**Cloud Computing CPU Allocation and Scheduling Algorithms Using CloudSim Simulator**

FCFS

* Algorithm is based on the arrival time of the resource request.
* AwT = (∑ Tn) / NT
* Where AwT - Average Waiting Time, Tn: Tasks Waiting time for execution, NT: Number of tasks

SJF

* Choose the task with the shortest execution time in order to take the lead of the queue and then FCFS way.
* AwT = (∑ Tn) / NT
* Where AwT - Average Waiting Time, Tn: Tasks Waiting time for execution, NT: Number of tasks

RR (static time quantum)

* Tasks submitted to VM sorted in ascending order based on burst time (execution time for each task)
* Computing time quantum
  + TQ = (NP \* MIPS) / 1000
    - TQ (Time Quantum), NP (Number of Processors), MIPS (Millions Instruction per second)
* For each task in the queue list
  + CPU allocates the time quantum for task execution
  + If task executed, sent to finished list
  + If task not executed, sent to waiting list.

RR (dynamic time quantum)

* Tasks submitted to VM sorted in ascending order based on burst time (execution time for each task).
* Computing time quantum
  + TQ = ∑ Tn / NT
    - TQ (Time Quantum), Tn (task burst time), NT(Number of tasks)
* For each task on the queue list
  + If task burst time < time quantum
    - Time quantum = task burst time
    - CPU allocates time quantum for task execution
    - Task executed, sent to finish list
  + If task burst time > time quantum
    - CPU allocates time quantum for task execution
    - Task sent to the waiting list
  + If waiting list not empty
    - Send tasks from waiting list to queue list.

**Analysis of Job Scheduling Algorithms in Cloud Computing**

FCFS

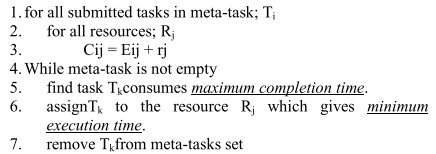
* Initialize tasks
* First task assigned to the queue and add task up to n numbers
* Add next tan in the main queue

SJF

* for i = 0 to i < main queue-size
  + if task **i+1** length < task **i** length
    - add task i+1 in front of task i in the queue

