International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 06 Issue: 08 | Aug 2019 www.irjet.net p-ISSN: 2395-0072

OPTIMIZATION OF COMPLETION TIME THROUGH EFFICIENT RESOURCE ALLOCATION OF TASK IN CLOUD COMPUTING BY ENHANCING THE GENETIC ALGORITHM USING DIFFERENTIAL EVOLUTIONARY ALGORITHMS

E. YASHASRI[1], K.M UMA MAHESWARI[2]

[1]E. YASHASRI & Department of Computer Engineering, SRM Institute of Science and Technology, Kancheepuram, Tamil Nadu, 603203, India.

[2]K.M UMA MAHESWARI, Assistant Professor, & Department of Computer Engineering, SRM Institute of Science and Technology, Kancheepuram, Tamil Nadu, 603203,

---

Abstract - *The primary purpose of our project is to develop technique for efficient resource allocation of a task in cloud computing that has a mono-objective aim which is to reduce the make span or completion-time. We are going to achieve this by a soft novel algorithm that is the "genetic algorithm". To enhance this objective parameter – minimization of completion time, we will combine genetic algorithm with the differential evolutionary algorithms. The laid down technical assumptions are that the directed acyclic graph which is represented by G (V, E), where the V (Vertices) is a set of nodes in the graph and E(Edges) denotes the precedence relations between the tasks. The nodes are weighted by the value of its computation time and the edges are weighted with the cost of communications between two tasks. If two tasks are assigned on the same processor the communication cost is zero. A genetic algorithm takes selection, mutation and crossover as its operations in the respective order, who's output acts as an input to the DE algorithm that has the same operations but in a reverse order of workflow. These double operational tasks refine the output and gives us the best task allocation that is required for the respective resource. This algorithm also handles real time parameters that eventually show an improvement by just working on the completion time.*

Key Words: **Resource Allocation, Genetic Algorithm, Differential Evolutionary algorithms, Cloud Computing, IaaS, Completion time, Load Balancing, DAG (Directed Acyclic Graph)**

1. INTRODUCTION

Cloud computing is a service which enables a delivery of computing resources over the Internet. Cloud computing is the latest trends in the world of Information Technology, The key word of cloud computing is the "cloud" itself, the exact meaning of the word cloud in this reference is that clouded cover computing is a group of web based computing resources that are delivered on demand by information services to users from any location in the world. The main idea of cloud computing is that it is a large group of dependent computers.

These computers can be personal computers or network servers. This service extends beyond a single company or enterprise. Cloud storage can be viewed as a model in which data is stored, managed and backed up remotely and made available to users over a network as web or typically the Internet. The service and the data served by the cloud are available to wide group of users. Access to these data are via the Internet. Only the authorized users can access these data's and the applications that are at any computer over any Internet connection.

In cloud computing, the allocation of resource is one of the processes used to assign the available resources to the needed application users, these are called the cloud resources they can be provisioned by either on-demand or a fine-grained or even in a multiplexer manner. In the cloud the resource allocation is mostly based on the following factors software, platform and the infrastructure respectively known as software as a Service (SaaS), Platform as a Service (PaaS) and Infrastructure as a service (IaaS).

In platforms of cloud, resource allocation mostly takes place at two major levels: The first level involves, when the cloud user wishes to be uploaded to the cloud, then the load balancer (which can by optimized) assigns the requested object to the remote physical computers, in order to balance the computational load of several applications across physical computers. The second level involves the application receiving multiple incoming requests, these requests should be assigned to an instance of application this is done to balance the computational load across a set of instances for the exact same application.

Our focus in this paper is to ensure the apt resource is assigned to the request both in the first and second level. Since computational cost will eventually be balanced through this method, we will be focusing on reducing the makespan. When a request is issued the load balancer also called as the resource monitor will use the proposed algorithm with the improved mathematical model and allocate resource that is fit for the job.

