International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

TASK SCHEDULING AND DYNAMIC RESOURCE ALLOCATION USING HEURISTICS ALGORITHM IN CLOUD COMPUTING

S. Jaya Surya[1]*, Mr. R. Sathish Kumar*[2]

[1]*Master of Computer Application, krishnasamy college of engineering &Technology, Cuddalore*
[2]*MCA. M.Phil. Assistant Professor, Master of Computer Application, krishnasamy college of engineering &Technology, Cuddalore*

ABSTRACT

Cloud computing has emerged as a technology that grease tasks by the dynamic allocation of virtual machines. Load balancing is a key aspect of cloud computing and avoids the situation in which some nodes become overloaded while the others are idle or have little work to do. Load balancing can improve the Quality of Service (QoS) metrics, including response time, cost, throughput, performance and resource utilization. Task Scheduling highly contributes to load balancing, and scheduling tasks much adheres to the requirements of the Service Level Agreement (SLA), a document offered by cloud developers to users. Important Task sheduling parameters such as Deadline are addressed in the LB (load balancer) algorithm. The proposed algorithm is aimed to optimize resources and improve Load Balancing in view of the Quality of Service (QoS) task parameters, the priority of VMs(virtual machine), and resource allocation. To analyze the performance of heuristic-based algorithms, the simulation is carried out and the results are produced efficiently. This heuristic always allocates tasks to virtual machines arbitrarily and then checks for the next available machine. The proposed LB algorithm addresses the stated issues and the current work gap based on the literature's findings. Results showed that the proposed LB (Load Balance) algorithm and heuristic algorithm results in an average of resource utilization compared to the existing Dynamic LBA algorithm. It also achieves good performance in terms of less Execution time and Makespan.To analyze the performance of heuristic-based algorithms, the simulation is carried out and the results are presented in detail. This heuristic always allocates tasks to virtual machines arbitrarily and then checks for the next available machine using genetic algorithm.

Keyword *- Quality of Service (QoS), LB (Load Balance),dynamic resource),virtual machine(vm).*

1. LITERATURE SURVEY

Sambit Kumar Mishra a[1]Scheduling or the allocation of user requests (tasks) in the cloud environment is an NP-hard optimization problem. According to the cloud infrastructure and the user requests, the cloud system is assigned with some load (that may be underloaded or overloaded or load is balanced). A brief explanation of considered performance parameters in the literature and their effects is presented in this work. To analyze the performance of heuristic-based algorithms, the simulation is carried out in CloudSim simulator and the results are presented in detail.

Mayanka Katyal[2]Cloud Computing is a new trend emerging in IT environment with huge requirements of infrastructure and resources. Load Balancing is an important aspect of cloud computing environment. Efficient load balancing scheme ensures efficient resource utilization by provisioning of resources to cloud user's on-demand basis in pay-as-you-say-manner. Infrastructure in the Cloud is made available to the user's On-Demand basis in pay-as-you-say-manner.

2. PROPOSED SYSTEM

Task Scheduling is a process that highly relates to workload balancing. Users send requests, the task is submitted through a cloud broker; this is where researchers should focus on providing an efficient algorithm. The proposed algorithm should efficiently submit jobs to appropriate VMs following essential parameters such as deadline to maintain a high quality of services and ensuring the requests sent by users are executed and completed within these specific requirements provided in the Service Level Agreement (SLA) document. The user sends requests via the Internet. These requests are stored in Virtual Machines (VMs), and CSP in every delivery model must maintain the QoS by ensuring the users' requests can be executed and completed within a specific deadline. This process depends highly on the scheduling policy's efficiency (Data Broker) which should be programmed to result in a high technique for balancing workload among the machines and servers. Efficient scheduling and utilization of resources can be achieved by designing and developing a dynamic load balancer (LB).A meta-scheduler that uses a multi-objective genetic algorithm (MO-GA) to findthe best scheduling based on three goals: Reducing energy consumption and emissions and (QoS) for the client, all with on-time termination applications and the constraints of the model. Although Task Scheduling is one of the main goals of providing an efficient Load Balancing and improving performance, most researchers focus on one or two aspects. For example, to enhance Load Balancing and considers few Task Scheduling parameters. Thus, only a few metrics are taken into consideration to improve the overall performance.The proposed algorithm is aimed to optimize resources and improve Load Balancing in view of the Quality of Service (QoS) task parameters, the priority of task, and resource allocation.They have performed an experimental evaluation of eminent state-of-the-art static tasks scheduling heuristic algorithm. These algorithms have been critically investigated in terms of resource utilization, task execution time, throughput, and energy consumption. Moreover, individual load-imbalance is computed and compared.

