ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com

International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 8, Issue 11, November 2021

A Study on Min-Min Max-Min and Hybrid Task Scheduling Algorithms in Cloud

Nagam. B[1], Dr.M. Mohamed Surputheen[2]

[1]Research Scholar in Computer Science, [2]Associate Professor
Department of Computer Science, Jamal Mohamed College (Autonomous)
(Affiliated to Bharathidasan University)
Tiruchirappalli 620 020, Tamil Nadu, India
[1]nagamrajkumarvignesh@gmail.com, [2]msurfudeen@yahoo.com

Abstract: In cloud computing, the user can access the shared resources over the network in a service-based environment. The on-request access to computing resources such as networks, servers, and applications is the major idea of cloud computing. With the development in computing technologies, cloud computing has added a new paradigm to user services that permits accessing Information Technology services on the bottom of pay-per-use at any time as well as any location. Owing to flexibility in cloud services, many organizations are shifting their business to the cloud and service providers are creating more data centers to offer services to users. However, it is vital to provide cost-effective execution of tasks and appropriate utilization of resources. Cloud computing manages a variability of virtualized resources, which makes scheduling a serious component. In the cloud, a client may utilize numerous thousand virtualized assets for all task. Consequently, manual scheduling is not a possible solution. Task scheduling is one of the vital techniques in the cloud computing environment. It is essential for allocating tasks to the appropriate resources and optimizing the overall system performance. The simple idea behind task scheduling is to slate tasks to minimalize execution time and maximize performance as well as resource utilization. In this paper we have surveyed Min-Min, Max-Min and Hybrid algorithm to analyse which algorithm gives better makespan and resource utilization.

Keywords: Cloud computing, Task Scheduling, Max Min algorithm, Min Min algorithm, Hybrid algorithm.

I. Introduction

Cloud computing could be exactly regarded as the symbol for the Internet. Cloud offers access from anywhere and at any time. Cloud computing eradicates the physical barrier for users while providing access to its resources, with users just needing internet connectivity from their end. Cloud grants admission to its resources to the users over the internet as a service. The cloud environment mostly adopts the pay-per-use model while offering its services to the users. There are numerous cloud service providers who could offer the demanded services to the cloud users. Users just need to log-in with their authorizations for accessing the cloud services. Cloud users are reassured from storing their data from their own private computers and instead could do so virtually in the cloud. They can also submit their applications to the cloud and make use of servers accessible at the cloud for getting their application operated and processed [1]. Such characteristics of cloud entice several individuals as well as corporates to make use of the cloud services for executing their applications.

Task Scheduling is a technique of finding the order in which tasks or activity should be accomplished. It is mapping the resources to the suitable task which is acquiesced for their conclusion to the cloud it’s come in the type of NP hard problem because of huge number of solution space and takes longer time for define the optimal solution. It is a technique for organisation of resources in cloud. Task scheduling is solved the problem of which resources is to be allotted to which task so that rise the resource utilization and drop the execution time. For an improved performance scheduling algorithm need to be effective. Users submitted their tasks for completion to cloud, these task need to allocate to the processor for their execution. Now the concern is that how the tasks are allocated to processor so that least execution time and maximum profit is received by the cloud owner. So here the task scheduling resolve the problem of allocating the tasks to the top suitable processor which considering the other factor [2]. Task scheduling is greatest methods for enhanced utilization of resource and attaining the economic efficiency.

II. Related Work

Naseem A.AL-Sammarraie et al., [3] have proposed a Performance and the Cost algorithm (PAC). It is hybrid algorithm. The simple concept is to construct the number of queues equivalent to the number of priority levels considered in the system. This model uses three different priority levels like High, Medium as well as Low. Henceforth, three different queues are desired for scheduler. Later calculating the priority of tasks, it is sent to the suitable queue in the scheduler based on their priority.

Pankajdeep Kaur et al., [4] proposed a Priority based Scheduling Algorithm which focuses on Fastest Completion Rate. In this work, authors have used two principles': (1) Higher priority tasks should be scheduled preceding to the one's having lower priority. (2) Complete the task as quick as possible which will lower the cost of using resources of the service provider and also attains QoS at user level.

