© 2018 IJSRSET | Volume 4 | Issue 1 | Print ISSN: 2395-1990 | Online ISSN : 2394-4099
Themed Section : Engineering and Technology

Resource Utilization of Workflow Scheduling Algorithms in Public Cloud

Dr. T. Lucia Agnes Beena
Assistant Professor and Head, Department of Information Technology, St. Joseph's College, Triuchirappalli, Tamil Nadu, India

ABSTRACT

The advent of Web 2.0 and Internet, transform the Grid Computing to Cloud Computing. The growing reputation of cloud computing has fascinated many IT organizations to move towards the Cloud which widens the Cloud Data centers. The raise in number of Data centers influence the need for the energy efficiency in Cloud. The energy efficiency can be achieved through energy-aware resource management, development of efficient policies and algorithms for virtualized data centers and eco-friendly technology. This paper focuses the resource utilization parameter of the resource management by efficient mapping of applications to Cloud resources. The algorithms discussed in this paper reduce the application execution time by assigning the appropriate resources, there by maximizing the resource utilization. Among the various algorithms discussed, Differential Evolution Algorithm for Workflow Scheduling in Public Cloud outstrips the other algorithms in minimizing the execution time of the application, minimizing the cost of executing the application and maximizing the resource utilization.

Keywords : Cloud Computing, Resource Utilization, Tasks Scheduling, Workflow Scheduling

I. INTRODUCTION

Cloud Computing provides various Services to user on pay-per-use-basis. The benefits of the Cloud attract many users day by day. As the demand for Cloud Services increases, the need for effective Cloud resource utilization becomes a vital issue. Efficient and appropriate allocation of resources to the Cloud applications results in better resource utilization.

Cloud Services are not restricted to a particular geographical area; it provides services across the different Countries of the World. Many computing service providers including Google, Microsoft, Yahoo, and IBM 2 are rapidly deploying data centers in various locations around the world to deliver Cloud computing services [1]. Infrastructure as a Service (IaaS) is one of the Services provided by the Cloud. IaaS involves managing resources like processing, storage, networks and other fundamental computing resources where the consumer is able to deploy and run arbitrary software, which can include operating systems and applications. Efficient management of the resources increase the number of cloud users. Hence there is a better profit for the Service Providers.

The competent resource organization involves two processes, namely, Scheduling and Resource allocation. Scheduling refers to mapping of application to the resources. Resource allocation involves effective, appropriate selection of resources that minimizes the application execution time as well as increasing the percentage of resource utilization. The effective scheduler can increase resource utilization, reduce queue waiting time and reduce operational costs. This paper analysis the resource utilization percentage of workflow scheduling algorithms such as Customer Facilitated Cost based Scheduling in Cloud (CFCSC) [2], Level Based Task Prioritization Scheduling (LBTP) [3], Workflow Scheduling for Public Cloud Using Genetic Algorithm (WSGA) [4] and Differential Evolution Algorithm for Workflow Scheduling in Public Cloud (DEWS) [5].

II. STATE OF THE ART

Zomaya et. al. [6] proposed two energy-conscious task consolidation heuristics, that maximize resource utilization using Task Consolidation method. These algorithms also save energy consumed by the Data Center, which indirectly reduces the operation costs. The task scheduling algorithms proposed by Sindhu et. al. [7] aims to schedule the tasks effectively. These algorithms reduce the turnaround time and improve resource utilization. The proposed algorithms exhibit good performance under heavy loads.

Sometimes modern virtualization technologies may affect the resource allocation policy. The aggressive consolidation and variability of the workload sometime does not allocate appropriate requested resource for the Virtual Machines (VMs). This results in performance loss in terms of increased response time, time outs or failures in the worst case. Hence, the Cloud providers have to concentrate on energy-performance trade-off. Buyya et. al. [8] solved this problem by performing the VM reallocation in two steps. The first step selection of VMs to migrate is considered separately for each optimization stage. In the second step, determining new placement of the selected VMs on physical hosts, are solved by application of a heuristic for semi-online multidimensional bin-packing problem. At the first step, the utilization of resources is monitored and VMs are reallocated to minimize the number of physical nodes in use and thus minimize energy consumption by the system.

Shakkeera et. al. [9] proposed a load balancing strategy for better resource utilization based on the various Quality of Service (QoS) performance metrics like cost, average execution time, throughput, CPU usage, disk space, memory usage, network transmission and reception rate, resource utilization rate and scheduling success rate for the number of virtual machines, and it improved the scalability among resources. Buyya et.al. [10] presented and evaluated energy-aware resource allocation algorithms utilizing the dynamic consolidation of VMs. Their experiment results have shown that their approach leads to a substantial reduction of energy consumption in Cloud data centers. This paper attempts to maximize the resource utilization of the Data Center by using static scheduling policies.

III. COMPARATIVE ANALYSIS OF THE WORKFLOW SCHEDULING ALGORITHMS

With the rapid advance of cloud computing, large scale data center plays a key role in cloud computing. Energy consumption of such distributed systems has become a prominent problem and received much attention. Among existing energy-saving methods, application scheduling can reduce energy consumption by replacing and consolidating applications to decrease the number of running servers. However, most application scheduling approaches did not consider the energy cost on network devices, which is also a big portion of power consumption in large data centers. Scheduling Algorithm for applications (business and scientific applications) to minimize the energy consumption of both servers and network devices can be developed [11].

Most of the business and scientific processes can be represented in terms of workflow. So a workflow can be described as the set of tasks which is used to complete some business process. Task invocation, task synchronization, and information flow are done in a specific order which is described by workflow management. The tasks of workflows are varying in nature. The main issue in workflow management system is workflow scheduling. The mapping and management of workflow task's execution on shared resources is done with the help of workflow scheduling.

One of the most challenging problems in Clouds is workflow scheduling, i.e., the problem of satisfying the QoS of the user as well as minimizing the energy consumption of workflow execution. Workflow scheduling is the problem of mapping each task to a suitable resource and of ordering the tasks on each resource to satisfy some performance criterion. The traditional scheduling methods try to minimize the execution time (makespan) of the workflows. However, in Clouds, there are many other potential QoS attributes such as execution time, reliability, security and availability. Along with these parameters resource utilization is also to be concentrated in the Cloud workflow scheduling algorithms. Due to the complexities of the development of a general multi-objective scheduling algorithm, many researchers try to propose bi-criteria scheduling algorithms [12].

This paper analyses the resource utilization of four different algorithms namely, CFCSC [2], LBTP[3], WSGA[4] and DEWS[5]. All the four algorithms are developed with the objective of minimizing the makespan (execution time) by maximizing the resource utilization. The CFCSC algorithm is a list scheduling algorithm. The list scheduling algorithms have two phases: the prioritizing phase, assigns a priority to each task and a processor selection phase allocates a suitable processor that reduces the heuristic cost function. The CFCSC algorithm, in the prioritizing phase, weights are assigned to the nodes and the edges in the first step. In the next step, the upward rank calculation is made. Then the nodes are sorted in the descending order using upward rank for prioritization. At the same time, the virtual machines are sorted in the ascending order based on their prices. This is done to balance the load and to utilize all the reserved resources in an efficient way. As a final step, the algorithm calculates the Modified Earliest Finish Time (MEFT) for all the tasks in every virtual machine and selects the virtual machine with minimum MEFT value, preserving the precedence constraints.

The LBTP is also a List Scheduling algorithm. In LBTP algorithm, for task prioritization, the Directed Acyclic Graph (DAG) is divided into different levels based on the task dependency with the entry node. At each level, the tasks are prearranged according to their computation cost in descending order. When more than one task have the same computation cost, the task with higher communication cost is given higher priority. In case of the same communication cost, the tasks are arranged in the topological order. In the resource selection phase, each task in the execution order list (prepared in the task prioritization phase) is assigned to the best resource to minimize the total completion time. The LBTP algorithm, applies three different procedures, namely, Earliest Finish Time (EFTi), Parent Resource Allocation (PRA) and Round Robin (RR) to find out the best possible allocation of resource to the task.

The WSGA is a meta-heuristic algorithm based on Genetic algorithm (GA). As the objective of this algorithm is to minimize the makespan and cost of executing the workflow by maximizing the resource utilization, the fitness function f(i) for the WSGA is given by equation 1.

$$f(i) = 1/(1 + (MS * AVGCOST)) \qquad (1)$$

where MS is the makespan, AVGCOST is the average resource cost of the virtual machines used to execute the chromosome i. The selection method, Roulette wheel and crossover method, single point crossover are used in this algorithm. After conducting experiments, the crossover rate and mutation rate are found to be 0.7 and 0.1 respectively. The results are converged in 200 iterations. From the initial population, after applying the reproduction operations like selection, crossover and mutation for 200 generations, the results have been observed.

The DEWS is a Differential Evolution Algorithm (DEA) that is a population based algorithm like genetic algorithm using the similar operators like crossover, mutation and selection. The fitness function for the DEWS algorithm is given as equation 2.

$$f(x) = Alpha*(MS_{maximum} - MS_{current}) + (1 - Alpha)*(C_{maximum} - C_{current}) \quad (2)$$

where $MS_{maximum}$ is the highest makespan value in the current generation, $MS_{current}$ is the makespan of the current schedule, $C_{maximum}$ is the cost of the schedule whose makespan is the highest in this generation and $C_{current}$ is the cost of the current generation. Alpha is a cost-efficient factor that represents the user's preference for the makespan and the cost. The value of Alpha ranges between 0 and 1. For the DEWS algorithm the Alpha value is varied from 0.5 to 0.9 in steps of 0.1 and it is found that 0.6 for Alpha optimizes schedule for the given workflow. The mask mutation operator Θ is used in this algorithm and hence $Q_a^G = P_b^G \Theta F(P_c^G \Theta P_d^G)$. Since the mutation scaling factor F in DEA is not applicable for workflow scheduling problems [13], F is considered to be one. Each mutant vector recombines with its respective parent through crossover operation to produce its final offspring schedule. The schedule is produced based on the crossover rate CR which is between 0 and 1. $R_j^G = Q_j^G, \ if \ rand(0,1) \leq CR$. Otherwise $R_j^G = P_j^G$. Then, a selection operation is performed. The fitness function value of each trial vector $f(R_j^G)$ is compared to that of its corresponding target vector $f(P_j^G)$ in the current population. If $f(R_j^G)$ of the trial vector is less than or equal to $f(P_j^G)$ (in a minimization problem) of the corresponding target vector, the trial vector will replace the target vector and enter the population of the next generation. Otherwise, the target vector will remain in the population for the next generation. By repeating the experiment from 1st generation to 300th generation, it was found that the optimal schedule for the given workflow is achieved in 100 generations.

In all the four algorithms, Resource utilization $RU(R_i)$ of resource R_i is calculated using equation 3

$$RU(R_i) = Makespan - \sum_{j=1}^{k} Idletime_j \ (R_i) \quad (3)$$

where $\sum_{j=1}^{k} Idletime_j \ (R_i)$ is the sum of all idle time slots of resource R_i. The average resource utilization (ARU) of all resources gives the overall utilization percentage of the Cloud resources. It is specified by equation 4.

$$ARU = \frac{(\sum_{I=1}^{p} RU(R_i))}{p} * 100 \quad (4)$$

where p is the total number of resources.

IV. RESULTS AND DISCUSSION

All the algorithms discussed in this paper are developed using Java in the Netbeans IDE 7.1 environment. The input for the algorithms is the arbitrary task graph generated by DAGEN Tool developed in Java [14]. The DAGEN Tool generates the needed virtual machine instance with various speeds randomly. The number of tasks to be generated and the number of virtual machines needed for executing the workflow are given as input. The Tool generates the arbitrary task graph as output. It is possible to generate Computation Intensive task graphs, Communication Intensive task graphs and mixed task graphs using this DAGEN Tool. The experiment was conducted by varying the number of tasks from 10 to 1000, for the Computation Intensive task graphs, Communication Intensive task graphs and mixed task graphs. Table 1 tabulates the average Resource Utilization percentage of the Computation Intensive Arbitrary Task Graphs.

TABLE I. AVERAGE RESOURCE UTILIZATION (%) OF THE ALGORITHMS FOR COMPUTATION INTENSIVE ARBITRARY TASK GRAPHS

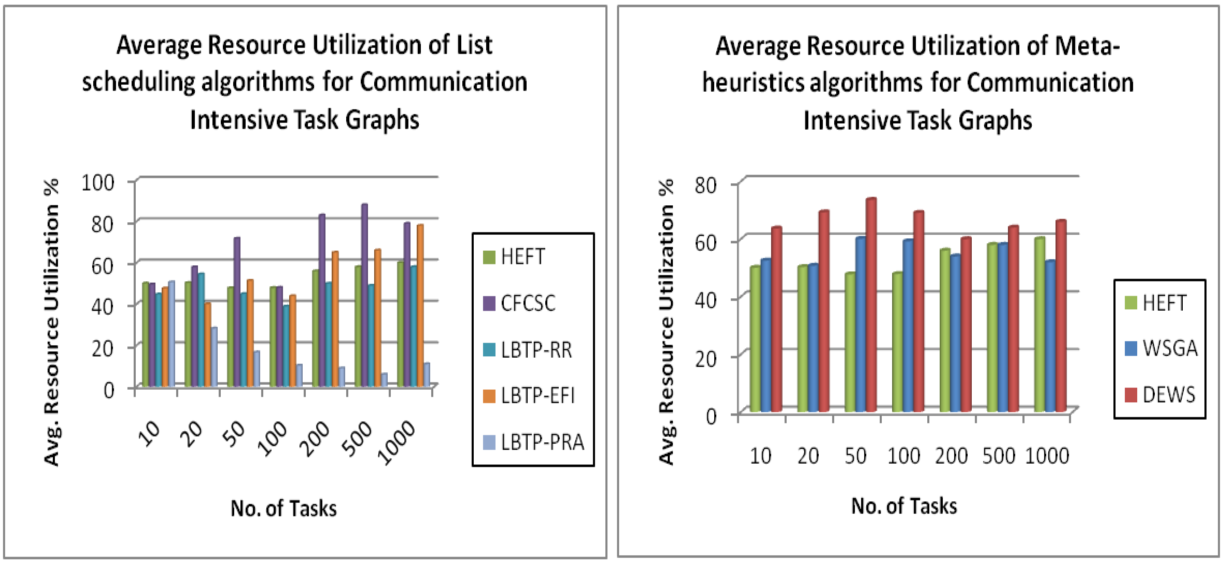
No. of Tasks	**No. of Resources**	**HEFT**	**CFCSC**	**LBTP**			**WSGA**	**DEWS**
				RR	**EFI**	**PRA**		
10	3	26	**29**	27	26	31	59	**65**
20	4	26	**21**	23	21	22	65	**67**
50	7	23	**33**	22	22	15	45	**45**
100	10	31	**28**	22	25	10	54	**61**
200	14	34	**35**	31	33	18	63	**70**
500	22	52	**59**	42	47	21	56	**64**
1000	31	47	**48**	36	44	67	56	**58**

TABLE II. AVERAGE RESOURCE UTILIZATION (%) OF THE ALGORITHMS FOR COMMUNICATION INTENSIVE ARBITRARY TASK GRAPHS

No. of Tasks	**No. of Resources**	**HEFT**	**CFCSC**	**LBTP**			**WSGA**	**DEWS**
				RR	**EFI**	**PRA**		
10	3	50	**50**	45	48	51	53	**64**
20	4	50	**58**	55	40	28	51	**69**
50	7	48	**72**	45	51	17	60	**74**
100	10	48	**48**	39	44	10	59	**69**
200	14	66	**83**	50	65	9	54	**60**
500	22	66	**88**	49	66	6	58	**64**
1000	31	80	**79**	58	78	11	52	**66**

TABLE III
AVERAGE RESOURCE UTILIZATION (%) OF THE ALGORITHMS FOR MIXED ARBITRARY TASK GRAPHS

No. of Tasks	**No. of Resources**	**HEFT**	**CFCSC**	**LBTP**			**WSGA**	**DEWS**
				RR	**EFI**	**PRA**		
10	3	40	**45**	33	27	51	38	**65**
20	4	32	**34**	38	32	32	60	**70**
50	7	40	**43**	43	32	15	47	**64**
100	10	31	**32**	36	31	12	62	**64**
200	14	41	**42**	37	40	8	60	**72**
500	22	53	**54**	43	49	62	53	**63**
1000	31	66	**66**	48	60	64	56	**66**

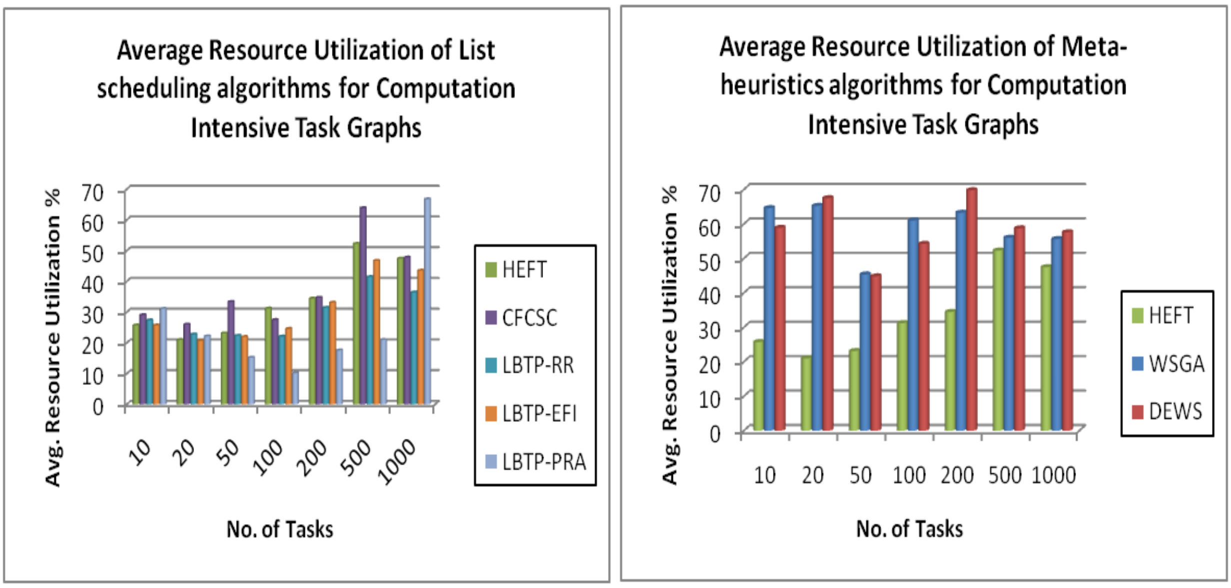


(a) **(b)**

Figure 1. Average Resource Utilization Percentage of Communication Intensive Task Graphs

The Average Resource Utilization percentage of the Communication Intensive task graphs and mixed graphs are tabulated on Table 2 and Table 3 respectively. It is observed that in the List scheduling category the CFCSC algorithm utilizes the Virtual Machines effectively. Though the LBTP-PRA algorithm reduces the makespan and cost of resources, the average resource utilization percentage is less compared to the other algorithms. This is because, in order to reduce the communication cost, the successor tasks are assigned the virtual machine in which the predecessor task is executed. Thus, only few resources are used. This has led to the reduction in the average resource utilization percentage.

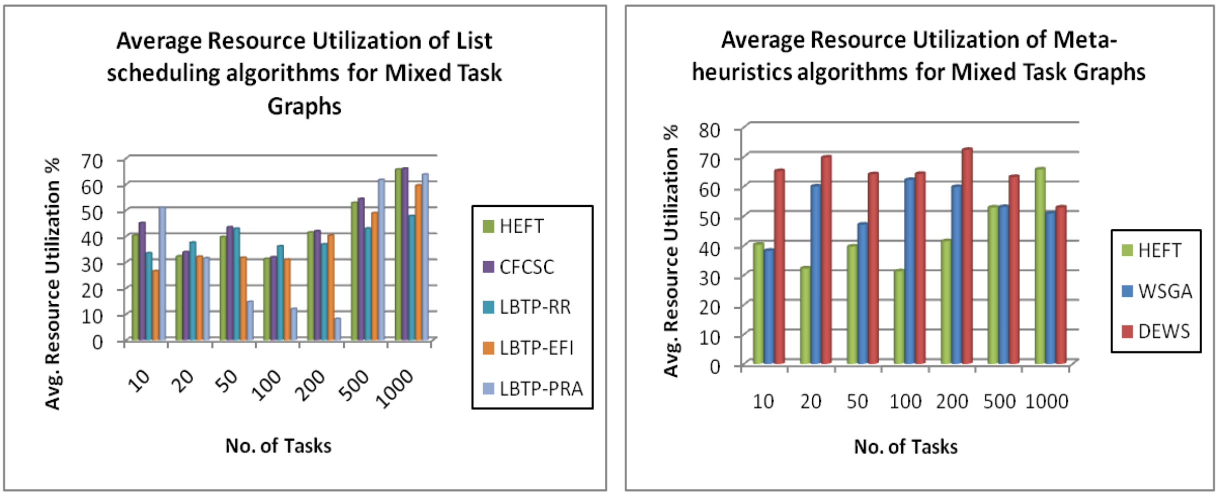
In the meta-heuristics category, DEWS algorithm utilizes the resource in a better way compared to the WSGA algorithm. Among all the algorithms discussed in this paper, DEWS algorithm outperforms all the other algorithms. The increasing usage of Cloud Resources leads to the installation of more Data Centers around the world by various Cloud Service Providers.



(a) (b)

Figure 2. Average Resource Utilization Percentage of Computation Intensive Task Graphs

The graphical representation of Average Resource Utilization Percentage of Communication Intensive Task Graphs is shown in Figure 1. Figure 2 and Figure 3 show the Average Resource Utilization of Computation Intensive Task Graphs and Mixed Task Graphs respectively.



(a) (b)

Figure 3. Average Resource Utilization Percentage of Mixed Task Graphs

As the number of Data Centers increases, there is a need for efficient energy consumption which aids in lower carbon footprints. This paper is an attempt to reduce the carbon footprints by effectively using the resources needed for the execution of the applications.

V. CONCLUSION

The benefits of Cloud computing such as scalability, cost reduction and optimal resource usage attract small as well as large businesses. This in turn causes an enormous rise of the energy usage. To address this problem, data center resources need to be managed in an energy-efficient manner. While allocating resources to the application, the Service Level Agreements must concentrate on energy usage in addition to Quality of Service. This paper analyses the resource utilization for four workflow scheduling algorithms. It is observed that the meta-heuristic algorithm DEWS outstrips the other algorithms and utilizes the resources effectively. By the effective resource utilization, energy consumption is also reduced. In future, energy-aware workflow scheduling algorithms can be proposed so that the carbon footprints can be reduced to support the Green Cloud environment.

VI. REFERENCES

[1] Buyya R, Beloglazov A, Abawajy J., “Energy-efficient management of data center resources for cloud computing: a vision, architectural elements, and open challenges”, arXiv preprint arXiv:1006.0308, 2010.

[2] George Amalarethinam D. I., Lucia Agnes Beena T., “Customer Facilitated Cost-based Scheduling algorithm (CFCSC) in Cloud”, Procedia Computer Science, Elsevier Publications 46C, pp. 660-667, April 2015.

[3] George Amalarethinam D. I., Lucia Agnes Beena T., “Level Based Task Prioritization Scheduling for Small Workflows in Cloud Environment”, Indian Journal of Science and Technology, Vol. 8, Issue 33, pp 1–7, 2015.

[4] George Amalarethinam D. I., Lucia Agnes Beena T., “Workflow Scheduling for Public Cloud using Genetic Algorithm (WSGA)”, IOSR Journal of Computer Engineering (IOSR-JCE), 18(3), V, 23-27, May-Jun. 2016.

[5] George Amalarethinam D. I., Lucia Agnes Beena T., “Differential Evolution Algorithm

for Workflow Scheduling (DEWS) in Public Cloud", International Journal of Control Theory and Applications, Vol. 9, No. 27, pp. 43 – 50, ISSN No:0974-5572, Oct 2016.
[6] Lee, Young Choon, and Albert Y. Zomaya. "Energy efficient utilization of resources in cloud computing systems." The Journal of Supercomputing, Vol. 60, Issue 2, pp. 268-280, 2012.
[7] Sindhu, S., and Saswati Mukherjee., "Efficient task scheduling algorithms for cloud computing environment.", High Performance Architecture and Grid Computing , pp. 79-83, 2011.
[8] Beloglazov, Anton, and Rajkumar Buyya. "Energy efficient resource management in virtualized cloud data centers." In Proceedings of the 2010 10th IEEE/ACM international conference on cluster, cloud and grid computing, IEEE Computer Society, pp. 826-831, 2010.
[9] Shakkeera, L., Latha Tamilselvan, and Mohamed Imran. "Improving Resource Utilization Using Qos Based Load Balancing Algorithm For Multiple Workflows In Iaas Cloud Computing Environment.", ICTACT Journal on Communication Technology 4, Vol. 02, pp. 750-757, 2013.
[10] Beloglazov, Anton, Jemal Abawajy, and Rajkumar Buyya. "Energy-aware resource allocation heuristics for efficient management of data centers for cloud computing.", Future generation computer systems, Vol. 28, Issue 5, pp. 755-768, 2012.
[11] Gaojin Wen, Jue Hong, Chengzhong Xu, Pavan Balaji, Shengzhong Feng, Pingchuang Jiang, "Energy-aware Hierarchical Scheduling of Applications in Large Scale Data Centers", International Conference on Cloud and Service Computing, IEEE Publications, pp. 158-165, 2011.
[12] Abrishami, S., Naghibzadeh, M., "Deadline-constrained workflow scheduling in software as a service Cloud", Computer Science & Engineering and Electrical Engineering, Scientia Iranica, vol. 19, issue 3, pp. 680–689, 2012.
[13] Abazar Shamekhi, "An Improved Differential Evolution Optimization Algorithm", International Journal of Research and Reviews in Applied Sciences, 15, 2013.
[14] George Amalarethinam D.I, Joyce Mary G. J. DAGEN – A tool to generate arbitrary Directed Acyclic Graphs used for Multiprocessor Scheduling. International Journal of Research and Reviews in Computer Science (IJRRCS), Vol. 2, Issue 3, pp. 782 – 787, 2011.