Task Scheduling Algorithms in Cloud Computing

Aliaa Kadhem, Hassan Aziz, Taha Yassen Gasim and
Haider Ismaeel

October 3, 2021

Task Scheduling Algorithms in Cloud Computing

1. **First Author Alia'a Kadheam[1], Second Author Hssain aziz[2] Third Author,Taha yaseen[3] Four Author,haider Ismaeel[4]**

Islamic Azad University South Tehran Branches

Abstract

A Cloud computing allows to users to get computing resources and services over the internet in the form of an on-demand service using virtualization technology. While cloud resources are built in different locations so they need new strategies to manage the running of them.

These strategies may contain more than one algorithm. In cloud computing system, scheduling system is the one of the main keys of system's performance.The scheduling system doesn't include task scheduling strategies only, but also it contains fault tolerance and load balancing strategies. Scheduling system is used in cloud management and handling failure. In this study, we discuss some scheduling algorithms and their advantages and disadvantages and compare with them.

Keywords:

- ✓ **Cloud Computing**
- ✓ **Scheduling Algorithm**
- ✓ **Task Scheduling**
- ✓ **Scheduling Approaches**

Figure 1: Cloud Computing

Introduction

Clouds are used for diffferent purposes and have numerous application areas. Clouds are defined as a large pool of easily accessible virtualized resources (such as hardware, development platforms and/or services) which can then be dynamically reconfigured to adjust to a variable load, allowing for optimum resource utilization **[1]**. One role of Clouds is to influence development process of software as a service. Currently, software engineering practices are far from ideal, and often the quality of the fifinal product sufferers Complexity of software development process is still very high, and many processes are still manually executed. An example of such process is the scheduling of hardware/software resources for multiple collaborating software engineering teams (e.g., development, system test, user-acceptance test/staging, etc.).Infrastructures based on Service-Oriented Architecture (SOA) increase the severity of the problem. The flexible setup of SOA systems, and dependencies on other services, make the deployment of these systems complex in staged environments. Virtualization can help to reduce the cost of physical hardware, and simulation of application behavior can reduce the dependency on back-end systems and external services **[2]**. These simulators can be deployed using cloud infrastructures to create a flexible platform to support the software development and test teams. However, to fully exploit the benefits of virtualization and virtualized services, one still needs to carefully manage dependencies between different parts of the system to ensure that all services are in place, resources that are not required are not launched (or being turned down) for optimal resource usage. Without automated scheduling this task is very complex and error prone, and potential savings cannot be achieved. Fig 1**[4]**

1 There are several interesting characteristics of cloud infrastructure architectures that render them appealing for future IT apps and services. [5]

1.1 . Self-service on-demand

Cloud resources, such as CUP time, storage, network connectivity, worker time, web apps, and so on, may be delegated to customers without any human association as required.

1.2 . Cost-efficiency

Cloud service providers' offerings are remarkably financially savvy, if not free. Pay according to usage is the charging model; there is no valid justification to acquire the base and, therefore, reduce maintenance costs.

1.3 . Pooling Capital

Into the Cloud are shared physical and simulated processing tools. As the client has no influence or details about their area, these services are not subject to the field.

1.4 . Scalability

Cloud computing's foundation is completely adaptable. Through small alterations to the cloud foundation and programming, cloud vendors will connect additional hubs and servers to the cloud.

1.5 . Dependability

It is done by the usage of multiple repetitive locations in cloud computing. High resilience renders the Cloud a perfect option for disaster relief and business-critical businesses.

1.6 . Virtualization of

Cloud infrastructure enables the user, from either terminal, to get service everywhere. Instead of a famous organization, the services it needs come from the cloud. Using a PC or a mobile phone, you can finish what you need through net service. Customers will safely accomplish or exchange it in a clear way, wherever, wherever. Clients should complete an assignment that cannot be accomplished on a single machine.

2 Cloud providers can provide three services in the cloud:

Software as a Service (**SaaS**)
Platform as a Service (**PaaS**)
Infrastructure as a Service (**IaaS**)
Fig. 2**[3]** illustrates these services.

Figure 2. Cloud-based services and deployment model

2.1 Public cloud

Most services are offered in a public environment in which consumers can access a resource pool that is managed by a host corporation [4]. Because of its existence, this type of environment will pose important concerns regarding security issues [4].

Private cloud

A third party vendor provides the services which distinguish it from public accesses [4]. Therefore it is better than the previous development model because it prevents unauthorized access.