The second main constrain that we focus in this paper is to ensure that the resource in this case the processor is never idle. Thus, we will be reducing the idle time and increase the efficiency of the entire cloud. Decrease in the makespan will make both the computational cost and load balancing efficient. As mentioned before the allocation of the resources can be either on-demand or fine-grained. As we will be dealing with dynamic requests, the on-demand technique is implemented in the algorithm.

The enhanced Algorithm model will improvise the Improved genetic algorithm and provide with better completion time than the existing genetic algorithm. Genetic algorithm is a biological concept that is used for generating population. It is inferred by Darwin's theory of evolution. In his theory Darwin the principle of "Survival of the fittest" is used as the method of scheduling in which the tasks are assigned to resources according to the value of fitness function for each parameter of the task scheduling process.

The three main principles which are used for in this algorithm are Initial population, Fitness function, Selection. The main techniques used would be Crossover, Mutation, Selection. The techniques are used in the traditional genetic algorithm to produce the best out of the lot, which can be mapped to the fittest value in the N number of iterations. Initial population points to the number of individuals or the resources that are used in the GA to find the optimal solution. The fittest value will be the maximum completion time of the resources that is apt for the provided request, this is selected criteria from the initial population.

Therefore, in this paper we will be changing the mathematical criteria required for the crossover and mutation and selection of the fittest value. The Differential evolutionary algorithm is like the GA, the steps are vice versa. Hence when we implement both these algorithms together, we will get the optimized combination of task to resource. The corresponding goal to optimize the completion time will be achieved. In the proposed system we will be running both the Improved and the Enhanced genetic algorithm with Differential evolutionary algorithm and tabulate and represent a graph for both the make span and cost for each of the algorithms to prove that optimization has occurred.

2. ARCHITECTURE

Cloud is the solution with dynamic way to scale up and scale down the services when required. It is based on the concept of implementing virtualization that includes making a layer between the resources (OS or VM) and host machine. It improves the concept of decoupling the dependency of OS and hardware.

A client sends a request to the cloud that can be broken into individual tasks. Every task is an interdependent sub-task that has its output to be the input for another task. That's why, it is very important in the perspective of the cloud to assign the right virtual machine to the unique request that has been raised by the client.

A cloud consists of processors or VM layer, under which we have the Virtualization layer, that is followed by hardware or servers' layer remotely located. A cloud also consists of a resource monitor that interacts between the enhanced genetic algorithm and gives the apt resource allocation to the client as per their requirement.

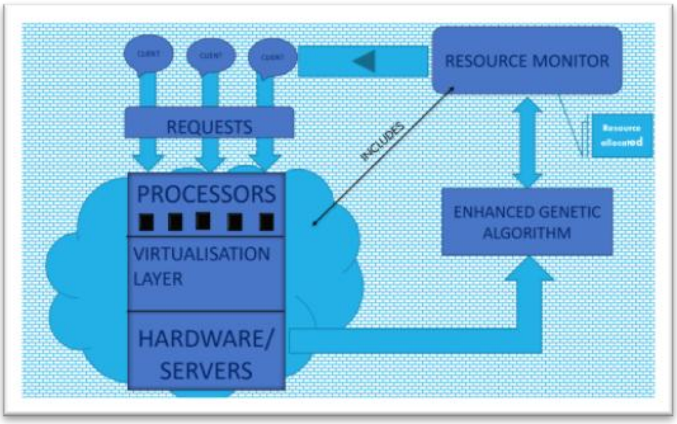

Figure 1: Architecture

3. RELATED WORK

When job scheduling itself is viewed it is vital to understand the conservation. Job scheduling model for cloud computing [1] describes on detail about energy consumption, profit of service, to recognize decision to analyze the applications. These parameters resulted in a fruitful output which benefited in many ways. The use of scheduling algorithms with the above criteria eventually helped in an optimized dynamic selection of jobs alias the tasks.

The basic study of Genetic algorithm based on resource allocation for cloud computing describes about the scheduling procedure that is done using the Virtual machine that uses the information of tasks to make the task scheduling, in this the model included the type of request as a main parameter for task scheduling. The main aim of the algorithm was in creating a criterion for task scheduling using quality evaluation.