Abdul Hussein et al., [5] proposed a genetic algorithm which aims on live migration of Virtual Machine. The proposed algorithm is a search heuristic algorithm which repeats the process of natural selection. The steps involved in Genetic algorithm include Population Coding, Fitness Function, Selection Strategy, Initialization of Population, Crossover and Mutation.

Zhang Qian, et al., [6] have proposed a Load balancing Task scheduling algorithmbased on feedback mechanism which is created on peer-to-peer cloud environment. Each node can obtain the tasks, evaluate the obtainable resources, schedule and execute the tasks. The algorithm makes usage of weighted random strategy, overload assessment and feedback and confirm that effective nodes are not overloaded when tasks are submitted to the top resource. It also confirms that resources with normal performance endure to execute the tasks. The algorithm will balance the load efficiently and balances the workload of the nodes in the network as well as presents a resolution in the cloud computing environment for the load balancing strategy.

Madni et al., [7] have matched Min-Min, Max-Min, MCT and Sufferage algorithms by using the parameters like load balance, cost, makespan and throughput in Infrastructure as a Service model. All the algorithms were performed in the both homogeneous as well as heterogeneous environments. The authors decided that the Min-Min algorithm performed better, though Max-min and sufferage algorithm produce improved results.

Alworafi et al., [8] proposed a cost based approach namely Scheduling Cost Approach (SCA) which calculates the cost of CPU, communication cost& bandwidth, storage, RAM. In this method, the tasks will be allocatedto the VMs based on the priority given by user. The user gives priority to the task will be based on their budget. The SCA has tried to rise the load balance by allocating each task to stated VM.

III. Task Scheduling

Task scheduling is a technique used to allot incoming tasks to the available resources. The key goal of tasks scheduling algorithms is to maximize the resources utilization without disturbing the service parameters of the cloud [9]. The chief challenge of the task scheduling mechanism is to rise resource utilization without affecting the quality of services. Figure 1 shows the simple task scheduling process which is done in the cloud environment.

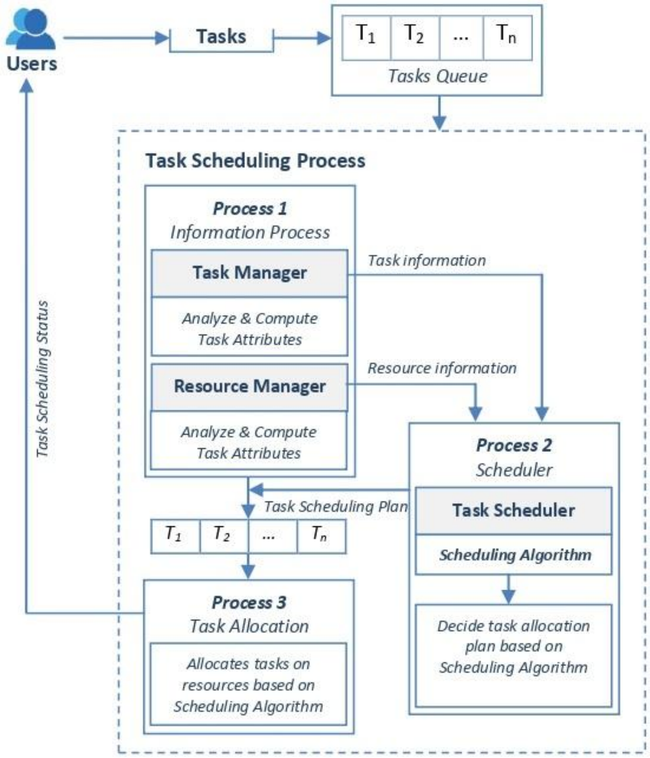

Figure 1. Task Scheduling Process

The figure displays that task scheduling is separated into three processes. The first process is the information process, in which the task scheduler gathers task information and resources information from the task manager and the resource manager. The second process is a scheduler, in which the target resource is nominated based on exact parameters of the resource and the task. These parameters comprise task size, task priority, activity-based cost, reliability factor, and dynamic slotted length of the tasks. Then, the task scheduler sends the task allocation strategy to the resource manager. The task allocation is the finishing process. In this process, the task manager allocates each task to the suitable resources.

