Journal of Electronic Research and Application

Research Article

# Extended Balanced Scheduler with Clustering and Replication for Data Intensive Scientific Workflow Applications in Cloud Computing

Satwinder Kaur, Mehak Aggarwal

Lala Lajpat Rai Institute of Engineering and Technology, Moga, India

**Abstract:** Cloud computing is an advance computing model using which several applications, data and countless IT services are provided over the Internet. Task scheduling plays a crucial role in cloud computing systems. The issue of task scheduling can be viewed as the finding or searching an optimal mapping/ assignment of set of subtasks of different tasks over the available set of resources so that we can achieve the desired goals for tasks. With the enlargement of users of cloud the tasks need to be scheduled. Cloud's performance depends on the task scheduling algorithms used. Numerous algorithms have been submitted in the past to solve the task scheduling problem for heterogeneous network of computers. The existing research work proposes different methods for data intensive applications which are energy and deadline aware task scheduling method. As scientific workflow is combination of fine grain and coarse grain task. Every task scheduled to VM has system overhead. If multiple fine grain task are executing in scientific workflow, it increase the scheduling overhead. To overcome the scheduling overhead, multiple small tasks has been combined to large task, which decrease the scheduling overhead and improve the execution time of the workflow. Horizontal clustering has been used to cluster the fine grained task further replication technique has been combined. The proposed scheduling algorithm improves the performance metrics such as execution time and cost. Further this research can be extended with improved clustering technique and replication methods.

***Keywords:*** scientific workflow; cloud computing; replication; clustering; scheduling;

## 0 Introduction

Cloud computing is a new paradigm that combines concepts, technologies and creates a platform for IT infrastructure and cost-effective business applications. The embracing of Cloud computing is increasing steadily for past few years in the technology market. By adopting cloud computing, IT (information technology) industries got benefited as cloud provides with less maintenance costs and infrastructure costs. Cloud computing; thus, may be defined as a multitenant environment that provides you with the resources abstracted from the fundamental infrastructure [1]. Services and resources are provided "on demand "and "at scale" in cloud environment. These terms used in definition could be elaborated further as:

1) ***Multitenant environment*** *- it's a type of environment that provides consumers with the resources from single implementation that saves the provider's costs (Azeez et al., 2010).*
2) ***On demand*** *- It means that resources can be provided when required, released when they are not required and estimated when only used.*
3) ***Auto-scale*** *- services provides with the infinite resources so that they are able to meet all the requirements they have demanded* [2].

In simple words, cloud computing is the endowment of computing resources through a reliable network. Enterprises have been trying to reduce the costs of computing and for that particular reason IT (information technology) started associating their IT operations and then on adapting virtualization technologies [3]. In order to reduce more there is a better technology and it is Cloud computing. Cloud computing has taken enterprises at new level and have also provided them with faster deployment cycles, improved utilization, reduced administration, and thus reducing the cost as well.

## *A. Scientific workflow*

A scientific workflow is specification of process to streamline automates and represents the schedule of integration, dataset selection, analysis and computation for the final presentation and visualization [4]. Scientific disciplines are knowledge driven with the help of data analysis and discovery pipeline. The series of data intensive and computational intensive tasks are designed composed and executed. The grid and cloud computing infrastructure attracts the scientist community the features such as sharing of computation, storage and software licenses. Multidisciplinary fields such as Bio-informatics, hem-informatics, geo-informatics etc. doing large investment on IT infrastructure. The communities of scientists are interested in robust middleware which could afford the requirements of scientific tasks.

## *B. Examples of Scientific Workflow*

### *1) EPIGENOMICS*

It is an information parallel work process. The Illumina- Solexa Genetic Analyzer gave us a beginning information which is in the form of DNA grouping paths. Different paths of DNA successions can be produced by every Solexa machine. DNA arrangements then mapped to the right areas in a reference Genome by workflow. This produces a guide that shows the grouping thickness demonstrating how often a specific arrangement communicates on a specific area on the reference genome [4].

### *2) MONTAGE*

The montage scientific workflow is used to compute the mosaics of the sky images. The images collected first re-projected as per the coordinates. Next rectification of background has taken place. The co-added has done to create a big picture of sky. The project has been funded by NASA [4].

## *C. Scientific Workflow Management System*

To perform execution of computational tasks in cloud environment the scientific workflow management system. The scientist community has vision to perform high computation task on the distributed system with large computations and data intensive applications. Front end can be enabling by many server side languages to input the task to the environment. Figure 2 describes the workflow management system.

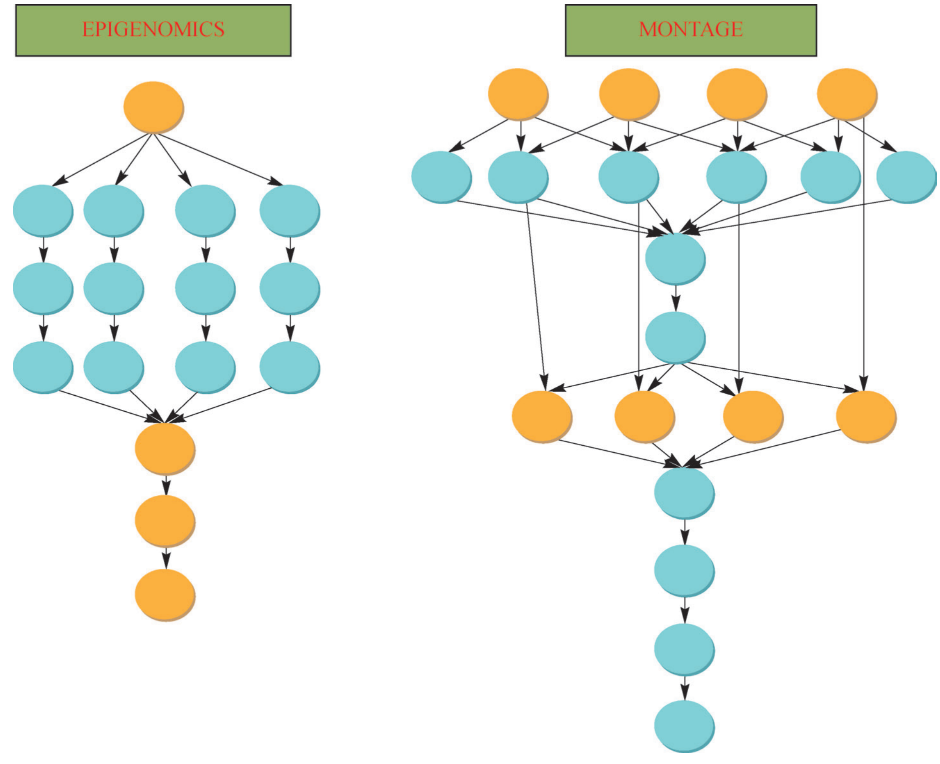


**Figure 1. Scientific workflow examples (Yu and Buyya,2005)**

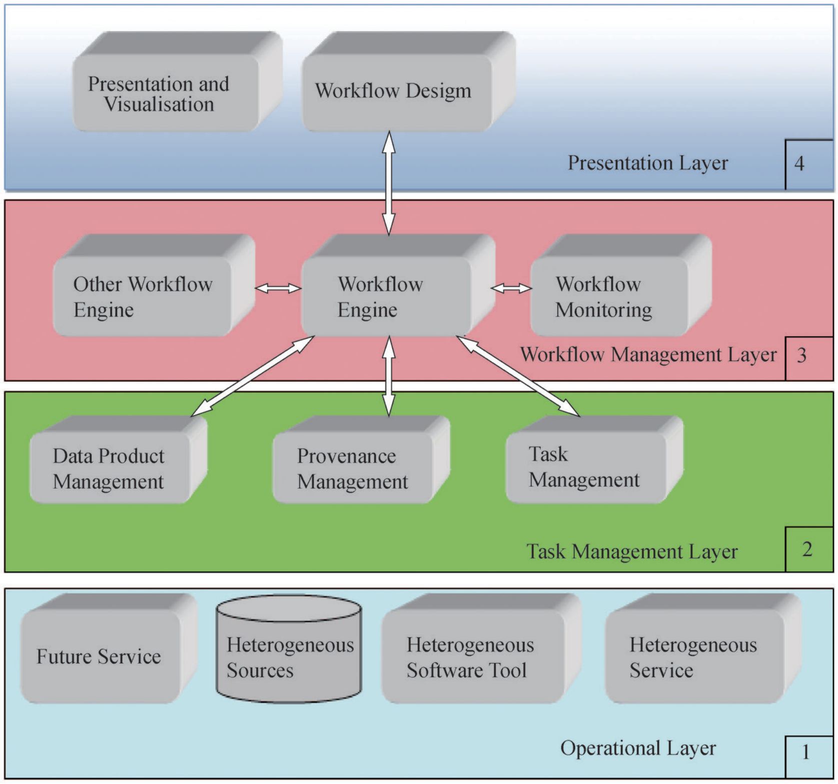


**Figure 2. Scientific workflow management system (Lin et al., 2008)**

### *D. Workflow Scheduling*

Workflow scheduling is doing the mapping and management of workflow task execution on them management system. Scheduling module plays a vital role in cloud computing systems. Task scheduling of cloud computing refers to dispatch the computing tasks to resource pooling between different resource users. Scheduling of tasks cannot be done on the basis of single criteria but under a lot of rules and regulations that we referred as an agreement between users and providers of cloud. This agreement is nothing but the quality of service that the user wants from providers [4]. The job scheduling problem can be viewed as the searching or finding an optimal mapping of set of subtasks of various tasks over available set of computing machines so that we can get desired goal for tasks. The scheduler is concerned mainly with:

1) ***Throughput*** *The total number of processes that complete their execution per time unit.*
2) ***Latency***
3) *Turnaround Time- Total time between submission of a process and its end.*
4) *Response Time- amount of time it takes from when a request was submitted until the first response is produced.*
5) ***Fairness*** *Equal CPU time to each process.*
6) ***Waiting time*** *the time the process remains in the ready queue.*

## 1 Review of Literature

Service providers are facing various scheduling issues while deploying the scientific workflow in cloud environment. Taking into consideration the issue, following review of literature has been conducted.

**Wang, Jun et al. (2014)** proposed the compute and storage intensive framework for hadoop based cloud. The load balancing and random placement strategy has been developed. The data grouping strategy has been designed for the data placement named as DRAW. The technique is learning the grouping data logs and clustering the data and organizing and reorganizing the data [5].

**Chase Qishi Wu, XiangyuLin (2014)** discussed about the development of prototype generic workflow system using influence of existing technologies for rapid calculation of scientific workflow optimization strategies. They plan a heuristic answer for this issue, and show its execution predominance over existing strategies through vast simulation also, genuine work flow tests in view of evidence of-idea execution and arrangement in a local cloud tested [6].

**Zhang et al. (2014)** tailored a multi-objective algorithms based on ordinal optimization (OO) for complex dynamic cloud environment. OO technique is combined with virtual cluster from the multi datacenters named as VOO. Monte Carlo and Blind Pick methods are used to search the optimal resource in previous algorithms used for the comparison [7].

**Deepak Poola, Saurabh Kumar Garg (2014)** presents three resource allotment policies with cost, robustness and makespan as its objective. For making the schedule robust by considering the budget and deadline constraint, the resource allotment policies adds slack time to it. The RCT strategy contributes a robust schedule with expenses hardly higher than the reference algorithm investigated [8].

**Ghafarian et al. (2015)** devised the technique to place and schedule the data intensive workflow on cloud resources as well as volunteer computing if the task is taking more time as compare to expect than the task has be schedule to the cloud [9]. Two strategies have been proposed, first technique reduce the cost by considering the deadline and second strategy improve the cost by considering the deadline.

**Zhao, Qing et al. (2015)** proposed the 2-stage placement strategy for data placement. First stage refers to the clustering of the task by considering the correlation of task. Second stage is runtime, the distribution and redistribution algorithm look after the layout of the data [10].

**Yong Zhao and Wenhong Tian (2015)** reference service framework is proposed for integrating scientific workflow management systems into different cloud platforms. In integrating the SWfMSs an implementation effort is also presented in this paper. The usage can rapidly be utilized for Open Stack as it is getting more demanding in scientific research area as well as in business applications. They are additionally researching the joining of different SWFMS into these different clouds [11].

**Weiwei Chen, Rafael Ferreira da Silva (2015)** Here, Theoretical analysis is conducted by the author in whom the analysis is of impact of transient failure on runtime performance of scientific workflow execution. They suggest general task failure modeling framework. This framework maximum likelihood estimation-based parameter for modeling workflow performance. To improve runtime performance of workflow execution they suggest three fault tolerant clustering strategies which works in faulty execution environments results [12]

**Chase Qishi Wu et al. (2015)** discussed about the development of prototype generic workflow system using influence of existing technologies for rapid calculation of scientific workflow optimization strategies. They plan a heuristic answer for this issue, and show its execution predominance over existing strategies through vast simulation also, genuine work flow tests in view of evidence of-idea execution and arrangement in a local cloud tested [13].

**Malawski et al. (2015)** present the model based on mathematical programming language a Mathematic Programming Language (AMPL) and Coin Mathematical Programming Language (CMPL) used to minimize the cost of workflow execution by taking consideration of deadline. Algebraic mathematical models used to define variables, constants. Mixed integer problem (MIP) is used along with mathematical model [14].

**Jianbing Dinga et al. (2016)** talks about Abacus. It is auction based resource assignment framework which is for cloud computing. So there can be the future work in which the dependent resources can be handled. This can be done by using dependent model [15].

**Cotes et al. (2016)** In this paper, author proposed a new tool that collaborates power model on the basis of communication design and computing that permit the atomization of advanced management strategies in power saving in the consideration of network cost, computing, reconfiguration, Quality of Service and also associate Dynamic Voltage Frequency Scaling (DVFS) [16].

**Zhao et al. (2016)** in this paper, a new offline task scheduling method in cloud was suggested in order to decrease consumption of energy. Moreover a model for energy consumption was designed [17]. Then an essential parameter Task Requirement Degree (TRD) was proposed by which resource utilization efficiency can be improved.

## 2 Problem Formulation

Cloud computing provides the heterogeneous environment for execution of scientific workflow. As many companies in the market provide the infrastructure, software and platform as a service, which enhance the competition in the market related to QoS and cost tradeoff. Many third party companies provide the trade-off for various application specific cloud provider. The balancer scheduler designed by the author [18], provides trade-off between cost and execution time. As scientific workflow is combination of fine grain and coarse grain task. Every task scheduled to VM has system overhead. If multiple fine grain task are executing in scientific workflow, it increase the scheduling overhead. To overcome the scheduling overhead, multiple small tasks can combine to large task, which decrease the scheduling overhead and improve the execution time of the workflow. Our aim is to design the scheduling algorithm which can optimize the cost to run scientific workflow on cloud computing.

***1) Scheduling Overhead***

The scientific workflow is group of multiple task small (fine grained) and large task (coarse grained), scheduling is process to transfer the task to specific computation node. When multiple tasks need to schedule it take lot of time of scheduler (milliseconds). For each and every task it has to look the appropriate machine.

***2) Execution time***

In terms of milliseconds, in scientific workflow we called it as makespan time, total time to execute between submission and execution in terms of milliseconds. Due to scheduling overhead, makespan time increases.

***3) Cost:***

In cloud computing, the total cost of Virtual machines in term of dollar ($) or either we can calculate in terms of rupees also, to process the given scientific workflow.

## 3 Research Methodology

The EBSCR **(Extended Balanced Scheduler with Clustering and Replication)** has been designed to provide the approximation for data intensive workflow. The customer can compare the execution time of various jobs and relative cost for deploying the application in cloud heterogeneous environment. It provides the execution time for various kinds of VMs, so that customer can compare the cost of various kinds of configuration and time consumed. Further customer can compare the QoS given by the service provider.

### A. Clustering of Jobs

Clustering of the jobs can reduce the scheduling overhead to great extending. By little bit effort, if customer can do the clustering of small jobs to large job. It will cost them less as compare to multiple small jobs. One drawback of the clustering is that if one task fails in entire job the whole job considered as fail. If the failure reluctant jobs are in scientific workflow, than it is a better strategy to run the bigger job.

**Horizontal Clustering Algorithm**

*1: Method HORICLUSTERIING(W,C)*
*2: Repeat step 3 to 5, if level <depth(W) then*
*3: TL = TasksLevel(W,level)*
*4: CL = TaskMerge(TL,C)*
*5: W = W-TL + CL*
*6: End Method*

**Task Merging Algorithm**

*1: Method TaskMerge(TL,C)*
*2: J= {empty}*
*3: CL= {empty}*
*4: Repeat Step 10 and 11 if TL is not empty*
*5: J.ADD(TL.pop(C))*
*6: CL.add(J)*
*7: return CL*
*8: End Method*

***B. Scheduling Algorithm***

Queue based technique has been used to perform the scheduling. The task from each level put in the queue by considering the replication as per replication algorithm.

**Proposed EBSCR Algorithm (Extended Balanced Scheduler with Clustering and Replication)**

***Input:*** *Workflow W, VM set*
***Output:*** *Scheduling plan for the given VM set*
*1:* ***For*** *l=1 to |L|*
*2: s = total number of descend from actual level*
*3: A ← {tasks in l}*
*4:* ***while*** *parents ≠ ø*
*5:* ***for*** *j=1 to s //for all descended*
*6. Task replication as per Replication algorithm*
*8: add task(s) to the q*
*9:* ***End***
*10:* ***End***
*3: A ← {tasks in l}*
*4:* ***while*** *parents ≠ ø*
*5:* ***for*** *j=1 to s //for all descended*
*6. Task replication as per Replication algorithm*
*8: add task(s) to the q*
*9:* ***End***
*10:* ***End***

## 4 Experimetne Setup

The workflowsim will be used for the experiment of the task. It has been developed on the cloudsim. It is widely used open source simulator for scientific workflow simulation in cloud environment. The proposed algorithm and existing algorithm will be compared and analysis will be performed.

### *A. Simulation Setup*

Workflowsim has been used for implementing the Robust Energy Aware Task Scheduling for Scientific Workflow in Cloud Computing. It is an open source workflow simulator. It is giving a workflow level support by amplifies cloudsim. Workflow will be modeled using DAG model in which DAG model demonstrate a model of delay happening in different levels of Workflow Management System, elaborating model of failure of node, stack and task clustering algorithm. Parameters are straight forwardly gained from hints of genuine execution. Below are the various results gathered during the experiments. Both the existing and proposed results are posted to make a comparison of how our technique performs better in terms of various performance attributes.

*a)* Application Modeling : A workflow is combination of various tasks according to the dependencies. It indicates the temporary relationship between the tasks. The sequence and parallelism performs the task concurrently. Each having their own characteristics.

*b)* Resource Modeling: Cloud Model having one dataset is taken. To effectively process the cloudlets without any overhead, four types of VM is used with homogeneous configuration. The actual machine that is available in data centers is host. Proposed algorithm saves the energy and also do not violate the SLA. Hosts and Virtual Machine each having its own specification that are given in table below:

**Table I. Resource Configuration: VM Setup**

| Resource | Capacity |
|---|---|
| Storage | 10000 MB |
| RAM | 512 MB |
| MIPS | 1000 |
| Bandwidth | 1000 |
| Number of CPUs | 1 |

## 5 Result and Analysis

### *A. Experiment 1*

Experiment - 1 is performed on the Montage workflow with different number of task with 5 VM with respect to existing technique. The experiment has been performed to check the runtime and cost of processing of Montage workflow.

**Table II. Comparison of Makespan Time for Montage Workflow**

| No. of Tasks | Existing | Proposed |
|---|---|---|
| Montage 50 | 753.38 | 393.45 |
| Montage 100 | 1671.25 | 710 |
| Montage 1000 | 5818.8 | 2815.22 |

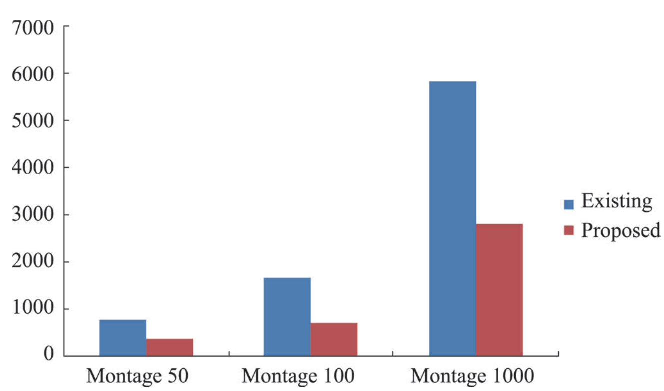


**Figure 3. Comparison of Makespan Time for Montage Workflow**

**Table III. Comparison of Makespan Cost for Montage Workflow**

| No. of Task | Existing | Proposed |
|---|---|---|
| Montage 50 | 1634.08 | 1669.77 |
| Montage 100 | 3439.38 | 3443.69 |
| Montage 1000 | 36133.21 | 35549.41 |

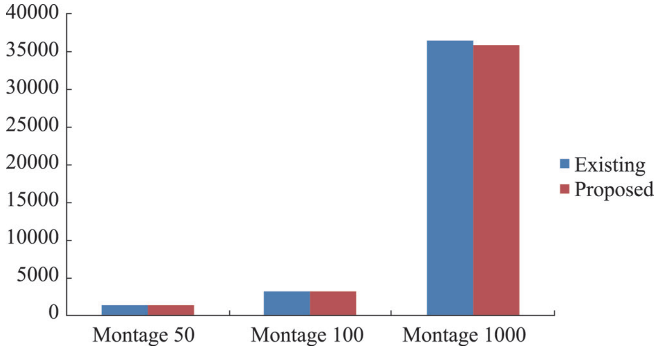


**Figure 4. Comparison of Makespan Cost for Montage Workflow**

Result shows that the our technique gives better through put with little bit of cost with less number of tasks as task grows the cost will be less as compare to previous technique.

### B. Experiment 2

Experiment - 2 is performed on the CyberShake workflow with different number of task with 5 VM with respect to existing technique. The experiment has been performed to check the runtime and cost of processing of Montage workflow.

**Table IV. Comparison of Makespan Time for CyberShake Workflow**

| No. of Tasks | Existing | Proposed |
|---|---|---|
| CyberShake_50 | 617.97 | 542.72 |
| CyberShake_100 | 1285.56 | 1110.99 |
| CyberShake_1000 | 6791.05 | 5047.24 |

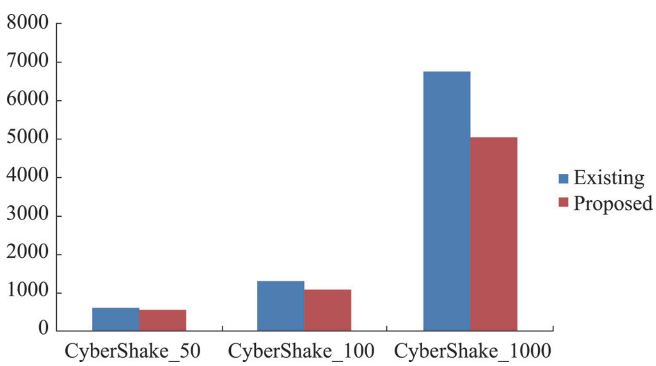


**Figure 5. Comparison of Makespan Time for CyberShake Workflow**

**Table V. Comparison of Makespan Cost for CyberShake Workflow**

| No. of Tasks | Existing | Proposed |
|---|---|---|
| CyberShake_50 | 9808.28 | 8867.98 |
| CyberShake_100 | 20017.78 | 18331.16 |
| CyberShake_1000 | 99420.85 | 73345.76 |

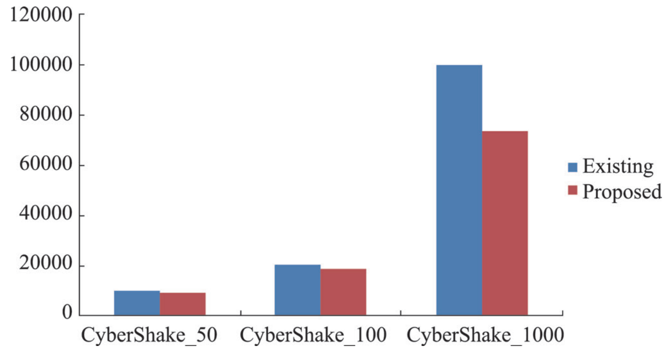


**Figure 6. Comparison of Makespan Cost for CyberShake Workflow**

## 6 Conclusion

The extended balanced scheduler has been designed which gives better makespan as compare to existing technique. Proposed technique is also very cost effective. In montage workflow it tends costly for less number of task. In future, the technique can be modified using vertical and adaptive clustering. Experiment has been performed using simulator. Further it can be implemented on cloud providers such as Amazon EC2, Google etc. Result shows that the our technique gives better throughput as well as cost effective with all number of task as compare to previous technique.

## References

[1] Voorsluys, William, James Broberg, and Rajkumar Buyya. "Introduction to cloud computing." *Cloud computing: Principles and paradigms* (2011): 1-44.

[2] Mell, Peter, and Tim Grance. "The NIST definition of cloud computing."*Communications of the ACM* 53, no. 6 (2010): 50.

[3] Wyld, David C. Moving to the cloud: An introduction to cloud computing in government. IBM Center for the Business of Government, 2009.

[4] Yu, Jia, Rajkumar Buyya, and Kotagiri Ramamohanarao. "Workflow scheduling algorithms for grid computing." *Metaheuristics for scheduling in distributed computing environments*. Springer Berlin Heidelberg, 2008. 173-214.

[5] Wang, Jun, Pengju Shang, and Jiangling Yin. "DRAW: A new Data-gRouping-Aware data placement scheme for data intensive applications with interest locality." In *Cloud Computing for Data-Intensive Applications*, pp. 149-174. Springer New York, 2014.

[6] Wu CQ, Lin X, Yu D, Xu W, Li L. " End-to-end delay minimization for scientific workflows in clouds under budget constraint. " Cloud Computing, *IEEE* Transactions on. 2015 Apr 1;3(2):169-81.

[7] Zhang, Fan, Junwei Cao, Keqin Li, Samee U. Khan, and Kai Hwang. "Multi-objective scheduling of many tasks in cloud platforms." *Future Generation Computer Systems* 37 (2014): 309-320.

[8] Poola D, Garg SK, Buyya R, Yang Y, Ramamohanarao K. "Robust scheduling of scientific workflows with deadline and budget constraints in clouds. " In Advanced Information Networking and Applications (AINA), 2014 *IEEE 28th International Conference* on 2014

[9] Ghafarian, Toktam, and BahmanJavadi. "Cloud-aware data intensive workflow scheduling on volunteer computing systems." *Future Generation Computer Systems* 51 (2015): 87-97.

[10] Zhao Y, Li Y, Raicu I, Lu S, Lin C, Zhang Y, Tian W, Xue R. " A Service Framework for Scientific Workflow Management in the Cloud. " Services Computing, *IEEE* Transactions on. 2015 Nov 1;8(6): 930-4427.

[11] Zhao, Qing, CongcongXiong, Xi Zhao, Ce Yu, and Jian Xiao. "A data placement strategy for data-intensive scientific workflows in cloud." In *Cluster, Cloud and Grid Computing (CCGrid), 2015 15th IEEE/ACM International Symposium on*, pp. 928-934. IEEE, 2015.

[12] Chen, W., Ferreira Da Silva, R., Deelman, E., & Sakellariou, R. (2015). Using imbalance metrics to optimize task clustering in scientific workflow executions. *Future Generation Computer Systems*, *46*, 69–84. http://doi.org/10.1016/j.future.2014.09.014.

[13] C. Q. Wu, X. Lin, D. Yu, W. Xu, and L. Li, "End-to-end delay minimization for scientific workflows in clouds under budget constraint," IEEE Transactions on Cloud Computing, vol. 3, no. 2, pp. 169–181, 2015.

[14] Malawski, Maciej, Kamil Figiela, Marian Bubak, Ewa Deelman, and Jarek Nabrzyski. "Scheduling multilevel deadline-constrained scientific workflows on clouds based on cost optimization." *Scientific Programming* 2015 (2015): 5.May 13 (pp. 858-865). IEEE.

[15] J. Ding, Z. Zhang, R. T. Ma, and Y. Yang, "Auction-based cloud service differentiation with service level objectives," Computer Networks, vol. 94, pp. 231–249, 2016.

[16] I. T. Cotes-Ruiz, R. P. Prado, S. Garc´ıa-Gal´an, J. E. Mu˜noz-Exp´osito, and N. Ruiz-Reyes, "Dynamic voltage frequency scaling simulator for real workflows energy aware management in green cloud computing," PloS one, vol. 12, no. 1, p. e0169803, 2016.

[17] Zhao, Qing, Cong Xiong, and Peng Wang. "Heuristic Data Placement for Data-Intensive Applications in Heterogeneous Cloud." *Journal of Electrical and Computer Engineering* 2016.

[18] Casas, Israel, Javid Taheri, Rajiv Ranjan, Lizhe Wang, and Albert Y. Zomaya. "A balanced scheduler with data reuse and replication for scientific workflows in cloud computing systems." *Future Generation Computer Systems* (2016).