Research of cloud computing task scheduling algorithm based on improved genetic algorithm [3] uses the algorithm energy aware improved genetic algorithm for task scheduling in the cloud computing focuses on the make span to optimize the cost, also to provide a quality of service with conserved energy. These three principles were the parameters which differed from the traditional genetic algorithm. The process implemented was using the voltage dynamic scheduling. This optimized energy conservation in a new level.

Eventually mono objective systems if not focused on the right systems will tend to create drawback in other aspects. Hence Genetic algorithm based bi-objective task scheduling in hybrid cloud computing [4] aimed on optimizing both the execution time and the execution cost for a hybrid system. This simultaneous optimization of the execution time and the cost of the scheduling in hybrid cloud provided a new perspective on task scheduling in a cross cloud.

Understanding the various types of resources that are in a virtualized form in a cloud, that are required for the dynamic scheduling is projected in Various priority-based task scheduling algorithms in cloud computing[5]. Analysis the types of resource provides the priority of scheduling this was captured and optimized. In Resource allocation in cloud computing [6] the take on the difference in the various software and hardware resources for multiple allocation algorithms was performed which gave a result of choosing the best duo for ensuring efficient resource allocation.

In order to achieve higher performance and better utilization of resources with minimum response time, Cloud pad computing should be able to execute maximum number of tasks for any type of complicated tasks in any environment. The paper Survey on scheduling algorithms in cloud computing and the paper A survey: to harness an efficient energy in cloud computing both suggest this technique.

Hybrid algorithms provided easier workflow; this was presented in workflow scheduling using hybrid GA-PSO algorithm in cloud computing [9]. In this paper the main aim was to increase make span and ensure load balance and optimize the fittest value calculation. This was achieved with integration of Genetic algorithm and Particle Swarm Optimization algorithms for improving the cost reduction and make span along with load balance on virtual machine. This however could be extended for dynamic resource allocation too.

Cloud task scheduling algorithm based on improved Genetic algorithm paper aimed to optimize both the execution and the computational cost of resource for efficient allocation. This work involved changed mathematical model than to that of a traditional Genetic algorithm. The usage of the three-state selection method helped converting the simple genetic algorithm to improved genetic algorithm, for improving the completion time. Hence the main goal of this work was to improve the completion time. This work is considered as the base for our proposed system. Similar to this work but implementing different mathematical model with different set of parameters and dataset we will achieve the objective of efficient flowtime and reducing the idle time.

Cloud computing, the longest hope for using a utility for computing has helped potentially convert a great part of the IT industry, it has enabled the software more portable and services more affordable. The major differentiation of cloud computing from other distributed computing techniques is the on-demand resource allocation. Something like the allocating of Electric power.

After reviewing several research papers, we've gained a fair amount of knowledge on the various algorithms and techniques implemented to increase the efficiency of the dynamic allocation of resources. With the knowledge gained by the data provided we hope to implement these ideas in the process of achieving our goal in project.

4. MODULES

- Datacentre configuration
- Production of Initialization
- Improved Genetic Algorithm
- GA-Differential Evolution Method (Proposed)
- Result and Analysis

5. MODULES DESCRIPTION

5.1 DATACENTER CONFIGURATION

- Input: Hosts, Vms, Tasks - configurations and parameter initialization.
- Output: Datacentre Environments.

In this module, the data center environments are configured by giving the values of MIPS, memory, storage, bandwidth, number of cores on an individual host and virtual machines and the number of servers (hosts) and virtual machines. And the workflow(DAG) tasks are submitted to schedule and execute on the datacenters. Also, the parameters for initialization and task scheduling values are entered.

5.2 PRODUCTION OF INITIAL POPULATION

In this module, in order to improve the quality of the initial population, Min-Min and Max-Min algorithms are introduced to initialize the population.

