International Journal of Computer Science and Engineering (IJCSE)
ISSN(P): 2278-9960; ISSN(E): 2278-9979
Vol. 6, Issue 1, Dec - Jan 2017; 79-88

International Academy of Science, Engineering and Technology
Connecting Researchers; Nurturing Innovations

LOAD BALANCING TECHNIQUES: ESSENTIALS, ISSUES AND MAJOR CHALLENGES IN CLOUD ENVIRONMENT - A METICULOUS REVIEW

HIRAL M. PATEL & RUPAL R. CHAUDHARI

Assistant Professor, Sankalchand Patel College of Engineering, Snakalchand Patel University, Visnagar, Gujarat, India

ABSTRACT

A revelation of cloud computing carries immense opportunities to entertain virtual resources at moderate cost without be obliged to possessing any kind of infrastructure. Cloud data centres consisting of heterogeneous servers treating multiple virtual machines having various specifications and variations of resource usages, which may provoke imbalanced resource utilization within servers that may be result into performance mortification. Load balancing is a methodology to distribute workload across multiple computers to gain maximum profits by optimal resource utilization. This paper presents a review of a few load balancing techniques in cloud computing. By investigation of such techniques with their innumerable repays, limitations and issues a new and competent practice for Load Balancing is instigated in future.

KEYWORDS: Fault Tolerant, Job Migration, Load Balancer, Virtual Machine (VM)

INTRODUCTION

Hasty growth of technology escort to the up gradation in processor, network, storage and computing resources. They turn out to be further influential and ubiquitous. This technological development make sure a new computing paradigm entitled as cloud computing. On demand service, rapid elasticity, scalability, and metered service are major characteristics of clouds computing. Managing high degree of data and other resources require several methodologies to optimize functions and provide improved quality of facility. It is essential to improve storage utilization and response time for users. One important concern associated with this field is dynamic load balancing or task scheduling. Load balancing is a move towards reassign the loads from overloaded nodes to underutilize nodes. It is generally dynamic in scenery because of traffic flood and need of server node is depending over the user request. Cloud data centres are broadening over different geographical regions. Users can subscribe cloud service from any location. Random generation of task create load imbalance in cloud platform that means some of the data centre as heavily loaded while others are in idle or underloaded.

Estimation of load, comparison of load, Information exchange, stability of different system, performance of system, interaction among the nodes, and temperament of job to be relocated ,selection of nodes, load balancing operation are the major factors of devising an effective load balancing algorithm.. The Numbers of load balancing techniques are available can be compared or characterized on following parameters:

Throughput – Number of finished user requests in finite time period.

Completion Time-The Maximum time unit required to complete a job

Communication Cost-The overall cost of transmissions and receiving of the data bits.

Resource Utilization-utilize a resource in such a way that it never get free.

Response Time-Maximum time required to execute a user request.

Scalability- future scope to extend the network resource.

Fault Tolerance - capability to complete unbroken load harmonizing in spite of unaware node or link damage.

Migration Time – time to ride the jobs from one protuberance to other.

ESSENTIALS OF LOAD BALANCING

As we know, Cloud Environment is a distributed Environment in which number of nodes exists. The jobs that arrive in the system are typically not homogeneously distributed. The Ventures of Load balancing are accomplishing user satisfaction in addition to resource utilization, making sure that no single node is weighed down, which will perk up the overall performance of the system. If load balancing used in a appropriate manner then it can attain most favourable resource utilization which will minimize the resource consumption. one more important advantage of using load balancing are put into practice fail-over, enabling scalability, avoiding bottleneck, reducing response time, accommodate future modification in the system , accommodate future modification in the system and achieving Green Computing in clouds.