Efficient task scheduling is desired to save the completion time and efficiently utilize the resources well. The task scheduling helps in:

i. Growing the usage of resources.
ii. Mapping jobs inavailable resources withfocuses on reducing response time and makespan.
iii. Balance between improving the quality of services and at the same time sustaining the effectiveness and fairness among the tasks.
iv. To maximize the resource utilization as well as minimize processing time of the tasks.

IV. Min-Min And Max-Min Algorithm

To implement task scheduling, there are numerous scheduling algorithms. We have studied and compared two algorithms as follows [10]:

A) Min Min Algorithm

A list of process is kept and minimum expected completion time is calculated for all the obtainable processes. A process with minimum execution time is allocated to the resource according to minimum estimated completion time. This algorithm is separated into following steps:

STEP 1: the estimated completion time of every process in the meta-process is calculated on all resource.

STEP 2: the process with minimum execution time is nominated and checked for corresponding minimum expected completion time and allocated to the resource, and the nominated process is detached from the meta-process.

B) Max Min Algorithm

A list of process is kept and minimum expected completion time is calculated for all the obtainable nodes. A process with maximum execution time is allocated to the resource according to the minimum estimated execution time, unlike Min-Min algorithm. This algorithm is separated into following steps:

STEP 1: the estimated completion time of every process in the meta-process is calculated on all resource.

STEP 2: the process with maximum execution time is nominated and checked for corresponding minimum expected completion time and allocated to the resource, and the nominated process is detached from the meta-process.

V. Comparative Study Of Various Proposals

Efficient Min-Min Algorithm (EMM): The key thing they focused in this work is the technique of resource allocation that directs how the task should be scheduled and allocated to the resources for execution so as to decrease the average response time. The Min-Min algorithm is a static resource allocation method that assigns tasks with minimum execution time and thereafter tasks with extreme execution time. The drawback in existing algorithm is that tasks with lengthiest execution time have to wait and there is interruption in their execution. To overcome this problem, the Efficient Min-Min approach is proposed in which the algorithm executes the longest and the smallest tasks alternatively so that the average response time can be decreased. EMM approach is divided as two phases: a) initially the task with minimum execution time is executed for one clock cycle then the task with maximum execution time is selected b) After one clock cycle, the tasks with remaining execution time will be executed as, first executing the task with minimum completion time till the sum of their execution time is less than or equal to the execution time of task having maximum execution time. After the execution of the entire task with minimum execution time, the task with maximum execution time will be executed. The experimental result shows that proposed EMM algorithm decreases the average response time by 45%. So, it is more effective than the existing Min-Min algorithm [11].

Rescheduling Enhanced Min-Min Algorithm (REMM): This algorithm strategically chooses the allocation of tasks in Meta task set based on speed of resources in the environment of cloud. LBMM algorithm overwhelms restrictions of Min-Min at some range but it produces the same makespan as Min-Min for some problematic sets. New Rescheduling Enhanced Min-Min Meta task scheduling algorithm is proposed to improvise the drawbacks of LBMM. The proposed REMM algorithm works as in two steps: a) the resources are sorted based on their speed. The less execution time task is assigned to the fastest resources and removed from the task set. Then another task with less execution time is assigned to the next fastest resources till the execution of all tasks, b) reschedule the task with maximum execution time from the resource then allocate it to the resource available which gives minimum completion time and compare MCT with the makespan. If MCT is lesser, then the task to be rescheduled or else the second maximum execution time task to be selected. The makespan achieved by REMM is 7s which is a better makespan than Min-Min (9s) and LBMM (8s). Also the resource utilization is better in REMM Algorithm. The experimental result of the Rescheduling Enhanced Min-Min algorithm proven that it is better in performance than the Min-Min and LBMM algorithms [12].

