IJCSE International Journal of Computer Sciences and Engineering Open Access

Research Paper Vol.-7, Issue-3, March 2019 E-ISSN: 2347-2693

Load Harmonization Using Media Parameters in Massive Clusters of Learning Grids

[1*]G. Manoharan, [2] K. Nirmala

[1,2]Manonmaniam Sundaranar University, Tirunelveli, India

Corresponding Author: manog73@yahoo.co.in Tel.: 9940293459

DOI: https://doi.org/10.26438/ijcse/v7i3.481485 | Available online at: www.ijcseonline.org

Accepted: 07/Mar/2019, Published: 31/Mar/2019

***Abstract*:** E-Content generally requires media components like Graphics/Video in addition to Texts. Independent Learning objects of different sizes having different media components may be alternatives to traditional single large e-content file. The main objective of this paper work is to optimize the load harmonization using media parameters based on grid process running time. This Proposed work is classified into four categories such as estimating grid process running time, harmonization while loading, clustering the media parameters based on instructional and media parameters and processing to optimize the load balancing. The main goal is to load the same shareable media object for the massive user on a particular time using harmonization and clustering. This proposed work produces less grid process running time for the same media object even if the number of the user is larger at the same time.

***Keywords*:** Harmonization, Victimization, Clustering, Optimization

I. INTRODUCTION

1.1 Grid Computing

Grid Computing technology is widely adopted for deploying e-learning content within the internet as any e-learning setting would demand big computing resources significantly for a vast pool of concurrent e-content users. Thus computer code and hardware area unit are required to be updated or revived again and again for this setting. Large learning grid could be a machine cooperative setting for effectively endeavor massive pools of e-learners of the online that's turning into vogue. The one among the basic problems in such a grid environments is job programming, that is required for achieving higher performance. As grid setting is mostly suburbanized and it consists of heterogeneous systems, economical programming technique would be noticeably required for sound acceptable resources for relevant e-content; say massive or little e-content.

1.2. Massive Open Online Course

Web course is aimed toward unlimited participation and open access via the online. Additionally, to ancient course materials like recorded lectures, readings, and downside sets, Several MOOCs offer interactive courses with user forums to support community interactions among students, professors, and teaching assistants (TAs) further as immediate feedback to fast quizzes and assignments. MOOCs area unit is a recent and wide research development in distance education that was initially introduced in 2006 and emerged as a well-liked model of learning in 2012. Early MOOCs typically stressed open-access options, like open licensing of content, structure and learning goals, to market the reprocess and remixing of resources. Some later MOOCs use closed licenses for his or her course materials whereas maintaining free access for college kids

II. RELATED WORK

The emergence of large Open Online Courses (MOOCs) has created a major interest from the upper education establishments. The aim of this study is to explore the lecturers' perceptions of victimization MOOCs as their pedagogics supported Technology Acceptance Model (TAM). This analysis explores, however, the perceived quality and perceived simple use have an effect on the particular use of MOOCs for teaching. Qualitative analysis has been conducted to analyze the perception of lecturers in victimization MOOCs. Findings have disclosed the most concern of lecturers towards victimization of MOOCs as a teaching tool and the way MOOCs will be redesigned to realize the larger social impact. The study has arranged a valuable foundation of information on the present state of MOOCs in education particularly within the MOOC style and implementation.

It is concerning about the grid computing or distributed computing refers to a way within which a central server breaks an oversized machine downside into a collection of freelance workloads and sends it to a variety of connected autonomous computing nodes having their own native memory, storage, and hardware. These nodes then operate the employment victimization their own computing resources, gather the machine results, send it back to the parent server and transfer a brand new set of employment for the process. Computing resources needed to resolve vast machine issues square measure terribly valuable and sophisticated. A price is various effective to distribute these workloads to an oversized range of non-public computers, whose house owners give the idle hardware time of their computers.

Learning Grid requires Heterogeneous Computing resources for different types and volumes of data, and processes.
Different nature of media components may require different capacities of resources for processing. Hence we are influenced.

