IOSR Journal of Engineering (IOSRJEN) **www.iosrjen.org**
ISSN (e): 2250-3021, ISSN (p): 2278-8719
Vol. 09, Issue 3 (March. 2019), ||S (III) || PP 11-17

Score-Based Max-Min Budget Constraint Workflow Scheduling Algorithm for Cloud System

Makhan Singh
Department of Computer Science & Engineering, UIET, Panjab University, Chandigarh. India
Corresponding Author: Makhan Singh

Abstract: Cloud computing is the paradigm that uses central remote servers and internet to host the data and perform distributed computing for large-scale applications. This is the latest concept emerging in Cloud computing for service provisioning in distributed system. The research community groups are keen to explore the merits and demerits of scientific application execution which involves workflow. In Cloud computing the scheduling of workflow is the prominent issue that maps and allows inter-dependent tasks to be executed on the distributed resources. Workflow scheduling allocates those resources to tasks which totally satisfy the quality of service (QoS) constraint that are defined by the Cloud users. In present scenario the workflow scheduling approaches are focusing on execution cost and time related QoS parameters only for allocating virtual machines to workflow applications. Sometimes virtual machines (VM) are not reliable at datacentres which results in the failure if workflow application that are scheduled on these. These resources frequently results into failure when workflow applications are scheduled on this VMs. Failure of workflow application is another key concern. This situation cannot be tolerated as workflow application of users may contain sensitive data. Hence proper scheduling can improve performance of the system. In this paper a score based Max-Min budget constraint workflow scheduling algorithms has been designed and simulated. This algorithm reduces the execution time and failure rate of workflow applications within user specified budget.

Keywords: Workflow Scheduling, Score, Virtual Machine, Budget Constraint, Cloudlet.

Date of Submission: 06-03-2019 Date of acceptance: 25-03-2019

I. INTRODUCTION

Cloud Computing is the paradigm that uses the central severs and internet to host the data and resources. This is the latest concept emerging in Cloud computing for service provisioning in distributed system. Cloud Computing facilitate software applications, platform, and hardware infrastructures as a service. Data and software packages are hosted on servers.

Workflow scheduling is one of the major issues in workflow management especially in Cloud and Grid workflow systems. It is a process that maps and manages the execution of workflow tasks on different distributed resources. It allocates required resources to suitable workflow tasks such that complete execution can be completed to satisfy constraints defined by the users. Proper scheduling can improve the performance of the system. However, in general, the mapping of tasks on the distributed resources is NP-hard problem. For such problems, no known algorithms are able to generate the optimal solution within polynomial time. Even then the workflow scheduling problem can be solved by using exhaustive search. In this case problem solving complexity is very high. In Grid Computing decisions about scheduling the tasks is taken in the shortest time possible, because number of individuals demand for resources, and time slots needed by one user could be taken by another user at any moment (Yu, J. et al., 2007). This equally applies to Cloud Computing environments.

Workflow scheduling is the problem of mapping of each task on suitable resources while satisfying the constraints imposed by the user. Proper workflow scheduling can have significant impact on the performance of the workflow management system.

