Task Scheduling in Cloud Computing: A Survey

Article · May 2020

DOI: 10.22214/ijraset.2020.5369

iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 8 **Issue:** V **Month of publication:** May 2020

DOI: http://doi.org/10.22214/ijraset.2020.5369

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.429
Volume 8 Issue V May 2020- Available at www.ijraset.com

Task Scheduling in Cloud Computing: A Survey

Gopika Venu[1], Vijayanand K S[2]
[1]*M. Tech. Student, Dept. of Information Technology, Government Engineering College Bartonhill, India*
[2]*Associate Professor, Dept. of Information Technology, Government Engineering College Bartonhill, Kerala, India*

Abstract*: *The on-demand availability of computer system resources such as data storage and computing power is cloud computing. Scheduling is the method of allocating jobs onto resources in time. Scheduling increases the efficiency and performance of cloud environment by maximizing the resource utilization. This scheduling process has to respect constraints given by the jobs and the cloud providers. Ordering the tasks by scheduler along with maintaining the balance between Quality of Service (QoS), fairness and efficiency of jobs is difficult. Scheduling algorithms are designed and implemented considering some parameters like latency, cost, priority, etc. The aim of this paper is a study of various types of job scheduling algorithms that provide efficient cloud services.

Keywords: ***Task scheduling, Cloud computing, on-Demand computing.***

I. INTRODUCTION

Cloud is a trendy term for a network or remote servers that can be accessed via an Internet connection store and manage information. Two types of models present such as deployment models and service models. The deployment model includes private, public and hybrid cloud. Private Clouds are data centers architecture owned by a single company. E.g. IBM's Blue Cloud, Sun Cloud, Window Azure, etc. Public Clouds are basically the Internet. To make resource available to general people, service provider use internet. E.g. Gmail, Office 365, Drop box, etc. In case of hybrid clouds, for instance during peak periods individual applications or portion of applications can be migrate to the public cloud. The service model includes Software as a Service (SaaS) such as Drop box, Google Apps, Platform as a Service (PaaS) such as Google Compute Engine, Apache Stratos and Infrastructure as a Service (IaaS) such as Virtual Machine, Storage, Servers.

Cloud provides on demand computational resources in the form of virtual machines .These Virtual Machines are deployed in a cloud provider's data centers. The computational resources are shared among different cloud consumers who pay for the service accessed as per the usage. Allocation of resources and proper scheduling has a considerable impact on the performance and efficiency of the system. The main goal of cloud computing is to provide efficient access to remote and geographically distributed resources. An efficient scheduling is a key to manage the access to different resources, load balancing as well as resource allocation. Different types of resource scheduling algorithms are available in cloud computing based on certain parameters like time, cost, and performance, utilization of resources throughput, bandwidth, resource availability, physical distances and priority.

The paper is organized as follows. Section II includes Scheduling in Cloud Computing. Section III describes literature survey on scheduling algorithms used in cloud computing. And section IV includes conclusion.

II. SCHEDULING IN CLOUD COMPUTING

Scheduling is the allocation of various jobs to available resources. The task is a minimum computational unit to run on a node or resource. Job is a computational activity made up of several tasks that could require different processing capabilities and could have different resource requirements such as CPU, number of nodes, memory, etc. Each job may have various parameters such as desired completion time often called the deadline, required data, expected execution time, job priority, etc. A resource is something that is required to carry out an operation. For e.g., a processor for data processing, a data storage device or a network link for data transporting, etc. Scheduling problem is related to two types of users cloud consumers as well as cloud providers. Cloud consumer wants to execute their jobs for solving problem of varying size and complexity. Cloud providers will contribute resources for executing consumer jobs. Cloud consumer will benefit by selecting and aggregating resources wisely while cloud provider benefit by maximizing resource utilization. The problem of mapping tasks on resources belongs to a class of problems known as NP hard problems. These are problems for which no known algorithms are able to generate the optimal solution within polynomial time. So solutions based on exhaustive search are impractical. Overhead of generating schedules is very high. Scheduling decisions must be made in the shortest time possible in clouds because multiple users are competing for the time and resources, so metaheuristic algorithms such as genetic algorithm, PCO etc are used.

A. *Framework Of Task Scheduling In Cloud Computing Environment*

There is a framework for task scheduling system in cloud computing. In Fig 1, the users are accessing the cloud environment through internet. The cloud part shows how the cloud is managing to serve various requests given by the consumers.

