A Research on an Efficient Cloud Scheduling with a Geo Microarray Data Set

Selvi S, Chandrasekar A, Dhipa M

Abstract— Investigations on micro-array organisms for various researches have made a non discrete dealing of thousands of gene expressions achievable. For any applications, the results would be more accurate only when maximum count is analyzed within a predictable time and it is one of the unseen challenges in the field of bio medicine. The purpose of this data analysis is to regulate and control the activities of thousands of genes in our body. This paper develops a scheduling analysis of how effectively gene molecular patterns are taken into experimentation. This motivated our investigation in a new dimension for a cloud environment. This paper is about applying our previous works such as Workflow Shuffling and Hole Filling Algorithm (WSHF) [13], Agent Centric Enhanced Reinforcement learning algorithm (AGERL) [14], Heuristic Flow Equilibrium based Load Balancing (HFEL) [15] and Dynamic Resource Provisioning and Load Balancing (DRBLHS) [16] algorithms collaboratively for a Gene Express Omnibus dataset as a case study. The gene data's plays an important role in monitoring the human activities and how well, the data has been processed in the cloud with minimum budget, time and minimum virtual machines. Finally, the efficiency of the system is analyzed in terms of resource utilization, completion time, response time, throughput and VM Migration time.

KEYWORDS- Cloud Provider, Execution Process, WSHF, AGERL, HFEL, DRBLHS, Gene Expression Omnibus Dataset, Resource Utilization, Workflow Completion Time, Response Time, Throughput and VM Migration Time

I. INTRODUCTION

Cloud computing is one the developing technology which provides the various services in terms of the software, infrastructure, security to the user. Among the various services, infrastructure services place a vital role because it reduces the processing cost for the users. The infrastructure services are being utilized in different commercial and scientific applications because it provides the storage and location for cloud customer request [1]. In addition, the infrastructure establishes the clusters and girds for managing the hosted services in the cloud. During this process, the cloud providers use the virtual machines and inter-connected systems for processing the user request and workflows in an effective manner. In those systems, the cloud providers need to manage the user requirements with a defined budget along with minimum computation time [2]. So, the cloud providers use the various job scheduling algorithms for satisfying the user requirements in the cloud. The main objective of deriving those algorithms is to utilize the defined resources effectively for fulfilling the user request with minimum time. The above discussed requirements were further optimized by forming the clusters based on the user request which reduces the maximum utilization of the resources in the cloud [3]. Even though, the infrastructure process ensures the effective utilization of the resources, load balancing should be taken into account by cloud provider for enhancing the user workflow process. Based on the above discussed properties in cloud, different scheduling and load balancing techniques are introduced earlier but those techniques consumed large number of resources as well as they acquire maximum amount of time [4]. For these reasons, this paper uses the optimized techniques [5] for analyzing the budget of the execution process based on WSHF; in addition it removes the irrelevant and unwanted request from the list. Then the appropriate resources are identified using the AGERL. The identified resources aids in forming the clusters of similar type of requests which gets improvised using the Hopkins statistical assessment process. Such clustered resources are allocated to the user applications and scheduled accordingly using HFEL and DRBLHS, where by decrementing the number of virtual machines should be taken care.

Based on the above introduced strategy, a bio-medical case study has been designed using the Gene Expression Omnibus (GEO) dataset. To end with, the efficiency of the system is evaluated with the help of the experimental results and discussions. The rest of the paper is organized as follows: section 2 analyzes the proposed scheduling and workflow execution process models, section 3 gives an overview on the microorganisms that has been used for investigation, section 4 examines the biomedical case study and section 5 concludes the work.

II. PROPOSED WORKFLOW EXECUTION MODEL

This section illustrates the proposed workflow execution model for resource allocation and scheduling process. Initially, user requested workflow [6] is submitted to the cloud provider. The provider applies WSHF for estimating the budget. Simultaneously, the irrelevant and performance pull workflows present in the ensembles are identified and eliminated. The resultant workflow is partitioned into various sub workflows and those workflows are divided into the ensembles. For those ensembles, virtual machine is determined, CPU usage and memory is predicted for each virtual machine. Then the detected virtual machine information [7] should be allocated to the budget estimation

Revised Manuscript Received on October 15, 2019.

