ISSN (ONLINE): 2454-9762
ISSN (PRINT): 2454-9762
Available online at www.ijarmate.com

IJARMATE
International Journal of Advanced Research in Management, Architecture, Technology and Engineering
(IJARMATE) Vol. 3, Special Issue 2, April 2017

Map Reduce For Traffic Aware Partitioning and Scheduling Using Big Data Application

V.Subhasri,R.Ramya,A.Banu Priya
Department of Computer Science and Engineering,
Indra Ganesan College Of Engineering
Trichy,Tamilnadu.

P.Suresh Pandi M.E,Assistant Professor,
Department of Computer science and Engineering,
Indra Ganesan College Of Engineering,
Trichy,Tamilnadu.

Abstract— **It is energy and cost-efficient with a short time period of scheduling using an VM Map Reduce cluster by renting multiple virtual Machine (VM) from a VM provider. To reduce the system complexity of overall network, we linearize the MINLP by changing some operational parameters of MINLP.MILP is a Joint Optimization of Task Assignment, Data Loading and Data Movement. Mixed Integer Non Linear Programming is the Non Joint Optimization problem. The important factor for minimizing cost is Task Assignment, Data Loading and Data Movement. Optimization of these three factors will reduce the overall cost of network. The data locality for both map tasks and reduces tasks, is improved to avoid job waiting, and improve job execution performance. Two variations of MINLP and MILP are further introduced to separately achieve a better map-data locality and a faster task assignment. Hadoop supports to evaluate and compare the two variations with scheduling algorithms. It outperform the other tested algorithms in terms of map-data locality, reduce-data locality, and network overhead without incurring significant overhead. The two variations are separately suitable for different MapReduce-work load scenarios and provide the best job performance among all tested algorithms.**

Keywords— VMs, Map Reduce

I. INTRODUCTION

The term 'Big Data' describes innovative techniques and technologies to capture, store, distribute, manage and analyze petabyte-or larger-sized datasets with high-velocity and different structures. Big data can be structured, unstructured or semi-structured, resulting in incapability of conventional data management methods. Data is generated from various different sources and can arrive in the system at various rates. In order to process these large amounts of data in an inexpensive and efficient way, parallelism is used.

Big Data is a data whose scale, diversity, and complexity require new architecture, techniques, algorithms, and analytics to manage it and extract value and hidden knowledge from it. Hadoop is the core platform for structuring Big Data, and solves the problem of making it useful for analytics purposes.

Hadoop is an open source software project that enables the distributed processing of large data sets across clusters of commodity servers. It is designed to scale up from a single server to thousands of machines, with a very high degree of fault tolerance. Big data is a term that refers to data sets or combinations of data sets whose size (volume), complexity (variability), and rate of growth (velocity) make them difficult to be captured, managed, processed or analyzed by conventional technologies and tools, such as relational databases and desktop statistics or visualization packages, within the time necessary to make them useful. The architecture is decomposed into three layers, including Infrastructure Layer, Computing Layer, and Application Layer from top to bottom.

Big data and big data analytics participate an important role in today's fast-paced data-driven businesses. The general characteristic of real-life applications is that they frequently have to agreement with a tremendous amount of data to obtain useful information. Achieving analytics and delivering accurate query results on such large amounts of data can be computationally expensive (long time for processing) and resource intensive. In common, overloaded systems and high delays are unsuited with a high-quality user experience, and the early estimated answers that are accurate sufficient are often of much greater value to users than tardy exact results.

Big data is certainly one of the biggest buzz phrases in IT today. Combined with virtualization and cloud computing, big data is a technological capacity that will force data centers to importantly transform and develop within the next five years. Related to virtualization, big data infrastructure is exclusive and can create an architectural upheaval in the way systems, storage, and software infrastructure are associated and maintained.