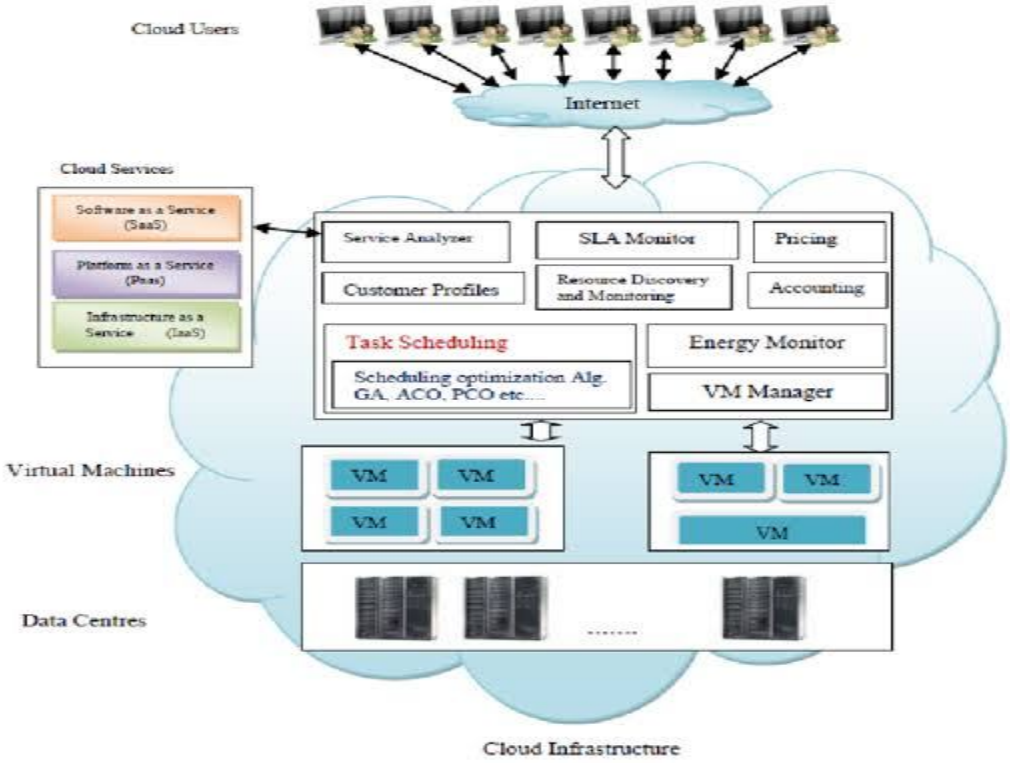


Fig1: Framework of task scheduling in cloud computing environment

1) *SLA Monitor:* The customer first submits the service request. Then the SLA monitor checks the submitted request for QoS requirements before determining whether to accept or reject the request. SLA Monitor is responsible to monitor the progress of the submitted job and if there are any violations, corrective actions has to be taken immediately.
2) *Resource Discovery And Monitoring:* Resource discovery may be described basically as the task in which the provider should find appropriate resources in order to comply with incoming consumer requests. The key features of cloud computing is capability of receiving and releasing resources on demand, therefore monitoring of resources has to be continuous.
3) *Task Scheduling:* The input of task scheduling algorithm is normally an abstract model .This abstract model defines tasks without specifying the physical location of resources on which the tasks are executed.
4) *Reschedule:* When a task can't be completed due to processor failure or other problems then uncompleted task would be rescheduled in the next computation.
5) *Scheduling Optimizer:* After acquiring information about available resources in the cloud a set of appropriate candidate is highlighted .The resource selection mechanism selects the candidate solution that fulfills all requirements and optimizes the usage of the infrastructure .The Resource selection may be done using an Optimization Algorithm. Many Optimization strategies may be used from simple and well known techniques such as simple metaheuristic algorithms. Genetic Algorithm, Ant Colony and Particle Swarm Optimization for clouds are examples.
6) *Advanced Resource Reservation Monitor:* Advanced resource reservation monitor provides QoS guarantees for accessing various resources across data centre. Users are able to secure resources required in the future. This is important to ensure successful completion of time critical applications such as real time and workflow applications for parallel applications requiring a number of processors to run. Provider can predict future demand and usage more accurately.
7) *Data Centers and VMs:* Data centers are the infrastructure or hardware part where all the physical servers present. Physical servers are not used as viable users. Through virtualization they are converted into different VMs and users jobs are going to run on these VMs in scheduling tasks.VM manager manages the VMs.

B. Different Types Of Scheduling

1) *Static vs. Dynamic Scheduling:* Jobs are pre scheduled in Static scheduling. Information regarding all resources as well as the entire task in the application is assumed to be available by the time the application is scheduled. The main drawback is no job failure and resources failure is assumed. In dynamic scheduling, jobs are dynamically available for scheduling overtime by the scheduler with no issues to be able of determining runtime in advance. The Dynamics of job execution which refers to the situation when job execution could fail due to some resource failure or job execution could be stopped due to the arrival in the system of high priority jobs in the case of preemptive mode is considered. The Dynamics of resources in which workload on resource can change continuously overtime
2) *Centralized, Hierarchical vs. Distributed Scheduling:* Centralized and decentralized scheduling differ in the control of the resources and knowledge of the overall system. In the case of centralized scheduling there is more control on resources. A scheduler has knowledge of the system by monitoring of the resource state. The main advantages are ease of implementation, efficiency and more control and monitoring of resources. The main disadvantage is lacks scalability. Hierarchical scheduling allows one two co-ordinate different schedules at certain level. Schedulers at the lowest level in the hierarchy have knowledge of the resources. The main disadvantage is lack of scalability and fault tolerance yet it is better and is more fault-tolerant than centralized schedulers. In Decentralized or distributed scheduling, no central entity controlling the resources .Scheduling decisions are shared by multiple distributed schedulers. This is less efficient than centralized schedulers.
3) *Preemptive vs. Non Preemptive Scheduling:* Preemptive scheduling criterion allows job to be interrupted during execution and a job can be migrated to another resource leaving its originally allocated resource and unused to be available for other jobs it is more helpful if there are constraints as priority to be considered. In case of Non preemptive, resources aren't being allowed to be reallocated until the running and scheduled job finish the execution.
4) *Immediate vs. Batch Mode Scheduling:* Immediate or online mode scheduler schedules recently arriving job as soon as it arrives with no waiting for next time interval on available resources at the moment. Batch or offline mode scheduler holds arriving jobs as group of problems to be solved over successive time intervals. This will helps in better mapping in job for suitable resource depending on its characteristics.
5) *Independent vs. Workflow Scheduling:* In independent scheduling, tasks can be run independently. In the case of workflow scheduling tasks are interdependent. Dependency means there are precedence orders existing in tasks .That is a task cannot start until all its parent are done. Workflows are represented by directed acyclic graph notation. Each task can start its execution only when all preceding tasks in DAG are already finished. In DAG nodes are tasks and connections or edges are dependencies between them.