Enhanced Max-Min algorithm (EMM): The proposed EMM algorithm upsurges efficiency in terms of lessening completion time as well as average waiting time by enhancing resource allocations in the cloud. Max-Min algorithm gives priority to the tasks having maximum execution time to be allocated first to the available resource than assigning the task having minimum execution time. In EMM approach, the task with maximum burst time is executed primarily then tasks with minimum burst time is selected for execution till their execution time is less than or equivalent to the execution time of just executed task. This procedure endures till all tasks in the meta-task set is executed completely.As the outcome of Max-Min algorithm: the average completion time is 30.75 & average waiting time is 21.75 and EMM algorithm: average completion time is 26.5 & average waiting time is 17.5. The outcome of the experiment shows that EMM algorithm is effective with less average completion time and average waiting time as compared to Max-Min algorithm [13].

Improved Max-Min Scheduling Algorithm (MMSIA): The proposed MMSIA algorithm is animprovement of Max-Min scheduling algorithm, which advances the completion time of the requests by using the "learned learning" machine learning, by gathering size of requests and clustering consumption percent of VMs. The algorithm then allocates the largest cluster requests to the VM with the minimum utilization percent, which is recurrent when the request list is vacant. In specific, the MMSIA algorithm has enhanced the completion time. As simulation results for 500 requests, the completion time of MMSIA Algorithm is 2625.95 which is less than the completion time of Max-Min (2664.22), Min-Min (3743.97) and Round Robin (3374.22). The simulation result shows that the proposed MMSIA algorithm attains less completion time which is better than the Min-Min, Max-Min and Round Robin Algorithm [14].

Hybrid Algorithm of Min-Min and Max-Min (HAMM): This hybrid algorithm is based on two heuristic algorithms: Min-Min & Max-Min. For computation, Cloudsim simulator is used with dissimilar optimization parameters; makespan, average of resource utilization, load balancing, average of waiting time as well as simultaneous execution among small length tasks and long size tasks. HAMM concentrates on Average Task Length (AvgTL) and two counters namely Minimum Task Length Counter (TLCmin) & Maximum Task Length Counter (TLCmax). Hence Max-Min Algorithm is better than Min-Min Algorithm; the HAMM applies Max-Min Algorithm which begins with Larger Task. Upcoming task at next loop will be decided according to the unscheduled tasks length. If TLCmin < TLCmax, then the scheduler chooses Min-Min algorithm to execute smaller task and if TLCmin > TLCmax, then the scheduler chooses Max-Min algorithm to execute larger task. The simulation result of HAMM achieves Makespan (44.74ms), average waiting time (44.69ms), average resource utilization (1.00) and Load balancing (1.00) which is the better result than Min-Min and Max-Min Algorithm [15].

Dynamic Min-Max Algorithm: This work addresses scheduling of the tasks energetically by taking into account the entire execution or else completion time of the tasks and resource utilization. The Cloud scheduler enquires the Cloud Information System to check for the accessibility of resources, knowing their properties, and then scheduling the resources as per the requirements of task. The proposed work examines the effectiveness of resource provisioning in the cloud computing environment. The objective of the proposed work is to attain the goal of minimizing the execution time of the tasks. Here, VM selection algorithm is used to select which VM is accessible and efficient to execute the tasks. The scheduler allots least execution time task to the selected VM. The dynamic Min-Max algorithm eradicates the drawbacks in Min-Max algorithm and produced improved result on resource allocation by handling requests dynamically which is against Min-Max algorithm. Results are sent back to users after the task completion [16].

Priority based Performance Improved Algorithm (PPIA): It is a Meta-Task Scheduling algorithm which deliberates the user priority of meta-tasks. The meta-task set having high priority is scheduled based on Min-Min algorithm and having normal priority is scheduled based on Max-Min algorithm. The proposed algorithm gives least makespan and improved resource utilization. The experimental outcomes of the proposed algorithm PPIA for several problems prove that it outperforms the existing Min-Min algorithm in terms of makespan as well as resource utilization. The PPIA achieves makespan as 30s which is lesser than the makespan of Min-Min 37.5s. The resource utilization rate of PPIA algorithm is 0.83 and Min-Min algorithm is 0.65 where PPIA algorithm is better in resource utilization than Min-Min. [17].