3. ARCHITECTURE

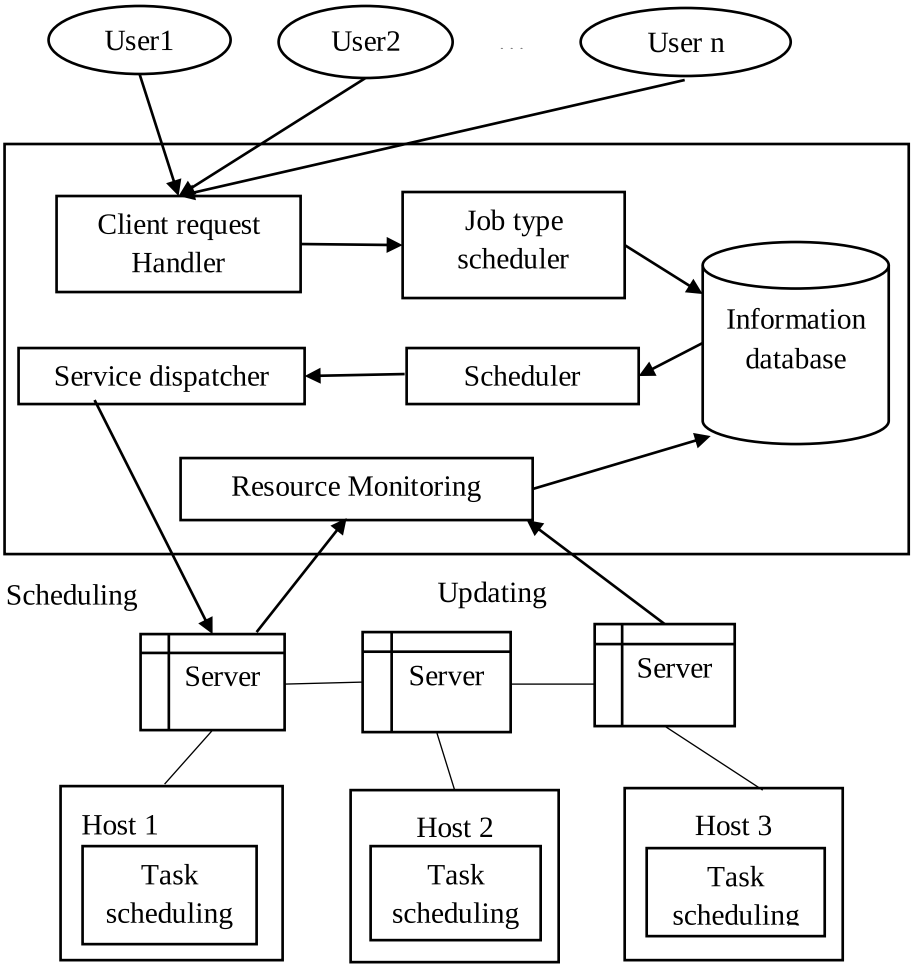

Figure:Dynamic Resource Allocation

4. METHODOLOGY

The proposed LB (Load Balance) algorithm aims to improve the cloud's performance by considering both aspects of Task Scheduling and Load Balancing. It utilizes all available CPUs in machines and schedules tasks appropriately to reduce Makespan, Execution Time, and maximize resource utilization. Below are the assumptions made in the proposed algorithm :

- One-to-many cloudlets (also known as task or user request) per Virtual Machine (VM).
- Cloudlets arrive in a random order (Arrival Time)
- Each Cloudlet has a length, a time to complete known as Deadline (included in Service Level Agreement document), a completion time, and finally, the arrival time.
- The proposed algorithm checks the completion time for each workload (a total of cloudlets) against the Deadline.

- If there is any violation, whereby the completion time exceeds the Deadline, then the proposed algorithm will reconfigure the VM's priority based on its CPU. If it is in a successful state, the cloudlets get scheduled else; it will migrate the VM's workload.
- Expected Completion Time is calculated by taking the cloudlet length (also known as Million instruction per second (MIPS)) and dividing it by Virtual Machine MIPS (also known as CPU).
- Initially, all VMs share an equal portion of the available CPU; then, it is reconfigured based on the violation status. The CPU is set to its full utilization in the proposed algorithm.