Load distribution is described in a variety of widely differing techniques and methodologies. These are broadly classified as[1]:

- **Task Assignment Approach** → Each process submitted by a user for processing is observed as a collection of related tasks and these tasks are scheduled to appropriate nodes hence improve performance.
- **Load Sharing** → Load can just be placed on inactive hosts.
- **Load Levelling** → Instead of demanding to get a rigorously equal distribution of load across all the nodes or merely making use of the idle nodes, load levelling tries to avoid congestion on any one host.
- **Load Balancing** → tries to guarantee that the workload on each host is within some small range of the workload present on all the other nodes in the system

When we go from load sharing to load levelling to load balancing we are in fact moving from a crude distribution to a keen distribution of load. load balancing is the premium type of load distribution[1].

LOAD BALANCING APPROACHES

Various algorithms, strategies and policies have been proposed, implemented and classified.

LOAD BALANCING ALGORITHMS

The Algorithms for Load Balancing Can Be Classified Into Two Categories Based on Environment: Static or Dynamic

Static Load Balancing Algorithm

Static load balancing algorithms assigns task of a parallel program to workstations based on either load at the time nodes are assigned task or based on an average load of workstation cluster. The Load balancing decisions are prepared at compile time when resource needs are calculated. An advantage of algorithm is the simple implementation and less overhead. Since there is no need to constantly keep an eye on the workstations for performance information. On the other

hand, static algorithms only work on form when there is not much variation in the load on the workstation. Static load balancing still have a number of faults: it is very difficult to estimate in an accurate way the execution time of different parts of a program. Occasionally there are communication interruptions that vary in an out of control way for some problems the number of steps to reach a solution is not known in advance [1].

Dynamic Load Balancing Algorithm

Dynamic load balancing algorithms make alterations to the distribution of work among nodes at runtime. They use present load information during making decision of load distribution. Dynamic environment is complicated to be simulated but is highly adjustable with cloud computing environment [1].

The Algorithms for Load Balancing can be Classified into Three Categories Based on Location: Centralized, Distributed and Hierarchical

Centralized Load Balancing

In centralized load balancing technique allocation and scheduling related decisions are ended by a single node which is totally is responsible for storing knowledge base of entire cloud network. In this scenario Master-Slave relation exists. This technique reduces the time requisite to examine different cloud resources but generate a immense overhead on the centralized node. The network is no longer fault tolerant in this set-up as breakdown intensity of the overloaded centralized node is far above the ground and resurgence might not be effortless when node failure.

Distributed Load Balancing

In distributed load balancing technique only one node is not responsible for making resource provisioning or task scheduling decision. Instead of single node for monitoring the cloud network multiple domains monitor the network to make precise load balancing decision. All nodes in the network maintain local knowledge base to make sure efficient distribution of tasks in static environment as well as relocation in dynamic environment. In distributed scenario, failure intensity of a node is not abandoned. For this reason the system is fault tolerant and balanced and also no single node is overloaded to make load balancing decision.

Hierarchical Load Balancing

Hierarchical load balancing includes different levels of the cloud in load balancing decision. This typeof load balancing techniques mainly function in master and slave relationship. These can be model using tree data structure in which each node in the tree is balanced under the control of its parent node. Master can use light weight mediator process to obtain statistics of slave nodes. By using the information gathered by the parent node provisioning or scheduling decision is made.

The Algorithms for Load Balancing can be based on Dependencies of Task

The execution of Dependent tasks is dependent on one or more subtasks. They can be executed just after completion of the subtasks on which it is completed. Consequently, scheduling of this type of task preceding to execution of subtasks is inefficient. Task dependency is represented using workflow based algorithms. Directed Acyclic Graphs can be used as knowledge base to represent task dependency. Based on single or multiple workflows are to be modelled or else single or multiple QoS factors are to be sustained in the system, algorithms are designed. Workflows can be categorized as Transaction Incentive where several instances of one workflow that have identical organization and Data Incentive

workflows in which size and quantity of data is bulky.

COMPARISON OF DIFFERENT TYPES OF LOAD BALANCING SCENARIO IN CLOUD ENVIRONMENT

Table 1 compares different type of load balancing scenarios in cloud computing environment. It specifies the knowledge base, usage advantages, drawbacks of each type of algorithm and issues addressed by these algorithms along with applicable environment.

