Vol 6 • Number 2 April - Sep 2015 pp. 269-274 **Impact Factor: 2.5** **Available at** www.csjournals.com

An architecture for Load Balancing Techniques for Fog Computing Environment

Manisha Verma, Neelam Bhardawaj, Arun Kumar Yadav
[1]MTech Scholar, UPTU University Agra, U.P. India
[2]Assistant Prof., Hindustan Institute of Tech. & Mgmt. Agra, U.P., India
[3]Associate Prof., ITM University, Gwalior, M.P., India
soni2185@gmail.com, 8.neelambhardwaj@gmail.com,arun26977@rediffmail.com

Abstract: Cloud computing is an emerging computing technology that uses internet to maintain large applications and data servers to provide services to end users belonging to different organization. It has make availability of resources in a secure and flexible manner thereby enhancing the throughput and performance .It has certain limitation that the load balancing of data and hosting of cloud data centers in the internet create large and unpredictable network latency. It is then resolved by new sort of computing model called Fog Computing. Fog is analogous to cloud only difference is that it is located at the edge of network. Another advantage is that the applications which requires location dependence can be make feasible through it.. In this paper, we are proposing new architecture based on load balancing algorithm in Fog Computing environment in a less complex and effective manner which because the number of end users are increasing every day at a very rapid rate.
Keywords: Cloud Computing, Fog Computing, Load balancing, Reliability, Throughput, Virtualization.

Introduction

Cloud computing is a newest computing paradigm which make resources like applications and files available through internet having thousands of computers interlinked together in a distributed and complex manner and you have to pay for the resources as you use them .Cloud Computing uses the concepts of virtualization, distributed computing, networking, software and web computing for its implementation. A cloud consists of several elements such as clients, datacenter and distributed server. The focus in cloud computing is to reduce execution time of tasks on the machines which are known as virtual machines that run in parallel with each other. Cloud Computing provide different service models according to different user needs like Infrastructure as a service known as IAAS, Platform as a service known as PAAS, Software as a service known as SAAS providing a cost effective manner to pay for services on utility costing basis. In this multiple users are able to access more than one server without purchasing and considering the updating of license for different software applications. So it is basically SOA (service oriented architecture). It offer many advantages like fault tolerance, high availability, reduced overhead, flexibility, scalability, reliability, location independence, elimination of system administrative functions.

Overview of Load Balancing Technique

Load balancing[1] is a systematic approach to reassign the total loads of the various overloaded servers to under loaded servers, data centers , hard drives or other computing resources there by helping cloud service providers to distribute application requests to various data centers. It is basically the process of distribution of site traffic among various servers with the help of network based device or load balancer like switch, router which intercepts traffic to target site or server and redirects the traffic or split it in to individual requests to desired replica servers on the basis of there availability always keeping in mind the performance of cloud computing environment. It is dynamic in nature because load varies according to the client or end users request. When the various servers get over loaded through user requests then we need a load balancing approach to distribute the load to unutilized servers.

The various factors here to be considered are estimation of total load, scalability of servers, throughput, performance of system, interaction between the servers, amount of work to be transferred, and selection of best nodes. The load can vary from CPU load, amount of virtual memory used, and network load.

Load balancing can be done through in many ways like static or dynamic, periodic or non periodic, centralized or decentralized In static load balancing algorithm, we have prior knowledge of load or system resources and it does not depend over the current status of the network whereas in dynamic algorithm load varies or changes on the server and it depends the current status of the network. It generally gives better performance then static algorithm.

Goals of Load Balancing Techniques [2] are

- Maximum Response Time – It is also called as Time to First byte or TTFB. It is the time interval between sending a request packet to a server and receiving first response.
- Maximum throughput-It is the Maximum time required to complete a transactions in per second.
- Optimum Resource utilization-In this we utilize the resources in such a way that none of them remain idle.
- Min network delay- Minimum delay that occurred between intermediate network devices such as switch, router etc.
- Performance factor- It checks in how much minimum time the algorithm complete the user requests.
- Fault tolerance-It means that in case of the loss of a service like network, some host, software crash there are other servers to manage this.
- Execution time- It is the time required to complete a user requests.
- Scalability- In this if the workload of a server exceeds the certain capacity of your existing software and hardware the system can scale up the system (software and hardware) to adjust the increased workload.
- Low Overhead-Low overhead refers to the processing time required by system which includes the operating and any utility that support application programs

Load balancing can be categorized in to various approaches

A. Based on type of Load Balancer

i. Hardware based load balancing can route TCP/IP packets to various servers in cloud .These type of load balancers are used to provide robust topology with high availability, uses circuit network gateway to route traffic but with high cost.

ii. Software based load balancing which is software based and is a combination of web server and application server software packages. It is less costlier than hardware based also it is flexible as it can be configured according to requirements, incorporate intelligent based routing based on various input parameters , but for doing this we have to add additional hardware to isolate the load balancer.