Table 1. Comparative Study of Various Proposals

AUTHORS	ALGORITHMS	PARAMETERS	OBJECTIVES	METHODOLOGY	RESULT
Pandaba Pradhan et al., [11]	Min-Min Algorithm	Execution Time Response Time	To minimize average response time of tasks	Instead of executing smallest execution time task all the time; the algorithm aims at switching among the longest and the smallest tasks execution	Reduces the average response time by 45% than existing Min-Min Algorithm
D. I. George Amalarethinam et al., [12]	Min-Min Algorithm	Execution Time Completion Time Makespan	To reduce the makespan and also increases the resource utilization	The allocation of tasks in Meta task set based on speed of resources	REMM provides better result than Min-Min & LBMM (on run time & resource utilization)
Pandaba Pradhan et al., [13]	Max-Min Algorithm	Execution Time Completion Time Waiting Time	To reduce completion time and average waiting time	First executing the task with highest burst time then the lowest burst time task is executed	EMM Average Completion Time is 26.5 & Average Waiting Time is 17.5 which is better than Max-Min
Tran Cong Hung et al., [14]	Max-Min Algorithm	Task Size VM Utilization Percentage Clustering	To improve the completion time of requests	Assigning the largest cluster requests to the VM with the least utilization percent, which is repeated till the request list is empty	Completion time of MMSIA is 2625.95 which is better than Max-Min, Min-Min and Round Robin
Ibrahim A. Thiyeb et al., [15]	Hybrid Algorithm Min-Min & Max-Min	Makespan Average of Resource Utilization Load Balancing Average of Waiting Time Concurrent Execution	To achieve better performance	HAMM chooses Max-Min initially then for all task i) TLCmin < TLCmax Min-Min algorithm applied to execute small tasks ii) TLCmin > TLCmax Max-Min algorithm applied to execute large tasks	The Makespan, Average waiting time, Average resource utilization, Load balancing of HAMM is better than Min-Min, Max-Min
ChethanVenkatesh	Min-	Total execution or	To improve	VM selection algorithm is	Resource allocation

et al., [16]	Max	completion time of the tasks Resource Utilization	response time and performance	used to find those VMs with the most efficient in executing tasks Allocates least execution time task to VM	has an accuracy of 99.3%
D. I. George Amalarethinam et al., [17]	Hybrid Algorithm Min-Min & Max-Min	Execution Time Completion Time Makespan Average time taken by the prioritized tasks	To minimize makespan and better resource utilization.	i) Consider the user priority ii) High priority meta-task set is scheduled based on Min-Min algorithm and normal priority meta-task set is scheduled based on Max-Min algorithm.	PPIA Algorithm proven that it outperforms the existing Min-Min algorithm based on makespan and resource utilization

VI. CONCLUSION

Cloud computing is a vast source, according to the requirement of customers, of computational power, storage, software and a number of other services. The main objective of the task in the cloud setting is the dispersal of available resources to the task in a suitable structure. In cloud environment, the key objective of task scheduling is to minimize the execution time of tasks and to maximize the utilization of resource. In this paper, we analysed Min-Min, Max-Min and hybrid of Min-Min Max-Min algorithms proposed by different authors which gives effective resource utilization and makespan. Here the comparative analysis has been studied based on different dimensions, conditions, results, processes, and instruments. Efficient scheduling algorithm can be attained by merging different parameters to existing algorithms which will increase their complete performance of cloud environment. Minimizing Makespan is the key objective of the Scheduling algorithm. According to this survey the hybrid algorithm with effective parameters was found to be the most effective algorithm with respect to Makespan as well as resource utilization compared to the other algorithms. The Hybrid algorithm eliminates the drawbacks of two individual algorithms by combining its constructive approaches.

REFERENCES

[1] Pradeep Krishnadoss, Gobalakrishnan Natesan and Javid Ali, “CCSA: Hybrid Cuckoo Crow Search Algorithm for Task Scheduling in Cloud Computing”, International journal of Intelligent Engineering and Systems, 2021.

[2] Kalka Dubey, Mohit Kumar and S. C Sharma, “Modified HEFT Algorithm for Task Scheduling in Cloud Environment”, Procedia Computer Science, Vol. 125, 2018.

