

# Hybrid Heuristic Load Balancing Algorithm For Resource Allocation In Cloud Computing

This paper was downloaded from TechRxiv (<https://www.techrxiv.org>).

LICENSE

CC BY 4.0

SUBMISSION DATE / POSTED DATE

22-09-2020 / 24-09-2020

CITATION

Shrestha, Anup; Chuprat, Suriayati; Mukherjee, Nandini (2020): Hybrid Heuristic Load Balancing Algorithm For Resource Allocation In Cloud Computing. TechRxiv. Preprint. <https://doi.org/10.36227/techrxiv.12991340.v1>

DOI

[10.36227/techrxiv.12991340.v1](https://doi.org/10.36227/techrxiv.12991340.v1)

# Hybrid Heuristic Load Balancing Algorithm For Resource Allocation In Cloud Computing

Anup Shrestha, Suriyati Chuprat, Nandini Mukherjee

**Abstract**— Cloud computing is becoming more popular, unlike conventional computing, due to its added advantages. This is because it offers utility-based services to its subscribers upon their demand. Furthermore, this computing environment provides IT services to its users where they pay for every use. However, the increasing number of tasks requires virtual machines for them to be accomplished quickly. Load balancing a critical concern in cloud computing due to the massive increase in users' numbers. This paper proposes the best heuristic load balancing algorithm that will schedule a strategy for resource allocation that will minimize make span (completion time) in any technology that involves use cloud computing. The proposed algorithm performs better than other load balancing algorithms.

**Index Terms**— Cloud Computing, Load Balancing, Heuristic Algorithm,.

## I. INTRODUCTION

Cloud computing technology is a web-based framework which helps allocate data, resources, and services to a user on demand. Both software and hardware resources are provided via the web, making cloud computing very volatile in communication technology progression and utilizing the web. Distributed computing has been overspread everywhere hence the need to store bulks of data that can be easily retrieved [4]. In this computing technique, a consumer gets access to the cloud services through subscription to whatever service they are in need of to use through a process that involves the signing of a contract between them and the provider of the cloud services which is referred to as the Service Level Agreement (SLA). SLA helps in the determination of the service that the consumer will receive and the parameters through which the services will be delivered [1] [3].

Cloud computing has become popular with time as the number of consumers has been increasing massively; therefore, the demand for storage facilities, hardware, and software has grown in equal measures. This increase has prompted the amplification of efficiency and utilization of resources so that resource starvation needs to be limited to keep up effective service. To prevent cases where the no heaps are available at the servers such that clients are given equal execution time on their various tasks, resources used in the cloud computing environment should be scalable and elastic enough to ensure proper assimilation of the loads' virtual machines in the background [7]. This process is referred to as load balancing. In this paper, the main focus will be on heuristic algorithms used for load balancing in cloud computing.

## II. LOAD BALANCING

Load balancing involves distributing tasks among the available resources to ensure that no resource is underutilized or over-utilized. In cloud computing load balancing resources are mainly data centers, physical machines, virtual machines, or any application software [5]. Load balancing in a distributed environment requires two essential issues to be handled:

- Allocating resources
- Scheduling tasks

### A. Allocating Resources

This is the process of mapping resources to different components of the cloud upon demand. This process needs to be precise such that every node in cloud computing environment is neither overloaded nor any resources present are not wasted in any way.

### B. Scheduling Tasks

After all the resources have been allocated to the different entities, then the need for planning the tasks occurs. This is the process that defines how the execution of various tasks will happen. Resource allocation is responsible for determining which resource will be available and meets the requirement of the available task. On the other hand, scheduling tasks determines how the available resources are allocated to the different tasks available.

### Need for load balancing in cloud computing

In cloud computing, tasks and other undertakings to the virtual machines are referred to loads. These loads are categorized into three categories which are;

- under-loaded
- over-loaded
- Balanced

Load balancing algorithms are responsible for trying to equalize the total workloads in the cloud computing environment [6]. This is achieved by transferring tasks from over-loaded nodes to under-loaded nodes thus maximizing the system's throughput.

## III. HEURISTIC ALGORITHMS FOR LOAD BALANCING

Heuristic mainly means an optimization technique used to solve problems quickly when the other old systems are slow and fail to find the expected solution. They are also referred to as the approximation algorithms. Heuristic algorithms are objective and are used to produce solutions to a particular problem in a reasonable time [5]. This paper will focus on Min-min, Max-min, and heuristic algorithms and how they are used for load balancing.