C. Scheduling Objectives

An objective function is designed for scheduling algorithm. The main aim is to minimize this objective function or maximize this objective function according to the criterion specified by the user. In traditional approaches a single objective function is considered but now multicriterion objective function is created. In this, simultaneously two or more parameters are considered. On the basis of users it can be Application Centric. For example the parameters makespan and economic cost can be application centric. The designed algorithm should minimize either makespan or economic cost or both. On the basis of providers it can be Resource Centric. For example the parameters resource utilization and economic profit can be resource centric. The designed algorithm should maximize either resource utilization or economic profit or both.

III.LITERATURE SURVEY

A. Various Existing Task Scheduling Algorithms

1) *First Come First Serve (FCFS):* The incoming task looks for the queue where the waiting time is the smallest. The queue is managed by FIFO mechanism (first come first out). In FCFS, place each incoming Task at the end of the service queue. The first task in the queue is assigned to VM when it is available. This algorithm is simple and easy to understand. The main disadvantage is large waiting time. The short jobs at the back of a queue will wait until large task in the front of queue is finished. It is based on single criterion for scheduling. This algorithm is non preemptive.
2) *Round Robin Algorithm:* Round Robin Algorithm is a pre-emptive algorithm .This algorithm distributes the jobs on the available VMs in a round form or cyclic manner, where the jobs are stored in a ring queue. Each job is given a quantum of time. If it is not completed within its turn, it would be interrupted. Then it is stored back in the tail of the queue, and wait for its next turn. The algorithm is repeated until each task in the queue is assigned to minimum one virtual machine. The pros of this

algorithm are no need for a preprocessing step to fetch the nominated VM, Distribute the load equally among VMs, Focuses on fairness among the scheduled tasks, Jobs are executed in turn and never waiting for previous job to finish execution (starvation free), The scheduler will not wait until all processing power of a VM is exhausted before it moves to next VM and at last, it is based on a simple rule. The major cons of this algorithm are long jobs take longer time to complete execution. Servers may be overloaded. Preemptive policies depend on the length of time slice and during short time slice this would cause many switching.

3) *Modified round robin [1]:* By dynamically taking into consideration the burst time for each incoming task entering the ready queue, modified round robin does a simple improvement on the standard round robin algorithm. The time slicing process is done on the basis of computing the average burst time of all the remaining waiting requests in the ready queue. For this purpose, two registers are there, Sreg and Areg. The total burst time of all request in the ready queue are stored in Sreg. The Average burst time by dividing the value of Sreg by the number of tasks residing in the ready queue is stored in the Areg. In the beginning the first job is allocated to virtual machine and all its burst time is taken, then the time slice of each incoming request is computed by the scheduler. When it is allocated to run on a virtual machine, each task would run for a time period that is same as the time slice granted to it by modified Round Robin, when it entered the queue. When the time slice is over the task is either removed from the ready queue or joins the ready queue again standing at the back of the queue. The scheduler adjusts the values of the registers by subtracting the burst time of the removed task from Sreg and adding the value of the new joined tasks to Areg.
4) *Shortest Job First (SJF):* Shortest job first scheduling (SJN) is used for ordering a set of tasks by putting the shorter task in the front of the queue and longer tasks at the end of the queue. This algorithm reduces the waiting time of the short jobs and increases the waiting time of the long jobs thereby reducing average waiting time .This can result in starvation for longer jobs when there are a large number of small jobs.
5) *Traditional Min-Min Heuristic Algorithm [2]:* This algorithm is based on the concept of Minimum Completion Time. In Min-Min algorithm, for each task determine its minimum completion time over all machines. For all tasks find the minimum completion time. Assign the task to the machine that gives this completion time. Then iterate till all the tasks are scheduled. The main advantage of this algorithm is smaller makespan, since tasks are scheduled on the fastest machines where they are completed earlier. The algorithm is operative for the task scheduling in cloud computing. This algorithm will increase the throughput. The disadvantages of this algorithm is, it increases the total completion time of all the tasks and hence increases the makespan. The long tasks have to wait for smaller tasks to end their execution and the load is unbalanced.
6) *QoS Guided Min-Min Heuristic Algorithm [3]:* QoS guided Min-min takes bandwidth requirement of tasks into account. Jobs requiring higher bandwidth are scheduled prior to others. Hence, if bandwidth requisite of all tasks vary tremendously, QoS guided Min-min is better. In this algorithm, tasks are classified under high and low bandwidth. Task required high bandwidth are scheduled first.
7) *Traditional Max-Min Heuristic Algorithm [4]:* This algorithm is based on the concept of Maximum Completion Time. In Max-min algorithm, for each task determine its minimum completion time over all the machines. Then find the maximum completion time for all. Assign the task to the machine that gives this completion time. Iterate till all the tasks are scheduled. This can reduce the waiting time of long tasks so they never starved. This can increase the utilization and the response time is minimized. The makespan is reduced since smaller jobs are executed concurrently while other longer jobs are executed. The main disadvantage of this algorithm is as it first selects the large tasks for execution the smaller tasks are delayed. This algorithm is not effective in load balancing.
8) *Improved Max Min Algorithm [5]:* Improved Max Min is based on the expected execution time and not on the completion time. Max-Min could execute multiple short tasks simultaneously while executing larger ones. In this scenario the total makespan is decided by the execution of long task. In cases where Meta tasks contain tasks having different completion time and execution time, the makespan is not decided by one of submitted tasks. Improved max-min task scheduling algorithm tries to minimize waiting time of short jobs by assigning largest tasks to be executed by slower resources, small tasks are executed concurrently on fastest resources to complete large number of tasks during finalizing minimum one large task on slower resource. Based on these cases they proposed a substantial improvement of Max min algorithm which leads to increase of max min efficiency. The suggested improvement increases the opportunity of simultaneous execution of tasks on resources. It works in two phases: The first phase is task with maximum execution time is selected (Largest Task).The second phase is selected task is scheduled over resource with minimum completion time (Slowest Resource).
9) *Enhanced Max-min Task Scheduling Algorithm [6]:* Sometimes largest task is too large compared to other tasks in Meta-task. In those cases, overall makespan is increased because too large task is executed by slowest resource first while other tasks are