2.2 Community cloud

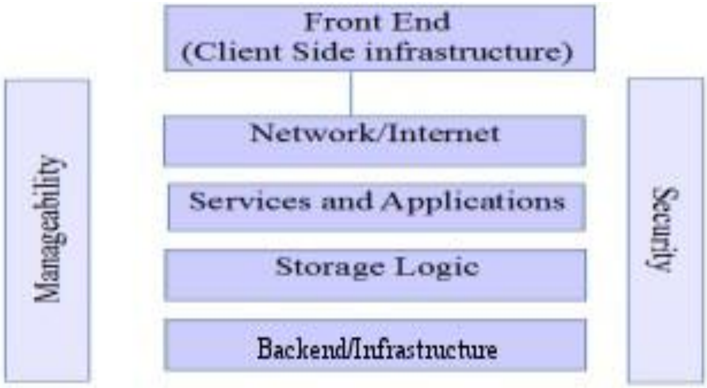
The cloud services are provided to a specified group where all members are entitled to equal access to the sheared services [4].

2.3 Hybrid cloud

The cloud services are provided as multiple cloud combustion (public cloud, private cloud, and community cloud) [4]. It might just inherit any kind of vulnerability or risk which resides within the parties listed above.

3 CLOUD ARCHITECTURE

The cloud computing architecture design has many elements and components as shown in Fig.3. All the elements are loosely coupled. The Front End components provide the applications and interface to access the cloud platform, which are clients, mobile devices.



CLOUD ARCHITECTURE **Fig3[13]**

4 TASK SCHEDULING [10]

Cloud Service Provider's Data Center has large number of Servers and other computing infrastructure. Thousands of Virtual Machines runs inside a data center to utilize the resources in best possible manner. Task Scheduling can be done to efficiently utilize the resources by allocating specific tasks to specific resources. It automatically improves the quality of service and performance. These task scheduling algorithms are categorized into two different categories, one is Batch Mode Heuristic Algorithms finds the task with lease execution time and then assigns the resource to that task which produces least execution time. If multiple resources provides the same amount of execution time, then resource is selected on a random basis.

Various BMHA examples are FCFS Algorithm, Round-Robin Algorithm, Min-Min Algorithm and Max-Min Algorithm.

Online Mode Heuristic Algorithms works in a way that tasks are scheduled as they are arrived in the system.

Some Important Terms that are used in these scheduling algorithms

A.T= Arrival Time

B.T= Burst Time

C.T= Completion Time

T.T = Turnaround Time = Completion Time - Arrival Time = C.T - A.T

W.T = Waiting Time = Turnaround Time - Burst Time = T.T - B.T

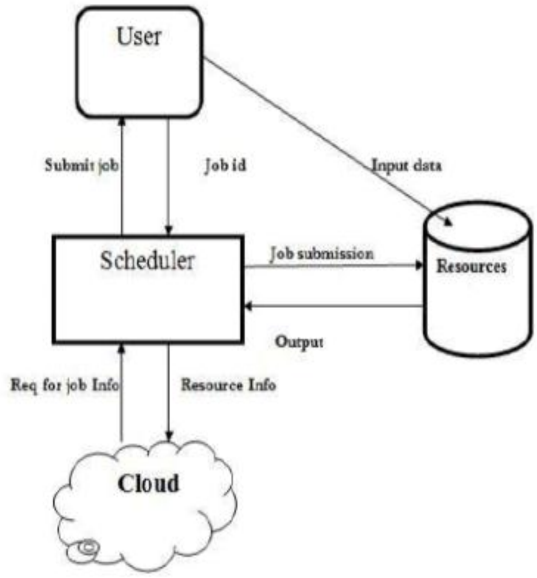


Fig4[11]Scheduling in cloud computing

4.1 TASK SCHEDULING IN CLOUD COMPUTING ENVIRONMENT [15]

Task scheduling is a procedure used to allocate incoming tasks to the available resources. The main goal of tasks scheduling algorithms is to maximize the resources utilization without affecting the service parameters of the cloud [14]. **Figure.12** shows the basic scheduling process which is done in the cloud environment. The figure shows that task scheduling is divided into three processes. The first process is the information providing process, in which the task scheduler collects task information and resources information from the task manager and the resource manager. The second process is a selection process, in which the target resource is selected based on specific parameters of the resource and the task. These parameters include task size, task priority, reliability factor, activity-based cost, and dynamic slotted length of the tasks. Then, the task scheduler sends the task allocation plan to the resource manager. The task distribution is the final process. In this process, the task manager allocates each task to the appropriate resources.