II. RELATED WORK

Workflow scheduling in Cloud is divided into two main categories as per literature (Bala, A., et al. 2011). First category is best effort based scheduling algorithms which tries to optimize the execution time and ignoring the other factors such as execution cost and other QoS constraints. Second one is QoS constraint based scheduling algorithms which tries to optimize the performance under QoS constraints for e.g. cost minimization under the deadline constraint or time minimization under the budget constraint. (Abrishami, S. et al., 2012) proposed workflow scheduling algorithm that is based on partial critical path approach in this they tried to reduce the cost of execution along with satisfy the user objective function deadline. This algorithm partial critical path is recursively schedule. (Abrishami, S. et al., 2013) presents the workflow scheduling algorithms first one is called IaaS Cloud Partial Critical Paths (IC-PCP) which is one phase algorithm, and second one is called IaaS Cloud Partial Critical Paths with Deadline Distribution (IC-PCPD2).These algorithms consider the main features like on demand resource providing and pay per use model of the current commercial Clouds. (Yu, J. et al., 2008) proposed a genetic approach based workflow scheduling algorithm which considering two QoS constraints Budget and Deadline. Proposed genetic algorithm schedule the workflow application either to minimize the Makespan while meeting user imposed deadline constraints or minimize the execution cost while meeting user imposed budget constraints. The performance of proposed genetic algorithm is compared with the existing genetic algorithms. This algorithm mainly consider heterogeneous environment and provide the solution for deadline and budget optimization problem. (Jayadivya et al., 2012) proposed a workflow scheduling algorithm which provides fault Tolerance. Fault tolerance is achieved by using resubmission and replication of tasks. The resubmission and replication of tasks depends upon the metric which is achieved by resubmission factor and replication factor. Priority of tasks is defined on the basis of the criticality of the task which is calculated by using parameters resubmission impact factor, earliest deadline. (Bittencourt et al., 2011) proposed scheduling algorithm for Hybrid Cloud which optimizes the cost of resources. This algorithm chooses which resources should be leased from the public cloud and aggregated to the private cloud to provide enough processing power to execute a workflow within a given execution time. Design algorithm can reduce costs while achieving the specified desired execution time. (Van den Bossche et al., 2010) consider the problem of outsourced tasks at the time of heavy load from an internal data center to a cloud provider. The main objectives this algorithm to increase the utilization of the internal data center and to minimize the cost of running the outsourced tasks in the cloud while satisfy the required QoS constraints. They consider this optimization problem in a multi-provider hybrid cloud setting with deadline-constrained and preemptible but non-provider-migratable workloads that are characterized by memory, CPU and data transmission requirements.

III. SCORE BASED MAX-MIN WORKFLOW SCHEDULING ALGORITHM

The proposed scheduling algorithm is based on the concept of score. In proposed scheduling algorithm Virtual machine (VM) score is defined on the basis of capability of the machine. In which each hardware component of the machine receives an individual score and final score of the machine is determined by lowest sub score of the component. Final score represents minimum performance of the machine, which is based on the capabilities of different parts of the machine including processing power, storage space and RAM. Similarly workflow tasks score is defined on the basis of number of instructions in workflow tasks. Score based algorithms looking for machines whose final score is equal to or more than the tasks score and execute the tasks on those machines within user specified budget. In proposed work score based Max-Min budget constrained workflow scheduling algorithm has been designed. Then performances of this algorithm has been compared with basic Max-Min budget constrained workflow scheduling algorithm with respect to their execution time, execution cost and failure rate.

Max-Min Scheduling Algorithm (Sidhu, A. K., et al., 2013) it starts with set of unassigned workflow tasks. Then all unassigned tasks are scheduled on those available resources which will execute them in maximum time (Kaur R, et al., 2013) This procedure is followed until all the workflow tasks are assigned to the machines. Here Max-Min scheduling algorithm has been designed with budget constraint i.e. tasks are executed within user specified budget. Score concept has also been introduced and only those machines are selected for scheduling which satisfy minimum task score. Then performance of score based Max-Min scheduling algorithm has been compared with basic Max-Min scheduling algorithm with respect to their execution time, execution cost and failure rate.

Algorithm BASIC_MAX-MIN (T, VM, B , C_i)

// T is Work Flow Tasks List, VM is Virtual Machines List, B is Budget, C_i is Virtual Machine Costs.

Step1:-Submit list of workflow tasks T :=$\{T_1, T_2, \dots, T_n\}$.

Step2:- Get available resources from data center VM :=$\{VM_1, VM_2, \dots, VM_n\}$.

Step3:- Assign Budget B to workflow tasks T.

Step4:- Arrange the VMs in ascending order and Ts in descending order.

Step5:- Repeat while T!= NULL

{

Step 5.1:- Pick VM from list.