executed by faster resource or when there is a huge difference among slowest and fastest sources in terms of processing speed or band width. Then the largest task is executed by the slowest resource which increases the makespan and load imbalance across resources. Therefore rather than selecting largest task, if we select average or closest greater than average task then overall makespan is reduced and also balance load across resources.

10) *Minimum Completion Time Algorithm:* The algorithm scans the available VMs to find the most suitable machine to assign a job. The VM is selection is based on the minimum completion time. This minimum completion time is calculated by taking into consideration the processing speed and the current workload on a machine. Therefore it is considered as successful heuristic that could be implemented in Cloud Computing. The main disadvantage is process of assigning a task to certain machine with minimum completion time is done in arbitrary order so each time a task is assigned to the fastest machine in the remaining resources pool.

11) *Suffrage heuristic:* Various steps to be performed are:

a) For each task determine the difference between its minimum and second minimum completion time over all the machines.
b) For each task determine the difference between its minimum and second minimum completion time over all the machines.
c) Over all the tasks find the maximum suffrage.
d) Assign the task to the machine that has resulted in obtaining minimum completion time.
e) Iterate till all the tasks are scheduled.

12) *Resource Awareness Scheduling Algorithm [7]:* Resource Awareness Scheduling Algorithm is a hybrid algorithm. It is a combination of Max-Min and Min- Min Algorithms. This algorithm is also known as RASA. RASA is based on the concept of Completion Time of each task. In Resource Awareness Scheduling Algorithm, for each task Expected Completion time is calculated. To schedule task, apply Max-min and Min-Min alternatively. If number of resources is even, apply Max-Min strategy first otherwise Min-Min strategy.

13) *Reliable RASA Scheduling Algorithm [8]:* To provide required QoS to user a resource reliability parameter is added to standard algorithms. Reliability plays a significant role in performance of grid. Sometimes performance in view of completion time is high but reliability of resource might be low. It is important to allocate tasks to highly reliable resources because low resource reliability indicates frequent resource failures. In this algorithm, resource reliability is provided in percentage. Resource manager sets reliability criteria. Appropriate resources which fulfill the criteria are selected. The result after applying this algorithm is risk associated with job failure reduced.

It works in three phases:

a) Expected Completion time is calculated for each task.
b) Average or closest greater than average task is selected.
c) Selected task is scheduled over resource with minimum completion time i.e., with slowest resource.
d) For each task determine the difference between its minimum and second minimum completion time over all the machines.
e) For each task determine the difference between its minimum and second minimum completion time over all the machines

14) *Genetic Algorithm [9]:* The Genetic Algorithm mainly works in eight phases:

a) Genetic Encoding: Two dimension coding is the coding of the population individual.
b) Genetic Decoding: The decoding scheme of the encoded chromosomes is, the first char in genetic encode is decoded directed as the host resource. Then, the following chars are decoded as the tasks order that are scheduled on this host resource.
c) Initial Population Generation: First genetic individual of initial population is generated.
d) Fitness Function: To measure the quality of the population individual, the fitness function of the population is used. Deadline, budgets etc can be considered.
e) Genetic Crossover: Obtain new individuals in the current population by combining and rearranging parts of the existing individuals. A crossover probability is selected to bring a better population individual by combining two fittest individuals.
f) Genetic Mutation: Two genetic mutation operations such as exchange mutation and replace mutation present. To select randomly a host and select two tasks on this host to exchange, exchange mutation is used. To reallocate available hosts to tasks in the population individual, replace mutation is used.
g) Genetic Selection: The Roulette wheel is used to implement the population individual.
h) Genetic Termination: The termination condition of the genetic operation is setting the maximal iteration number.