[3] Naseem A. AL - Sammarraie, Mohammed F. Al - Rahmawy, Magdi Z. Rashad, “A Scheduling Algorithm to Enhance the Performance and the Cost of Cloud Services”, Computer Engineering and Intelligent Systems, Vol.6, issue 8, 2015.

[4] Pankajdeep Kaur, Parampreet Singh, “Priority based Scheduling Algorithm with Fast Task Completion Rate in Cloud”, Advances in Computer Science and Information Technology (ACSIT), Volume 2, No.10, pp. 17-20, April-June, 2015.

[5] Abdul hussein, Abdul mohson, Sudha Pelluri and Ramachandram Siranda, “Energy Efficient Load Balancing of Virtual Machines in Cloud Environments”, International Journal of Cloud-Computing and Super-Computing, Vol. 2, No. 1, pp. 21-34, 2015.

[6] Zhang Qian, GeYufei and Liang Hong, Shi Jin, "Load Balancing Task Scheduling Algorithm based on Feedback Mechanism for Cloud Computing", International Journal of and Distributed Computing, Vol. 9, No. 4, 2016.

[7] Madni, S. H. H., Latiff, M. S. A., Abdullahi, M., Abdulhamid, S. M., Usman, M. J., “Performance comparison of heuristic algorithms for task scheduling in IaaS cloud computing environment”, Public Library of Science (PLOS), 2017.

[8] Alworafi, M.A., Dhari, A., Al-Hashmi, A. A., Basit Darem, S. A., “Cost-aware task scheduling in cloud computing environment”, International Journal Computer Network Information, pp. 52–59, 2017.

[9] Hebasaleh, Hebanashaat, Walaa saber and Hany m. harb, “IPSO Task Scheduling Algorithm for Large Scale Data in Cloud Computing Environment”, IEEE Access, Vol. 7, 2018.

[10] Pandaba Pradhan, Jashmin Swain and Sonali Sahoo, “A Comparative Analysis of Resource Allocation Algorithms in Cloud Computing”, International Journal of Advanced Research in Engineering and Technology (IJARET), Vol. 11, Issue. 7, pp. 466-476, 2020.

[11] Pandaba Pradhan, Prafulla Ku. Behera and B. N. B. Ray, “Efficient Min-Min Algorithm for Resource Allocation in Cloud Computing”, International Journal for Research in Engineering Application & Management (IJREAM), Vol. 06, Issue. 08, Nov 2020.

[12] D. I. George Amalarethinam and S. Kavitha, “Rescheduling Enhanced Min-Min (REMM) Algorithm for Meta-task Scheduling in Cloud Computing”, Springer, pp. 895–902, Jan 2019.

[13] Pandaba Pradhan, Prafulla Ku. Behera, B N B Ray, “Enhanced Max-Min Algorithm For Resource Allocation In Cloud Computing”, International Journal of Advanced Science and Technology, Vol. 29, No. 8, pp. 1619-1628, Jan 2020.

[14] Tran Cong Hung and Le Ngoc Hieu, “MMSIA: Improved Max-Min Scheduling Algorithm for Load Balancing on Cloud Computing”, Proceedings of the 3rd International Conference on Machine Learning and Soft Computing, pp. 60-64, Jan 2019.

[15] Ibrahim A. Thiyeb, Sharaf A. Alhomdy, “HAMM: A Hybrid Algorithm of Min-Min and Max-Min Task Scheduling Algorithms in Cloud Computing”, International Journal of Recent Technology and Engineering (IJRTE), Vol. 9, Issue. 4, Nov 2020.

[16] ChethanVenkatesh, Shiva Murthy G, “Dynamic min-max algorithm for resource provisioning in cloud environment”, Indian Journal of Computer Science and Engineering (IJCSE), Vol. 12, No. 3, May 2021.

[17] D. I. George Amalarethinam and S. Kavitha, “Priority based Performance Improved Algorithm for Meta-task Scheduling in Cloud Environment”, International Conference on Computing and Communications Technologies (ICCCT’17), pp. 69-73, May 2020.