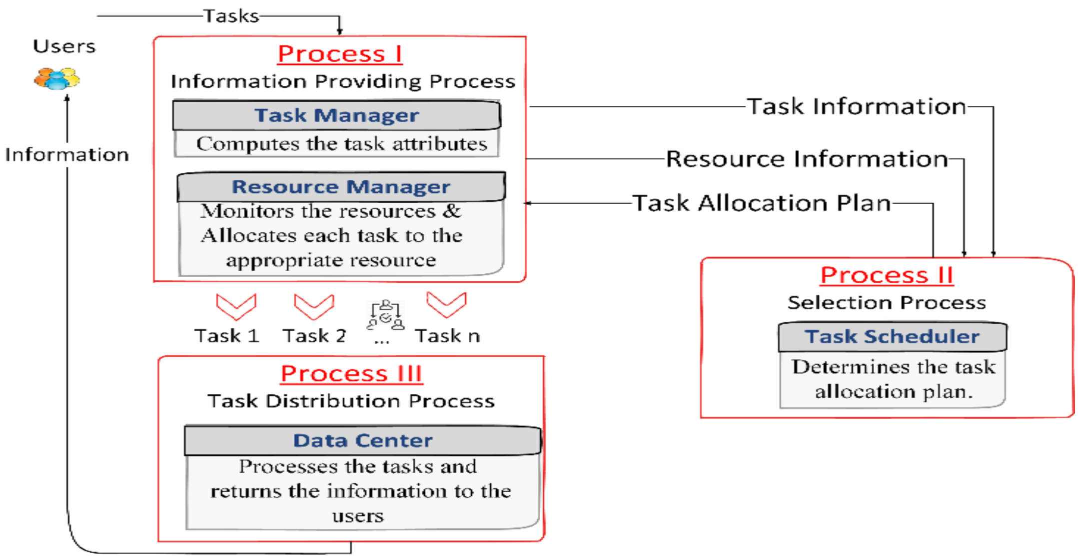


Figure. 12 The figure shows that task scheduling is divided into three processes

Scheduling in cloud computing is considered as a problem with a large solution space. Thus, it takes a long time to find an optimal solution. Deterministic scheduling algorithms are much easier and faster to implement because they are all designed based on one or a few rules for managing and arranging the tasks. However, these algorithms cannot find the optimal solution in a reasonable time, especially when the problem becomes complex or the number of tasks is too large.

4.2 Tasks scheduling algorithms overview[8]

Tasks scheduling algorithms are defined as the mechanism used to select the resources to execute tasks to get less waiting and execution time.

4.3 Tasks scheduling algorithms definition and advantages[8]

Tasks scheduling algorithms are defined as a set of rules and policies used to assign tasks to the suitable resources (CPU, memory, and bandwidth) to get the highest level possible of performance and resources utilization.

- Manage cloud computing performance and QOS.
- Manage the memory and CPU.
- The good scheduling algorithms maximizing resources utilization while minimizing the total task execution time.
- Improving fairness for all tasks.
- Increasing the number of successfully completed tasks.
- Scheduling tasks on a real-time system.
- Achieving a high system throughput.
- Improving load balance.

5 Different Task Scheduling Approaches[10]

5.1 First-come-first-serve scheduling Algorithm

- In this algorithm, tasks that arrived first are served first. Jobs when enter the queue are inserted into the tail of the queue. One by one each process is taken from the head section of the queue. This algorithm is straightforward and quick Characteristics.
- There is no prioritization at all and this makes every process to eventually complete before any other process is added.
- This type of algorithm does not work well with delay sensitive traffic as waiting time and delay is relatively on the higher side.
- As context switches only occurs when a process is terminated, therefore no process queue organization is needed and there is very little scheduling overhead.

5.2 Shortest Job First scheduling Algorithm

SJF Algorithm is a pre-emptive that selects the waiting process that has the least execution time. The process is then allocated to the processor that has the least burst time.

Characteristics

- One of the problems that SJF algorithm is that it has to get to know about the next processor request.
- It reduces the average waiting time as it executes small processes before the execution of large ones.
- When a system is busy with so many smaller processes, starvation will occur.
-

5.3 Round Robin scheduling Algorithm

In this type of algorithm, processes are executed just like in FIFO, but they are restricted to processor time known as timeslice. If the process is not completed before the expiration on processor time, then the processor takes the next process in the waiting state in the queue. The preempted or new process is then added to the rear of the ready list and then new processes are inserted in the tail of the queue.