Using GA for tasks scheduling, each job vector is represented by chromosomes and the positions in this vector are tasks. The population shows various mappings for tasks to machines and GA performs heuristic search to find the optimal solution. The fitness function measures the quality of solution. Usually algorithm imitates the mechanism of natural selection strategy which consists of four steps Selection crossover mutation and evaluation.

The main disadvantages are complexity in computations and long time requirement. The accuracy of algorithm can be decreased by trial/error.

B. Enhanced Methods For Task Scheduling Procedures In Cloud Environment

For making the scheduling process more efficient, various hybrid algorithms and methods are proposed.

1) *Particle Swarm Optimization Algorithm:* PSO is a type of meta- heuristics algorithms. This algorithm applies self-adaptive global search for optimization. It starts with random initialization for position and velocity for the practices population. Looking at the problem of task scheduling, tasks are considered as the particles and number of tasks in the workflow is the dimension of these particles. Each dimension has a value that indicates the resource where the tasks workflow is heading. So the mapping between tasks and resources is shown by a particle in PSO. Like GA each particle is evaluated using fitness function. The traffic workload using PSO is balanced. As it could be used with any number of tasks and resources, this algorithm is scalable. It can find near optimal solutions for mapping all tasks in the workflow to the set of available resources. Less use of mathematical operators compared with GA and consequently less need for parameters tuning. This algorithm is Simple and effective. This can be used in wide applications with little computation overhead compared to GA. The major disadvantages are, it is easy to fall into local optimum in large search space and slow convergence.
2) *Cloud Task Scheduling Based on Ant Colony Optimization [10] [11]:* ACO mainly simulate the food searching behaviour of ant colonies. While searching for food ants use kind of chemical for communicating with each other. This Chemical is called pheromone. In the beginning ants search for their foods randomly, once they find a food source, they leave pheromone on the path. Any ant can reach the food source by following the trail of pheromone. As this process continues, ants try to find the shortest path as there is large amount of pheromones accumulated on the way. Some of the advantages of this algorithm are they use positive feedback mechanism, inner parallelism and this algorithm is extensible. The disadvantages are stagnation phenomenon, all individuals find the same solution, searching to a certain extent, unable to search for a solution space, making the algorithm reach a local optimal solution.

ACO can be applied to any problem if it defines:

a) The problem representation allows ants to incrementally build or modify solutions.
b) A constraint satisfaction method which forces the construction of possible solutions.
c) A pheromone updating rule has to be there which shows how to modify trails on the edges of the graph.
d) A transition rule of the heuristic desirability and of pheromone trail.

3) *A New Flower Pollination based Task Scheduling Algorithm in Cloud Environment [12]:* For scheduling tasks in cloud environment here uses nature-inspired algorithm called Flower pollination Algorithm. This task scheduling approach is proposed in order to map tasks and resources in best optimized way, thus minimizing make-span as a consequence. Set of tasks of different completion times and set of resources of different processing powers are considered. In this algorithm, flowers are single solutions. An initial population is randomly generated. To decide between local pollination and global pollination a Switch probability is considered. For local pollination, the probability value is more. Global pollination and local pollination can be calculated using the single solutions. If switching probability is less than a random number generated then global pollination is applied, otherwise local pollination is applied. Then evaluate new solutions as per fitness Function. If new solutions are better than existing ones then update them in population, otherwise ignore.
4) *A Hybrid Bio-Inspired Algorithm for Scheduling and Resource Management in Cloud Environment [13]:* Cloud receives client's tasks in a random manner. Allocation of resources to these tasks must be handled in an intelligent way. In this method, first allocate the tasks to the virtual machines in an efficient manner. This is done using Modified Particle Swarm Optimization algorithm. Then allocation or management of resources, as demanded by the tasks, is handled by the Hybrid Bio-Inspired algorithm. Hybrid Bio-Inspired algorithm is a new hybrid approach. This incorporate two existing approaches such as Modified Particle Swarm Optimization (MPSO) and Modified Cat Swarm Optimization (MCSO).This approach focuses on two goals. First one is to provide efficient load balancing in task scheduling by building enhanced PSO algorithm. Second one is to provide dynamic resource allocation and resource management by building hybrid approach using MPSO and MCSO algorithms.

5) *Monkey Search Algorithm for Task Scheduling in Cloud IaaS [14]:* This Monkey Search Algorithm is a dynamic learning inspired task allocation algorithm. This algorithm is used to minimize the overall completion time and to improve resource utilization. Existing algorithms are mainly static or dynamic in nature. They may suffer from network traffic delay which leads to request failure. Network aware monkey search algorithm with fine execution time, with lease execution cost and network delay will provided better QoS than existing static, dynamic and learning based algorithms. This monkey search algorithm for task allocation is divided into 4 phases such as Initialization, Climb, Watch and Jump and Somersault. In initialization, initial population is generated using a set if tasks and VMs. In climb phase, monkey designated as cloudlet. The mountain is designated as vm. The monkey will climbs/schedule on the mountain whichever comes first in its path. In Watch and Jump phase, monkey searches in its local domain to find most optimal solution. This can be related as when a monkey gets on the top of the mountain, it will check whether other mountains around it higher than its present positions. If present, then jump to that higher mountain. In somersault phase, each monkey will find a locally maximal mountaintop around its initial point i.e., least execution cost path is found and monkey rolls down from the mountain. In future better output can be produced using this algorithm and network bandwidth can also be considered.

