

An Optimal Solution in cloud computing for Load Balancing using Genetic Algorithm

Shweta Sharma, Shweta Pandey



Abstract: Cloud computing is a research trend which bring various cloud services to the users. Cloud environment face various challenges and issues to provide efficient services. In this paper, a novel Genetic Algorithm based load balancing algorithm has been implemented to balance the load in the network. The literature review has been studied to understand the research gap. More specifically, load balancing technique authenticate the network by enabling Virtual Machines (VM). The proposed technique has been further evaluated using the Schedule Length Runtime (SLR) and Energy consumption (EC) parameters. Overall, the effective results has been obtained such as 46% improvement in consuming the energy and 12 % accuracy for the SLR measurement. In addition, results has been compared with the conventional approaches to validate the outcomes.

Index Terms: About four key words or phrases in alphabetical order, separated by commas.

I. INTRODUCTION

The applications in a cloud computing hosted in an internet based approach where several these files are served. It is an emerging technology which is efficient for parallel computing and large scale distribution .A framework in cloud computing has been established to enable the applications so that it works on the virtualized resources and it is easily accessible by following standards and protocols. It provides an extremely flexible way of accounting and infrastructure resources and services, which can be arranged in a typical way, depending on the end user's requirement. Over the last decade, cloud computing has a solid platform for spreading the web, providing virtual hardware and software infrastructure over the Internet, thanks to its exponential growth [1]. Cloud uses high-speed internet to deploy remote PCs or Data centres from local or custom PCs. The cloud service provider's use cloud computing service individually or industrially, anywhere in the world. A paid model for Cloud's student service has attracted more users for better utility computing [2]. Another reason why companies and end users are involved is reducing costs, ensuring that they are depreciated. Providing QoS (better and faster service) and adapting to end-users 'timely resource requirements is one of the major cloud service providers' problems.A computational

style that provides massively scalable IT capabilities, such as Internet access to many international clients across the globe. Thus, providing QoS (better and faster service) and adapting to end-users 'timely resource requirements is one of the major cloud.

Cloud computing is a development of the previous distributed model such as Grid Computing..The goal behind its expansion is to transform the computing resource for its service [3]. Cloud computing is based on the concept of software, application, information, management and maintenance service. Cloud computing includes computation of various models. Cloud computing is not limited to the services provided to users processing power and storage, web hosting , database support and more.

Cloud computing caters to the following needs of the user.

- Dynamism- It is provided in thecloud computing which facilitates, and demanding as per need of the users.Virtual Machines plays an important role in cloud environment
- Abstraction – End users can avail the facility of abstraction in the cloud environment. In addition, they don't need to cater with web security and plug-ins.
- Resource Sharing – An optimised utilisation of resources possible through this feature of the cloud environment.

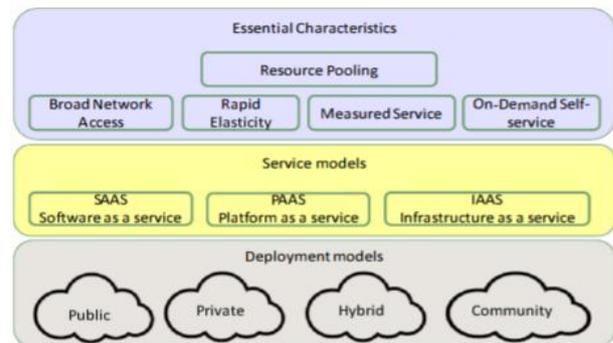


Fig.1 Cloud Computing Model [2]

Any cloud service provider (CSP) serves various services such as hardware, computing, and software services. Cloud Service Provider (CSP) is solely responsible to provide QoS. All the features maintained in the cloud computing requires efficient data centres, then cloud computing can be a glorious future for almost the next decades. Apart from that, still there are many problems in the cloud environment such as Load balancing. Load balances can be shared evenly at multiple distributed intersections in a distributed environment. The balance of load in a distributed environment must also take into account two key issues.

Revised Manuscript Received on October 30, 2019.

* Correspondence Author

Shweta Sharma*, Asst. Prof, Goswami Ganesh Dutta Sanatam, Information Technology.