Characteristics

- If we apply a shorter time-slice or quantum, then in that case there will be lower CPU efficiency.
- If we apply a long time-slice or quantum, then it will result in poor response time.
- As waiting time is high, there will be a very rare chance that deadlines met.

5.4 Priority scheduling algorithm

In this algorithm, priority is assigned to each process and processes are executed on the basis of priority. Priorities having same priority use FCFS.

Characteristics

- If there is a large number of equal priority processes, then it results in large waiting time.
- Processes with higher priority results in least waiting time and lesser delay.
- Low prioritized processes can see starvation.

5.5 Min–Min algorithm: [12]

This algorithm chooses small tasks to be executed firstly, which in turn large task delays for long time.

5.6 Max – Min algorithm: [12]

This algorithm chooses large tasks to be executed firstly, which in turn small task delays for long time.

5.7 Most fit task scheduling algorithm: [12]

In this algorithm task which fit best in queue are executed first. This algorithm has high failure ratio.

Consider the following set of processes, with the length of the CPU-burst time in milliseconds is shown in Table 1.

TABLE 1
PROCESS WITH ITS ID AND BURST TIME

Process ID	Burst Time (ms)
P1	10
P2	2
P3	8
P4	6

a. First Come First Serve

P1	P2	P3	P4

0 10 12 20 26

b. Shortest Job First

P2	P4	P3	P1

0 2 8 16 26

Figure 14: Gantt chart for SJF

c. Round Robin

Assign time quantum as 5 ms for each process

P1	P1	P1	P1	P1	P1	P1

0 5 7 12 17 22 25 26

Figure 15: Gantt chart for RR

Priority is assigned for each process as follows:

TABLE 2

PROCESS WITH ITS ID, BURST TIME AND PRIORITY

Process ID	Burst Time (ms)	Priority
P1	10	3
P2	2	1
P3	8	4
P4	6	2

P2	P4	P1	P3

0 2 8 18 26

Figure 16: Gantt chart for PS

TABLE 3
WAITING TIME FOR INDIVIDUAL PROCESS AND AVERAGE WAITING TIME FOR EACH SCHEDULING WAITING TIME (ms)

Process ID	FCFS	SJF	Round Robin	Priority Scheduling
P1	0	16	12	8
P2	10	0	5	0
P3	12	8	17	18
P4	20	2	20	2
Avg Waiting Time	10.5	6.5	13.5	7

TABLE 4
TURNAROUND TIME FOR INDIVIDUAL PROCESS AND AVERAGE TURNAROUND TIME FOR EACH SCHEDULING TURNAROUND TIME

Process ID	FCFS	SJF	Round Robin	Priority Scheduling
P1	10	26	22	18
P2	12	2	7	2
P3	20	16	25	26
P4	26	8	26	8
Avg Turnaround Time	17	13	20	13.5