Table 1: Comparative Analysis of Load Balancing Approaches

Algorithm	Pre-Knowledge Base	Applicable Environment	Issues to be Addressed	Advantages	Drawbacks	Examples
Static	Each node's status and user's Requirements must be known in advance	Homogeneous	More power consumption, poor resource utilization, less throughput and response time, not scalable	Simplicity no overhead no constant monitoring	Not Flexible Not Scalable Not compatible with changes in load	-Round Robin -MaxMin -MinMin etc
Dynamic	Use current or recent load information when making distribution decisions.	Heterogeneous	cost of collecting and maintaining load information performance	Improvement in performance cloud cannot rely on the prior knowledge whereas it takes into account run-time statistics highly adaptable	Difficult to be simulated complex	-Ant Colony -Genetic Algorithm etc
Centralized	All the allocation & scheduling decision is made by a single node. And it is responsible for storing knowledge base of entire cloud network	Used in small network having less load	High Failure Intensity Difficulty in Recovery Throughput	Reduces the time required to analyse different cloud resources	Great overhead on the centralized node Not fault Tolerant	-Round Robin -MaxMin -MinMin -Genetic Algorithm etc
Distributed	Every node in the network maintains local knowledge base	Used in Heterogeneous and large network	Fault Tolerant Interprocess communication Migration time	No single node is overloaded to take balancing decision	Complex Algorithm Communication overhead	-Honeybee foraging -Ant Colony
Hierarchical	Nodes operate	Used in	Migration Time,	Different levels	Complex	-Map-

	in Master-Slave mode so based upon the information gathered by the parent node scheduling decision is made	Heterogeneous and medium-large network	Failure Intensity Information exchange criteria	of the cloud reduces the load balancing overhead with communication optimized hierarchy	less Fault Tolerant	Reduce -LBMM
Task Dependent	Directed Acyclic Graph is used as knowledgebase to represent task dependency	Homogeneous as well as Heterogeneous	Fault Tolerant, Execution time, Migration time, Transaction Incentive workflow, Data Incentive workflow, Multiple workflow	Allocation of suitable resources to workflow tasks to achieve objective.	Difficult to model Maintenance of knowledge base is complex	-Cost based scheduling algorithm

LOAD BALANCING POLICIES

Transfer Policy: A transfer policy verifies whether a machine is in a appropriate state to take part in a task transfer either as a sender or a receiver.

Selection Policy: This policy decides which task is to be transferred once the transfer policy decides that a machine is in a heavily-loaded state. Selection policies can be classified into two types: pre-emptive and non pre-emptive. A pre-emptive policy picks a partly executed task so it is required to transfer the task state. Thus transferring operation is costly where as in non-pre-emptive policy only tasks that have not begun execution are selected as a result it does not require transferring the state of task.

Location Policy: The purpose of this policy is to locate a proper transfer associate for a machine, once the transfer policy has determined that the machine is in heavily loaded state or lightly loaded one. General location policies are random selection, dynamic selection as well as state polling.

Information Policy: This policy find out at what time the information regarding the status of other machines have to be collected, from where it has to be collected along with what information is to be collected.

LITERATURE REVIEW

Following section includes survey of related papers.

An Optimal Load Balancing Technique for Cloud Computing Environment Using Bat Algorithm

Shabnam Sharma et. al[2] proposed an algorithm based on echolocation behaviour of Bats. When the bat tries to discover its food, it will adjust its position, speed and rate of pulse production depends on the distance between the food and itself. This concept is implemented in Bat algorithm to find the optimal server among all the available servers, for the execution of incoming jobs. When any task arrives in the job pool, load balancer will invoke the bat algorithm to find the best server which suits to the requirement of incoming task. The bat algorithm considers job type and resource required, while selecting the optimal VM for execution of task. Once an appropriate VM is selected, it assigns the load to that

machine. If the load is higher than the load of all other servers, then the task is distributed to more than one server. Bat algorithm based load balancing technique has been implemented for minimizing the response time and per-forms load balancing without causing any delay By using Bat algorithm, optimum and best result can be obtained, by executing the algorithm over multiple iterations.