The cloud technology is emerging as an infrastructure appropriate for building large and complex systems. Storage and compute resources provisioned from converged infrastructure and distributed resource pools present a cost-effective different to the traditional in-house data center. The cloud offers new levels of scalability, flexibility and availability, and permits easy access to data from several locations and any device. In addition, the cloud representations a data model of objects that contain data integrated among its user-defined and system-defined metadata as a single part. Thus, the cloud is an attractive model for a recent form of scalable predetermined content applications that involve rich metadata.

The Cloud computing model has happen to an established option to contribute on foundations for data storage and computation resources. Together trends, ubiquitous sensing and Cloud computing, balance each other in a normal way. Sensor networks gather information regarding the physical environment, but in general require the resources to store and process the collected data over long periods of time. Cloud computing elastically presents the missing storage and computing resources.

The present technologies namely grid and cloud computing have all proposed to access massive amounts of computing power with aggregating resources and providing a single system view. Between these technologies, cloud computing is attractive a powerful architecture to achieve large-scale and complex computing, and has transformed the way that computing infrastructure is theoretically and utilized. Additionally, a significant goal of these technologies is to distribute computing as a result for tackling big data, such as large scale, multi-media and high dimensional data sets.

Big data and cloud computing are together the rapidly-moving technologies. Cloud computing is connected with new concept for the provision of computing infrastructure and big data processing technique for all types of resources. Additionally, several new cloud-based technologies include to be approved since dealing with big data for concurrent processing is complex. One important quality of cloud computing is in aggregation of resources and data into data centers on the Internet. The present cloud services (IaaS, PaaS and SaaS) take in better execution effectiveness by aggregating application execution environments at different levels involving server, OS and middleware levels for distributing them.

A different approach of aggregating data into clouds has also been initiated, and it is to analyze such data with the efficient computational capability of clouds. As cloud computing services become popular, more and more academic, enterprise, and personal computing applications are deployed in the shared computing environment. Users of cloud services try to minimize the execution time of their submitted jobs without exceeding a given budget under the specified requirements, while cloud providers try to maximize the use of resources and achieve more profits.

Infrastructure as a Service (IaaS) clouds have greatly reduced the investment risk of owning an infrastructure. It offers computers as physical or more often as virtual machines (VMs). A cluster of VMs, virtual cluster, is often requested as a platform for users to run parallel or distributed applications such as MapReduce and Dryad applications. In order to get high throughput, fast response, load balance, low cost, and low price, many topics on VM configuration, VM placement, VM consolidation, and VM migration are explored. The network topology of a virtual cluster has a significant impact on the execution of applications running on it because the physical nodes where VMs are located can be linked in different ways.

For example, some nodes are located in the same rack while others in different racks through a slow link. Another special architecture is the hierarchical network where two physical nodes may lie in different local area networks. Furthermore, the characteristics of different applications have different requirements for the network topology. Some applications create tasks running on different VMs which need to exchange large amount of data frequently, while others create tasks which execute independently or exchange a little data.

MapReduce and MapReduce-like models are widely used to process "Big Data". Applications based on such models place heavily data-dependency or communication on VMs, so network traffic becomes the bottleneck of jobs. The following are three phases of data exchange in the execution process of an application based on MapReduce model. It provisioning a virtual cluster according to the position relationship between VMs so as to decrease the network traffic and improve the performance of MapReduce and MapReduce-like applications rather than modifying the job scheduling strategies or VM configurations. By optimizing the architecture of virtual clusters, cloud users can get a more efficient platform with the same resource request and cost, and cloud providers can obtain a higher resource utilization ratio.

The affinity by defining the distance of a virtual cluster is measured. The online heuristic VM placement algorithm and the global sub-optimization algorithm are compared by simulations. The former has lower time complexity while the latter returns shorter average distance for multiple requests.Two metrics of application runtime and cluster affinity show the efficiency of virtual cluster optimization.

II. PROPOSED METHODOLOGY

The original large-scale problem is decomposed into several sub problems and it is solved in parallel manner using distributed algorithm. The system designed to postpone the migration operation until the cumulative traffic cost exceeds a threshold. The system investigate network traffic reduction within MapReduce jobs by jointly exploiting traffic-aware intermediate data partition and data aggregation among multiple map tasks. It offers computers as physical or more often as virtual machines (VMs).

A cluster of VMs, virtual cluster, is used as a platform to run parallel or distributed applications such as MapReduce and Dryad applications. In order to get high throughput, fast response, load balance, low cost, and low price, many topics on VM configuration, VM placement, VM consolidation, and VM migration are explored. The network topology of a virtual cluster has a significant impact on the execution of applications running on it because the physical nodes where VMs are located can be linked in different ways.

For example, some nodes are located in the same rack while others in different racks

ISSN (ONLINE): 2454-9762
ISSN (PRINT): 2454-9762
Available online at www.ijarmate.com

IJARMATE
International Journal of Advanced Research in Management, Architecture, Technology and Engineering
(IJARMATE) Vol. 3, Special Issue 2, April 2017

through a slow link. Another special architecture is the hierarchical network where two physical nodes may lie in different local area networks. Some applications create tasks running on different VMs which need to exchange large amount of data frequently, while others create tasks which execute independently or exchange a little data. MapReduce and MapReduce-like models are widely used to process "Big Data".

Applications based on such models place heavily data-dependency or communication on VMs, so network traffic becomes the bottleneck of jobs. The following are three phases of data exchange in the execution process of an application based on MapReduce model. The targets is to provisioning a virtual cluster according to the position relationship between VMs so as to decrease the network traffic and improve the performance of MapReduce and MapReduce-like applications rather than modifying the job scheduling strategies or VM configurations.

By optimizing the architecture of virtual clusters, cloud users can get more efficient platform with the same resource request and cost, and cloud providers can obtain a higher resource utilization ratio. The shorter the distance, the closer the virtual cluster. The shortest distance problem by formulating it into an integer linear programming is sloved.

The online heuristic VM placement algorithm and the global sub-optimization algorithm are compared by simulations. The former has lower time complexity while the latter returns shorter average distance for multiple requests.In the experiment, the different virtual cluster architectures to test different MapReduce applications is adopted.

Two metrics of application runtime and cluster affinity show the efficiency of virtual cluster optimization. The distance difference between two central node selection strategies. Heuristic distance is mapped to the virtual cluster with the most appropriate central node built by our online VM placement algorithm. Shortest distance with a random central node is mapped to the same virtual cluster, however, the central node is chosen randomly. It is clear that even if the same virtual cluster, but not the same position of the central node, the distance difference is also great.

For MapReduce applications, the selection of the central node is the same important with the architecture of the virtual cluster. Each physical node has different capability and each request has different requirement. For MapReduce applications, it is very important to match the request with a virtual cluster and appropriate central node so as to make full use of the advantage of data-locality and reduce the network traffic. The MapReduce and Mapreduce-like models are based on master-slaves network topology. When the requests arrive randomly, their service time are also random, and the cloud resources are enough to meet multiple requests at one time, the virtual cluster provisioning is got according to our online heuristic algorithm and global sub-optimization algorithm.

The two scenarios of requests is simulated. One uses the same request configurations as the previous simulations and the other uses a request sequence with a relatively small number of VMs.

III.METHODOLOGIES

III.a) DATA UPLOADING

The master schedules map tasks in the workers by taking into account of data locality. The output of the map tasks is divided into as many partitions as the number of reducers for the job. Entries with the same intermediate key should be assigned to the same partition to guarantee the correctness of the execution. All the intermediate key/value pairs of a given partition are sorted and sent to the worker with

ISSN (ONLINE): 2454-9762
ISSN (PRINT): 2454-9762
Available online at www.ijarmate.com

IJARMATE

International Journal of Advanced Research in Management, Architecture, Technology and Engineering
(IJARMATE) Vol. 3, Special Issue 2, April 2017

the corresponding reduce task to be executed. Default scheduling of reduce tasks does not take any data locality constraint into consideration. As a result, the amount of data that has to be transferred through the network in the shuffle process may be significant.

III.B). SEGMENTATION

The system consider a typical MapReduce job on a large cluster consisting of a set N of machines.Let d(x,y) denote the distance between two machines x and y, which represents the cost of delivering a unit data. When the job is executed, two types of tasks, i.e., map and reduce, are created. The sets of map and reduce tasks are denoted by M and R, respectively, which are already placed on machines. The input data are divided into independent chunks that are processed by map tasks in parallel. The generated intermediate results in forms of key/value pairs may be shuffled and sorted by the framework, and then are fetched by reduce tasks to produce final results.

III.C) TASK ASSIGNMENT

The access tier is made up of cost-effective Ethernet switches connecting rack VMs. The access switches are connected via Ethernet to a set of aggregation switches which in turn are connected to a layer of core switches. An inter-rack link is the most contentious resource as all the VMs hosted on a rack transfer data across the link to the VMs on other racks. Our VMs are distributed in three different racks, and the map-reduce tasks are scheduled.

III.D) DATA LOADING

The number of reducers is changed from 1 to 6. The highest network traffic is achieved when there is only one reduce task under all algorithms. That is because all key/value pairs may be delivered to the only reducer that locates far away, leading to a large amount of network traffic due to the many-to-one communication pattern. As the number of reduce tasks increases, the network traffic decreases because more reduce tasks share the load of intermediate data.

III.E) PROCESSING OF TASK

The system divide the execution of a MapReduce job into several time slots with a length of several minutes or an hour. Them$_j^p$(t) and α_j(t) denote the parameters collected at time slot t with no assumption about their distributions. As the job is running, an existing data partition and aggregation scheme may not be optimal anymore under currentm$_j^p$(t) and sα_j(t). To reduce traffic cost,need to migrate an aggregator from machine j to j0 with a migration cost jj 0. Meanwhile, the key assignment among reducers is adjusted.Let reducer k0 process the data with key p instead of reducer k that is currently in charge of this key, the use of function $\phi_{kk'}(\sum_{\tau=1}^{t} \sum_{j \in A} \sum_{K \in R} {}^{gp}{}_{jk}(\tau))$ to denote the cost migrating all intermediate data received by reducers so far.

III.F) EVALUATION PROCESS

The system evaluate the performance of proposed algorithm under online cases by comparing it with other two schemes: OHRA and OHNA, which are online extension of HRA and HNA, respectively. The default number of mappers is 20 and the number of reducers is 5. The maximum number of aggregators is set to 4 and it also vary it to examine its impact. The key/value pairs with random data size within [1-100] are generated randomly in different slots. The total number of physical machines is set to 10 and the distance between any two machines is randomly choose within.

ISSN (ONLINE): 2454-9762
ISSN (PRINT): 2454-9762
Available online at www.ijarmate.com

IJARMATE
International Journal of Advanced Research in Management, Architecture, Technology and Engineering (IJARMATE) Vol. 3, Special Issue 2, April 2017

IV.EXPERIMENTAL RESULT

It evaluate and compare MILNP and MILN with three scheduling algorithms provided by Hadoop, including the FIFO algorithm (FIFO for short), the fair scheduling algorithm (Fair for short), and the capacity algorithm.It established a VM Map Reduce cluster by hich is a privately owned VM provider. The Hadoop master server and is located at a data center. There maintain N number of VM act as slaves. VMs are located at a data center . Each VM runs Ubuntu10.04 with two CPU cores, 2 GB RAM, and 48 GB SSD storage space. Each VM has a map slot and a reduce slot. It use Hadoop MRv1, which is widely adopted in production settings, as the implementation of MapReduce, and modify the source code of Hadoop-0.20.2 to realize MILNP and MILN.

Reduce-data locality rate, which is defined as the percentage of input data that a reducer can obtain from its local data center. The value ranges from 0 to1. A value of one is desirable.Job turnaround time (JTT for short), which starts when a job is submitted to the cluster and finishes when the job is completed. A short JTT is desirable.VM load, which shows the average number of map tasks executed by each VM and the corresponding standard deviation. With this metric, know the load balance among VMs. VMs are worked shortly and schedule the jobs are very fast manner.

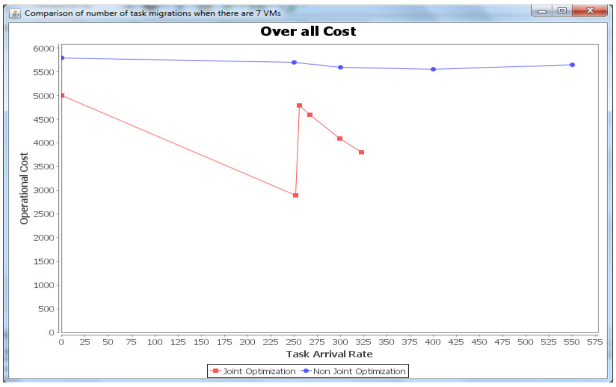

Fig.iv.A.Describes the evaluation of defined algorithm.

VI. CONCLUSION

A high efficient virtual cluster can minimize the network overhead and maximize the system performance. The simulation demonstrates the feasibility of the MILP algorithm and MINLP algorithm. Meanwhile the experiment results show the improvement of virtual cluster optimization for MapReduce applications. The optimization method can be extended to MapReduce-like applications where a master node distributes tasks on several different slave nodes and these slave nodes work collaboratively. In the future, the distance between physical nodes is to be investigated at first. It is measured and configured statically in this paper. How to compute their values when some VMs are down or reconfigured is critical for the VM placement policy. Second, from the experiments, the data-locality and the shuffle-locality are two important factors to affect the execution time in addition to the affinity of the virtual cluster. The integration of more fine-grained virtual cluster provisioning methods and MapReduce scheduling strategies needs to be explored.

VII. FUTURE ENHANCEMENT

Fault tolerance mechanisms either consume significant extra energy to detect and

ISSN (ONLINE): 2454-9762
ISSN (PRINT): 2454-9762
Available online at www.ijarmate.com

IJARMATE
International Journal of Advanced Research in Management, Architecture, Technology and Engineering (IJARMATE) Vol. 3, Special Issue 2, April 2017

recover from the failures .Fault-tolerant describes a computer system or component designed so that, in the event that a component fails, a backup component or procedure can immediately take its place with no loss of service. Fault tolerance is not just a property of individual machines; it may also characterize the rules by which they interact.

REFERENCES

[1] S. Chen and S. W. Schlosser, "Map-reduce meets wider varieties of applications," Intel Research Pittsburgh, Tech. Rep. IRP-TR-08-05,2008.
[2]J. Dean and S. Ghemawat, "MapReduce: Simplified data process-ing on large clusters," Commun. ACM, vol. 51, no. 1, pp. 107–113, 2008.
[3] Z. Guo, G. Fox, and M. Zhou, "Investigation of data locality in mapreduce," in Proc. 12th IEEE/ACM Int. Symp. Cluster, Cloud Grid Comput., May 2012, pp. 419–426.
[4] J. Jin, J. Luo, A. Song, F. Dong, and R. Xiong, "BAR: An efficient data locality driven task scheduling algorithm for cloud computing," in Proc. 11th IEEE/ACM Int. Symp. Cluster, Cloud Grid Comput., May 2011, pp. 295–304.
[5]Ming-Chang Lee, Jia-Chun Lin, and RaminYahyapour, "Hybrid Job-Driven Scheduling for Virtual MapReduce Clusters".VOL. 27, NO. 6, JUNE 2016.
[6]B. White, T. Yeh, J. Lin, and L. Davis, "Web-scale computer vision using mapreduce for multimedia data mining," in Proc. 10th Int. Workshop Multimedia Data Mining, Jul. 2010, pp. 1–10.
[7]Y. Wang, W. Wang, C. Ma, and D. Meng, "Zput: A speedy datauploading approach for the hadoop distributed file system," inCluster Computing (CLUSTER), 2013 IEEE International Conferenceon. IEEE, 2013, pp. 1–5
[8] T. White, Hadoop: the definitive guide: the definitive guide. " O'ReillyMedia, Inc.", 2009.