6) *Bacteria foraging based task scheduling algorithm in cloud environment [15]:* For cloud resource scheduling, this algorithm is an optimization technique based on bacterial foraging. The scheduling parameters used are number of hosts, number of cloudlets, bandwidth and number of VMs per machine. The main steps are, a virtual machine list is obtained from the data center after the provisioning of request of the user. Then a random appropriate solution and task lists are initialized. Choosing the optimal heuristic task is started from the low level. Here each micro organism from the large number of microorganisms represents an initial solution. Chemotaxis helps microorganisms to construct heuristic steps. At every decision point a fitness function health is computed. Health will compute and swimming process will start until bacteria have not climbed to long. This swimming process is continued until the cost has been reduced.

7) *Development and Analysis of a three phase cloudlet allocation [16]:* Consists of 3 phases:

a) VM categorization phase: In this phase, first select a suitable VM of certain MIPS according to cloudlet length. Then defines the cloudlet acceptability range for each VM. When cloudlet arrives, DCB selects the VM according to cloudlet's length.

b) Two round busy checking phase: DCB checks whether the VM is available or not. If available, then check for the balance threshold condition. If it satisfies the balance conditions, then task is allocated to the VM. If it is not available, two rounds are performed. Choose a VM with higher MIPS. If VM with higher MIPS present, checks for balance condition and allocates the task to it. If no, chooses a VM with lower MIPS. Continue the steps.

c) Cloudlet still not allocated phase: If VM is not allocated after the two round busy checking phase, checks for the VM with earliest finish time. If both VM have same earliest finish time, VM which becomes free earlier is chosen. The disadvantage of this method is it does not consider the total length of global queue of cloudlets. Lack in finding common completion time among heterogeneous VMs which may ensure good load balancing, in case of large and small cloudlets is an another problem of this method.

8) *ANGEL: Agent-based scheduling for real-time tasks in virtualized clouds [17]:* This method is a novel agent-based scheduling mechanism in cloud computing environment to allocate real-time tasks and dynamically provision resources. Three phases are present in the collaborative process i.e., basic matching phase, forward announcement-bidding phase and backward announcement-bidding phase.

Three agents present:

a) Task Agent
b) VM Agent
c) Manager Agent

Three phases:

i) Basic matching phase: In this basic matching phase when the task arrives, a new task agent is generated. Basic task requirement information such as task ID, task type, etc. are sent to manager agent from task agent. Task agent is then matched with the VM agents from VM information board to choose those VMs that satisfy the basic requirements posted by task agents. The matching process is done by manager agent after receiving the task requirement information. The selected VM information is sent to corresponding task agents by the manager agent.

ii) Forward announcement-bidding phase: The manager agent sent the VM's information to the task agent set. Each task agent generates forward announcement information. This includes its arrival time, length, deadline, priority, etc. Then send it to relevant VM agents. The VM agents receive the task's announcement information .Using this information calculate the corresponding bidding values based on some rules. The task agents receive the VM agent's bidding values and make forward awarding contracts for VM Agents.

iii) Backward announcement-bidding phase: The task agents receive backward announcement from vm agent. This will make forward contracts with them. The task agents receive the VM agent's announcement information .Then it will calculate the corresponding bidding values based on some rules. The VM agents receive the task agent's bidding values and then it will make backward awarding contracts. The task is allocated to the VM if a bidirectional contract is built between a VM agent and a task agent.

Agent-based scheduling algorithm consists of an algorithm for Manager Agent where the manager agent schedules the tasks that arrive at the same time instant as one batch. Then the manager agent chooses those VMs that satisfy basic requirements of task agents, and sends the selected VMs' information to those corresponding task agents. The announcement-bidding phase repeats in until all the tasks are allocated or rejected.

Another algorithm for task agents is present where each task agent generates forward announcement information to relevant VM agents. The VM agents return the corresponding bidding values. If it still fails to finish the task before its deadline, the task will be rejected.

Then an algorithm for VM agent is present. In this algorithm, the VM agents send backward announcement to task agents that send the forward contract with them, and the task agents return the corresponding bidding values. Then, the VM agents choose a bidder using the MAX Strategy or the P Strategy, and make backward awarding contracts. By now, a bidirectional contract is built and the task is allocated to the VM.

9) Dynamic Cloud task Scheduling Based On a two Stage Strategy [18]:

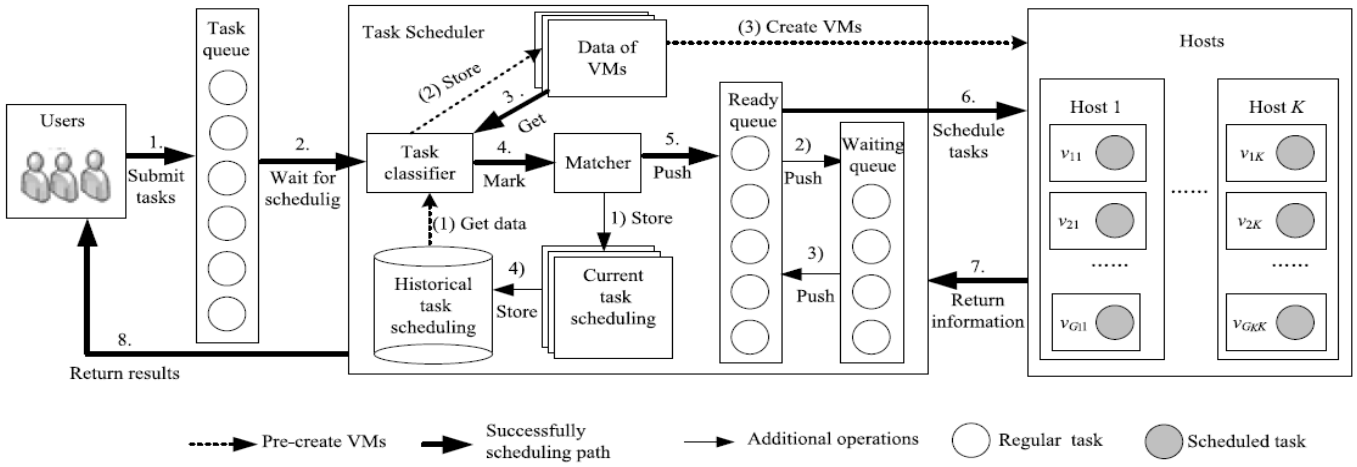


Fig 2: Cloud task scheduling framework based on a two-stage strategy.

In Dynamic cloud task scheduling based on a two stage strategy two stages are present. A Cloud task scheduling framework based on a two-stage strategy is shown in Fig.2.First stage is to classify tasks based on historical scheduling data .This can be done using a classifier based on the principle of Byes classifier's design. On the basis of this, certain numbers of virtual machines of different types are accordingly created. Time of creating VMs during task scheduling can be saved using this method. Second stage is matching tasks with concrete VMs dynamically. Dynamic task scheduling algorithms are accordingly proposed to improve performance of task scheduling. Maintain high level workload balance among all cloud resources in dynamic cloud environment. Dynamic Cloud Task Scheduling Based on a Two-Stage Strategy will schedules tasks at the idle time slot of VMs as early as possible. All constraints are also taken into consideration. For tasks of different types, this method has its obvious advantage. When compared to other methods, when the number and scale of tasks continue to increase, this method increases makespan and average waiting time more slowly. Using this method, failure rate is lower. Workload of VMs of each VM type as obtained by this method is better balanced than that by its peers. This method offers higher guarantee ratios of priority and ordinary tasks than its peers do.

IV. CONCLUSION

In cloud computing environment, resources with different characteristics are served virtually. Efficient scheduling is very important to manage this type of heterogeneous resources in an optimized way. Various scheduling algorithms and methods of scheduling have been presented in this paper. This paper deals with some traditional scheduling algorithms and its modified forms. Some enhanced scheduling methods are also discussed in this paper. This comparative study will be helpful in selection of appropriate scheduling algorithms for different cloud consumers as well as cloud service providers. To enhance the overall efficiency of cloud computing environments in future, understanding various scheduling methods may useful for development of other more efficient algorithms.

REFERENCES

[1] Ruba Abu Khurma, Heba Harahsheh and Ahmad Abdel-Aziz Sharieh, "Task Scheduling Algorithm in Cloud Computing Based on Modified Round Robin Algorithm," Journal of Theoretical and Applied Information Technology ,, 2018.

[2] S.Devipriya and C.Ramesh ,"Improved Max-Min Heuristic Model For Task Scheduling in Cloud", International Conference on Green Computing, Communication and Conservation of Energy (ICGCE),, 2013.

[3] HE XiaoShan, SUN XianHe and Gregor yon Laszewski,"QoS Guided Min-Min Heuristic for Grid Task Scheduling", Journal of Computer Science and Technology, 2003.

[4] Kobra Etminani and Prof. M. Naghibzadeh, "A Min-Min Max-Min Selective Algorithm for Grid Task Scheduing", Third IEEE/IFIP International Conference on Internet, Uzbekistan, 2007.

[5] O.M.Elzeki, M.Z.Reshad and M.A.Elsoud, "Improved Max-Min Algorithm in Cloud Computing", International Journal of Computer Applications, July, 2012.

[6] Upendra Bhoi1and Purvi N. Ramanuj, "Enhanced Max-Min Task Scheduling Algorithm in Cloud Computing,", International Journal of Application or Innovation in Engineering Management, 2013.

[7] Saeed Parsa and Reza Entezari-Maleki, "RASA:A New Grid Task Scheduling Algorithm,", International Journal of Digital Content Technology and its Applications, 2009.

[8] Sunilkumar Nakum, C. Ramakrishna and Amit Lathigara, "Reliable RASA Scheduling Algorithm for Grid Environment,",2014 IEEE International Conference on Computer Communication and Systems, 2014.

[9] Yang Cui and Zhang Xiaoqing, " Workflow Tasks Scheduling Optimization Based on Genetic Algorithm in Clouds ",, 3rd IEEE International Conference on Cloud Computing and Big Data Analysis, 2018.

[10] Airo Alonso Giraldo and M. Passino, "Honey Bee Social Foraging Algorithm for Resource Allocation", Springer Handbook of Computational Intelligence, 2015.

[11] H.W.Hashem and R. Rizk, "Honey Bee Based Load Balancing in Cloud Computing,", Transaction on Internet Information System, 2017.

[12] Jaspinder Kaur and Brahmaleen Kaur Sidhu, "A New Flower Pollination based Task Scheduling Algorithm in Cloud Environment", 4th International Conference On Signal Processing Computing And Control, 2017.

