Qualitative Study on the efficiency of Load balancing algorithms in Cloud Environment

Ramya S Gowda,

Assistant Professor, Sikkim Manipal University, Bangalore

Abstract: *Load balancing in cloud is different from the typical architecture of load balancing techniques. This opens up new opportunities and challenges. Resource management is the use of available processors in the most efficient way possible. A CPU scheduling algorithm is said to efficient when it keeps the CPU and disks busy. The Scheduling algorithm is of greater interest if the technique is dynamic.The dynamic technique is of larger interest because of its dynamic adaptability to the current state of the system. In dynamic technique, weights are designated to servers and by searching a lightest server in the whole network, the load is balanced.*
Key words: *Cloud, Cloud environment, Load Balancing, Qualitative, distributed computing, efficiency*

I. Introduction

Load Balancing is a method to distribute workload across one or more servers, network interfaces, hard drives, or other computing resources [14]. Cloud computing is increasingly treated as a flexible alternative to in-house computing resources [2] [14][12]. In this technology the load on the computing server (CPU Load) depends on the requirement of the customer [15].

In general, resource management is the use of available processors in the most efficient way possible. A CPU scheduling algorithm is said to efficient when it keeps the CPU and disks busy [23].

A Cloud consists of an array of computing resources (CPU and Memory) provided on demand by a third party over an internetwork. A cloud is a type of parallel and distributed system consisting of a collection of interconnected and virtualized computers that are dynamically provisioned and presented as one or more unified computing resources.

Many methods to address load balancing in a dynamic way have been proposed. Few of the popular ones include - Particle Swarm Optimization, hash method and genetic algorithms. One such dynamic technique which is in wide use in cloud environment is the Ant colony optimization [14]. This technique has been codenamed based on ant behaviour [14].

Another technique which is in use in cloud environment is the semi-distributed technique. In this clusters are formed by a group of the nodes of the system and the load balancing carried out in each cluster in a centralized way. This technique uses Virtualization concept too. Clustering algorithm of Physical Machines and Virtual Machines can be enhanced with their performance by applying central node algorithm for the cluster [2].

In any scheduling mechanism the rescheduling algorithm is expected to handle machine failure, and network anomaly. However, as the scheduler calls rescheduling frequently, the rescheduling algorithm should be low-complexity [20].

Load balancing is an important problem in many large scale distributed systems to achieve both high performance and to meet service level objectives for throughput and latency [4]. Small, fast front end cache can ensure effective load balancing regardless of the query distribution. In contrast to this Markatos examined the relationship between cache size and performance gain when caching search engine results [8]. This work focuses on improving performance by reducing redundant work. In support of the work from BinFan, server based caching achieves better load balancing by replicating a small amount of data from highly loaded servers to proxies that are close to data requesters [11]. Server-based caching assumes an exponential popularity model for analysis [4]. This is different from Bin, Lim, Anderson and Kaminsky in that they uses adversarial access patterns that incur a worst case load distribution, which is important to define and meet service level agreements.

Need of Load Balancing

In a single processor system only one process can run at a time; any others must wait until the CPU is free and can be rescheduled [19].
We need load balancing [14]
- To improve the performance of the cloud
- To have a backup plan in case of system fails even partially
- To maintain the system stability
- To accommodate future modification in the systems

Without load balancing, users could experience delays, timeouts and possible long system responses [22].

Types of Algorithms
The classification of load balancing algorithm depending upon [14]
• Who initiated the process
• Current status of the system
The load balancing Algorithms can be classified into three categories depending upon who initiated the process. They are
• Sender initiated: if the load balancing algorithm is initialized by the sender.
• Receiver initiated: if the load balancing algorithm is initiated by the receiver.
• Symmetric: it is the combination of both sender initiated and receiver initiated.
Depending upon the current state of the system, it can be classified as
• Static: it does not depend on the current state of the system but prior knowledge of the system is needed.
• Dynamic: it is better than static approach because it is depended on current state of the system. No prior knowledge about the system is needed.

In comparison to both the algorithms, dynamic approach is normally used as effective load balancing technique though round robin algorithms are based on simple rule, more loads conceived on servers and creates imbalanced traffic is discovered [17]. One more Benefit of dynamic load balancing is that if any node is fail it will not halt the whole network but it will affect the system performance [2].
According to [20] a load-balancing algorithm that follows the approach of the dynamic task allocation by balancing the load on cluster node and then reduces the cost of performance, increases the efficiency of the system.

II.   Literature Survey
CPU scheduling algorithms [23]: this algorithm has two parts
• Dispatcher provides the basic mechanism of running the processes.
• Scheduler is a operating system code that decides how long the processors will run and the priorities of the processors.