**Improved Max-Min Algorithm in Cloud Computing**

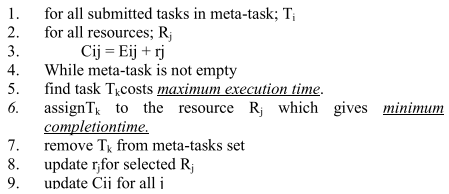
Max-Min



8. update rj for selected Rj

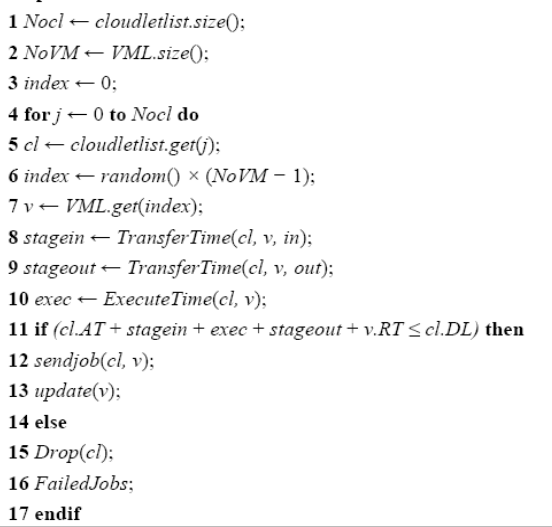
9. update Cij for all j

Improved Max-Min

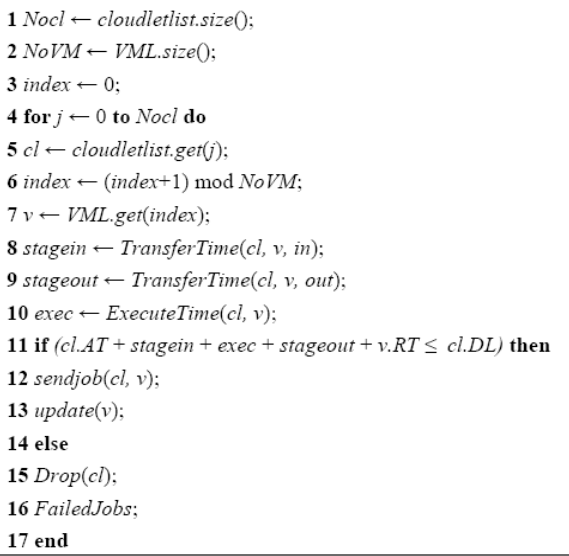


**COMPARATIVE STUDY OF SCHEDULING AL-GORITHMS IN CLOUD COMPUTING ENVIRONMENT**

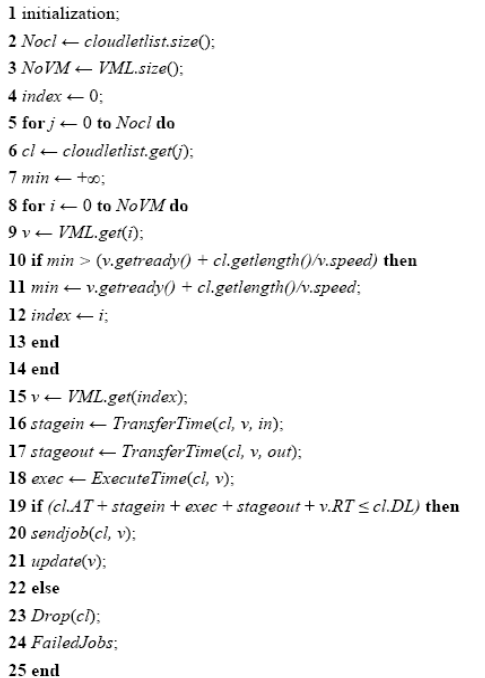
Random Algorithm



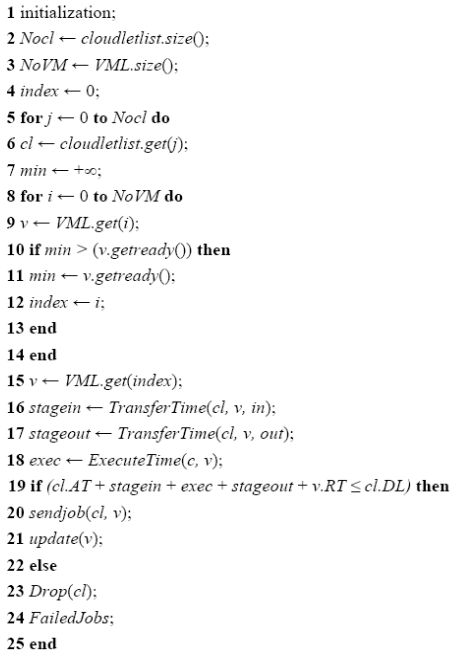
Round Robin algorithm



Minimum Completion Time

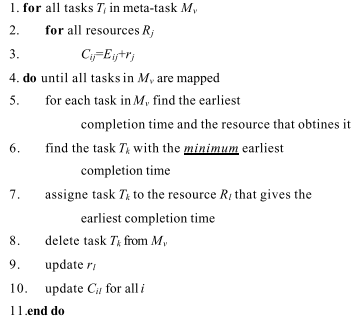


Opportunistic Load balancing algorithm

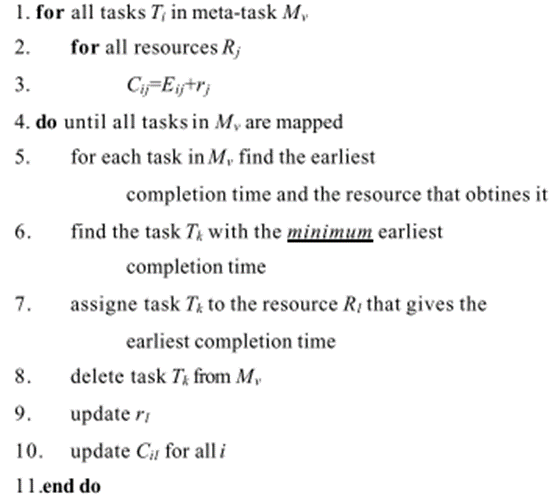


**RASA: A New Task Scheduling Algorithm in Grid Environment**

Min-Min

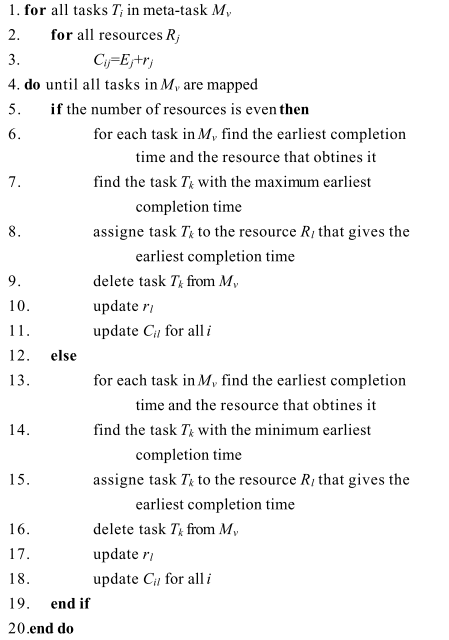


Max-Min



maximum

Resource aware scheduling algorithm



**Efficient Task Scheduling Algorithms for Cloud Computing Environment**

Longest Cloudlet Fastest Processing element (LCFP)

* Sort the cloudlets in descending order of length.
* Sort the PEs across all the hosts in descending order of processing power.
* Create virtual machines in the sorted list of PEs by packing as many VMs as possible in the fastest PE.
* Map the cloudlets from the sorted list to the created VM.

Shortest Cloudlet Fastest Processing element (SCFP)

* Sort the cloudlets in ascending order of length.
* Sort the PEs across all the hosts in descending order of processing power.
* Create virtual machines in the sorted list of PEs by packing as many VMs as possible in the fastest PE.
* Map the cloudlets from the sorted list to the created VM.

Task Scheduling in Cloud Computing: A Survey

A Comparison Study of Eleven Static Heuristics for Mapping a Class of Independent Tasks onto Ileterogeneous Distributed Computing Systems

A SURVEY ON ECONOMIC CLOUD SCHEDULERS FOR OPTIMIZED TASK SCHEDULING

Study and Analysis of Various Task Scheduling Algorithms in the Cloud Computing Environment

Dynamic Mapping of a Class of Independent Tasks onto Heterogeneous Computing Systems