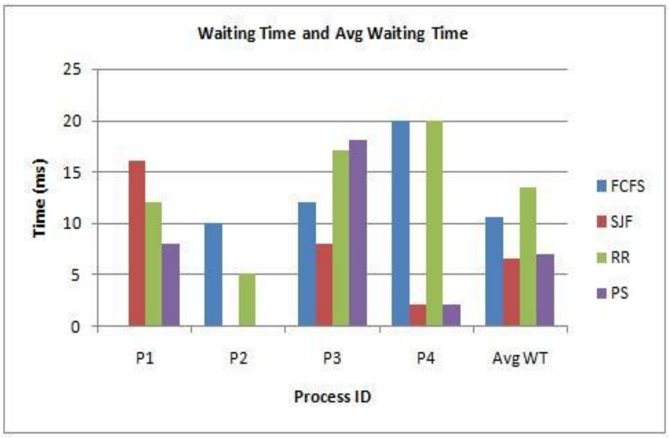


Figure 17: Comparison of Fundamental scheduling Algorithm-waiting Time

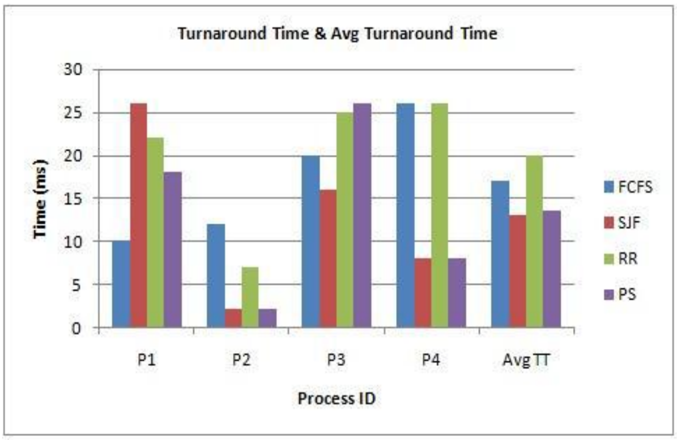


Figure 18: Comparison of Fundamental scheduling Algorithm-Turnaround Time

For example, turnaround time for the process is calculated as time of submission of a process to the time of completion of the process is obtained through Gantt chart for SJF scheduling.

Turnaround time for process **P1, P2, P3** and **P4** is observed as **26, 2, 16 & 8** and average turnaround time is **(26+2+16+8)/4=13 ms**.

The waiting time for the process is calculated as time taken by the process to wait in the ready queue is observed from Gantt chart for SJF scheduling. Waiting time for **process P1, P2, P3 and P4** is obtained as **16, 0, 8 & 2**

respectively and average waiting time is **(16+0+8+2)/4 =6.5 ms.**

Similarly the turnaround time and waiting time is calculated for all other algorithms and summarized in **Table 3 and Table 4.**

From the above discussion it is clear that First Come First Serve (FCFS) & Shortest Job First (SJF) is generally suitable for batch operating systems and Round Robin (RR) & Priority Scheduling (PS) is suitable for time sharing systems.

SJF algorithm is optimum for all type of scheduling algorithm. Hence, it is an algorithm with an optimum criteria and suitable for all scenarios.It is clearly observed that turnaround time, waiting time and response time of the processes are optimum for SJF scheduling algorithm compared to all other fundamental algorithms from Figure 13, Figure14, Fig. 15, Fig. 16, (Figure13 to Figure 16 are Gantt Chart) Figure 17 and Figure 18. It can also be observed that throughput and

CPU utilization rate are optimum.From above analysis and discussion, we can say that the FCFS is simple to understand and suitable only for batch system where waiting time is large. SJF scheduling algorithm gives minimum average waiting time and

average turnaround time. The priority scheduling algorithm is based on the priority in which the highest priority job can

run first and the lowest priority job need to wait though it will create a problem of starvation The round robin scheduling algorithm is preemptive which is based on round robin policy one of the scheduling algorithm which follows the interactive system and the round robin scheduling algorithm is deal with the time sharing system.

evaluate a scheduling algorithm to code it and has to put it in the operating system, only then a proper working capability of the algorithm can be measured in real time systems

Experiment Result[11] table

S. No	Algorithms	Evaluation Criteria	Advantages	Disadvantage
1	First Come First Serve	Waiting and turnaround time is high, Low Response Time	Easy implementation	No other criteria for scheduling
2	Round Robin	Medium waiting time, medium turnaround time, High Response time	Less complex and Appropriately load is balanced	Preemption is required
3	Shortest Job First Scheduling	Less waiting time, Small execution time, maximum throughput	CPU is allocated to the process that has the minimum amount of burst time.	Difficult to understand
4	Priority based	High turnaround time and high response time	It is designed on the basis of multiple criteria decision making model.	Difficult to understand and code

RESULTS AND DISCUSSION

It is find out that SJF provides a better turnaround and waiting time than all other algorithms. Throughput and CPU utilization with SJF is also seems to be optimum. FCFS involves largest waiting and turnaround time as it has short processes wait for a longer intervals. With large waiting time even for smaller processes, it is not recommended where delay sensitive traffic is involved. In RR, each job gets an equal amount of time, but there are some cases where average waiting time can be a problem. Time Slice or Quantum has to be decided carefully, otherwise a larger time-slice or quantum can affect the processing. Priority Queuing can become problem if there are large sets of processes with similar priority as then it can work like FCFS too.

CONCLUSIONS

The SJF scheduling algorithm is to serve all types of job with optimum scheduling criteria. The treatment of shortest process in SJF scheduling tends to result in increased waiting time for long processes. And the long process will never get served, though it produces minimum average waiting time and average turnaround time.

The shortest job first scheduling algorithm deals with different approach, in this algorithm the major benefit is it gives the minimum average waiting time.

It is recommended that any kind of simulation for any CPU scheduling algorithm has limited accuracy.