Shweta Pandey, Research Scholar, Desh Bhagat University ,CSE, Mandi Gobindgarh

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

An Optimal Solution in cloud computing for Load Balancing using Genetic Algorithm

These are resource allocation, and task scheduling. The energy consumption in a cloud environment is other interesting fact which needs to cater using various metaheuristic approaches [4]. Let us consider a given example:-

E_c be taken as a consumption of energy in a given task or activities and E_t be the energy used for the transmission of data. The cloud environment assumes consumption of energy in the following formula

$$P_{cloud} = E_t + Db/BW \quad (1)$$

Where, Db is the data bytes and BW is the bandwidth of the network

The generated output is given as G which is assumed to be smaller than Db . Noe, smart devices perform activities which would be given as follows:-

$$D_{smart} = E_c * I + E_t * Db/BW \quad (2)$$

Here, I is the numeral for the execution of various tasks. Thus saved energy will be given as:-

$$E_s = E_t * Db/BW - E_c * I - E_t * Db'/BW \quad (3)$$

$$E_s = [E_t * Db/BW] - E_c * I - [E_t * (Db * F)]/BW \quad (4)$$

Since energy saved in the above equation appears positive which offloads the task. Load balancing in cloud computation is an important task. Load balancing algorithm can be centralized, or distributed, and static or dynamic, with their advantages and disadvantages. It is easy to apply and control the static load balancing scheme. On the contrary, it is difficult to apply dynamic algorithms, but it is best adapted to the heterogeneous environment [5]. Unlike the centralized algorithm, all allocation and scheduling decisions are performed with a single node. Load balancing algorithm in shared mode can be established to perform various activities together which can be balanced using the fuzzy set approach [6]. These intersections are interconnected to achieve load balancing. The benefits of the distributed algorithm is that it provides better error tolerance, but requires a higher replication. Specifically, distributed load balancing can be both co-operative and non-co-operative.

The load balancing also helps to schedule activities over different nodes in the cloud environment. In addition, load balancing is a typical optimization method. It provides the distribution of overall workload equally provided in the system for each system to work. When a special key is overloaded, it should be neutralized by using a loaded node. In addition, acyclic graphs has been utilized to balance the load [7]. Hence, by customizing load balancing algorithms, service providers can maintain their resources and help manage QoS and reducing the response time.

II. RELATED WORK

Berwal and Kent 2015 proposes an efficient algorithm to analyse different load balancing algorithms to define a new stronger algorithm. Its purpose is to offer develop a hybrid

algorithm that combines features impartial random selection algorithms and weighted round robin preparation algorithm.

Mishra 2012 tried to solve the problem of load balancing using the ant colony optimization. Heuristic algorithm using the mixed colony optimization concept. The authors stress that the pheromone update is proven to be an optimally balanced load. However, the proposed technique does not solve the problem of high tolerance, and more work is being done to fulfil these aspects.

Kumar 2019 discuss the various challenges of the cloud environment. The problems related to various load balancing techniques has been discussed to understand the gap in the previous studies. Authors also presents a comprehensive and comparative study of various load balancing algorithms. The study also portrays merits and demerits of all the latest schemes that can be used to improve load balancing algorithms.

Xu et al 2013 discusses the load balancing model for the public cloud. Thus, the model is shared in Cloud Global where multiple divisions of clouds was considered in a cloud environment. The main controller selects the appropriate divisions by two techniques such as balancing type and balancing strategy.

Lu et. al. 2011 presented a 10in-Idle-Queue (JIQ) algorithm for load balancing in a distributed systems which offer different advantages. In particular, the authors state that the JIQ algorithm does not communicate through dispatchers and processors on the arrival of jobs. Finally, the authors suggest that an extension of the basic JIQ algorithm must face very high loads by using only local server load information.

Abed and Younis 2019 presents a Telemetry transmission to messaging message (MQTT) and Hypertext Transfer Protocol (HTTP). This material aims to make efficient use of resources. Thus, cloud load balancing is a combination of two types of algorithms: dynamic algorithm and static algorithm (weighted round robin). The results of the proposed algorithm show improvement in resource utilization, productivity growth, and reduced response time.

Priya et. al. 2019 introduces an integrated resource allocation and load balancing algorithm which is efficient for managing cloud service delivery. The method establishes a Fuzzy-based Multidimensional Resource Scheduling model to ensure resource planning efficiency in cloud infrastructure. A load balancing algorithm balance the load using simulation of data. The simulation analysis shows that the method improves resource efficiency by 7% and also reduces the reaction time by 35.5% compared to the last works.

Mohanty 2019 presents a cloud computing based technique which is a jargon in the era of information technology. It acts as a metaphor for the Internet. However, it has some problems with providing resources, security, spreading events, server consolidation, virtual machine migration. Load balancing is one of the most critical challenges in the industry. Here is the basic purpose of balancing the load is to minimize reaction time and improve overall system performance. In this article, a JAYA algorithm is used to balance load into a cloud that uses less control parameters and provides a better optimized result.