#### IV. THE MIN-MIN TASK SCHEDULING ALGORITHM

This heuristic algorithm starts with a (set U) of tasks that are unmapped. A subset M of tasks with minimum (least) completion times for each job in the set U is found from this set. Furthermore, the task with the least completion time from the subset M is chosen and is allocated to the available machine (thus the name Min-min algorithm). After the task is mapped to the available device, it is deleted from U. This process iterates up to the point where all the tasks have been mapped such that set U is empty [9]. When these tasks are allocated to machines which will execute them faster and complete the process quickly, then smaller make span can be obtained. However, this algorithm's main problem is that it gives first priority to tasks that are small, thus increasing the completion time for large tasks.

#### V. MIN-MIN ALGORITHM IN LOAD BALANCING

This heuristic algorithm is simple but produces schedules that help minimize the make span than other algorithms; however, the algorithm does not effectively utilize resources. The algorithm operates in two phases: First, the Min-Min algorithm goes through execution, then, the tasks are arranged to utilize under- utilized resources in an effective way.

At first, the algorithm finds the execution time for every tasks available in a given set of tasks. Secondly, the task with the minimum execution time is selected from all the other tasks. Furthermore, that task is assigned to a resource that will ensure that the task is executed in the least time possible. This process iterates until all the tasks in the set have been scheduled using the Min-Min algorithm [6]. Since this algorithm first organizes the smaller tasks to resources with higher computational power, it produces schedules that are not optimal if small tasks are more than the larger tasks.

#### VI. THE MAX-MIN TASK SCHEDULING ALGORITHM

Just like in the Min-Min algorithm, this Max-Min heuristic algorithm begins with a set made up of unmapped tasks. Assigning the unmapped tasks to a set U, a subset M of all the tasks with the least completion time is generated from set U. The algorithm later picks the tasks with the highest (maximum) make span and allocates it to the next available virtual machine hence the name. When this task has been mapped to the best machine, it is emptied from the set U of tasks. The entire process is iterates until all the tasks have been emptied (set U is empty) (Shaikh and Rathod, 2015). Mapping a task with a maximum execution time to the high capacity machine ensures that the task can be executed concurrently with shorter execution times. Max-Min task scheduling algorithm helps in improving the make span. On the other hand, improved make span ensures a more balanced load across all the machines in the cloud computing environment.

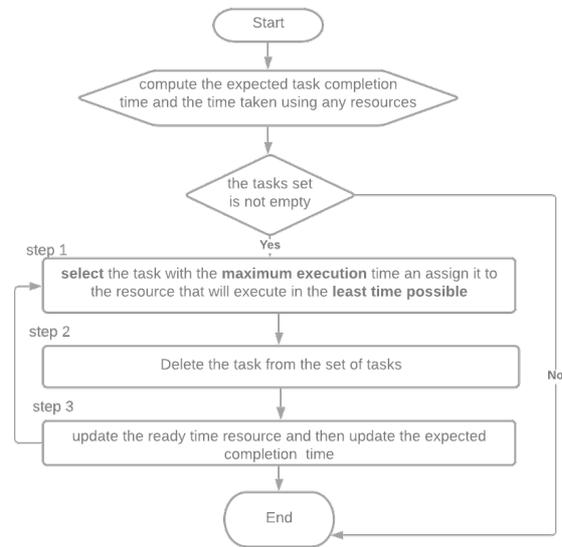
Pseudocode for the Max-Min algorithm

1. For all the tasks in the set U;  $T_i$
2. For all the resources ;  $R_j$
3.  $C_{ij} = E_{ij} + r_j$

4. While the tasks set U has tasks (not empty)
5. Select a task  $T_k$  with the highest completion time.
6. Allocate that task to a resource ( $R_j$ ), which will give the minimum execution time.
7. Empty the task  $T_k$  from set U.
8. Repeat steps 1 to 7 until the set U is empty.

From the above pseudocode,  $r_j$  represents the resource  $R_j$  which is ready for executing a task.  $C_{ij}$  represent the expected completion time and  $E_{ij}$  the execution time [2].

The overall process the algorithms in cloud computing can be simplified using the flow diagram below.