Step 5.2:- Pick next VM from list.

Step 5.3:- If (VM Cost < Budget Cost)

{

Step 5.3.1:- Assign workflow task to VM.

Step 5.3.2:- Select next task from list and Goto Step 5.
}
Else
Step 5.3.1:- Goto Step 5.2.
}
Step 6:- Return map to simulation.

The figure 1 shows the flow diagram of Basic Max-Min Budget Constrained Workflow Scheduling Algorithm.

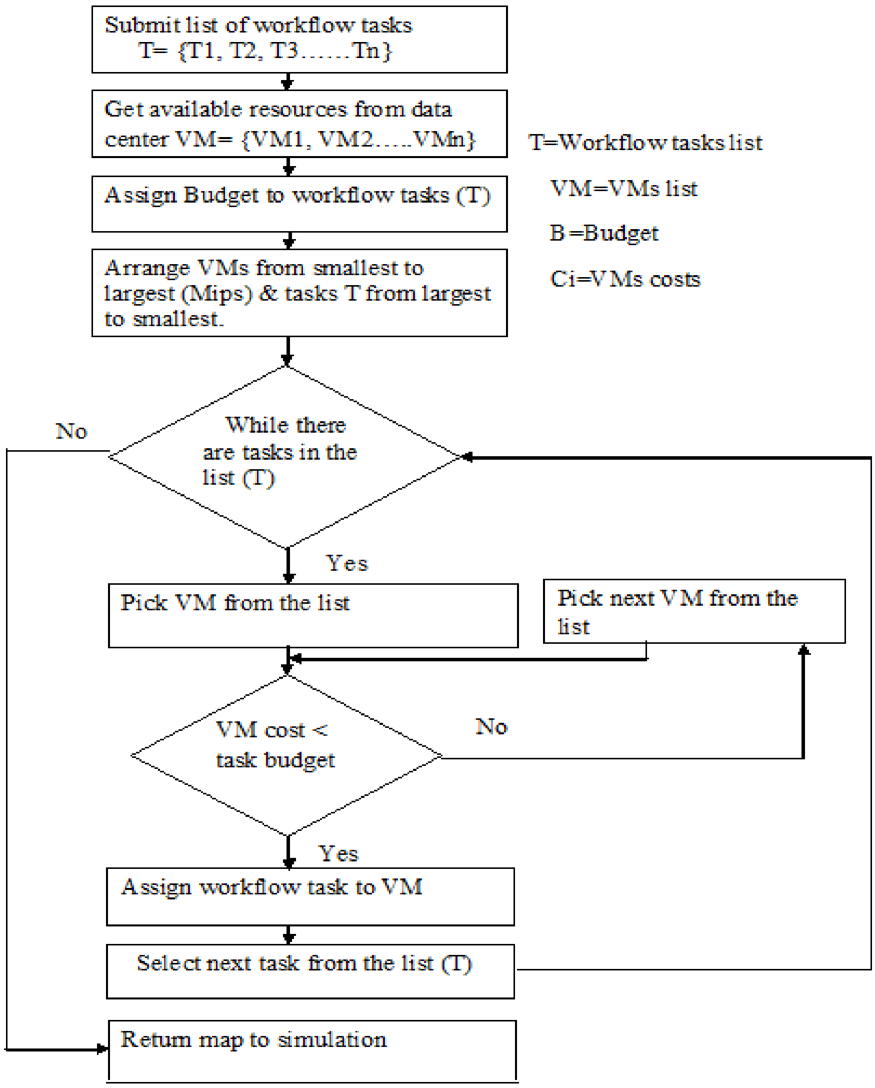

Figure 1: Basic Max-Min Budget Constrained Workflow Scheduling

Step by step description of score based Max-Min budget constrained workflow scheduling algorithm is presented below:

Algorithm SCORE_BASED_MAX-MIN (T, VM, B , C_i , S_v, S_t)
// T is Work Flow Tasks List, VM is Virtual Machines List, B is Budget, C_i is Virtual Machine Cost, S_v is VM Score, S_t is Task Score.
Step1:-Submit list of workflow tasks T :={ $T_1, T_2, \ldots, T_n$ }.
Step2:- Get available resources from data center VM :={ $VM_1, VM_2, \ldots, VM_n$ }.
Step3:- Assign Budget B to workflow tasks T.
Step 4:- Obtain Scores of VMs (S_v).
Step 5:- Obtain task scores (S_t) based on instruction length
Step6:- Arrange the VMs in ascending order and T in descending order.

Step7:- Repeat while T!= NULL
{
Step 7.1:- Pick VM from list that satisfy the task score.
Step 7.2:- Pick next VM from list that satisfy the task score.
Step 7.3:- If (VM Cost < Task Budget)
{
Step 7.3.1:- Assign workflow task to VM.
Step 7.3.2:- Select next task from list and Goto Step 7.
}
Else
Step 7.3.1:- Goto Step 7.2.
}
Step 8:- Return map to simulation.

The figure 2 shows the Flow Diagram of Score Based Max-Min Budget Constrained workflow Scheduling Algorithm