Ant Colony Optimization Technique
This algorithm has been codenamed based on ant behaviour [14].
• According to Ratan Mishra, ant colony optimization algorithm is based on dynamic load balancing algorithm [14]. This paper discusses about load distribution.
• Method: this algorithm is based on the behavior of unsophisticated insects (ants).
• Has limited memory
• Exhibits individual behavior
• They are based on self-organization
• This algorithm is based on shortest path method
• Shortcoming: This approach is only appropriate for routing in symmetric network. As for asymmetric network the cost from to and from may be different.

Guide to Dynamic Load Balancing in cloud
According to [3] Ali. Alakeel, in his "A guide to dynamic load balancing in distributed computer systems", presents and analyses many issues that need to be considered in the development or study of a dynamic load balancing algorithm.
The issues are load estimation, load levels comparison, performance indices, system stability, amount of information   exchanged among nodes, job resource requirements estimation, job's selection for transfer, remote nodes selection.

Semi Distributed Algorithm
In this algorithm clusters are formed by a group of nodes of the system and the load balancing carried out in each cluster in a centralised way. This technique uses virtualisation concept too.
• According to AjithSingh N and M. Hemalatha, in their " An approach on Semi-Distributed load balancing algorithm for cloud computing system, this paper proposes to design a better load balancing which can be applied in every central node of the cluster [2].
• Shortcoming: it is done only for center node of the cluster.

Greedy approach and Un-coordinated algorithm

According to [9], Goncalves, Endo, Palhares, Santos, Kelner and Sadok, on the load balancing of virtual networks in distributed clouds, the paper talks about the allocation of computing and network resources in the Distributed cloud (D-Cloud) with the objective of load balancing in virtualized infrastructure. The constraints considered are: processing power, storage, memory and network delay.

This algorithm uses Greedy approach.

Shortcoming: This algorithm achieves only approximated solutions for each time in a specific time.

The strategy works better when the nodes of the topology are fairly connected

Multi- Resource load balancing algorithms

According to [21], Yu, Ivan, Ricardo, Marta and Dianfu, the paper aims at an algorithm which balances both CPU and memory resources among cache instances by re-distributing stored data.

Performance measures: reconfiguration cost, minimize load imbalances

Shortcoming: This algorithm works efficiently with homogeneous cache nodes with equal capacities of CPU and Memory

III. Methodology

More than 40 papers were referred in this area out of which some 30 papers are relevant. For this paper, 10 papers were chosen which were very closely related based on which some research gaps were noted. Upon finding the research gap, it was noted that there were many limitations and which can be overcome by certain methodology which is presented below.

Also there are various challenges faces by cloud. They are long latency, horizontal scaling, do not know when a queue is created for the firt time, duplicate message, potential node failures, in determininistic eventual consistency windows [10].

IV. Research gap

From the above papers we got many research gaps which are listed below. In the present clustering algorithms, it is difficult to measure the result as the implementation is costlier. Load balancing has no design for clustering algorithms.

- The limitation of the Ant Colony technique is that it is for individual server in the cloud. Therefore the efficiency is comparatively less. This technique does not consider the fault tolerance issues. This technique is efficient for symmetric Network [14].
- In the present clustering algorithms, it is difficult to measure the result as the implementation is costlier. Load balancing algorithms has no design for clustering algorithm [2].
- In a real world platform, the network state and the cluster workload change frequently, so it is necessary to update the scheduling strategy by an efficient rescheduling algorithm [2].
- The un-coordinated algorithm works efficiently when the nodes of same topology are connected. Considerations of other topologies and an evaluation of the original algorithm can be done [9].
- In the multi- resource algorithm, the algorithm works efficiently with homogeneous cache nodes with equal capacities of CPU and Memory. So it is necessary to develop an algorithm that works with heterogeneous nodes [21].

Any one or the combination of one or more gaps can be converted into research topics or as proposed work.

- To improve the resource utilisation in a cloud system using better load balancing techniques: In a real world platform, the network state and the cluster workload change frequently, so it is necessary to update the scheduling strategy by an efficient rescheduling algorithm.

- **Areas of focus**
- Improvise the existing load balancing techniques by combining and testing the performance and efficiency.
- Adding Security Features to the existing Load balancing mechanisms

Proposed methodologyto carry out the research

Simulation technique (network simulator (ns-2))

- Ns-2 is a discrete event simulator targeted at networking research.
- Ns-2 provides substantial support for simulation of TCP, routing, and multicast protocols over wired and wireless (local and satellite) networks

V.   Conclusion

Combination of two different scheduling technique we can maintain high efficiency of load balancing on cloud environment. A high efficiency will allow the utilization of available resources in the most possible way and maintain the stability in the cloud. Efficiency is a definable term and importantly need to note the meaning as the parameters to observe efficiency differs based on the definition. Already there are around seventeen algorithms which were having their own advantages and disadvantages based on the eight parameters. If we map all the benefits and shortcomings then one can realize the weaker area and which algorithm is stronger in which parameter. That will be another study by itself.