[13] Shridhar Domanal, Ram Mohana Reddy Guddeti and Rajkumar Buyya, "A A Hybrid Bio-Inspired Algorithm for Scheduling and Resource Management in Cloud Environment", IEEE TRANSACTIONS ON SERVICES COMPUTING, 2016.

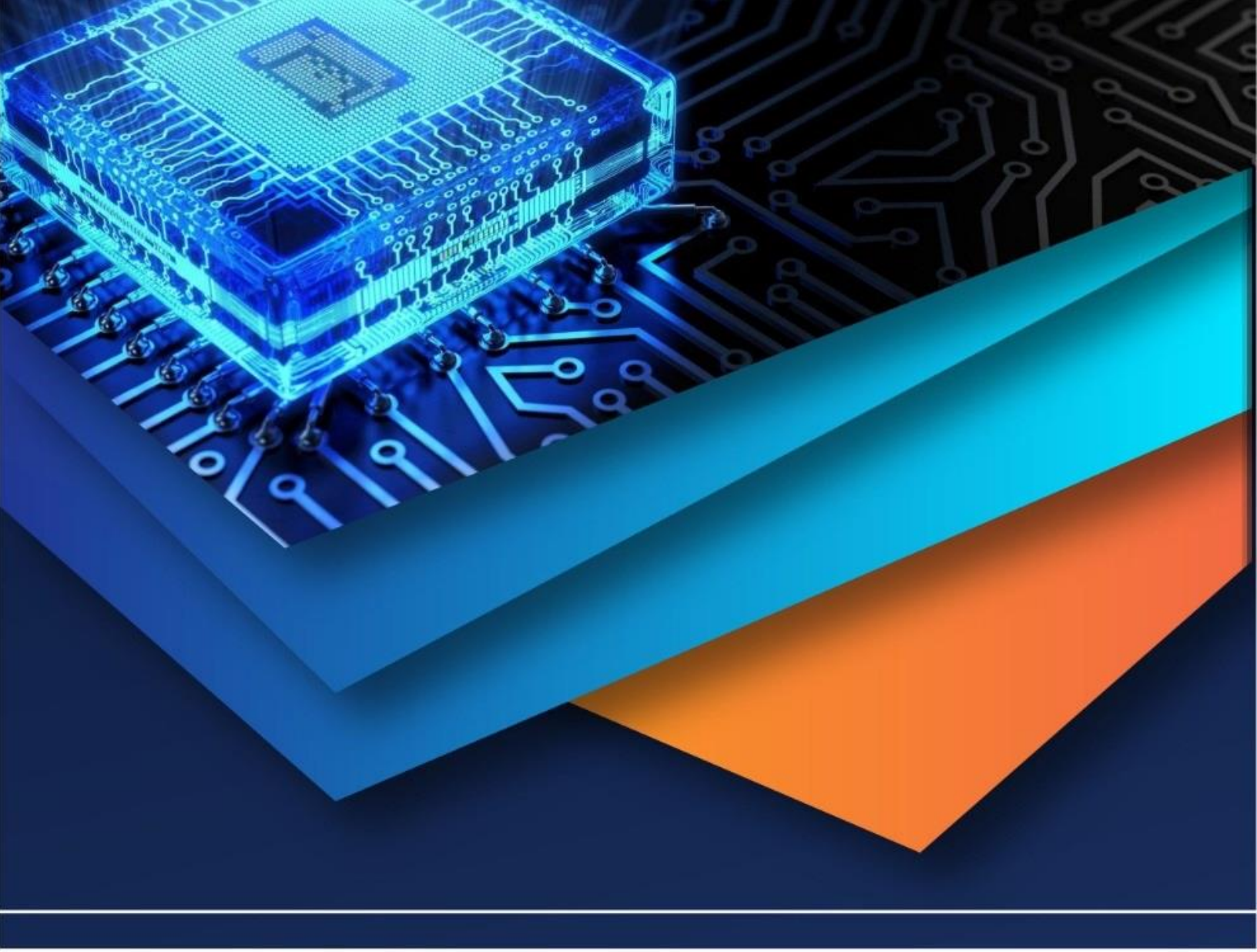
[14] Punit Gupta and Prateek Tewari, "Monkey Search Algorithm for Task Scheduling in Cloud IaaS",Fourth International Conference on Image Information Processing, 2017.

[15] Juhi Verma, Srichandan Sobhanayak and Suraj Sharma,"Bacteria Foraging Based Task Scheduling Algorithm in Cloud Computing Environment", International Conference on Communication and Automation, 2017.

[16] Sudip Roy,Sourav Banerjee,K.R.Chowdhury and Utpal Biswas, "Development and Analysis of a Three Phase Cloudlet Allocation Algorithm,", Journal of King Saud University - Computer and Information Sciences, 2017.

[17] Xiaomin Zhu, Chao Chen, Laurence T. Yang and Yang Xiang, "ANGEL: Agent-Based Scheduling for Real-Time Tasks in Virtualized Clouds", IEEE Transactions on Computers, 2015.

[18] PeiYun Zhang and and MengChu Zhou, "Dynamic Cloud Task Scheduling Based on a Two-Stage Strategy," IEEE TRANSACTIONS ON AUTOMATION SCIENCE AND ENGINEERING, 2018.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY