

Optimization Load Balancing Over an Imbalance Datacenter Topologies



K Siva Tharun, K. Kottilingam

Abstract: *The quick demand of cloud resources, responsible for design a highly dynamic and flexible Cloud, has become a main challenge in datacenter deployment. A huge number of virtual machines will be available in Datacenter. Further Datacenter will be divided into a greater number of clusters. Each cluster is grouped to same type of Virtual machines. The virtual machines inside the cluster is homogeneous and heterogeneous to other cluster. Any virtual machine can be allocated to end user. If an unhealthy and less energy virtual machine is allocated to user, it will completely degrade the performance of the machine. To overcome this issue, we use an efficient load-balancing algorithm to allocate virtual machine to end user. The Fuzzy Optimized load-balancing algorithm uses the bandwidth, memory, CPU utilization are the key metrics. An efficient algorithm increases the number of hosts allocated to each end user.*

Index Terms: *Datacenter, Memory, CPU, Bandwidth, Fuzzy, Load Balancing..*

I. INTRODUCTION

From most recent couple of decades, Datacenters has rapidly radiate not exclusively regarding equipment assets and administrations however furthermore from the investigation motivation behind read.

For sure, numerous and various administrations square measure made open to clients, from on-line stockpiling to a spread of applications. Every one of these administrations need assets that square measure set some place inside the Cloud. Clients couldn't care less wherever they will be, they exclusively care they exclusively measure that are present when required and wanted quality and security levels.

To meet quickly developing necessities clients, another amazing Cloud Computing [1] configuration has risen, where clients could raise and get data and administrations progressively through the system. With the exceedingly request of Cloud Services unendingly expanding, receiving an adaptable, adaptable, versatile and random plan of the cloud, ready to meet the jumbled requesting of the snazzy time of

overall data, has transformed into a certifiable test for Data Center structures.

In this structure, the chief critical issue to act naturally tended to is depicted by the conservative utilization of capacity, calculation and system assets for the portion of virtual machines in an amazingly conveyed and random situation [2]. In this paper, we will in general strategy a one of a kind methodology for VM allotment bolstered representative rationale. Moreover detail, entirely unexpected usage of our Fuzzy algorithmic program. We contrast and two variations are Multi-Objective allocators likewise as a basic Mono-dimensional algorithmic program. In addition, we will in general work in three diverse portion strategies, especially Best work, Initial Work and Worst Fit.

The clarification behind our proposition is that the quality not exclusively of the improvement strategy, anyway even the distinguishing proof of exact capacities to be diminished. To be sure, its intense to diagram a generally acknowledged cost perform since very surprising parameters to be thought about, for example, RAM, CPU and DISK and it is difficult to evaluate the pertinence of every issue. On the contrary hand, human experience is developing thus a characteristic gratitude to take advantage of heuristic dependable guidelines is through the work of representative rationale, a sort of numerous esteemed rationale that manages thinking that is surmised as opposed to attached and real.

Many Entrepreneurs are now collaborative with Amazon EC2[3], Salesforce[4], Microsoft Azure [5], VMware [6]. To distribute the VM load, Load balancing came in to play. Virtual assets could be included or expelled from the cloud stage whenever. The expense of the virtual asset is pay-as-you-go premise. The upside of Cloud Computing is easy to scale up and scale down with less human effort. It brings you the less down time and high availability. We can optimize the cloud cost by using an efficient utilization of resources.

Fuzzy based algorithm have applied in many areas, very few ways the fuzzy based algorithm implements on cloud computing. For instance[7], authors are oversee dynamic asset re-distribution in distributed computing Data focus utilizing a multi-specialist variant of the fluffy rationale controller. Many burden adjusting calculations on Round Robin Scheduler [8], Dynamic Monitoring Load Balancer [8], Min-min Scheduler [9] and Hungarian Algorithm.

In this paper the proposed methods are compared using CloudSim simulator tool which provides a clear information about the project works in real time environment.

Manuscript published on 30 September 2019

* Correspondence Author

K Siva Tharun*, PG Scholar, Department of Information Technology, SRM Institute of Science and Technology, Kattankulathur, India. tharun4ru@gmail.com

Dr. K. Kottilingam, Associate Professor, Department of Information Technology, SRM Institute of Science and Technology, Kattankulathur, India. k.kottilingam@gmail.com

© The Authors. Published by Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP). This is an [open access](https://creativecommons.org/licenses/by-nc-nd/4.0/) article under the CC-BY-NC-ND license [http://creativecommons.org/licenses/by-nc-nd/4.0/](https://creativecommons.org/licenses/by-nc-nd/4.0/)



By using this tool all the different possibilities are checked to prove the proposed method is more efficient than the existing method.

The remaining paper consists as follows. In Section II presents the few existing Load balancing techniques in the Literature Review. Section III presents the Problem definition. Section IV describes the cloud architecture used in simulation.

Section V describes the proposed algorithm in detail. Experimental analysis represents in Section VI, Section VII describes the Conclusion and Section VIII explain in details about References

II. REVIEW ON LITERATURE

Virtual server farm will have an eccentric client traffic which requests a decent burden adjusting procedures. The total heap of the server farm is dispersed to a couple of servers, remaining servers will be in non-working. It naturally increment the reaction time expected to serve a solicitation. A portion of the heap adjusting procedures are clarified quickly beneath:

A. Round Robin Load Balancer (RR-LB):

This helps a champion among the best harsh kind weight balancing with a fundamental working framework. It sends the client deals to accompanying after VM with no priority learning. It initiates from the primary virtual machine and proceeds with the undertakings to be allocated to last virtual machine goes rundown.

B. Min-min Load Balancer (MM-LB):

In Min-min load balancer, the fastest reaction time is used to design the errand to proper virtual machine. It requires the most negligible [9] speculation taken to serve a sales a virtual machine. It checks the present response time to each and every virtual machine. Present reaction time is the actual required time to serve the end customer request. This framework explains the response time of virtual machine is lesser than the highest response time

C. Active Monitoring Load Balancer (AM-LB):

Dynamic Monitoring Load Balancer [8] is high grounded than the Round Robin Load Balancing. It viably screens the present weight condition of every virtual machine and distributes errand to the less stacked virtual machine. Calculation of virtual machines gives a graph of exact figuring. Exactly when information gets a requesting, it immediately makes an inquiry to the Active Monitoring Load Balancer (AMLB) to achieve the sensible virtual machine. Dynamic weight balancing looks at present errand assignment of each virtual machine and returns the virtual machine with least endeavor partition status. By then it extends its errand divide counter by 1[10].

D. Max-min Load Balancer (MM-LB):

At first for all the current errands are created to the framework and least achievement time for every one of them are determined, at that point among these undertakings the one which is having the finishing time, the most extreme is chosen and that is allocated to the relating machine. In the event that the assignment is a solitary long tedious

undertaking, at that point, Max-Min calculation runs little errands parallel alongside the long tedious assignment. It further considers the undertaking with most extreme finishing time though Min-Min considers the base culmination time at first. This calculation execution is influenced because of starvation where the undertakings are having the outrageous completing time will get performed first while abandoning the assignments having the base culmination time. [11]

E. Distributed dynamic priority-based algorithm:

It is utilized for adjusting the heap on occurrences adequately, improves the framework consistency, limits reaction time and builds the throughput. Dispensing the assets on virtual machines dependent on need accomplishes the better reaction time and handling time. Burden adjusting ensures all events in a hub in the systems to do the indistinguishable measure of work at any moment of time. Need based asset arrangement to improve the use of assets and lessening reaction time of cloud administrations. [6].

III. PROBLEM DEFINITION

Virtual Machine combination calculation should be planned so that there will be least vitality utilization, least infringement of SLA, productive virtual machine relocation and least number of dynamic in a given time. Virtual machine development causes administration level understanding encroachment since when a virtual machine is moved from one host to other it needs to trade its fundamental memory to the objective have and in the trade method the referenced CPU could not be passed on, as the virtual machine will be in a change state. Therefore, alongside power utilization, we have to ensure that the quantity of virtual machine relocation is negligible which will in reality diminish the SLA infringement.

An ideal virtual machine solidification approach will lessen vitality utilization and just as, it will diminish the negative impact on Quality of Service. To address these issues, virtual machine blend has been considered as a compartment squeezing issue in specific investigates, e.g., [3, 17, 19]. Then again, there are explores where VM union has been separated in discrete issues where receptacle pressing arrangement considered as one of the sub-issues of VM associations, i.e., virtual machine circumstance [1, 2, 4, 8, 9, 12, 15, 16]. There are several virtual machine allocation approaches are existed, namely Random Selection, Round Robin, Active monitor Min-min algorithms. Virtual machine union has been separated in four sub-issues and managed in explores are the followings:

- ✓ Identifying the underutilized virtual machine and give them as first preference to distribute the load to Virtual Machines.
- ✓ Determine the host capacity and allocate virtual machines from one data center to another datacenter.
- ✓ Identifying the usage of the resources based on the region
- ✓ Virtual machines should be flexible to update and upgrade the existing infrastructure.

IV. ARCHITECTURE

The complete cloud computing architecture is an inter-related hierarchical structure. The sub connected components of the virtual data center with highly flexible, scalable and available are the important essential highlights of the cloud vendor. Fig 1. demonstrates fundamental form of a virtual server farm. In this, virtual machines are the essential processing squares. Here the end clients allude too by and large conduct of the clients over the all-inclusive. Clients process the solicitation to the virtual machine through the base web. Web is in charge of making the irregular inertness conditions to the system. The complete de can be portrays as follows:

$$T_{total} = T_{latency} + T_{transfer} \tag{1}$$

Where $T_{latency} = \text{NetworkLate}$

$T_{transfer} = \text{Time to transfer a fixed load of data from origin to end}$

$$T_{transfer} = D / \text{Bandwidth per user} \tag{2}$$

$$\text{Bandwidth per user} = \text{Bandwidth total} / N_r \tag{3}$$

Where Band width total is the finished accessible transfer speed in the datacenter. Nor is the quantity of solicitations at present in transmission. Server farm Controller is the focal module to the datacenter. It deals with the framework asset alongside the VM organizations and the systems administration assets with in the Virtual Data Center A unique bit of Data Center Control specifically, the Virtual Mahine Load Balancer is Accountable for spreading the titanic proportion off load to the VM sin away that asserts a perfect respond time for the customer queries

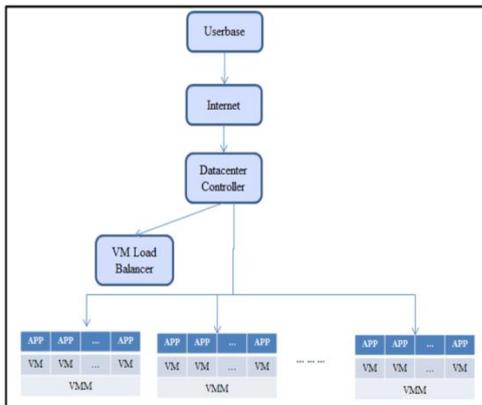


Fig. 1. Basic Virtualized Datacenter

V. PROPOSED ALGORITHM:

To improve the heap adjusting limit of the virtualized datacenter, we proposed Optimized Fuzzy Based Distributed Load Balancer. The heap status of the virtual machines are completely theoretically loose. The relationship between different parameters such as disk space, memory, bandwidth and the heap should be considered less to gauge the heap of virtualized server farm. So here proposed the Optimized Fuzzy Based Distributed Load Balancer that comparatively delivers preferred outcomes over the algorithm gave in literature.

A. Optimized Fuzzy Based Distributed Load Balancer (OFBD-LB):

When an end user request reaches the datacenter to allocate the virtual machine, the main objective is to improve the network traffic without any loss of load. So, each end user request is divided into three parameters: CPU, Memory and Disk Space. If two or many servers are available, then the load distributes according to any one of three policies.

- 1) Best Fit (BF): It is a good combination arrangement, choosing the server with the least accessibility of assets.
- 2) FirstFit (FF): It describes solidification strategy picking the very first applicant server from a pre-requested rundown.
- 3) WorstFit (WF): It describes heap adjusting strategy, choosing the server with the most elevated accessibility of assets.

Signifying with CPU_s , $RAMS_s$, $DISK_s$ assets left on servers after the effective Conceivable arrangement to process the load to virtual machine.

- 1) No server is available: The load demand to VM will be rejected.
- 2) One server is available: Requests should be processed into queue based.
- 3) More than one Server is available: Requests should be processed in to current policy

Algorithm 1 provides the simple flow of Virtual machine consolidation. Initially the hosts are created. Then collect the real cloud data. Depends on the collective data, virtual machines and cloudlets are created. Finally assign the virtual machines to host and cloudlets are assigned to virtual machines. For every specific time collect the health status of all the virtual machines and stores the metrics into load balancer and sprays the load to multiple servers in a sequential order.

Algorithm1. Simple VM consolidation:

1. Given no. of cloudlet;
2. Concatenate with current cloud information;
3. Virtual machine is generated and dedicated to hosts;
4. Cloudlet is generated and allocate to virtualmachines;
5. For definite time interval
6. Get health conditions of VMs;
7. Identify the under-utilized and available host through metrics;
8. Distribute load to available VMs;
9. Monitor the metrics properly;
10. End

Algorithm 2: Optimized Fuzzy Based Distributed Load Balancer (OFBD-LB):

1. Calculate the current allocation counter, $vc(i)$;
2. Var User Requests, r ;
3. Given Number of virtual machines, n ;
4. Variables:
5. RAM Usage, $R_v = \{ \text{veryLow, Least, Moderate, High,} \}$



- VeryHigh }
6. BW Usage , $BW_v = \{veryLow, Least, Moderate, High, VeryHigh \}$
 7. DS Usage , $DS_v = \{veryLow, Least, Moderate, High, VeryHigh \}$
 8. VM statue , $S_v = \{veryBusy, Busy, Moderate, Aavailable, HighAavailable \}$
 9. Request Processing:
 10. Datacenter receives, r from end-user through internet
 11. Datacenter queries the Optimized Fuzzy Based Distributed Load Balancer
 12. For every V_m
 - a. Identify the existing R_v, BW_v, DS_v
 - b. Calculates the available R_v, BW_v, DS_v for the requested user r
 - c. $VM\ status(i) = FIS(R_v, BW_v, DS_v)$
 13. end
 14. Pick the least loaded VM, $vm\ out = minimum(VM\ status)$.
 15. OFBM-Load Balancer sends request to DCC.
 16. Datacenter update $Vt(vmout) = Vt(vmout) + 1$;
 17. When request sent,
 18. a. Datacenter update, $vt(vmout) = vt(vmout) - 1$;
 19. Continue from step2
 20. End

Algorithm2 at the moment that server ranch gets customer requests, it advances to the advantage necessities of that sales to the OFBD-LB to get the suitable VM where the errand can be arranged. OFBD-LB by then channels the memory, data trade point of confinement and circle space status of each VM (step 12.a). By then asset necessities are added with current asset use to get the regular asset status of each VM (step 12.b). After that the fresh estimation of memory, transmission utmost and plate space are changed over to fleecy respect. We develop a delicate enlistment structure (FIS) with 75 rules. For instance, if memory use is very Low, Bandwidth use is Low, and disk space use is Low, by then the virtual memory load status is very Busy. We shut the hard and fast standard table by virtue of room obstacle. OFBD-LB assesses the stack status, of each VM utilizing Fuzzy Inference Rules (step 12.c). Our story check picks the least Busy VM and sends it to the Data Center Controller (mastermind 15-16). Server farm Controller produces the present undertaking task counter of the picked VM (Step 19). Right when the errand has been done, the DCC decreases the undertaking task counter as appeared in stage 18. After that DCC sets itself up to dole out another client demand (step-19).

VI. EXPERIMENTAL RESULTS

In real time the highly demand internet application is highly impossible to perform as it requires a large-scale infrastructure. We used cloudsim simulator to analazye the performance and brings the efficiency in the proposed algorithm. Figure 2,3,4 describes the efficiency between advanced virtual machines and load balancer virtual machines to distribute the load across the virtual machines. Figure 5 describes the consuming of virtual macines that can be distributed to load.

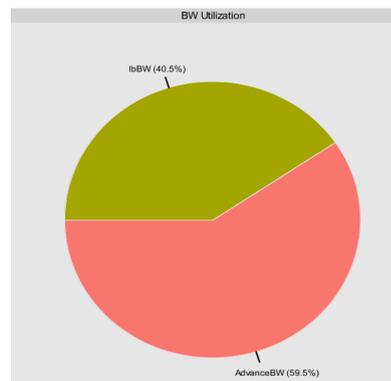


Fig 2. Band Width Utilization

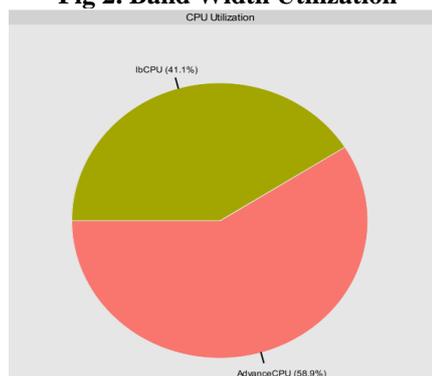


Fig 3. CPU Utilization

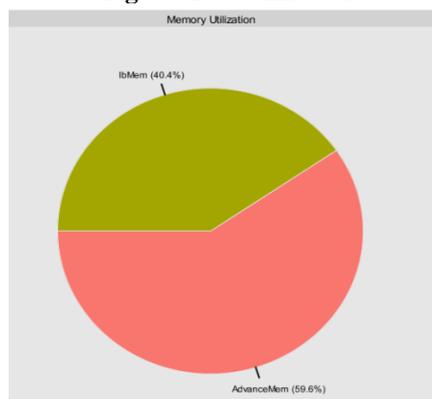


Fig 4. Memory Utilization

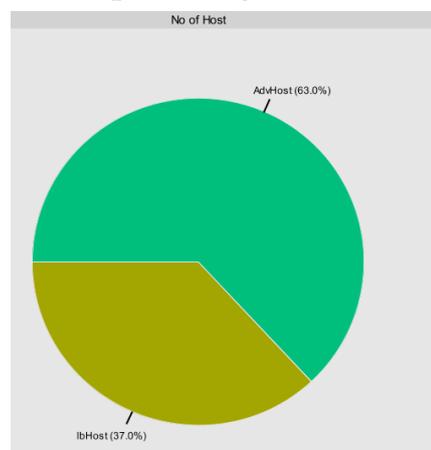


Fig 5. Mean of the VMs

VII. CONCLUSION

This paper has proposed an Optimized Fuzzy Based Distributed Load Balancer Algorithm, which compares the existing algorithms performance and brings the efficiency over the virtualized datacenter. We mainly focused three parameters such as CPU, MEMORY and DISK to handle the load to virtual machines. For future work, the resources can be utilized properly and can be save energy of the hard ware by implementing an efficient algorithm to the Data Center.

REFERENCE

1. C. J. Burgess, C. Reilly, L. Steward-Harrison, F. Balouch, and P. J. Lewindon, "Utility of proactive infliximab levels in paediatric Crohn's disease," *Arch. Dis. Child.*, vol. 104, no. 3, pp. 251–255, 2019.
2. N. Leavitt, "Really Ready," *Growth Lakel.*, vol. 42, no. January, pp. 15–20, 2009.
3. W. S. Setup, "Setting up your AWS account," no. November, pp. 1–10, 2013.
4. J. Dubinsky, R. D. Howell, T. N. Ingram, D. N. Bellenger, and S. Job, "Salesforce Socialization Unique Characteristics of the," vol. 50, no. 4, pp. 192–207, 2014.
5. R. Barga, V. Fontama, and W. H. Tok, *Predictive Analytics with Microsoft Azure Machine Learning*. 2015.
6. O. Krieger and P. Mcgachey, "Enabling a Marketplace of Clouds : VMware ' s vCloud Enabling a Marketplace of Clouds : VMware ' s vCloud," *Challenges*, pp. 103–114.
7. D. Minarolli and B. Freisleben, "Virtual machine resource allocation in cloud computing via multi-agent fuzzy control," *Proc. - 2013 IEEE 3rd Int. Conf. Cloud Green Comput. CGC 2013 2013 IEEE 3rd Int. Conf. Soc. Comput. Its Appl. SCA 2013*, pp. 188–194, 2013.
8. A. Wickremasinghe, R. N. Calheiros, and R. Buyya, "CloudAnalyst: A cloudsim-based visual modeller for analysing cloud computing environments and applications," *Proc. - Int. Conf. Adv. Inf. Netw. Appl. AINA*, pp. 446–452, 2010.
9. X. Kong, C. Lin, Y. Jiang, W. Yan, and X. Chu, "Efficient dynamic task scheduling in virtualized data centers with fuzzy prediction," *J. Netw. Comput. Appl.*, vol. 34, no. 4, pp. 1068–1077, 2011.
10. S. A. E.H. Mandani, "Mamdani1999.Pdf," vol. 1. 1999.
11. R. Buyya, C. S. Yeo, S. Venugopal, J. Broberg, and I. Brandic, "Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility," *Futur. Gener. Comput. Syst.*, vol. 25, no. 6, pp. 599–616, 2009.