Selvi S, Department of Computer Science and Engineering, Erode Sengunthar Engineering College, Erode, Tamil Nadu, India (Email: selvi.me08@gmail.com)

Chandrasekar A, Department of Computer Science and Engineering, Malla Reddy Institute of Technology and Science, Secunderabad, Telangana, , India (Email: chandru.as76@gmail.com)

Dhipa M, Department of Electronics and Communication Engineering, Jairupaa College of Engineering, Tiruppur, Tamil Nadu, , India (Email: dhipachandrasekar@gmail.com)

process with specified deadline constraints, if the ensembles is true it passed to the queue else, the unallocated space is calculated. This process is repeated for determining the entire workflow details. Then the ensemble present in the queue is continuously read, divided into sub ensembles and the details are stored in the hole as a log record. From the log, the holes should be mapped with the sub ensembles, at the same time deadline constraints are calculated, if it is satisfied send to queue for further processing and updating should be performed continuously else the ensembles need to be rejected. After estimating the budget and rejection [8] of the ensembles, the distance between the workflow present in the queue has been calculated for any two nodes with the help of the Manhattan distance(U_m) for a node which is computed as follows [14],

$\boldsymbol{U_m}$ = Manhautten Distance $\left(\boldsymbol{p_i}, \boldsymbol{p_j}\right)$

Manhautten Distance $\left(\boldsymbol{p_i}, \boldsymbol{p_j}\right) = |\boldsymbol{x_1} - \boldsymbol{x_2}| + |\boldsymbol{y_1} - \boldsymbol{y_2}|$ (1)

where $(\boldsymbol{x_1}, \boldsymbol{y_1})$ and $(\boldsymbol{x_2}, \boldsymbol{y_2})$ are the coordinates of $\boldsymbol{p_i}$ and $\boldsymbol{p_j}$ respectively.

Similarly the Manhattan distance (V_m)for adjacent node is computed for any two points p_i and p_j.

After calculating the neighboring workflows or ensembles information in the queue, Hopkins Statistics are estimated as follows,

$$\boldsymbol{H} = \frac{\sum \boldsymbol{U_m}}{\sum \boldsymbol{U_m} + \sum \boldsymbol{V_m}} \quad (2)$$

H value for every node is computed in the similar manner and if H(avg) > 0.5, then the cluster to be restructured would be a meaningful cluster. After forming the Agent Centric Enhanced Reinforcement learning algorithm based cluster, the resources has been allocated to the scheduling process using the Heuristic Flow Equilibrium based Load Balancing (HFEL). In this process, the clustered tasks [9] are assigned to virtual machines which process the user requested task by defined load. So, the similar tasks or requests have been clustered using the previous step, and the cluster related load has been calculated as follows [15]:

$$\boldsymbol{L_{VMi}} = \frac{\boldsymbol{N(T,t)}}{\boldsymbol{VM_{bwj}(t)}} \quad (3)$$

where, $\boldsymbol{L_{VMi}}$ refers to amount of load accumulated in each VM.

Based on equation (3), resource utilization of each VM is calculated as below:

$$\boldsymbol{R(T_i)} = \boldsymbol{ET_i} * \left[\frac{(\boldsymbol{n} * \boldsymbol{R_k})}{\boldsymbol{m}}\right] \quad (4)$$

where $\boldsymbol{R(T_i)}$ refers to resource utilization of each VM.

Allocate the task to virtual machine which is accommodated with minimum load and available with ample amount of idle resources. This process is repeated for executing the user requested workflows, simultaneously minimizing the number of VMs is considered into account with the help of DRBLHS approach. The CPU and memory usage computation [10] is updated in a dynamic and continuous manner in order to lessen the usage of VMs. The trust rank between the user requests is also estimated for task-resource mapping; if it is high then the task has been linked with the resource manager and the virtual machine scheduler starts schedule the task with particular machine. The available space has been continuously monitored by the scheduler for allocating the resources to the next incoming task. Thus the process reduces the number of virtual machines, resources with minimized budget flow. The above process is applied to the GEO dataset related case study for designing and explains how the introduced process workflow methodologies process the user requested gene expression dataset.