B. Based on different state of Cloud environment

i. Static Algorithm –It divides the traffic equally between all servers and once the task being assigned to a virtual machine it will keep executing on it. In this performance of processors is determined using prior knowledge of coming tasks. Its drawback is that it may be possible that certain information may not get available during allocation of tasks which may lead to incomplete execution of tasks.

ii. Dynamic Algorithm- In this decision of allocation of tasks to virtual machines is made at runtime. Tasks are buffered on the queue and executed on various virtual machines according to there availability. The tasks are continuously shifted from one virtual machine to another during the execution which make it little complex. It too can be divided in to two parts

- Centralized Algorithm: -In this decision of distribution of workload is taken by a single node of the system but if that main node get fails by any reason then the whole system will halt down.
- Distributed Algorithm: -In this decision of distribution of workload is taken by many nodes at a time and task of load balancing is shared by them , its pros is that if in any case any node fails then the whole system will not halt only performance get degrade to a extent.

C. Based on who initiated the process

i. Sender Initiated –In this initiation of load balancing is caused by heavily loaded nodes which gather information of load assigned to different nodes and according to their workload a load is assigned to low loaded node.

ii. Receiver Initiated-In this initiation of load balancing is caused by low loaded nodes which gather information of heavily loaded nodes and retrieve work from them. In system it is easy to gain information of heavily loaded as compared to low loaded nodes.

iii. Symmetric-In this initiation balancing of workload can be very well coordinated both by Sender and Receiver depending on the conditions.

D. Based on Policy

i. Information Policy- It defines what workload information is to be collected from nodes and from where and how.
ii. Resource Type Policy - In this resource is defined as either server or receiver of process according to its availability.
iii. Location Policy- It defines which destination node is to be selected for transferring the workload.
iv. Transfer Policy- It defines which task is to be selected for transferring from local node to remote node.
v. Selection Policy- It defines which processors are involved in Load Exchange.

Related Work

A. Honey bee behavior inspired load balancing of tasks in cloud computing environment

Dhinesh babu L.D et al. [3] proposed an algorithm based on HBB-LB model. In this algorithm analogous behavior of Honey bee is used. In this two types of bees are there one is scout bees which forage for food sources and then they come back to beehive and inform forage beehive through a waggle/tremble/vibration dance about the quality and quantity of food and distance from the beehive. After knowing these forage bee's follows the same path of scout bees to the food source location, and in the same manner they inform other bees too and process goes on vice versa telling about how much food is left.

In HBB-LB algorithm same concept is used for load balancing, here the tasks are represented as Honey bees and Virtual Machines are represented as food sources. Here Virtual Machine can be in three situations Balanced, Overloaded, Low Overloaded .The removed tasks from overloaded VM's act as honey bees and then these tasks are submitted to Under loaded VM's based on how many high priority tasks and tasks are running on that under loaded Virtual Machines. Only that under loaded Virtual Machine's is selected which has least priority tasks and low load and after that information regarding it is globally updated so that other priority tasks get there suitable under loaded Virtual Machines. Its advantages are good resource utilization, maximum throughput, and Quality of Service is based on the task priority. Limitations are that low priority tasks have to wait to for long time in the queue thus unbalancing the balancing workload.

B. Dynamic Load balancing in distributed virtual environment using Heat diffusion

Yunhua deng et al. [4] proposed an algorithm that is based on selecting best efficient cell in conjugation with two heat based diffusion algorithm called Global and Local diffusion which is quite a simple concept. In Heat diffusion concept heat flow from high temperature to low temperature. In the same manner here transfer Of the load or traffic from Overloaded VM to Low Over loaded VM is done. In this we divide the Virtual environment into number of cells and each square cell have objects, here every node in cell send load information to its neighbor node in each iteration.