References

[1].   Bestavros (1997): WWW traffic reduction and load balancing through server-based caching. Concurrency, IEEE, 5(1):56-57.
[2].   Ajith Singh N and M. Hemalatha, An Approach on Semi-Distributed Load Balancing Algorithm for Cloud Computing System, International Journal of Computer Applications (0975 – 8887), Volume 56– No.12, 2012.
[3].   Ali M. Alakeel, A guide to dynamic load balancing in distributed computer system, IJCSNS International Journal of Computer Science and Network Security, Vol 10 No.6 2010.
[4].   Bin Fan, Hyeontaek Lim, David G. Andersen, Michael Kaminsky: "Small cache, Big effect: provable Load Balancing for randomly partitioned cluster services
[5].   Charles Border, CloudComputing in the curriculum: Fundamental and Enabling Technologies,
[6].   David Excalante and Andrew J. Korty, Cloud services: policy – Assessment, EDUCAUSE Review. Vol 46, no. 4, 2011.
[7].   DervisKaraboga, An idea based on honey bee swarm for numerical optimizatIon, Technical report-tr06, october, 2005
[8].   E.P Markatos (2001), On caching search engine query results. Computer communications, 24(2):137-143, ISSN 0140-3664
[9].   Glauco Estacio Goncalves, Patricia Takako Endo, Andre Almeida Palhares, Marcelo Anderson Santos, Judith Kelner, DjamelSadok, On the Load Balancing of Virtual Networks in Distributed Clouds, 2013
[10].   Huan Liu, Dan Orban (2011): Cloud MapReduce: a MapReduce Implementation on top of a Cloud Operating System. Proceeding of 11th IEEE/ACM international symposium on cluster, cloud and grid computing, 2011
[11].   J.Dean (2009), Challenges in building large-scale information retrieval systems: invited talk. In proceeding of the second ACM international conference on websearch and datami9ning, WSDM '09, ACM.
[12].   Martin Randles, David Lamb, A. Taleb-Bendiab, "A Comparative Study into Distributed Load Balancing Algorithms for Cloud Computing," IEEE 24th International Conference on Advanced Information Networking and Applications Workshops, pp551-556, 2010
[13].   NarimanMirzaei, Cloud Computing, Fall 2008
[14].   Ratan Mishra, AnantJaiswal (2012): Ant colony OptimizationL A Solution of load balancing in cloud", international journal of Web and Semantic Technology (IJWesT) Vol.3, No.2, 2012.
[15].   R. Buyya. C.S Yeo and Venugopal (2008): Market oriented cloud computing: vision, hype and reality for delivering IT services as computing utilities". Proceeding of the 10th IEEE international Conference on the high performance computing and communications, 2008
[16].   R. Jeyarani, R. Vasanth Ram, N. Nagaven (2010): Design and Implementation of an efficient Two-level scheduler for cloud computing environment. In the proceeding of the 10th IEEE/ACM International Conference on Cluster, Cloud and Grid Computing, 2010.
[17].   R.X. T and X.F.Z., A load balancing Strategy based on the combination of static and dynamic, in database technology and applications (DBTA), 2010 2nd international workshop, pp.1-4, 2010.
[18].   R. Shimonski (2003), Windows 2000 & Windows Server, 2003 Clustering and Load Balancing. Emeryville. McGraw-Hill Professional publishing, CA, USA (2003), p2.
[19].   Silberschatz, Gaven and gagne, Operating systems,Wiley publishing house, IISN No: 978-81-265-0962, Chaprer 5, 2009
[20].   Vaishali W. Thawari, Sachin D. Babar, and Nitin A. Dhawas, An Efficient Data Locality Driven Task Scheduling Algorithm for Cloud Computing, SHIV SHAKTI International Journal in Multidisciplinary and Academic Research (SSIJMAR), vol. 1, No. 3, September-October (ISSN 2278 – 5973), 2012.
[21].   Yu Jia, Ivan Brondino, Ricardo Jimenez Peris, Marta Patino Martinez, Dianfu Ma, A Multi-Resource Load Balancing Algorithm for Cloud Cache Systems
[22].   ZenonChaczko, VenkateshMahadevan, ShahrzadAslanzadeh and Christopher Mcdermid (2011): Availability and Load Balancing in Cloud Computing, proceedings of international conference on Computer and Software Modeling IPCSIT vol 14(2011), IACSIT press, Singapore.
[23].   http://pages.cs.wisc.edu/~bart/537/lecturenotes/s11.html (2013): lecture notes of the course CS 537 Notes, Section #11: Scheduling and CPU Scheduling retreived on 26/02/2013