work execution.
Xiao Yu et al [6]	Bi-Hadoop scheme to Optimum extension of Hadoop	Introducing minimization of overload caused due to data transfer
Xiaoyi Lu et al [7]	Performance of generic Hadoop RPC architecture	Employs message size locality for avoiding manifold memory allocations
Rasooli, A. et al [9]	Hadoop schedulers schemes and Fair sharing approaches	Hybrid solution that chooses proper scheduling schemes for heterogeneous kind of Hadoop systems
Chen He et al [11]	Optimization using Open Science Grid approach	Enhance faults allied with Hadoop's tolerability for huge data
Mingli Wu et al [12]	Resource monitoring in Hadoop using Ganglia and Nagios	Developed a resource monitoring approach of Hadoop, that can easily facilitate real-time monitoring system
Kewen Wang et al [13]	Guided configuration optimizer scheme	This system employs the experience factors of Hadoop with MapReduce scheme for process optimization
Kejiang Ye et al [14]	a virtual cluster platform called vHadoop	This enhanced system was based on the large-scale MapReduce-oriented parallel data processing technique.
Sadasivam, G.S. et al [15]	HPSO-GA a parallel programming scheme	This approach facilitating higher load balancing facility and thus optimum resource utilization
Jinshuang Yan et al [16]	System for enhancing the execution time period of MapReduce process	The authors developed 3 models 1. Minimize the time factor in process initialization that was exhibited by enhancing its setup and cleanup tasks. 2. Replacement of pull-model task handing over scheme with a new model called push-model. 3. Replacement model for instant message communication at the place of heartbeat-based communication scheme.
Shafer, J. et al [18]	Reasons for bottleneck	evaluate the differences or tradeoffs between the replacement efficiency, its portability and the performance factor in HDFS

Here in our proposed systems also, we have considered the nodes in its heterogeneous state. Some other work [20] illustrated that regardless of optimization in stragglers, the overall performance of MapReduce frameworks remains as poor only as considered even with load balancing approach employed for MapReduce that could cause extreme and burst scenario in communication. The predictive load balancing approach was employed to eliminate such problems. In certain specific consideration of our proposed work, here we do not emphasize on solving the problem of reliability, but here we are much concerned with the issue of overall cloud performance while taking into account of resource allocations, capacity optimization, latency optimization, etc. in some other work [21] a MapReduce framework was proposed in a global wide-area volunteer surroundings. Then while this approach was used with the BOINC framework where all the comprising data were hold by the unitary central scheduler. In this research proposal, we are not considering such restrictions. Luo et al. [22] in their work developed a multi-cluster MapReduce scheme while emphasizing on compute intensive tasks which could as for resources in multiple clusters for higher computational power. A scheme called pipelining MapReduce was proposed in [23] with introducing certain modification in Hadoop workflow so as to get optimized responsiveness and overall performance. Their proposed system enables the shuffling of intermediary data devoid of storing it to the disk. On contrary in this research proposal the scheme being proposed might be employed for deciding in which data link there should be flow and their approach might be employed for optimizing the data transfer. In [24] a system called purlieus system was proposed that considers MapReduce framework in a single cloud. Even this system was unique in that case that this approach emphasizes on locality in the shuffling phase and further focuses on the pairing between the localization of tasks with VMs and the data or resource. Then while, this system couldn't facilitate an end-to-end system optimization for data flow in MapReduce framework.

This is matter of fact that the previous researches have made better effort to optimize the overall performance of Hadoop cloud model, but in fact most of these approaches have been emphasized with the goal of MapReduce optimization with certain definite performance factors and an overall optimization based system could not be explored. Numerous researches have implemented their system on generic and low level cloud platform that cannot justify the robustness of their developed system with real time Big Data processing scenario. Even very few systems have been proposed for scheduling process of MapReduce. Therefore it is required to develop a highly robust kind of system model which can facilitate uniform load distribution and MapReduce process efficiently with optimized performance parameters. This research work has been emphasized on developing such system model to ensure QoS performance of Hadoop based Big Data applications.

