Computing services that are provided by data-centers over the internet are now commonly referred to as cloud computing.

* In order to process the jobs/user's request we need to schedule the process to satisfy the request of customers.
* The classic approach to scheduling is static, in which each operation is mapped to

a clock cycle at compile-time, but recent years have seen the emergence of dynamic scheduling, in

which an operation’s clock cycle is only determined at run-time. Both approaches have their

merits: static scheduling can lead to simpler circuitry and more

resource sharing, while dynamic scheduling can lead to faster

hardware when the computation has non-trivial control flow. In

this work, we seek a scheduling approach that combines the

best of both worlds.

static tasks scheduling algorithms in cloud computing environment :

FCFS

FCFS: the order of tasks in task list is based on their arriving time then assigned to Virtual Machines.

Advantages

-> Most popular and simplest scheduling algorithm.

-> Fairer than other simple scheduling algorithms.

-> Depend on FIFO rule in scheduling task.

-> Less complexity than other scheduling algorithms.

Disadvantages

-> Tasks have high waiting time.

-> No-priority is assigned to tasks. That means when we have large tasks in the begin tasks list,

all tasks must wait a long time until the large tasks to finish.

->Resources are not consumed in an optimal manner.

SJF

Tasks are sorted based on their priority.

Priority is given to tasks based on tasks lengths and begins from (smallest task ≡ highest priority).

Advantages

-> Wait time is lower than FCFS.

-> SJF has minimum average waiting time among all tasks scheduling algorithms.

Disadvantages

Unfairness to some tasks when tasks are assigned to Virtual Machines, due to the long tasks tending to be left waiting in the task list while small tasks are assigned to VM.

-> Taking long execution time and total finish time.

MAX-MIN

In MAX-MIN tasks are sorted based on the completion time of tasks; long tasks that take more completion time have the highest priority.

Then assigned to the VM with minimum overall execution time in VMs list.

Advantages

-> Working to exploit the available resources in an efficient manner.

-> This algorithm has better performance than the FCFS, SJF, and MIN-MIN algorithm.

Disadvantages

-> Increase waiting time to small and medium tasks; if we have six long tasks, in MAX-MIN scheduling algorithm they will take priority in six VMs in VM list, and short tasks must be waiting until the large tasks finish.

-> Unfairness to some or most small and medium tasks when tasks are assigned to VM.

MIN-MAX:

Opposite to MAX-MIN