A Novel Approach for Dynamic Load Balancing with Effective Bin Packing and VM Reconfiguration in Cloud (DLBPR)

Dinesh Komarasamy et. al[3] proposed an approach in which Physical Machines(PM) and Virtual Machines(VM) are considered as bin and items respectively. VMs are packed in a PMs.The Vms are grouped based on the processing speed of the VM as small, medium and large clusters based on their processing speed with the support of VM live migration. The system supports the load balancing during the execution of deadline based jobs. Initially, the deadline based job scheduler categories and prioritizes the jobs to complete within the deadline.DLBPR approach clusters the VM at the runtime using the receiver-initiated approach. The VM live migration reconfigures the VM based on the required processing speed of the job. The approach outperforms the existing scheduling algorithm by migrating the VMs from one cluster to another. It mainly focused on load balancing that automatically improves the throughput and also increases the utilization of the resources

A Heuristic Clustering- Based Task Deployment Approach for Load Balancing Using Bayes Theorem in Cloud Environment

Jio Zhao et.al [4] implemented a heuristic Clustering based Load Balancing in Cloud using Bayes Theorem and called it as Load Balancing based on Bays and Clustering (LB-BC). LB-BC has short out the disappointment quantity of task disposition events perceptibly, improved the throughput, and optimized the external services performance of cloud data centres. LB-BC first has narrowed down the search scope by comparing performance values. Then, LBBC has utilized Bays theorem to obtain the posterior probability values of all candidate physical hosts. LB-BC has combined probability theorem and the clustering idea to pick out the optimal hosts set, where these physical hosts have the most remaining computing power currently, for deploying and executing tasks by selecting the physical host with the maximum posterior probability value as the clustering centre and thus to achieve the load balancing effect from the long-term perspective.

Load Balancing Through Arranging Task with Completion Time

Palash Samanta et. al[5] propose a scheduling algorithm, Load Balancing through Arranging Task with Completion Time(LBATCT), which coalesces lowest completion time along with load balancing strategies. LBATCT assign tasks to computing nodes according to their resource capability. LBATCT can provide efficient utilization of computing resources and maintain the load balancing in cloud computing environment .However, the load balancing of cloud computing network is utilized, all calculating result could be included primary by the second level node prior to sending back to the supervision. Thus, the goal of load balancing and improved resources management could be accomplished.

A Load Balancing Task Scheduling Algorithm based on Feedback Mechanism for Cloud Computing

Zhang Qian et. al [6] proposed a weighted random scheduling algorithm based on the peer-to-peer cloud computing environment. The resource attributes are divided into two parts static and dynamic to be evaluated respectively.

The algorithm uses the weighted random strategy, overload assessment and feedback to ensure that the nodes with excellent performance will not be overburdened based on submitting tasks first to the resources with the best performance. When the computing nodes with excellent performance are busy then tasks are assigned to the other computing resources with better performance according to the feedback mechanism. The algorithm improves the resource utilization of the system and balance the load of the system dynamically.

Efficient Load Balancing Methodology for Future Internet Based on Game Theory

Shaoyi et.al [7] implemented an efficient Load Balancing methodology for Future Internet based on Game Theory. During the study of future internet it is observed that load harmonizing processes and job distributions are main investigation teething troubles in regions of reserve organization of upcoming internet. Authors have invented static load balancing problem in the model proposed an uncooperative game among users and cooperative game among processors and based on this model they developed a load balancing algorithm for computing centres. The compensation of the model are healthier scalability, educating system presentation, and low cost on maintenance of system material.

Shared Resource Clustering for Load Balancing

Dr. Vinay Chavan et al. [8] have employed this technique for load balancing in cloud. Jobs are made up of different tasks. Virtual Machines are required to implement this set of tasks. Clustering of tasks result in producing fewer numbers of jobs and improves the level of performances by balancing load as well as number of VM required to carry out tasks gets trim down. If tasks are not clustered then single jobs are assigned to virtual machines and need of deploying these virtual machines with dynamic formation increases in real time. This preparation provides proficient CPU utilization and load sharing.