Min-Min Algorithm: Min-Min scheduling is based on the concept of assigning a task having minimum completion time (MCT) first for execution on the resource, which has the minimum completion time (fastest resource). This algorithm is divided into two steps. In first step, expected completion time of each task in the metatask is calculated on each resource. In second step, the task with minimum expected completion time is selected and assigned to the corresponding resource, and then the selected task is removed from the metatask. This process repeats until all the tasks in metatask get mapped. Min-min algorithm is a simple algorithm but it gives the fast result when the size of task in metatask is small as compared to large size task. On the other hand, if large size tasks overlay the number of smaller tasks, it gives a poor resource utilization and large make span because large size tasks have to wait for the completion of smaller tasks.

Max-Min Algorithm: Max-Min algorithm performs the same steps as the Min-Min algorithm but the main difference comes in the second phase, where a task ti is selected which has the maximum completion time instead of minimum completion time as in min-min and assigned to resource Rj, which gives the minimum completion time. Hence it named as the Max-Min algorithm. This process is repeated until the metatask get empty or all the tasks are mapped.

The main aim of max-min scheduling algorithm is to reduce the waiting time of large size jobs. In this algorithm, small size tasks are concurrently executed with large size tasks, hence reducing the make span and providing better resource utilization.

5.3 IMPROVED GENETIC ALGORITHM:

There are three main operations of simple genetic algorithm for population: Selection, Crossover, and Mutation.

Selection operation: in each generation, individuals who adapt to the environment are selected by the fitness function, and the next generation is produced through these individuals.

Selection of Fitness Function

In genetic algorithm, the selection of fitness function is very important, and it is directly related to the convergence of the algorithm and the search of the optimal solution.

$$f = \max (TVi) \qquad i \in \{0,1,2,\ldots,M-1\} \quad \text{--- (1)}$$

Where, TVi represents the time that the virtual machine i execute all the cloud tasks assigned to it, and M represents the number of virtual machines.

Selection operation(mathematically)

The selection operation is to select the good individuals and eliminate the bad individuals. The average fitness fa and the random fitness fr are introduced into the selection operation.

$$sum = \Sigma f(i) \; i \in \{1,2, 3,\ldots,n\} \qquad \text{--- (2)}$$

$$fa = sum / n \qquad \text{--- (3)}$$

$$fr = fa.\ Math.\ Random () \qquad \text{--- (4)}$$

In these equations, sum represents the total finishing time, including all individuals in this population, f(i) is the fitness of individual i, and n is population size, by formula (3) and (4) we can see fa>fr.

The three-stage selection method is adopted in this paper; each selection retains the best individual of the contemporary population and makes different grades for other individuals. If the individual fitness is f<fr, the individual is defined as a excellent individual and retained; Individuals with fitness fr<f<fa are defined as sub excellent individuals and replaced by variants of the optimal individual; Individuals with fitness f>fa are poor individuals, these individuals will be eliminated and replaced by new individuals. This not only ensures the stability and diversity of the population, but also helps to find the global optimal solution.

Crossover operation

Two different individuals in a population exchange gene in the same position to produce new individuals. Suppose the individual A crosses with the individual B, and the crossover region length is L, in this region, the number of same genes of two individuals is K, and the crossover region similarity is: S = K/L

If the crossover region length is 9, the genes LA=022130111, LB=012112031, then K=4, crossover region similarity S=0.44, the crossover region similarity threshold μ=0.5 is specified in this paper. Two individuals perform crossover operation, and when the crossover region similarity is greater than the threshold, we decide that the crossover operation is meaningless and the two individuals will reproduce in asexual form: individual recombine genes randomly in crossover region then add it to the next iteration.

Mutation operation

The genes of certain individuals in the population may change, which may make individuals more adaptable and, of course, weaker.

5.4 GA-DIFFERENTIAL EVOLUTION METHOD (PROPOSED)

In the proposed algorithm, the operators of Genetic Algorithm and Differential Evolution Method are combined to find the optimal solutions to minimize the make span of the tasks execution and find the best task schedule in the cloud environments. The directed acyclic graph which is represented by G(V, E), where the V(Vertices) is a set of nodes in the graph and E(Edges) denotes the precedence relations between the tasks. The nodes are weighted by the value of its computation time and the edges are weighted with the cost of communications between two tasks. If two tasks are assigned on the same processor the communication cost is zero. The directed acyclic graph contains the tasks are T0, T1, T2, . . . , T10 , and T0, is entry task and T10 is exit task. In this proposed algorithm the Genetic Algorithm (GA) is applied, their selection, crossover and mutation operations and produce a set of solutions, these solutions are considered as an initial population to Differential Evolution(DE) method and the DE operations applied on it and produce a solution as next population to GA. This operation processes are repeated for a number of times and when the DE terminate the operations all the solutions are updated. To avoid the priority violence, it must be sorted from left to right, but when the new solutions priority is violated after the production of children. In this algorithm the premature convergence is avoided by comparing the produced children with their parents, if the fitness value is better than the parents then the parents are replaced by children else it is aborted. Once the initial population is generated, the fitness values of the solutions are evaluated. In each iteration of the algorithm, the termination condition is checked. If the termination condition is satisfied, then the algorithm is produced optimal solutions. Otherwise, GA and DE algorithms operators applied to the individuals respectively. The GA-DE algorithm implements the operations as:

1) Production of initial population

The initial population contains individuals and the sizes of the chromosomes are fixed. In this method, the initial population is considered for having a better solution from MIN-MIN and MAX-MIN method. The priorities found for the DAG tasks are considered as the initial population. The remaining of the individuals are generated randomly. The first and last places in the chromosome which they related to the start and exit nodes are established in the chromosome, respectively and the remaining tasks are randomly generated and they are sorted selected from left to right in a way that priorities are not violated the precedence constraints.

2) Selection Operator

The roulette wheel selection function is used to select an individual for a good fitness values and the genetic operations are implemented on the selected individuals. The individual which is having high fitness value gets more chances to be selected.

3) Crossover Operator

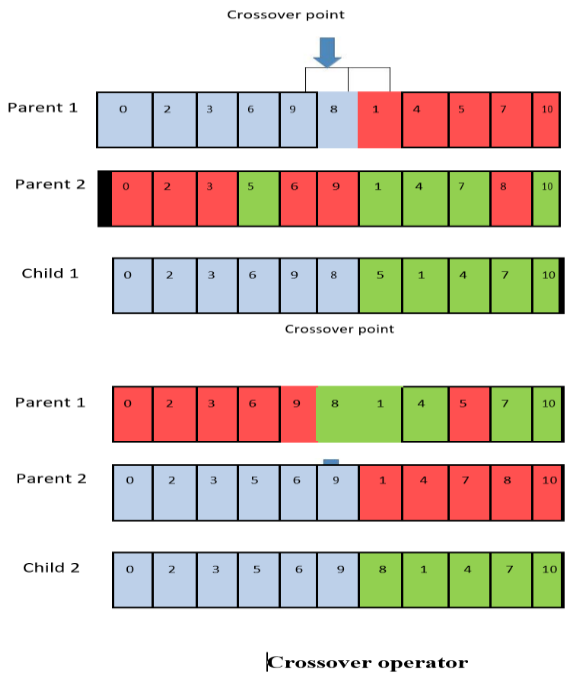

Figure 2: Crossover Operator

Crossover replaces same genes of one individual with genes of another to produce two valid offspring. The powerful single-point crossover operator is used. A random point between l and n is generated and the crossover is applied if the genes of both parents from entry node to crossover point are not identical. The crossover single point which is equal to 5 produces the two new offspring. The left side of the children inherit from their ' genes of parents in the same genes location then selected genes are removed from the parent and the remaining genes are imported to the child from left to right. And the produced children will be valid. After producing the children, the fitness of them obtained by fitness function. Then the fitness values of children are compared with parents and if fitness values of children are better than parents, the children will replace parents.

4) Mutation Operator

Mutation operator obtain a new chromosome by changing two genes in a way that precedence constraint is not violated. In this operator, at first, a gene is selected randomly. Then, first successor of the selected task (tj) from the mutation point to the end is found. If there is m th gene which it is a member of [i+1, j-1] and the predecessors of tm are not in front of ti, ti and tj can be changed with each other. If in mutation function these conditions do not happen, so, the mutation operator algorithm will be run from the beginning. Finally, the fitness value of a child is measured for the child and if the fitness value of the child betters the parent's fitness then the child will replace the parent.

5) Termination Conditions

The genetic and DE algorithms are considered as evolutionary algorithms, which can be executed for unlimited time. But for achieving solutions in the algorithm we need to terminate processes. Hence, termination condition must be considered. Some of the policies to terminate algorithms are considering the fitness evaluations or the running times of the system or by exploring the population diversity. In this paper, the algorithm is terminated when the algorithm is iterated for 30 times.

Table 1: Parameters

Number of Hosts	100
Number of Virtual Machines	40
Selection Technique	Roulette wheel
Crossover Rate	Single point
Mutation Factor	m gene → {i+1, j-1}

5.5 RESULT AND ANALYSIS

[1] Input: Number of Tasks on X-axis and Completion Time (Make span) on Y-axis

Output: Comparison of Make span Versus the Number of Tasks

A graph is drawn between the number of tasks on X-axis and the corresponding make span of the applications on Y-axis. The Genetic Algorithm-Differential Evolution algorithm is compared against the existing IGA scheduling algorithms in terms of the make span by 100, 200, 300, 400 and 500 tasks. The number of tasks increases in the graph, the proposed algorithm shows the better optimal make span values compared to existing improved Genetic Algorithms.

Table 2: Optimization of Execution time

ALGORITHMS	NUMBER OF TASK 30	NUMBER OF TASK 50	NUMBER OF TASK 100
SIMPLE GENETIC ALGORITHM	675.25 seconds	798.67 seconds	827.36 seconds
IMPROVED GENETIC ALGORITHM	413.677 seconds	520.12 seconds	627.067 seconds

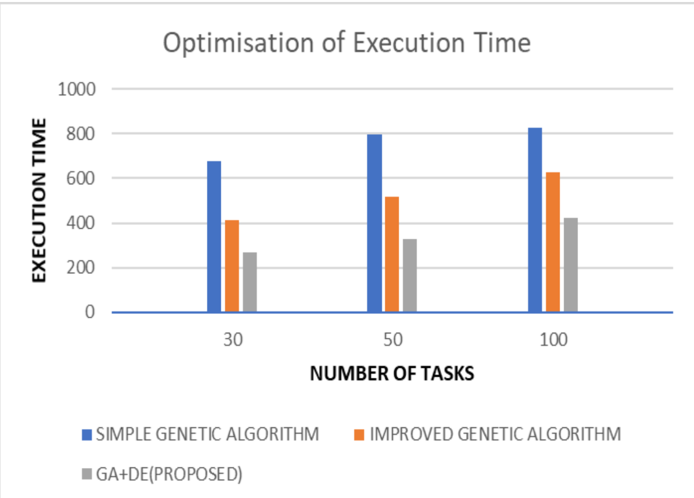

Figure 3: Graph for Optimization of Execution time (Table 2)

[2] Input: Number of Tasks on X-axis and cost of expenses on Y-axis.

Output: Illustrated a chart of comparison of cost and number of tasks.

Another graph is plotted between the number of tasks and the cost of expenses to execute the tasks in the allocated data Centre based on the utilization of resources. It shows the minimization of the cost of expenses compare to the existing algorithm.

Table 3: Comparison of Execution Cost

ALGORITHMS	NO OF TASKS (30) COST	NO OF TASKS (50) COST	NO OF TASKS (100) COST
SIMPLE GENETIC ALGORITHM	$ 1020.567	$ 1700.53	$ 1940.632
IMPROVED GENETIC ALGORITHM	$ 784.98	$ 540.315	$ 971.954
ENHANCED GENETIC ALGORITHM WITH DE ALGORITHM	$ 418.886	$ 501.448	$ 651.659

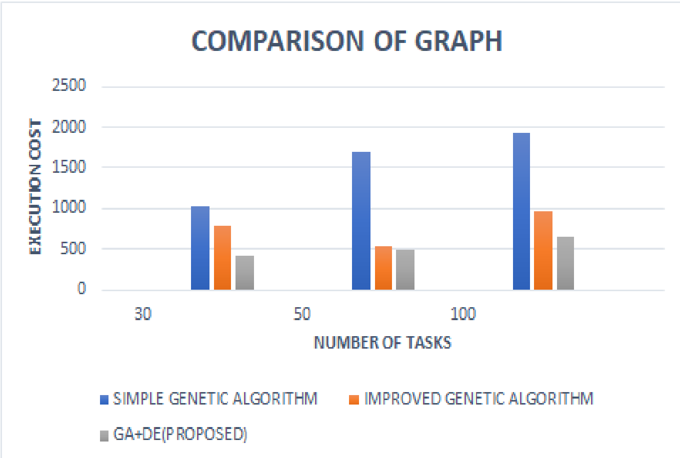

Figure 4: Graph for Comparison of Cost (Table 3)

6. CONCLUSION

The completion cost has reduced to a great extent when compared to simple genetic algorithm and improved genetic algorithm. The execution cost has reduced to a large extent when compared to the SGA and IGA. By achieving these two goals we can conclude by stating, the resource allocation of the tasks in a cloud environment has been optimized.

7. REFERENCES

1) Mizan, M. and Latip. “Modified bees life algorithm for job scheduling in hybrid cloud.” International Journal of Engineering and Technology, 2, 974–9.
2) Leena V. A., A. B. A. S. and S., R. M. “Genetic algorithm based bi-objective task scheduling in hybrid cloud platform.” International Journal of Computer Theory and Engineering.
3) Junwei G, Y. Y. “Research of cloud computing task scheduling algorithm based on improved genetic algorithm.” In Proceedings of 2nd International Conference on Computer Science and Electronics Engineering, 2134–2137.
4) Here Comes Hybrid Cloud, Gartner. http://blogs.gartner.com/thomas_ bittman/2012/09/24/mind-the-gap-here-comes-hybrid-cloud/. Accessed: 2016-03-29.
5) Ms. Pooja, P.Vasani, Mr. Nishant And S. Sanghani, (2013) "Literature Review: Various Priority Based Task Scheduling Algorithms In Cloud Computing," Journal Of Information, Knowledge And Research In Computer Engineering, (2) (2), 298-302.
6) Shivani Sharma and Dhanshri Parihar, (2014) "A Review on Resource Allocation in Cloud Computing," International Journal of Advance research, Ideas and Innovations in Technology.
7) M.Padmavathi, Shaik. Mahabbob Basha and Srinivas Pothapragada, (2014) "A Survey on Scheduling Algorithms in Cloud Computing," Journal of Computer Engineering (IOSR-JCE), (16) (4), 27-32.
8) Malathi.P And Arumugam.S, (2015) "A Survey: To Harness An Efficient Energy In Cloud Computing," International Journal Of Ubicomp (Iju), (6) (3), 1-7.
9) Workflow Scheduling Using Hybrid GA-PSO Algorithm in Cloud Computing Ahmad M. Manasrah1 and Hanan Ba Ali2 Wireless Communications and Mobile Computing Volume 2018, Article ID 1934784.
10) Cloud Task Scheduling Algorithm based on Improved Genetic Algorithm Hu Yao, Xueliang Fu*, Honghui Li, Gaifang Dong, Jianrong Li College of Computer and Information Engineering, Inner Mongolia Agricultural University, Hohhot, 010018, China. Available online at www.ijpe-online.com vol. 13, no. 7, November 2017.