V. Proposed Work

Here in this paper, we propose SBLsM: QoS oriented Scheduling Based Load Distribution Scheme for MapReduce application for optimizing overall performance of Hadoop cloud model while taking into account of performance optimization for overall latency, capacity, memory optimization, and computational overheads. Here in this research proposal, we have proposed system implementation of optimized and scheduled MapReduce framework with Microsoft Azure cloud platform integrated with Hadoop model. Here, we propose the implementation of HD Insight technology for creating Virtual Machines (VMs) using CENT OS, running on individual machines. With the consideration of HD Insight technique here we propose the cloud system simulation with numerous grids in cloud network and for accessing each grid we integrate Hadoop model which itself is the clusters of VMs. In this proposed research work we intend to consider Hadoop model with its components as the general Hadoop Distributed File System (HDFS) integrated with our optimized and scheduled MapReduce framework. Here in the proposed system model, we have emphasized our research towards Quality of Service (QoS) oriented Hadoop optimization for Big Data applications in real time scenario. Here, our proposed system aims to bring optimum enhancements in the performance parameters of Hadoop based cloud computing environment with enriched latency, delay, overheads, bottlenecks and optimum resource optimization and thus coming up with the optimum QoS enriched cloud system.

VI. Conclusions

This paper is a survey on the research in the area of optimizing big data processing with Hadoop. We have proposed a QoS performance oriented highly robust scheduling scheme for MapReduce framework of Hadoop cloud model. As per the review conducted and retrieved analysis data, our proposed system represents itself as the optimum approach for accomplishing best QoS for Big Data applications.

References

[1] Radu Tudoran; Alexandru Costan; Ramin Rezai Rad; Goetz Brasche; "Adaptive File Management for Scientific Work flows on the Azure Cloud"; IEEE Big Data in 2013.

[2] Cairong Yan; Ming Zhu; Xin Yang; Ze Yu; Min Li; Youqun Shi; Xiaolin Li; "Affinity-aware Virtual Cluster Optimization for MapReduce Applications"; in 2012.

[3] Manjula L; Sreedevi M; "Automated Cloud Based File Storage Nodes Balancer"; International Journal of Advanced Research in Computer Science and Software Engineering; Volume 3, Issue 9, September 2013.

[4] Kurazumi, S.; Tsumura, T.; Saito, S.; Matsuo, H., "Dynamic Processing Slots Scheduling for I/O Intensive Jobs of Hadoop MapReduce," *Networking and Computing (ICNC), 2012 Third International Conference;* pp.288-292, on 5-7 Dec. 2012.

[5] Sathyanarayana, M.V. ; Kavya, N.P. ; Naveen, N.C., "Intelligent mininig for capturing processes through event logs to represent workflows using FP tree", *International Conference on , Intelligent and Advanced Systems, 2007. ICIAS 2007*

[6] Xiao Yu; Bo Hong, "Bi-Hadoop: Extending Hadoop to Improve Support for Binary-Input Applications," *Cluster, Cloud and Grid Computing (CCGrid), 2013 13th IEEE/ACM International Symposium;* pp.245-252; on 13-16 May 2013

[7] Xiaoyi Lu; Islam, N.S.; Wasi-Ur-Rahman, M.; Jose, J.; Subramoni, H.; Hao Wang; Panda, D.K., "High-Performance Design of Hadoop RPC with RDMA over InfiniBand," *Parallel Processing (ICPP), 2013 42nd International Conference;* pp.641-650 on 1-4 Oct. 2013.

[8] Reeja, S.R.; Kavya, N.P.; "Real time video denoising," *Engineering Education: Innovative Practices and Future Trends (AICERA), 2012 IEEE International Conference on Digital Object Identifier*: Publication Year: 2012 , PP: 1 – 5, Oct 2012

[9] Rasooli, A.; Down, D.G., "A Hybrid Scheduling Approach for Scalable Heterogeneous Hadoop Systems," *High Performance Computing, Networking, Storage and Analysis (SCC), 2012 SC Companion;* pp.1284-1291 on 10-16 Nov. 2012.