An Efficient Local Hierarchical Load Balancing Algorithm (ELHLBA) in Distributed Computing

Rafiqul Zaman Khan et.al[9] discover Efficient Local Hierarchical load balancing approach absorbs the qualities of both centralized along with decentralized approach by eliminating the short comes of centralized and decentralized approaches. In ELHLBA hierarchical topology is chosen for load balancing which it is easy to manage and maintain the network because the whole network is separated into tiny segments known as cluster and error detection and correction is also easy and if one cluster is broken, other cluster will continue to work. The parents of leaf nodes act as front end nodes which execute the tasks if the leaf nodes are overloaded. The algorithm produces better result than existing algorithms in respect of response time and throughput against system utilization

Heat Diffusion Based Dynamic Load Balancing

Yunhua deng at el. [10] anticipated an algorithm that is based on the rule of heat diffusion used for load balancing. In heat diffusion concept heat dissemination has happened from high temperature to low temperature. This same observable fact is used for load balancing purpose in the terms of VM's. The traffic flow of user request is from overloaded VM to underloaded VM. According to algorithm the virtual environment is separated into number of cells and each cell has objects, every node in cell conveys load information to its neighbour node in particular iteration. In heat diffusion load balance environment amount of load migrates is bare minimum. This algorithm proficient for multiprocessor network. Network latency is minimized for load transfer between the cells. Restrictions related with this technique are usage of high computational and communication methods, small connectivity for large scale graphs or cells, network delay, more time

wastage in case of more iterations.

A Fast Adaptive Load Balancing Method

Dongliang Zhang et al.[11] proposed method based on binary tree search to improve the performance of distributed simulation system. This fast adaptive load balancing method is used to adjusting the workload among the processors from local region to global region. Simulation region is partition into sub - domains using binary tree structure. Domain decomposition is based on discrete approximation method, finite element method and binary element method. Benefits of this technique are lower communication overhead, fast balancing speed and soaring efficiency and the shortcoming is it cannot preserve the topology of cells.

Table 2 shows comparative analysis of various algorithms discussed above.

Table 2: Comparative Analysis of Load Balancing Algorithms

No	Author	Publisher	Technology Used	Advantages	Issues
1	**Shabnam Sharma et. al**	Indian Journal of Science and Technology (2016)	Bat Algorithm	-Minimizing the response time -No Delays -Simple flexible and easy to implement.	-Become problematic for higher-dimensional problems
2	**Dinesh Komarasamy et. al**	Indian Journal of Science and Technology (2016)	Bin Packing and VM Reconfiguration	-Improves the throughput -Increases the utilization of the resources -Waiting time Reduction	-VM Reconfiguration is subject to resource and placement constraints.
3	**Jia Zhao** et.al	IEEE (2016)	Heuristic Clustering Based on Bays Theorem Implemented.	-Reduced the failure number of task deployment events -Improved the throughput -Optimized the external services performance.	-Highly Complex Method.
4	**Palash Samanta et. al**	International Journal of Grid and Distributed Computing (2016)	Arranging Task With Completion Time	-Good resource utilization	-More calculations required
5	**Zhang Qian et. al**	International Journal of Grid and Distributed Computing (2016)	Task Scheduling Algorithm based on Feedback Mechanism	-Avoided the system bottleneck effectively -Self-adaptability -Improve the resource utilization	-Complex implementation
6	**Shaoyi Song et.al**	Journals of Applied Mathematics (2014)	Future Internet based Load Balancing Technology.	-Better Scalability. -Improving system performance -Low cost on maintaining system information.	-Network Contention

7	**Dr. Vinay Chavan** et.al	IEEE (2014)	Shared Resource Clustering	-Efficient CPU utilization -Improved scalability resource sharing	-Apriori specification of the number of cluster centres required
8	**Rafiqul Zaman Khan** et.al	Indian Journal of Science and Technology (2013)	Hierarchical Topology	-Improved response time and throughput System utilization	-Implemen tation complex
9	**Yunhua deng** el. al	ACM (2010)	Heat Diffusion Based Dynamic Load Balancing	-Require less amount of calculation,-High speed	-Wastage of time, Network delay
10	**Dongliang Zhang** et al	ELSVIER (2009)	binary tree structure	-Faster balancing speed -Low communication overhead,-Efficient	-Fail in marinating Topology of cells