Comparisons are made with other evolutionary approaches to observe the effectiveness of the proposed algorithm. Puthal et. al. 2018 proposes a new load balancing technique to verify the network edges and find the least charged node for assignment.

The proposed load balancing technique is more efficient than other existing approaches in finding the least allocated nodes assignment. The proposed approach not only improves load balancing efficiency; also strengthens security by validating destination nodes. The proposed load balancing technique is more efficient than other existing approaches in finding the least allocated node. The proposed approach not only improves load balancing efficiency; but also reduces computation time.

III. PROPOSED WORK ARCHITECTURE

The proposed architecture balances the load using the Genetic Algorithm and other points on the Virtual Machine. They are given as follows:-

- The cost values differ as three VMs require to process the applications.
- The application applied on the parent task executed
- The ranking of the children and their execution depends upon it.
- Network has been balanced in the proposed architecture.

Every job has been initiation time and completion time for the job. The allocation of VM for one job while task has not been assigned to another job before finishing the prior task. The jobs has been executed through the parallelization. In addition, the prioritization can be done through the Genetic Algorithm

The basic steps of the Genetic Algorithm are given as follows:-

- Step 1: To reset random population having chromosomes.
- Step 2: To compute suitability function in the population.
- Step 3: To develop the novel population of chromosomes.
- Step 4: To choice parent chromosomes for greatest fitness function.
- Step 5: To do crossover to have the duplicate of parents.
- Step 6: To do the alteration to mutate novel off springs or child.
- Step 7: To place novel offspring in the people.
- Step 8: To repeat the steps to get a fulfilled solution.

Algorithm 1: Load Balancing

```

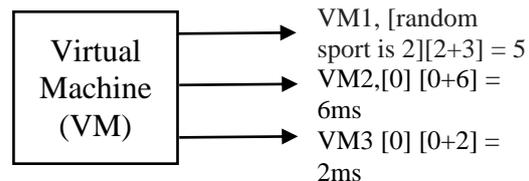
Genetic algorithm function GA ( )
{
    Initialize the population size;
    Calculate the fitness function of GA;
    Selection function
    - Current function to select a particular value;
    Crossover function - Value may be crossed;
    Mutation function - Mutated value;
    While (fitness value of fitness function termination criteria)
    {
        Function set_priority (jobs, Loadcosting) {initialize Load
            = 0; Initialize total Load = 0;
            initialize priority_order = [];
            3. for j = 1: Job
            4. for i = 1: HighConnection.count
            5. Load = Load + Loadcosting (j, i);
            6. End for
            7. End for
            8. Total Load
            = Load/(job * (HighConnection.count)); Priority_order
            = Zeros (1.numel (Jobs)); False_Fitness value_Source = false;
            10. Mutation function = (Jobs * Power Jobs);
    }
}
    
```

11. For $1 \leq i \leq \text{High Connection}$: Fitness Value / High Connection
12. Crossover Value = $\sum \text{EnergyNodes}(k)/i$;
13. if (Fitness value * random () < Load * random ())
14. False_Crossover value = true;
15. end if
16. if False_Crossover value == true
17. temp = (Priority_order[last];
18. Mutate (Priority_order one position);
19. else
20. Priority_order [K] = Mutation Value [i];
21. end if

Every activity or task has a start time and completion time. The start time denoted by T_s and completion time denoted by T_c . The priority of the various tasks has been ordered in the specified area. The order of the jobs has been set by using Cuckoo fitness function. All the eggs has been broke if there is a possibility of faultiness in any egg. The threshold value has been taken randomly where average cost depends upon energy consumed by the performing every job. If the condition to be said satisfied then it is positioned at its place otherwise it is positioned at the last place. The number of proceeding order are given as follows:-

Job – 2,5,3,6,4

The precedence order specifies that 2nd job transfers the data as follows:-



The second job has been scheduled which increase the processing cost such as VM1, VM2, and VM3.

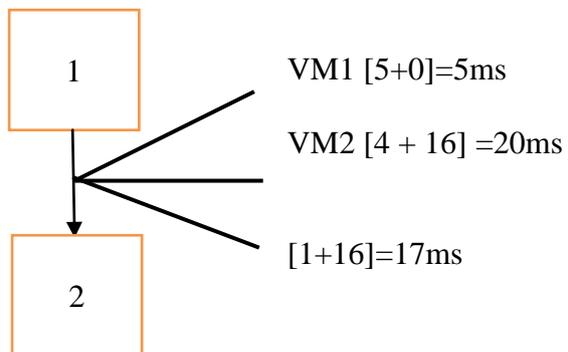


Table 1. Cost of VM for different jobs

Job Number	Cost of Job at different VM		
	VM1	VM2	VM3
1	2	5	2
2	5	3	5
3	2	4	3
4	3	2	6
5	5	2	3
6	7	2	7
7	12	12	12
8	16	7	7
9	3	2	4
10	3	1	6

The job structure specified like considering the job 2 first and then further executed. The allocation of job 2 can be done to any VM, following specifications requires to meet.

- a) There must be a slot provided to process the job request in a VM
- b) The cost of the VM has been compared to other VMs.
- c) Less energy consumption availed through the low time consumption.

IV. RESULTS AND ANALYSIS

The results has been evaluated based on the following parameters such as :

- Schedule Length Runtime (SLR): it is defined as the completion of the jobs at all the different virtual machines.

$$SLR = \sum_{m=0}^k Execution_time$$

In this equation, k represents the total number of jobs.

- Schedule Length to VM (SLV): It is defined as the ratio of SLR to the time at different virtual machines
 $SLV = SLR / \text{Jobtime of 1st VM}$
- Consumption of Energy (E_c) : It is the consumption of total energy for the execution of all tasks and activities.

Table 1. Proposed SLR

Total number of jobs in job model	Proposed SLR in ms	SLR Dag
100	0	0
300	400	500
700	800	900
1100	1200	1350
1500	1400	1550

Table 1. presents the proposed SLR which is compared with the conventional approach [17].It is seen that the average SLR attained through the proposed approach is 760 ms while that of other technique, it is 860. Thus, it seen that improved results has been obtained which reduces the consumption of time.

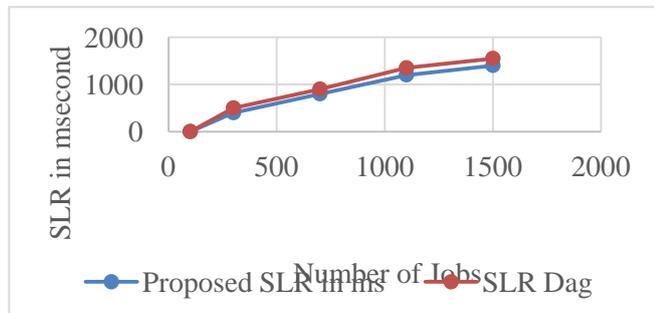


Fig.2 Proposed SLR in comparison to past approach [17]
 Fig.2 represents the comparison of the proposed and conventional approach. It has been analysed that for the jobs less than 500 the acquired SLR for the proposed approach is

400 while that of conventional approach, it is 500. Thus, better results has been attained through the proposed technique. The average obtained accuracy is $\frac{860-760}{860} \times 100 = 12\%$.

Table 2. Proposed Energy Consumption

Total number of jobs in	Proposed Energy Consumption	Energy Consumption through Dag
100	50	120
400	130	230
700	170	300
1100	225	420
1500	240	440

Table 2. represents the proposed energy consumption which is further compared with the conventional approach [17].It is seen that the average energy consumption attained through the proposed approach is 163 while that of other technique, it is 302. Thus, it seen that improved results has been obtained which reduces the consumption of energy. The effective results has been obtained through the proposed approach.

Fig.2 represents the comparison of the proposed and conventional approach. It has been analysed that the results achieved through the proposed approach follows an increasing trend and then remains constant. However, results obtained through the conventional approaches remains steady from above 1500 jobs. The average obtained accuracy is $\frac{302-163}{302} \times 100 = 46\%$.

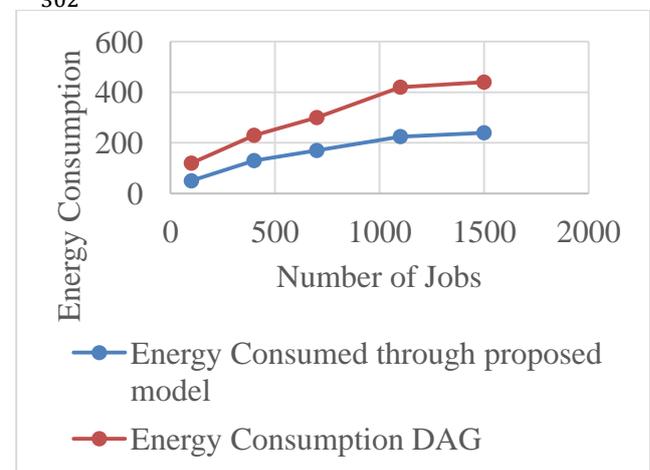


Fig.2 Proposed Energy Consumption in comparison to energy consumption acquired through the conventional approaches [17]

V. CONCLUSION

This paper presents a robust model based on the Genetic Algorithm to combat the issues of load management over the Virtual Machines (VM).In this paper, load balancing methods balances the load in the cloud environment using metaheuristic approaches.

The proposed structure has been further evaluated using the SLR and EC. An efficient results has been obtained in a hostile environment, where VMs has been settled down using the start time and completion time.

The proposed architecture has been further compared with the conventional approaches. The attained results has been improved in terms of energy consumption and time measurement for balancing the load.

REFERENCES

1. Kumar, D.S., Raj, D.P. and George, E., 2018. A literature review on load balancing mechanisms in cloud computing. *International Journal of Advanced Research in Computer Science*, 9(1).
2. Mishra, S.K., Sahoo, B. and Parida, P.P., 2018. Load balancing in cloud computing: a big picture. *Journal of King Saud University-Computer and Information Sciences*.
3. Chawla, A. and Ghumman, N.S., 2018. Package-Based Approach for Load Balancing in Cloud Computing. In *Big Data Analytics* (pp. 71-77). Springer, Singapore.
4. Dam, S., Mandal, G., Dasgupta, K. and Dutta, P., 2018. An ant-colony-based meta-heuristic approach for load balancing in cloud computing. In *Applied Computational Intelligence and Soft Computing in Engineering* (pp. 204-232). IGI Global.
5. Ragmani, A., El Omri, A., Abghour, N., Moussaid, K. and Rida, M., 2018. A performed load balancing algorithm for public Cloud computing using ant colony optimization. *Recent Patents on Computer Science*, 11(3), pp.179-195.
6. Kumar, N. and Shukla, D., 2018. Load Balancing Mechanism Using Fuzzy Row Penalty Method in Cloud Computing Environment. In *Information and Communication Technology for Sustainable Development* (pp. 365-373). Springer, Singapore.
7. D. Kliazovich, J. E. Pecero, A. Tchernykh, P. Bouvry, S. U. Khan and A. Y. Zomaya, "CADAG:Communication-AwareDirectedAcyclic Graphs forModelingCloudComputingApplications,"2013 IEEE Sixth International Conference on Cloud Computing, SantaClara, CA, 2013, pp. 277-284.
8. M. Berwal and C. Kant, "Load Balancing in Cloud Computing," vol. 6, no. 2, pp. 52-58, 2015.
9. R. Mishra, "Ant colony Optimization: A Solution of Load balancing in Cloud," *Int. J. Web Semantic Techno!*, vol. 3, no. 2, pp. 33-50, Apr. 2012.
10. Kumar, P. and Kumar, R., 2019. Issues and Challenges of Load Balancing Techniques in Cloud Computing: A Survey. *ACM Computing Surveys (CSUR)*, 51(6), p.120.
11. Hota, A., Mohapatra, S. and Mohanty, S., 2019. Survey of Different Load Balancing Approach-Based Algorithms in Cloud Computing: A Comprehensive Review. In *Computational Intelligence in Data Mining* (pp. 99-110). Springer, Singapore.
12. G. Xu, J. Pang, and X. Fu, "A load balancing model based on cloud partitioning for the public cloud," *Tsinghua Sei. Technoi.*, vol. 18, no. 1, pp. 34-39, 2013.
13. Y. Lu, Q. Xie, G. Kliot, A. Geiler, J. R. Larus, and A. Greenberg, "Join-Idle-Queue: A novel load balancing algorithm for dynamically scalable web services," *Peiform. Evai.*, vol. 68, no. 11, pp. 1056- 1071, Nov. 2011.
14. Abed, M.M. and Younis, M.F., 2019. Developing Load Balancing for IoT-Cloud Computing Based on Advanced Firefly and Weighted Round Robin Algorithms. *Baghdad Science Journal*, 16(1), pp.130-139.
15. Priya, V., Kumar, C.S. and Kannan, R., 2019. Resource scheduling algorithm with load balancing for cloud service provisioning. *Applied Soft Computing*, 76, pp.416-424.