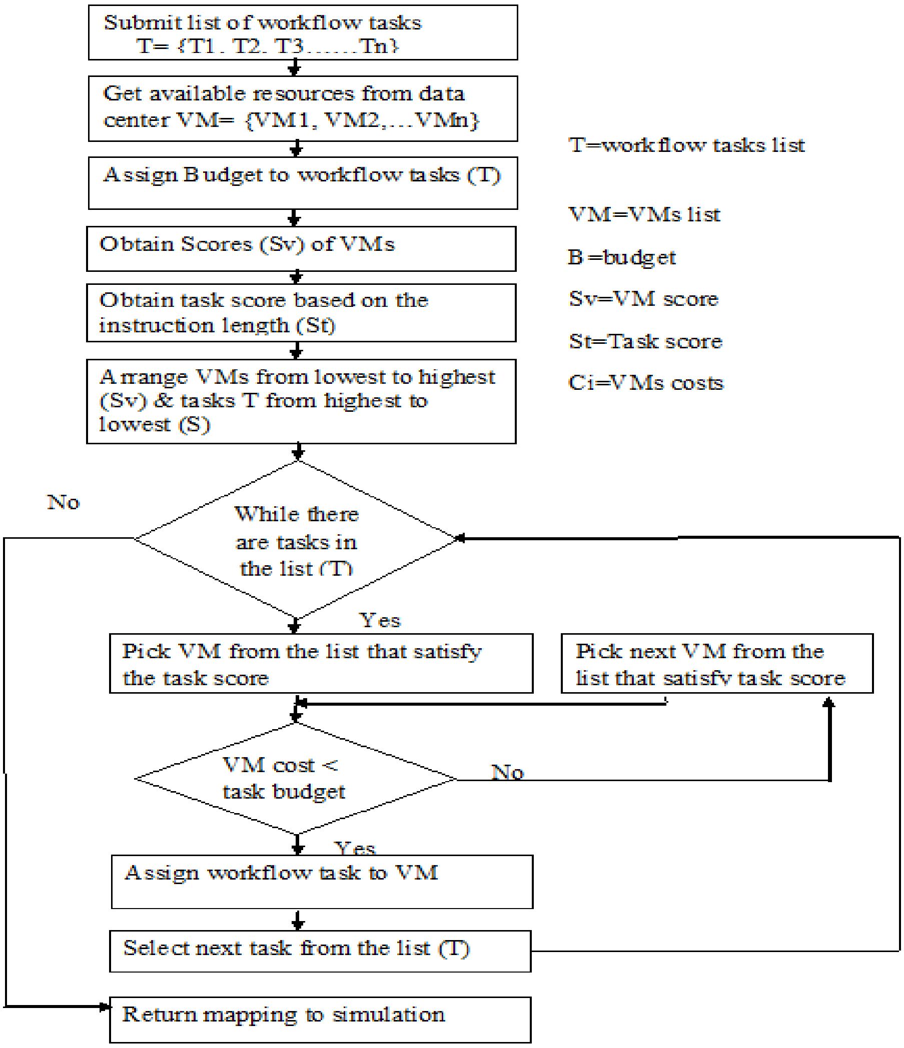

Figure 2: Score based Max-Min Budget Constrained Workflow Scheduling Algorithm

IV. RESULTS AND PERFORMANCE ANALYSIS

The software that has been used for the simulation of scheduling algorithm are Window 7, Java (JDK-6) , Eclipse-JUNO version and CloudSim-3.0 (Buyya, R., et al.. 2009). The parameters for Cloud Simulator are set as per Table 1.

Table 1: Parameters of Cloud Simulator

Type	Parameter	Value
Datacenter	Number of Datacenter	1
	Number of Hosts	4
	Type of Manager	Time-Shared
Virtual Machine(VM)	Total Number of VMs	30
	MIPS of PE(processing element)	1000-21000
	Number of PEs per VM	1
	VM Memory(RAM)	256-2048 MB
	Storage Space	1-21 GB
	Score	1-10
Cloudlet	Number of Workflow Application	1
	Number of Cloudlets(Tasks)	5-25

In the experimental results performance of score based Max-Min budget constrained workflow scheduling algorithm has been compared with basic Max-Min budget constrained workflow scheduling algorithm with respect to their execution time, execution cost and failure rate.

The test case in Table 2 shows the effect on execution time of Max-Min budget constrained workflow scheduling algorithm by varying the number of cloudlets when it is implemented with score and without score concept. Here cloudlet refers to workflow application tasks.

Table 2: Showing Execution Time and Number of Cloudlets

Max-Min Budget Constrained Workflow Scheduling Algorithm		
Number of Cloudlets	**Execution Time**	
	Max-Min (score)	**Max-Min(basic)**
5	2800	3006
10	4386	4960
15	6248	6625
20	8072	8536
25	9728	9945

Simulation results shown in Figure 3 indicate that score based Max-Min budget constrained workflow scheduling algorithm exhibit less execution time for the workflow application as compared to basic Max-Min budget constrained workflow scheduling algorithm.

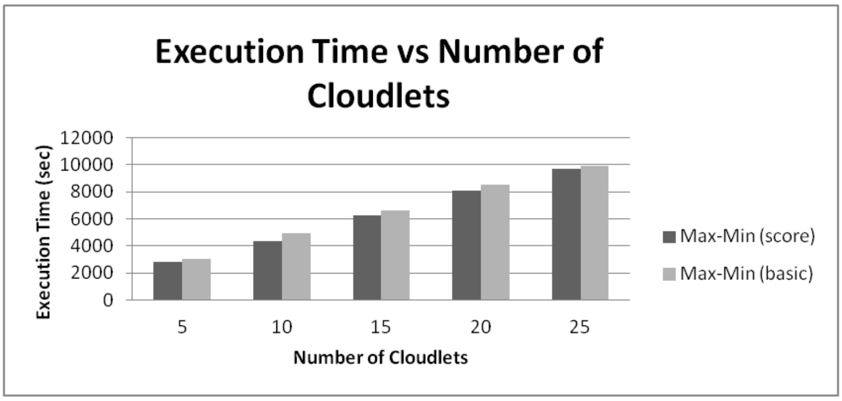

Figure 3: Showing Execution time vs. Number of Cloudlets

The test case in Table 3 shows the effect on execution cost of Max-Min budget constrained workflow scheduling algorithm by varying the user specified budget when it is implemented with score and without score concept. Here cloudlet refers to workflow application tasks.