References//

- [1] Vaquero, L.M., Rodero-Merino, L., Caceres, J., Lindner, M., A break in the clouds: towards a cloud definition.,SIGCOMM Comput. Commun. Rev. 39 (2008) 50–55
-
- [2] Nizamic, F., Groenboom, R., Lazovik, A., Testing for highly distributed service-oriented systems using virtual environ-ments., In: Proceedings of 17th Dutch Testing Day. (2011)
-
  - [3]Ahmed Alrehaili ,Abid Mir, Mir junaid ARetrospect of Prominent Cloud Security Algorithm ,International Journal of innovation Technology and Exploring Engineering Volume-9 Issue-3 January 2020
  -
  - [4] Tanweer Alam. Cloud Computing and its role in the Information Technology. IAIC Transactions on Sustainable Digital Innovation (ITSDI), 1(2 April), 108-115, 2020. DOI: https://doi.org/10.34306/itsdi.v1i2.103
  -

- [5] Electronic cop**https://ssrn.com/abstract=3790662** https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3y available at: 790662
- [6]. Soni, Ankit Narendrakumar, Diabetes Mellitus Prediction Using Ensemble Machine Learning
- Techniques (July 3, 2020). Available at SSRN: https://ssrn.com/abstract=3642877 or
- **http://dx.doi.org/10.2139/ssrn.3642877**.
- [7]**https://www.researchgate.net/publication/324201161_Scheduling_Strategies_in_Cloud_Computing_Methods_and_Implementations**
- [8]**https://www.intechopen.com/chapters/71902**
- [9]nternational Journal of Latest Trends in Engineering and Technology Vol.(9)Issue(3), pp.217-223
- DOI: http://dx.doi.org/10.21172/1.93.37
- e-ISSN:2278-621X DIFFERENT TASK SCHEDULING ALGORITHMS IN CLOUD COMPUTINGDIFFERENT TASK SCHEDULING ALGORITHMS IN CLOUD COMPUTING
- [10] R. Durga Lakshmi and N Srinivasu, " A Review and Analysis of Task Scheduling Algorithms in Different Cloud Computing Environments, " IJCSMC,
- Vol.: 4(12), pp: 235 – 241, December 2015.
- [11]Vijayalakshmi A. Lepakshi and Dr. Prashanth C S R," A Study on Task Scheduling Algorithms in Cloud Computing ," International Journal of
- Engineering and Innovative Technology (IJEIT), Vol.: 2(11), pp: 119-125, ISSN: 2277-3754, May 2013.
- [12] **https://www.ijcsmc.com/docs/papers/December2015/V4I12201558.pdf** A Review and Analysis of Task Scheduling Algorithms in Different Cloud Computing Environments. Research Scholar, Department of Computer Science & Engg, K L Uni-

versity, Vijayawada, AP, India *Professor, Department of Computer Science and Engineering, K L University, Vijayawada, AP, India

- [13] A REVIEW – DIFFERENT SCHEDULING ALGORITHMS IN CLOUD COMPUTING ENVIRONMENT
- [14] E. Djebbar, and G. Belalem, “Tasks scheduling and resource allocation for high data management in scientific cloud computing environment,” in *Proc. Springer International Conference on Mobile, Secure and Programmable Networking*, Paris, France, pp. 16–27, Jun. 2016.
- [15] **https://ieeexplore.ieee.org/iel7/6287639/6514899/08594653.pdf** IPSO Task Scheduling Algorithm for Large Scale
- Data in Cloud Computing Environment
- [16] Neeta Patil and Deepak Aeloor, "A REVIEW - DIFFERENT SCHEDULING ALGORITHMS IN CLOUD COMPUTING ENVIRONMENT", 11th International Conference on Intelligent Systems and Control (ISCO) , pp: 182-185, IEEE, 2017.
- [17] Pushpraj Singh, Vinod Singh and Anjani Pandey, "Analysis and Comparison of CPU Scheduling Algorithms," International Journal of Emerging Technology and Advanced Engineering, Vol.: 4(1), pp: 91-95, ISSN 2250-2459, January 2014.
- [18] Mr. Sindhu M, Mr. Rajkamal R and Mr. Vigneshwaran P, “An Optimum Multilevel CPU Scheduling Algorithm”. 978-0-7695- 4058-0/10 $26.00 © 2010 IEEE
- [19] Black, D.L., 1985. Scheduling support for concurrency and parallelism in the mach operating system. Computer, 23:123-126.