CONCLUSIONS

Load harmonizing is one of the key challenges in cloud computing environment. hence it is required to distribute the load evenly among nodes of cloud. A tremendously overcrowded service provider may fail to provide effective services to its customers. Therefore with appropriate load balancing algorithm system response, service and throughput can be augmented. This paper shows a comparison in order to evaluate various existing load balancing techniques. Using this comparison, performance can be improved of any existing techniques by implementing some new ideas as this table provides what is there in algorithm and what is missing. The Various Load Balancing techniques are analysed here. Since Load Balancing techniques requires a lot of computational overhead, lot of techniques are implemented to solve the issues. Here by analysing these techniques their various advantages and limitations an efficient technique for Load Balancing can be implemented in future.

REFERENCES

1. http://shodhganga.inflibnet.ac.in/bitstream/10603/26264/5/chapter 3.pdf.
2. Shabnam Sharma, Ashish Kr. Luhach,Sinha Sheik Abdhullah,” An Optimal Load Balancing Technique for Cloud Computing Environment using Bat Algorithm”, Indian Journal of Science and Technology, Vol 9(28), DOI: 10.17485/ijst/2016/v9i28/98384, July 2016.
3. Dinesh Komarasamy, Vijayalakshmi Muthuswamy,” A Novel Approach for Dynamic Load Balancing with Effective Bin Packing and VM Reconfiguration in Cloud”, Indian Journal of Science and Technology, Vol 9, Issue 11, DOI: 10.17485/ijst/2016/v9i11/89290, March 2016.
4. Jia Zhao, Kun Yang,” A Heuristic Clustering- Based Task Deployment Approach for Load Balancing Using Bayes Theorem in Cloud Environment”, IEEE Transaction on Parallel and Distributed Systems, Volume 27, Issue 2, February 2016.
5. Palash Samanta, Ranjan Kumar Mondal,”Load Balancing Through Arranging Task With Completion Time”,

International Journal of Grid and Distributed Computing Vol. 9, No. 5, pp.273-282,2016,http://dx.doi.org/10.14257/ijgdc.2016.9.5.23.

6. Zhang Qian, Ge Yufei, Liang Hong, Shi Jin," A Load Balancing Task Scheduling Algorithm based on Feedback Mechanism for Cloud Computing", International Journal of Grid and Distributed Computing Vol. 9, No. 4, pp.41-52,2016,http://dx.doi.org/10.14257/ijgdc.2016.9.4.04.
7. Shaoyi Song, TingjieLv, Xia Chen," Load Balancing for Future internet: An Approach Based on Game Theory", Journal of Applied Mathematics, 2014.
8. Dr. Vinay Chavan, Parag Ravikant Kaveri "Clustered Virtual Machines for Higher Availability of Resources with Improved Scalability in Cloud Computing" Networks & Soft Computing (ICNSC), First International Conference, IEEE,2014 .
9. Rafiqul Zaman Khan, Md Firoj Ali," An Efficient Local Hierarchical Load Balancing Algorithm (ELHLBA) in Distributed Computing", International Journal of Science, Engineering and Computer Technology, November 2013 , Vol 3, Issue 11, pp 427-430.
10. Yunhua Deng, Rynson W.H. Lau, "Heat diffusion based dynamic load balancing for distributed virtual environments", in: Proceedings of the 17th ACM Symposium on Virtual Reality Software and Technology, ACM, 2010, pp. 203 – 210.
11. Dongliang Zhang, Changjun Jiang, Shu Li, "A fast adaptive load balancing method for parallel particle -based simulations", Simulation Modelling Practice and Theory 17, Elsevier, pp 1032– 1042, 2009.