Table 3: Showing Execution Cost and User Budget

Max-Min Budget Constrained Workflow Scheduling Algorithm		
User Budget	**Execution Cost**	
	Max-Min (score)	**Max-Min(basic)**
3000	2250	2015
6000	4970	4568
9000	8008	7590
12000	10645	9834
15000	13645	12800

The results obtained in Figure 4 indicate that the score based Max-Min budget constraint workflow scheduling algorithm although incur more cost compared to basic Max-Min but still execute the workflow application within user specified budget.

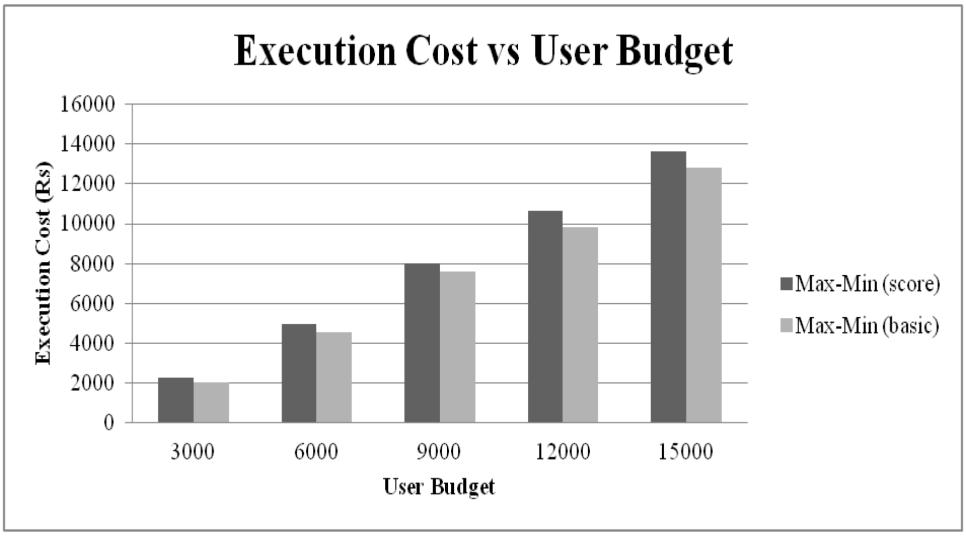

Figure 4: Showing Execution Cost vs. User Budget

This test case in Table 4 shows the effect on failure rate of Max-Min budget constrained workflow scheduling algorithm by varying the number of iterations when it is implemented with score and without score concept. Here cloudlet refers to workflow application tasks.

Table 4: Showing Failure Rate and Number of Iterations

Max-Min Budget Constrained Workflow Scheduling Algorithm		
Number of Iterations	**Failure Rate**	
	Max-Min (score)	**Max-Min(basic)**
10	0	0.1
20	0.1	0.3
30	0.12	0.3
40	0.12	0.45
50	0.12	0.5

Result obtained in Figure 5 shows score based Max-Min budget constrained workflow scheduling algorithm exhibit less failure rate as compared to basic Max-Min budget constrained workflow scheduling algorithm.

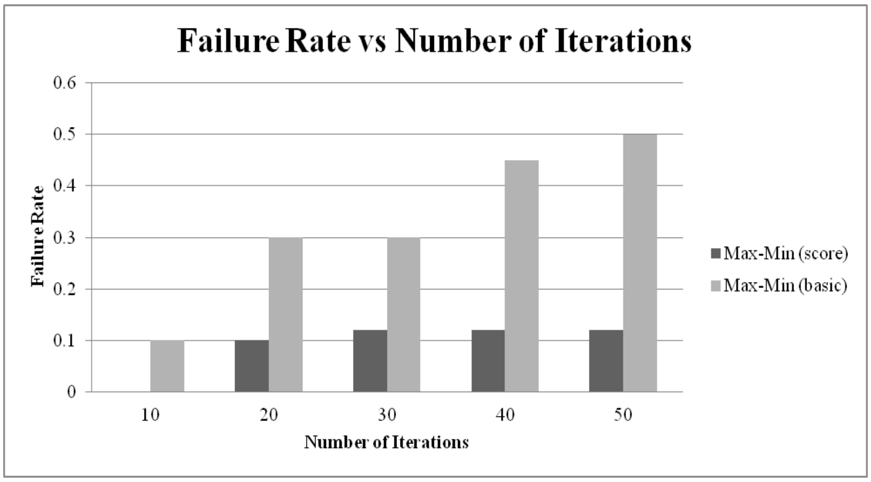

Figure 5: Showing Failure Rate vs. Number of Iterations

V. CONCLUSION

In this research work, we have proposed score based Max-Min budget constrained workflow scheduling algorithm. This algorithm schedule the workflow application within user specified budget constrained. The performance of score based Max-Min budget constrained workflow scheduling algorithm have been compared with basic Max-Min budget constrained workflow scheduling algorithm on the basis of following parameters i.e. execution time, execution cost, failure rate. Simulations results show that Score based Max-Min budget constrained workflow scheduling algorithm is more efficient than basic Max-Min budget constrained workflow scheduling algorithm. Score based workflow scheduling algorithm not only reduce the execution time of workflow application but also reduce the failure rate of workflow applications within user specified budget constraint.

REFERENCES

[1]. Abrishami, S., & Naghibzadeh, M. (2012). Deadline-constrained workflow scheduling in software as a service cloud. *Scientia Iranica*, 19(3), 680-689.

[2]. Abrishami, S., Naghibzadeh, M., & Epema, D. H. (2013). Deadline-constrained workflow scheduling algorithms for infrastructure as a service clouds. *Future Generation Computer Systems,* 29(1), 158-169.

[3]. Bala, A., & Chana, I. (2011, November). A survey of various workflow scheduling algorithms in cloud environment. *In 2nd National Conference on Information and Communication Technology (NCICT)*, 26-30).

[4]. Bittencourt, L. F., & Madeira, E. R. M. (2011). HCOC: a cost optimization algorithm for workflow scheduling in hybrid clouds. *Journal of Internet Services and Applications,* 2(3), 207-227.

[5]. Buyya, R., Ranjan, R., & Calheiros, R. N. (2009, June). Modeling and simulation of scalable Cloud computing environments and the CloudSim toolkit: Challenges and opportunities. *In 2009 international conference on high performance computing & simulation*, 1-11.

[6]. Jayadivya, S. K., Nirmala, J. S., & Bhanu, M. S. S. (2012). Fault tolerant workflow scheduling based on replication and resubmission of tasks in Cloud Computing. *International Journal on Computer Science and Engineering*, 4(6), 996.

[7]. Kaur, R., & Luthra, P. (2013). Load balancing in cloud computing. *Int. J. of Network Security*, 1-11.

[8]. Sidhu, A. K., & Kinger, S. (2013). Analysis of load balancing techniques in cloud computing. *International Journal of computers & technology*, 4(2), 737-741.

[9]. Van den Bossche, R., Vanmechelen, K., & Broeckhove, J. (2010, July). Cost-optimal scheduling in hybrid iaas clouds for deadline constrained workloads. In *2010 IEEE 3rd international conference on cloud computing*, 228-235.

[10]. Yu, J., Buyya, R., & Ramamohanarao, K. (2008). Workflow scheduling algorithms for grid computing. In *Metaheuristics for scheduling in distributed computing environments*, Springer, Berlin, Heidelberg, 173-214

[11]. Yu, J., Buyya, R., (2007). Workflow Schdeduling Algorithms for Grid Computing, *Technical Report, GRIDS-TR-2007-10, Grid Computing and Distributed Systems Laboratory*, The University of Melbourne, Australia.

Makhan Singh. "Score-Based Max-Min Budget Constraint Workflow Scheduling Algorithm for Cloud System." IOSR Journal of Engineering (IOSRJEN), vol. 09, no. 03, 2019, pp. 11-17.