#### VII. WHY HEURISTIC ALGORITHMS ARE THE BEST IN CLOUD COMPUTING FOR LOAD BALANCING

Heuristic algorithms such as the Min-Min and Max-Min are the best in cloud computing for balancing loads due to their superior performance. Their performance is rated according to the following attributes related to these algorithms:

- These algorithms are very efficient in utilizing any systems resources such as in cloud computing, which maximizes the utilization of the virtual machines and the CPU.
- These algorithms also have strategies that help them make intelligent decisions on whether any task needs to be assigned to a low-capacity virtual machine or rather wait for a high capacity machine running.
- They also can schedule all the tasks in the waiting mode more efficiently[11].
- Assignment of tasks to resources using these heuristic algorithms mainly ensures that all the tasks are executed quickly, thus minimizing the total make span of the virtual machine in use [12].

The max-min algorithm mainly aims to reduce the waiting time for larger tasks and assign them to virtual machines' higher capacity. After all the large tasks have been assigned to their respective devices, the small tasks are assigned to either idle or underutilized machines. This ensures that all the machines are used evenly, reducing the chances of machines being either overloaded with tasks or being underutilized [10].

On the other hand, the min-min algorithm assigns small tasks to machines that will execute them very fast, hence ensuring that devices are available for use in the execution of the remaining large tasks[13][14].

## VIII. CONCLUSION

Load balancing has been a major concern in cloud computing, thus finding the best algorithm that will ensure that all tasks have been allocated to virtual machines will help reduce the make span of the entire system. This paper compares the min-min and the max-min heuristic algorithms since they are the best load balancing algorithms for use in cloud computing. From the comparison results, we can conclude the max-min algorithm outperforms the min-min algorithm when huge tasks are higher in number than the small tasks. However, when the small tasks are higher than the large tasks min-min algorithm becomes the best choice.

## REFERENCES

- [1] Adhikari, M., & Amgoth, T. (2018). Heuristic-based load-balancing algorithm for IaaS cloud. *Future Generation Computer Systems*, 81, 156-165.
- [2] Amalarethinam, D. G., & Kavitha, S. (2016). Enhanced Min-Min Algorithm for Meta-task Scheduling in Cloud Computing. *International Journal of Control Theory and Applications*, 85-91.
- [3] Vigneshwaran, P., Umamakeswari, A., Gurubaran, S., & ShaileshDheep, G. (2017). A study of various meta-heuristic algorithms for scheduling in the cloud. *Intl. J. Pure Appl. Math.*, 115, 205-208.
- [4] Delavar, A. G., & Aryan, Y. (2014). HSGA: a hybrid heuristic algorithm for workflow scheduling in cloud systems. *Cluster computing*, 17(1), 129-137.
- [5] Ghomi, E. J., Rahmani, A. M., & Qader, N. N. (2017). Load-balancing algorithms in cloud computing: A survey. *Journal of Network and Computer Applications*, 88, 50-71.
- [6] Tadapaneni, N. R. (2018). Cloud Computing: Opportunities And Challenges. *International Journal of Technical Research and Applications*.
- [7] Kumar, S., & Mishra, A. (2015). Application of Min-min and Max-min algorithm for task scheduling in the cloud environment under time shared and space shared vm models. *International Journal of Computing Academic Research (IJCAR)*, 4(6), 182-190.
- [8] Madni, S. H. H., Abd Latiff, M. S., Abdullahi, M., Abdulhamid, S. I. M., & Usman, M. J. (2017). Performance comparison of heuristic algorithms for task scheduling in the IaaS cloud computing environment. *PLoS one*, 12(5), e0176321.
- [9] Shima, K., Kumar, K., & Khanna, D. Major Role of Fog Computing in Cloud Security.
- [10] Tadapaneni, N. R. (2019). Role of Fog Computing in the Internet of Things. *International Journal of Scientific Research and Engineering Trends*.
- [11] Patel, G., Mehta, R., & Bhoi, U. (2015). Enhanced load-balanced min-min algorithm for static meta task scheduling in cloud computing. *Procedia Computer Science*, 57, 545-553.
- [12] Priyadarsini, R. J., & Arockiam, L. (2014). Performance evaluation of min-min and max-min algorithms for job scheduling in federated cloud. *Int. J. Comput. Appl.*, 99(18), 47-54.
- [13] Rao, M. L., Gulraiz, J. J., Farooq, A., & Rehman, S. (2020). Future Challenges, Benefits of Internet of Medical Things and Applications in Healthcare Domain.
- [14] Shaikh, S. J., & Rathod, S. B. (2015). A QoS load balancing scheduling algorithm in the cloud environment. *Int. J. Comput. Trends Technol.(IJCTT)*, 30(1), 1-5.