[10] Kala Karun, A.; Chitharanjan, K.; "A review on Hadoop-HDFS infrastructure extensions," *Information & Communication Technologies (ICT), 2013 IEEE Conference;* pp.132-137 on 11-12 April 2013.

[11] Chen He; Weitzel, D.; Swanson, D.; Ying Lu, "HOG: Distributed Hadoop MapReduce on the Grid," *High Performance Computing, Networking, Storage and Analysis (SCC), 2012 SC Companion;* pp.1276-1283 on 10-16 Nov. 2012.

[12] Mingli Wu; Zhongmei Zhang; Yebai Li; "Application research of Hadoop resource monitoring system based on Ganglia and Nagios"; *Software Engineering and Service Science (ICSESS), 2013 4th IEEE International Conference;* pp.684-688 on 23-25 May 2013.

[13] Kewen Wang; Xuelian Lin; Wenzhong Tang, "Predator-An experience guided configuration optimizer for Hadoop MapReduce," *Cloud Computing Technology and Science (CloudCom), 2012 IEEE 4th International Conference;* pp.419-426 on 3-6 Dec. 2012.

[14] Kejiang Ye; Xiaohong Jiang; Yanzhang He; Xiang Li; Haiming Yan; Peng Huang, "vHadoop: A Scalable Hadoop Virtual Cluster Platform for MapReduce-Based Parallel Machine Learning with Performance Consideration, "*Cluster Computing Workshops (CLUSTER WORKSHOPS), 2012 IEEE International Conference;* pp.152-160 on 24-28 Sept. 2012.

[15] Sadasivam, G.S.; Selvaraj, D., "A novel parallel hybrid PSO-GA using MapReduce to schedule jobs in Hadoop data grids," *Nature and Biologically Inspired Computing (NaBIC), 2010 Second World Congress;* no., pp.377-382 on 15-17 Dec. 2010.

[16] Jinshuang Yan; Xiaoliang Yang; Rong Gu; Chunfeng Yuan; Yihua Huang, "Performance Optimization for Short MapReduce Job Execution in Hadoop," *Cloud and Green Computing (CGC), 2012 Second International Conference,* pp.688-694, 1-3 Nov. 2012.

[17] Raj, A.; Kaur, K.; Dutta, U.; Sandeep, V.V.; Rao, S.; "Enhancement of Hadoop Clusters with Virtualization Using the Capacity Scheduler"; *Services in Emerging Markets (ICSEM), 2012 Third International Conference;* pp.50-57, 12-15 Dec. 2012.

[18] Shafer, J.; Rixner, S.; Cox, A.L., "The Hadoop distributed file system: Balancing portability and performance," *Performance Analysis of Systems & Software (ISPASS), 2010 IEEE International Symposium;* pp.122-133 on 28-30 March 2010.

[19] M. Zaharia, A. Konwinski, A. D. Joseph, R. H. Katz, and I. Stoica, "Improving MapReduce performance in heterogeneous environments," in Proceedings of OSDI, 2008, pp. 29–42.

[20] F. Ahmad, S. Chakradhar, A. Raghunathan, and T. N. Vijaykumar, "Tarazu: Optimizing MapReduce on heterogeneous clusters," in Proceedings of ASPLOS, 2012, pp. 61–74.

[21] F. Costa, L. Silva, and M. Dahlin, "Volunteer cloud computing: MapReduce over the internet," in Proceedings of IEEE IPDPSW, 2011, pp. 1855–1862.

[22] Y. Luo, Z. Guo, Y. Sun, B. Plale, J. Qiu, and W. W. Li, "A hierarchical framework for cross-domain MapReduce execution," in Proceedings of ECMLS, 2011, pp. 15–22.

[23] T. Condie, N. Conway, P. Alvaro, J. M. Hellerstein, K. Elmeleegy, and R. Sears, "MapReduce online," in Proceedings of NSDI, 2010, pp. 313–327.

[24] B. Palanisamy, A. Singh, L. Liu, and B. Jain, "Purlieus: locality-aware resource allocation for MapReduce in a cloud," in Proceedings of ACM SC, 2011, pp. 58:1–58:11