In local diffusion local decision or scheduling of virtual resources is done and user's request is locally satisfied using the local decision. In global diffusion global level of decision making is used for assigning VM according to user request and Virtual Machines overloading is managed at global level. In heat diffusion Global method for load balancing is better than Local method. This algorithm is simple and efficient as communication overhead is less because it produce low number of migrated users and quite feasible for multiprocessor environment and also computation time is less as compared to other algorithms and also it has good convergence threshold. Limitations associated with this technique are network delay on a path and when several iterations are used lot of time wastage occurs.

C. Decentralized scale free network construction and load balancing in massive multiuser virtual environments

Markus Esch et al.[5] proposed the concept of self organized and scale free backbone for the interconnection of machines to do load balancing in the Hyper verse architecture which is based on a two tier P2P architecture. It is having two network overlays one loosely structured (used by user clients) and other structured (used by public servers) .In this non uniform load distribution area is subdivided in to cells and each cell is maintained by public servers voronoi scheme. In which we assign a virtual position of the world surface to each public server. According to load of clients and other web services high capacity and low capacity machines are introduced, also the virtual positions keeps on changing according to absolute workload in the cell, bandwidth, computational power and payload of its immediate neighbors.

In this scale free link structure between network nodes in a decentralized manner is used. Many extra links to the other nodes are provided in case of failure or load increase so that fast routing of load to all positions never stops. The advantage of this method is that network become fault tolerant, reliable, short average path length

and resilient which is must in global scale view. Limitations of this it sometime lack global synchronization and more time is used in it as load constantly been transferred between cells.

D. Genetic Based Algorithm Load balancing strategy

Kousik Dasgupta et al. [6] proposed a dynamic based algorithm for load balancing which is used to find the globally optimal solution in complex or vast search space also it does not get trap in local optimal solutions by using artificial intelligent techniques that are quite feasible for effective search and optimization.

It consists of three operations selection, genetic operation and replacement , this algorithm work as follows first initialize processing of unit is done by encoding them into binary strings then evaluation of the fitness value is done based on some initial mutation probability then optimal solution is found considering chromosome with lowest fitness twice and eliminate it with chromosome of highest fitness to construct mating tool which is called selection and then single point crossover to form new offspring called crossover then mutate new offspring with probability of 0.05 called mutation and process goes on placing new offspring as new population. Here ultimate goal is to reduce the cost function.

E. A Fast adaptive load balancing method for parallel practical based simulations

Dongliang zhang et al. [7] proposed the scheme of binary tree structure to improve the performance of simulated systems. Here partition of the simulated region in to various sub domains or cells is done, each having its processor associated with it. In this each leaf node represents a cell and a parent node in a binary tree constitutes the area having all cells which are child node basically. In this arrangement of the child nodes or cells into hierarchy is done. The main characteristic of this algorithm is to transfer or adjust the workload between various processors from local domain to global domain through compressing and stretching the cells according to net difference of workload between adjacent cells.

Here the indexes of the cells on the left of binary tree and the top are smaller than that of right and bottom.

Advantage of this technique is that there is a lower communication overhead, faster speed and high efficiency in distributed environment. Disadvantage of this technique that maintaining network topologies of cells is somehow sometimes become trivial.

F. A dynamic and adaptive load balancing strategy for parallel file system with large-scale I/O servers

Bin dong et al. [8] proposed an algorithm based on distributed architecture for dynamic and adaptive file migration as various problems occur in centralized system for large and dynamic file system. To get rid out of it a new algorithm has been proposed called as a SALB (self active load balancing algorithm). SALB runs on each I/O server and estimate future load on various servers by on line load prediction model. It takes decisions on which new or other server load should be transferred along with this. It is also aware of network transmission rate which it generally avoid to take load decision as only one network transmission is considered. SALB ability to predict future load help other servers to take decisions thus reducing decision delay an network transmission.

Its threshold value can also be adjusted dynamically and also SALB works silently and does not interrupt the services provided by whole system. SALB algorithm main characteristic is a decision making quality for distribution of workload in the distributed system. In beginning when central decision making, system are used for load balancing they give poor or no response if central server get fails accidentally.

In group decision making system is divided into groups, which is having its own decision making ability but here it lacks in the global view of system load, so because of these pitfalls distributed decision making system is preferred which provide us scalability, availability, throughput, high speed processing, elasticity, good response time, good resource utilization, ability to handle large file system, and load migration without affecting the system. Its limitation is that due to continuous file migration the system performance get degrade to some extent.

G. A Load Balancing in Cloud Computing using Stochastic Hill Climbing – A Soft Computing Approach

Kousik Dasgupta et al. [9] proposed a optimized and dynamic based algorithm for load balancing which is centralized in nature that is fewer messages are required by the central node to take the decision of distributing the workload, but it has some consequence too in case if central node fails the whole system performance will degrade which can be solved if workload is distributed effectively to give maximum throughput by optimizing it. Optimization can be done in two ways first is complete method in which valid assignment of values to all variables is done to find the solution or if any one assignment is wrong then it will not be considered as solution. The other is incomplete solution in which probability of more correct answers for all input parameters been considered as solution. This approach is simple, effective, and having good speed in solving problems. So it is being considered in Stochastic Hill Climbing approach for solving optimization problem.

In this a loop is taken which constantly generate random values in increasing order which is analog to move to the uphill. A value generation is stopped when it gains a unique upper value or peak which no other neighbor

value is having. This value chosen at random from the uphill moves its probability of selection depends on the steepness of the uphill move.
In this new values are mapped to old values by doing some minor changes to old values. Each assignment of values of a set are considered to be validated if they fit into some predefined criteria. The best value of the set is selected for next assignment and this process goes on until a solution is found or stopping criteria is acquired.
It has two main components a candidate generator which map solution of one candidate set to other successor values and other is evaluation criteria which select best solution and keep on giving ranks to the solution so that it can lead to further improvements to get best solution. It can be further improved by using other soft computing approaches.

Limitations of Cloud Computing

Though cloud computing offer many advantages but it has certain limitations too providing high speed reliable internet connectivity, high latency, multi-homing to avoid link outages ,requirement of high capacity bandwidth, security and privacy as data has to travel a long path from server to users though if we encrypt data. The emerging trend of Internet of things[10] that want every device or objects to be connected on the Internet, sensor network , real time applications, mobile data applications that require fast transfer of data and bandwidth put limitation on cloud computing. Real time applications which require high speed streaming of data when user directly interact with them suffer from delay jitter caused by latency in network which need to be solved by cloud computing.
These shortcomings can be resolved by the introduction of Fog Computing which provides better quality of service in terms of reduced data traffic, low latency, location awareness, low bandwidth, mobility which is made feasible by making fog computing systems very close to end users.
The other difference between fog computing and cloud computing is that in former the distance between client and server require one hop in while in later it requires multiple hops as the signal get weak while passing through various servers, also cloud computing required leased line while fog computing is entirely based on wireless network. Despite of many differences both Fog and Cloud complement each other as Cloud Computing will be preferred for high end batch processing jobs and Fog Computing will be preferred for more resilient and providing more security to the emerging technology.

Introduction to Fog Computing

Fog Computing is a term which is introduced by the Cisco systems, it is basically extended or new model of cloud computing in which we will connect enormous wireless data objects in distributed environment by placing the data and resources at the edge of cloud rather than hosting and working from a centralized cloud. In this bandwidth can be reduced as there is no need for transferring every bit of information over cloud network rather we can aggregate data at some access points which lower costs , increases efficiency, throughput ,reduced data traffic ,lower power delay also provide security as data transfer has not to follow a long path.
Fog is nothing but a virtualized platform like cloud only difference lies in the fact it is closer to the ground. Hence Fog Computing is called as an edge computing because Fog System operate from network ends. In this decision are taken as close to the place where data is generated as much as possible and stop it sending to reach Global network which is done initially by training fraction of data on machine learning models after the results are found accurate then this model is implemented in to the devices.

Proposed Architecture for Load Balancing in Fog Computing Environment

We are here proposing an Architecture for Load Balancing in Fog Computing environment. During the accessing of resources in Cloud Computing Environment the client time increases as servers are far located. So to get rid out of this problem we bring the resources as close as to the end users in Fog Environment. First the users sent the request to the nearby Fog servers which are consistently maintaining the frequently used data. In case, the users don't get the desired resource in nearby Fog Servers who are also communicating with each other, then that request is forwarded to the cloud servers. To implement this load balancing algorithm we will be using modified Honey Bee Based algorithm. In addition to this other algorithm good factors will also be used in Fog Computing environment.

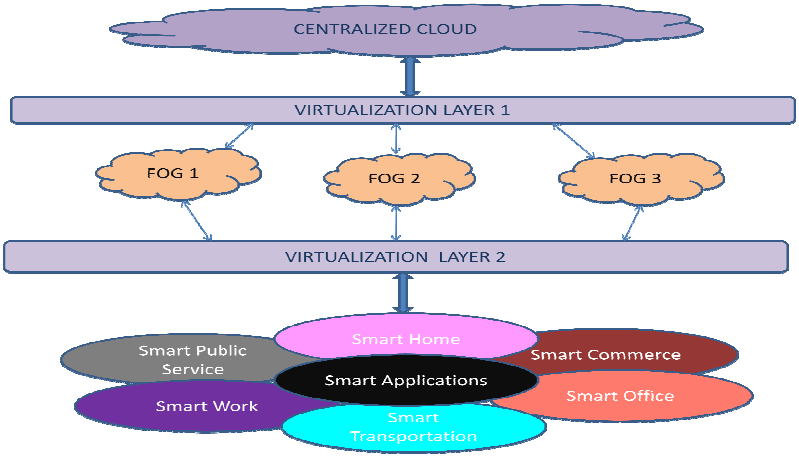

Figure 1. Load Balancing in Fog Computing Environment

Conclusion

This proposed architecture will help in bandwidth utilization, reduced cost, fast access speed, maximum throughput and many other factors. It will improve the computing needs of Internet of things. Therefore each and every device will be able to get connected to the internet. It will also resolve the problem of server availability which frequently occurs in Cloud Computing environment.

Future Work

Fog Computing can be thought of as an emerging trend that can provide various services immediately also it support applications that need location awareness , low latency , real time interactions, mobility , low delay jitter and many more by deploying applications closer to end users in distributed way managing the challenges of resource management, security , monitoring, accounting , testing . It can give enormous amount of speed to smaller organization whatever the devices they are using to establish quicker and closer connection to compute resources.

We can use the HBB-LB algorithm along with artificial intelligent and neural model for improving performance and also can add on some other features in such that low priority tasks does not have to wait for a long time and data availability become very efficient in Fog Computing environment.

This paper can be extended with a detailed architecture, tested on simulation tools and have compare with other algorithms. Distributed load balancing is a broad and highlighted area for researchers to go ahead with new algorithms and improvement of existing one. Researchers can also propose new scheduling mechanism for reassignment of resource in Fog Computing environment.

References

[1] Rukman Palta, Rubal Jeet" Load Balancing in the cloud computing using virtual machine migration" International Journal of Application or Innovation in engineering and management Vol. 3, Issue 5, May 2014.

[2] Rajesh George Rajan, V. Jeyakrishnan "A survey on Load Balancing in Cloud Computing Environment". International Journal of Advanced Research in Computer and Communication Engineering Vol. 2, Issue 12, December 2013.

[3] Dhinesh Babu L.D, P. Venkata Krishna,"Honey bee behavior inspired load balancing", Applied Soft Computing 13(2013) 2292-2303.

[4] Yunha Deng Rynson W.H. Lau," Heat diffusion based dynamic load balancing for distributed virtual environments",in:Proceeding of the 17th ACM Symposium on Virtual Reality Software and Technology,2010,pp.203-210.

[5] Markus Esch , Eric Tobias, " Decentralized scale-free network construction and load balancing in Massive Multiuser Virtual Environments", in: Collabrative Computing: Networking,Applications and work sharing, Collaborate Com,2010, 6th International Conference on IEEE , 2010,pp 1-10

[6] Kousik Dasguptaa, Brototi Mandalb, Paramartha Duttac, Jyotsna Kumar Mondald, Santanu Dame "A Genetic Algorithm (GA) based Load Balancing Strategy for Cloud Computing" in: International Conference on Computational Intelligence: Modeling Techniques and Applications (CIMTA) 2013.

[7] Dongliang Zhang, Changjun Jiang,Shu Li, , "A fast adaptive load balancing method for parallel particle- based simulations", Simulation Modeling Practice and Theory 17 (2009) 1032–1042.

[8] Bin Dong, Xiuqiao Li, Qimeng Wu, Limin Xiao, Li Ruan, "A dynamic and adaptive load balancing strategy for parallel file system with large-scale I/O servers", J. Parallel Distribution Computing. 72 (2012) 1254–1268.

[9] Brototi Mondala, Kousik Dasguptaa, Paramartha Duttab " Load Balancing in Cloud Computing using Stochastic Hill Climbing-A Soft Computing Approach" in: Procedia Technology 4 (2012) 783 – 789.

[10] Flavio Bonomi, Rodolfo Milito, Jiang Zhu, Sateesh Addepalli "Fog Computing and its Role in the internet of things" in: http://conferences.sigcomm.org/sigcomm/2012/paper/mcc/p13.pdf