III. GEO DATASET – AN OVERVIEW

The Gene Expression specifies the methodology of building proteins and it is used to manipulate the various important activities in the body. Proteins help the normal functioning of the body like digestion, building energy and growing. RNA molecules are used to categorize the amino acids that build individual proteins. Bioassays help in diagnosing the potential of tissues and detect the occurrence of biological hazards if any. Genetics and medicines are helpful in analyzing the genetic disorders. The sample details are as follows:

- DNA & RNA – 27,34,632 number of nucleotide sequences

(source:www.ncbi.nlm.nih.gov/genbank/samplerecord/)

- Genetic Medicine – 2300 number of samples

(Source: ftp://ftp.ncbi.nlm.nih.gov/pub/medgen/)

- Protein – 5300 number of samples

(Source: ftp://ftp.ncbi.nlm.nih.gov/blast/db/)

- Bioassays – 525 number of samples

(Source: https://pubchem.ncbi.nlm.nih.gov/assay/assaydownload.cgi/)

- Gene – 4500 number of samples

(Source: ftp://ftp.ncbi.nlm.nih.gov/refseq/H_sapiens/RefSeqGene/)

IV. CASE STUDY AND ANALYSIS & RESEARCH

The process workflow execution scenario is implemented with the help of the cloud provider where the user should fetch and examine the gene data and the relevant gene structure in human beings. In this case study, Gene Expression Omnibus [11] based biomedical data set has been used to process the structure of the gene by providing the gene workflow as the query. The dataset consists of the DNA, RNA details, Homology information, Protein patterns, Sequence Analysis, Taxonomy information, Genetics and Medicine information [12]. These details helps to user search the genomic information, variations and effects are examined from the third party cloud provider by using some sample gene expression data. At the time of analyzing the process, the data should be processed with minimum budget as well as within the predetermined time to get the genomic information as output. So, the proposed model approach discussed in the section 2 has been utilized in an effective manner, thereby reducing the cost and time. The structure of the proposed work flow is shown in the Figure 1.

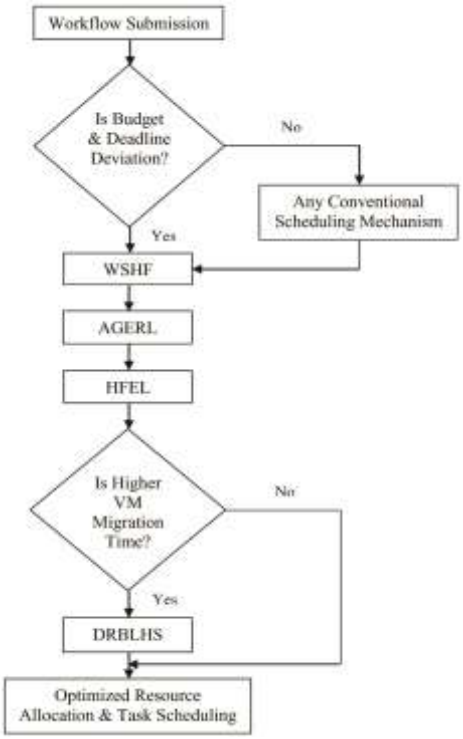

Figure 1. Proposed workflow processing structure

Initially the user provided gene information is passed to the cloud provider, they analyze the complexity present in the execution process in terms of how much memory needed to be accommodated, CPU usage time, budget involved etc., are examined with the help of the WSHF [13]. This methodology results in rejecting the irrelevant information that was derived from the user request. After that the appropriated resources are provided to the genomic data, by forming the clusters. During the cluster estimation process [14], neighboring genomic information is gathered for reducing the dissimilarity information processing that indirectly enhances the processing time. The similar gene information is clustered which is then allocated to the virtual machine by implementing HFEL algorithm [15]. As a next step, the number of virtual machines has been minimized using the ranking process. If the particular clustered genomic information has rank, the virtual machine has been allocated by calculating the CPU usage, memory and budget [16]. The virtual machine process the user request (gene workflow) and generate an appropriate gene pattern, structure and changes present in the genome is also listed. Finally, the efficiency of the case study is examined using the resource utilization, workflow completion time, response time, throughput and VM migration time metrics. The sample GEO data samples are shown in Figure 2.

Figure 2. Sample GEO database details

Based on GEO dataset, the obtained resource utilization of the different gene data by processing the proposed model,

probabilistic load balancing model and active clustering load balancing method is shown in the Table 1 and Figure 3.

Table 1. Resource utilization

Molecular Substrates	Resource Utilization (%)		
	Probabilistic Load Balancing	Active Clustering Load Balancing	Proposed Dynamic Resource Provisioning and Load Balancing
DNA and RNA	54.3	43.12	23
Genetic Medicine	53.1	41.09	25
Bioassays	49.32	39.31	26
Proteins	47.21	38.43	22.3
Genes	43.2	36.1	24.31

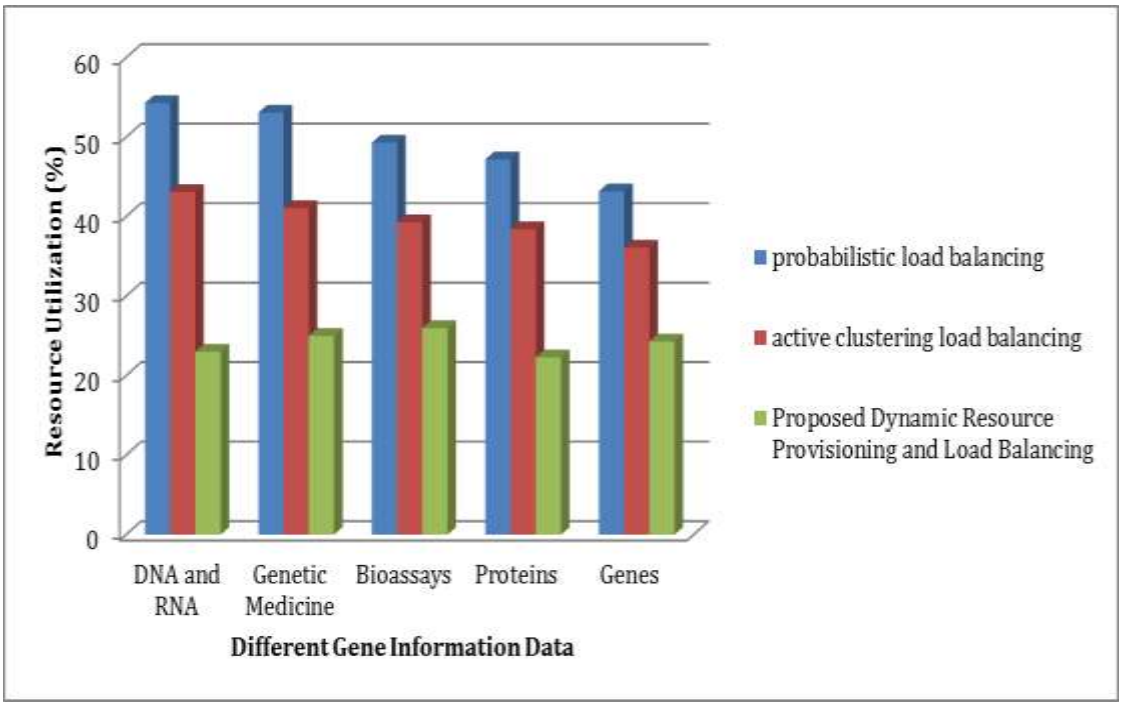

Figure 3. Resource utilization

According to the above Figure 3, it shows that the different gene data present in the GEO database, DNA and RNA (23%), Genetic Medicine (25%), Bioassays (26%), Proteins (22.3%) and Gene expressions (24.31%) of resources are utilized while analyzing that genomic information by using the proposed methodology. Thus the proposed system consumes minimum resources by processing the user request when compared to the other traditional methodologies such as probabilistic load balancing model consumes DNA and RNA (54.3%), Genetic Medicine (53.1%), Bioassays (49.3%), Proteins (47.2%) and Gene expressions (43.2%) of resources and active clustering load balancing method consumes DNA and RNA (43.12%), Genetic Medicine (41.09%), Bioassays (39.31%), Proteins (38.43%) and Gene expressions (36.1%) of resources. In spite of these methods consuming minimum amount of resources, the workflows should also be processed with minimum completion time and loads should also be evenly balanced by minimal VM migration which is shown in the Table 2 and Figure 4.

Table 2. Time(s)

Molecular Substrates	Probabilistic Load Balancing			Active Clustering Load Balancing			Proposed Dynamic Resource Provisioning and Load Balancing		
	Completion time	Response Time	VM Migration Time	Completion time	Response Time	VM Migration time	Completion time	Response time	VM Migration time
DNA & RNA	32.13	32.98	31.35	28.83	29.74	28.03	12.34	10.38	12.1
Genetic Medicine	29.46	28.56	29.13	27.45	26.31	28.94	13.23	10.21	13.42
Bioassays	33.1	31.89	29.43	26.31	25.14	26.46	12.12	11.23	12.3
Proteins	25.74	28.934	27.83	26.987	28.43	27.67	11.87	12.01	10.32
Genes	27.1	26.98	27.81	24.87	25.31	24.41	13.2	11.21	12.3

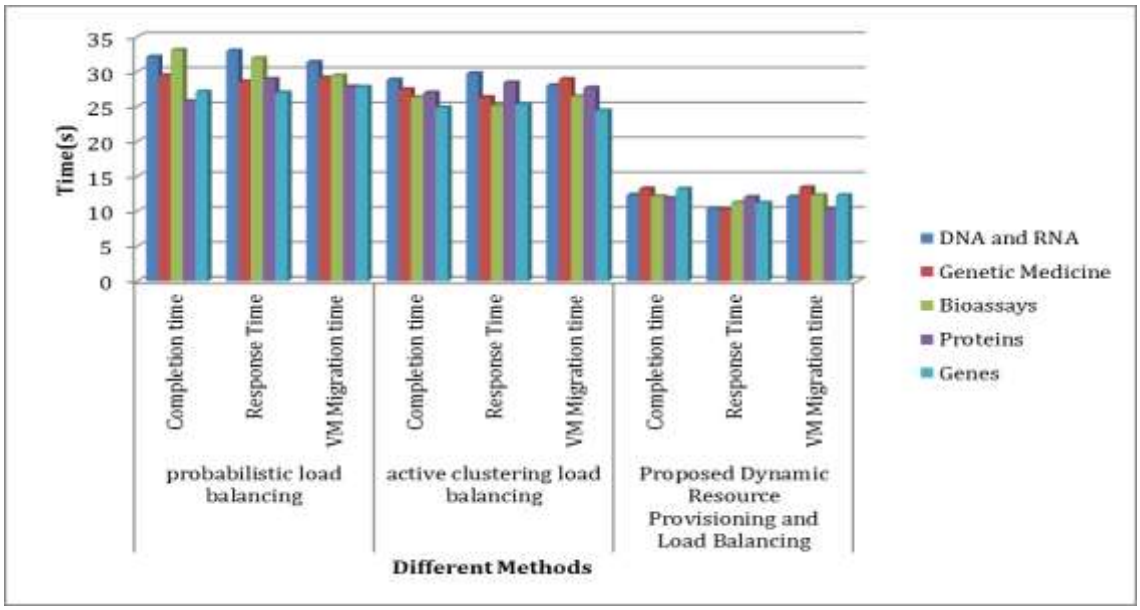

Figure 4. Processing and response time

Based on the above Figure 4, it clearly indicates, that the proposed method process the different gene data by responding, load balancing with VM migration and completing the workflow with minimum time are successfully achieved when compared to the other methods. Even though it process with minimum time, it has to be ensured with highest throughput while processing different kind of gene data as shown in the Table 3 and Figure 5.

Table 3. Throughput

Molecular Substrates	Throughput (%)		
	Probabilistic Load Balancing	Active Clustering Load Balancing	Proposed Dynamic Resource Provisioning and Load Balancing
DNA and RNA	92.8	93.2	95.32
Genetic Medicine	93	93.17	95.56
Bioassays	94	92.89	96.1
Proteins	93.4	94.67	96.89
Genes	94.7	95.1	97.32

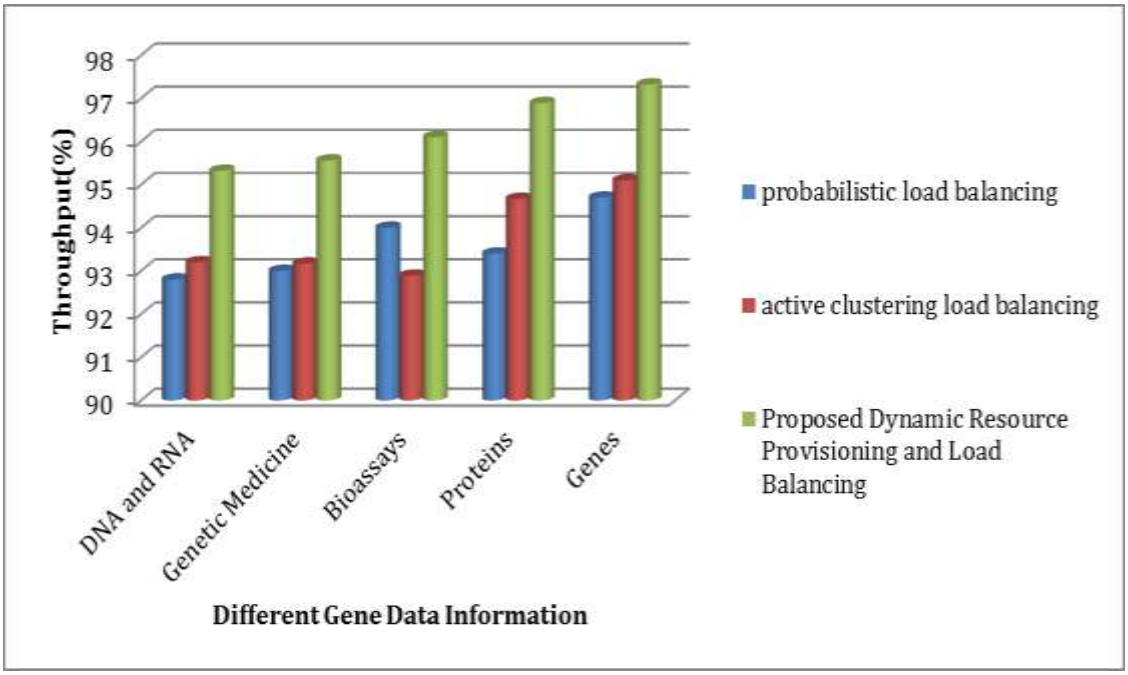

Figure 5. Throughput

According to the above Figure 5, it shows that the different gene data present in the GEO database, DNA and RNA (95.32%), Genetic Medicine (95.56%), Bioassays (96.1%), Proteins (96.89%) and Gene expressions (97.32%) of throughput while analyzing that genomic information by using the proposed methodology. Thus the proposed system consumes high throughput by processing the user request when compared to the other traditional methodologies such as probabilistic load balancing model consumes DNA and RNA (92.8%), Genetic Medicine (93%), Bioassays (94%), Proteins (93.4%) and Gene expressions (94.7%) of throughput and active clustering load balancing method consumes DNA and RNA (93.2%), Genetic Medicine (93.17%), Bioassays (92.89%), Proteins (94.67%) and Gene expressions (95.1%) of throughput. Based on the above discussions, the proposed model effectively process the user gene data with highest throughput rate with minimum time which is demonstrated with the help of the case study.

V. CONCLUSIONS

In this paper, a case study approach has been experimented to evaluate the user workflow request in the cloud. During this process, the cloud provider uses our previous algorithms such as WSHF, AGERL, HFEL and DRBLHS approaches in a step by step manner to allocate the request to the suitable virtual machines with an effective scheduling mechanism. Using these methodologies, the gene expression data set is taken as an input and experimented. The results picture clearly with an achievement of minimum resource utilization, workflow completion time, response time, VM migration time and higher throughput value. On an average, for the same set of samples, resource utilization rate is 24.12% when compared to PLB (49.42%) and ACLB (39.61%) methods. Regarding completion time, response time and VM migration time, the proposed approach acquires minimized 2.35 sec, 2.71 sec and 2.40 sec when compared to PLB methods, similarly minimized 1.98 sec, 2.45 sec and 2.24 sec when compared to ACLB method. In future, our work can be extended by using effective clustering techniques in the cloud.

REFERENCES

1. James Broberg, Rajkumar Buyya & Zahir Tari, (2009) "MetaCDN: Harnessing 'Storage Clouds' for high performance content delivery", Journal of Network and Computer Applications, Vol. 32, No. 5, pp 1012-1022.
2. Kapil Kumar Gupta, Baikunth Nath & Ramamohanarao Kotagiri, (2010) "Layered Approach using Conditional Random Fields for Intrusion Detection", IEEE Transactions on Dependable and Secure Computing, Vol. 7, No. 1, pp 35-49.
3. Gengbin Zheng, Esteban Meneses, Abhinav Bhatele & Laxmikant V Kale, (2010) "Hierarchical Load Balancing for Charm++ Applications on Large Supercomputers", In Proceedings of the 2010 39th International Conference on Parallel Processing Workshops, San Diego, CA, USA, pp. 436-444.
4. Ioan Raicu, Ian Foster & Yong Zhao, (2008) "Many-Task Computing for Grids and Supercomputers", In proceedings of 2008 Workshop on Many-Task Computing on Grids and Supercomputers, Austin, TX, USA.
5. Ewa Deelman, Gurmeet Singh, Miron Livny, Bruce Berriman & John Good, (2008) "The Cost of Doing Science on the Cloud: The Montage Example", In Proceedings of the 2008 ACM/IEEE conference on Supercomputing, Austin, TX, USA, pp 1-12.
6. James Dinan, Brian Larkins D, Sadayappan P, Sriram Krishnamoorthy & Jarek Nieplocha, (2009) "Scalable Work Stealing", In Proceedings of the Conference on High Performance Computing Networking, Storage and Analysis, Portland, Oregon.
7.]Ioan Raicu, (2009) "Many-Task Computing: Bridging the Gap between High Throughput Computing and High Performance Computing", Computer Science

Department, University of Chicago, Doctorate Dissertation.
8. Laxmikant V Kalé, (1988) "Comparing the Performance of Two Dynamic Load Distribution Methods", In Proceedings of the 1988 International Conference on Parallel Processing, pp 8-11.
9. National Center for Biotechnology Information, U.S. National Library of Medicine 8600 Rockville Pike, Bethesda MD, 20894 USA.
10. https://www.ncbi.nlm.nih.gov/geo/info/datasets.html.
11. Suraj Pandey, William Voorsluys, Mustafizur Rahman, Rajkumar Buyya, J Dobson & Kenneth Chiu, (2009) "A Grid Workflow Environment for Brain Imaging Analysis on Distributed Systems", Concurrency and Computation Practice and Experience, Vol. 21, No. 16, pp 2118-2139.
12. Lavanya Ramakrishnan, Shane Canon, Krishna Muriki, Iwona Sakrejda, & Wright N J, (2011) "Evaluating Interconnect and Virtualization Performance for High Performance Computing", ACM SIGMETRICS Performance Evaluation Review, Vol. 40, No. 2.
13. Selvi S & Kalaavathi B, (2017) "Reducing Rejected Workflows using WSHF Algorithm in Cloud", Asian Journal of Research in Social Sciences and Humanities, Vol. 7, No. 3. pp 397-405.
14. Selvi S & Kalaavathi B, (2016) "AGERL Based Enhanced Map Reduce Technique in Cloud Scheduling", SSRG International Journal of Computer Science and Engineering (IJCSE), Vol. 3, No. 10, pp 60-65.
15. Selvi S & Kalaavathi B, (2016) "Enhanced Scheduling Approach Using Heuristics Flow Equilibrium Based Load Balancing Algorithm in Cloud", International Journal of Control Theory and Applications (IJCTA), Vol. 9, No. 2, pp 1023-1034.
16. Selvi S & Kalaavathi B, (2016) "Mutlilayer Intensive Resource Allocation in Cloud Using Leftist Heaps with VM Migration", International Journal of Printing, Packaging & Allied Sciences, Vol. 4, No. 4, pp 2763-2775.

AUTHORS

S.Selvi did her B.E(CSE) in Kongu Engineering College, Perundurai, Tamil Nadu and M.E(CSE) in Velalar College of Engineering and Technology, Thindal, Tamil Nadu. She has completed her Ph.D. under Anna University, Chennai in the year 2019. Her area of interest includes Cloud Computing, Compiler Design and IoT. She has published about 17 papers in various International Journals.

A.Chandrasekar received B.Sc. Degree in Computer Science from Nagamalai Navarasam Arts and Science College, Bharathiar University, Tamil Nadu, India in 1998, M.Sc. Degree in Computer Technology from K.S.R. College of Technology, Anna University, Tamil Nadu, India in 2000, M.E. in Computer Science and Engineering from K.S.R. College of Technology, Anna University, Tamil Nadu, India in 2006. He also obtained his Ph.D. Degree in Information and Communication Engineering from Anna University, Tamil Nadu, India in 2016. He is having 16 years of teaching experience in various institutions. He has published 17 papers in various International Journals. His area of interest includes Mobile Computing, Design and Analysis of Algorithms and Internet of Things.

M.Dhipa completed B.E(EIE) in Easwari Engineering College, Madras University, Chennai in 2004 and M.E (Applied Electronics) in K.S.R. College of Technology, Anna University, Chennai in 2006. She is pursuing Ph.D. under Anna University, Chennai. She is having 12 years of teaching experience in various institutions. She has published about 8 papers in various International Journals. Her area of interest includes Heterogeneous Networks, Mobile Computing and IoT.