The proposed algorithm gives better output in terms of energy efficiency, cost and also all the VMs are allocated tasks. The cost is improved by making the following changes in the cost function:

Cost=w1*ProcessingCost()*(FreePes () /NumberOfPes())+w2*Delaycost()

w1 and w2 are predefined weights, pes: processing elements.

5. CONCLUSION

The primary objective of the proposed algorithm is to minimize MakeSpan and improve fitness function. Improving the Load Balancing process through Task Scheduling can result in efficient utilization of cloud resources. The objective of this proposed work was to provide an enhanced Load Balancing algorithm. Results proved that our algorithm reduces Makespan and provide efficient resource utilization of compared to existing Dynamic LBA (Load Balancing Algorithm). It also shows that the proposed heuristic algorithm can function in a dynamic cloud environment where user requests arrive in random order and where there are many changes in the length of the user requests. In the future, various other metrics like throughput, average time, resource utilization, waiting time, etc. can be considered. In the future, authors will work to optimize the cloud resources further and enhance cloud-based application performance, such as considering more SLA (Service Level Agreements) parameters. For example, the algorithm will be tested based on the number of violations and the migration count for better performance

REFERENCES

[1] H. Shukur, S. Zeebaree, R. Zebari, D. Zeebaree, O. Ahmed, and A. Salih, ''Cloud computing virtualization of resources allocation for distributed systems,'' J. Appl. Sci. Technol. Trends, vol. 1, no. 3, pp. 98–105, Jun. 2020, doi: 10.38094/jastt1331.

[2] M. Agarwal and G. M. Saran Srivastava, ''Cloud computing: A paradigm shift in the way of computing,'' Int. J. Mod. Educ. Comput. Sci., vol. 9, no. 12, pp. 38–48, Dec. 2017, doi: 10.5815/ijmecs.2017.12.05.

[3] N. Zanoon, ''Toward cloud computing: Security and performance,'' Int. J. Cloud Comput.: Services Archit., vol. 5, no. vol. 5, nos. 5–6, pp. 17–26, Dec. 2015, doi: 10.5121/ijccsa.2015.5602.

[4] C. T. S. Xue and F. T. W. Xin, ''Benefits and challenges of the adoption of cloud computing in business,'' Int. J. Cloud Comput.: Services Archit., vol. 6, no. 6, pp. 1–15, Dec. 2016, doi: 10.5121/ijccsa.2016.6601.

[5] D. A. Shafiq, N. Jhanjhi, and A. Abdullah, ''Proposing a load balancing algorithm for the optimization of cloud computing applications,'' in Proc. 13th Int. Conf. Math., Actuarial Sci., Comput. Sci. Statist. (MACS), Dec. 2019, pp. 1–6, doi: 10.1109/MACS48846.2019.9024785.

[6] S. K. Mishra, B. Sahoo, and P. P. Parida, ''Load balancing in cloud computing: A big picture,'' J. King Saud Univ.–Comput. Inf. Sci., vol. 32, no. 2, pp. 149–158, 2020, doi: 10.1016/j.jksuci.2018.01.003.

[7] I. Odun-Ayo, M. Ananya, F. Agono, and R. Goddy-Worlu, ''Cloud computing architecture: A critical analysis,'' in Proc. 18th Int. Conf. Comput. Sci. Appl. (ICCSA), Jul. 2018, pp. 1–7, doi: 10.1109/ICCSA.2018.8439638.

[8] A. Jyoti, M. Shrimali, and R. Mishra, ''Cloud computing and load balancing in cloud computing -survey,'' in Proc. 9th Int. Conf. Cloud Comput., Data Sci. Eng. (Confluence), Jan. 2019, pp. 51–55, doi: 10.1109/confluence.2019.8776948.

[9] S. H. H. Madni, M. S. Abd Latiff, M. Abdullahi, S. M. Abdulhamid, and M. J. Usman, ''Performance comparison of heuristic algorithms for task scheduling in IaaS cloud computing environment,'' PLoS ONE, vol. 12, no. 5, May 2017, Art. no. e0176321, doi: 10.1371/journal.pone.0176321.

[10] M. Adhikari and T. Amgoth, ''Heuristic-based load-balancing algorithm for IaaS cloud,'' Future Gener. Comput. Syst., vol. 81, pp. 156–165, Apr. 2018, doi: 10.1016/j.future.2017.10.035.