"Content" means Context allude to Shapes, Texture, Color or some other Information that can be gotten from Image itself. "Content Based" means Search examines the Contents of the picture as opposed to the metadata, for example, Tags, Description or Keywords related with the Image. An “Image Retrieval System” is a Computer System for Searching and Retrieving image from a Database of Digital Image. Many Features of Content Based Image Retrieval but Four of them are considered such as Color, Texture, Shape and Spatial Properties. “Low Level Feature” means Image like color, Texture, Shape can be extracted from the Image.

Regular-frequent patterns are an important type of regularities that exist in transactional, time-series and any other types of databases. A frequent pattern can be said regular-frequent if it appears at a regular interval given by the user specified threshold in the transactional database. The regularity calculation for every candidate pattern is a computationally expensive process, especially when there exist long patterns.

2.1 Load Balancing

Load Balancing is the tasks which are distributed onto multiple computers. For example, I create a regular communications protocol request from a shopper to access an online application; it gets directed to multiple internet servers. Primarily happens here is that the application’s workload is distributed among multiple computers, creating it additional scalable. It additionally helps in providing redundancy, therefore let’s say one server fails in a very cluster, the load balancer distributes the load among the remaining servers. But once miscalculation happens, and therefore the request is moved from a failing server to a practical server it is referred to as-as “failover”.

2.2 Clustering

Grouping or combining together is called clustering. Clustering is classified into two categories based on their file size and memory utilization. Two Categories are:

a) **Instructional Category:** which are classified into four such as Definable, Demonstrable, Solvable and Perceivable based on their file size and memory utilization.
b) **Media Category:** which is mainly focused on Text, Animation, and graphics to determine the load harmonization for massive users.

III. CLASSIFICATION OF ALGORITHMS

Algorithms for Load Harmonization for Massive open online courses dynamically used in the real world are extreme. A sample of Recursive Partition Based K-Means algorithm, G/S/M Model and Load Harmonization are Considered for the comparative study.

3.1 Recursive Partition Based K-Means (RPKM) Algorithms

The Recursive Partition based K-means algorithm (RPKM) is a technique that approximates the solution of the K-means problem through a recursive application of a weighted version of Lloyd’s algorithm over a sequence of spatial based-thinner partitions of the dataset:

Dataset partition induced by a spatial partition, Given a dataset TD and a spatial partition of its smallest bounding box, the partition of the dataset D induced by B is defined as,

P = B (D),

Where B (D) = {B (D)} B∈B and B (D) = {x∈ D: x lies on B ∈B}

Applying a weighted version of K-means algorithm over the dataset partition P consists of executing Lloyd’s algorithm over the set of centers of mass (representatives) of P,
P for all P ∈P, considering their corresponding cardinality (weight), |P|, when updating the set of centroids.

3.2 G/S/M Model

This model is based on an incremental tree. First, for each site, we create a two-level subtree. The leaves of this subtree correspond to the computing elements of a site, and the root is a virtual node associated to the site. These subtrees, that correspond to sites of a cluster, are then aggregated to form a three-level sub-tree. Finally, these sub-trees are connected together to generate a four-level tree called load balancing generic model. This model is denoted by G/S/M, where G is the number of clusters that comprise a grid, S the number of sites in the grid and M the number of Computing Elements. This model can be transformed into three specific models: G/S/M, 1/S/M and 1/1/M, depending on the values of G and S. The generic model is a non-cyclic connected graph where each level has specific functions.

3.3 Load Harmonization

Load Harmonization on media parameters for the massive open online course is estimated using poisson distribution. This Proposed work is classified into four categories such as grid process running time, harmonization, clustering and load balancing. Each shareable object file size and memory utilization can be calculated using java programming if the massive user using the same shareable object that determine for each experimental trials calculate the poisson distribution.

$$\mathbf{P(\lambda, k) = (\lambda k{*}e^{-\lambda})/k!}$$

Find out the chances of occurrence of a number of events in some given time interval or given space conditionally that the value of the average number of occurrence of the event is known.

IV. PROBLEM DESCRIPTION

Information Technology has been developed over the last two decades. Due to Increase of Information Technology, the usages of online course have been increased gradually. So online course are categorized into instructional courses and media courses. Consider a single file (shareable object) viewed by the number of the person at the same time. Due to an increase in a number of the person, loading time of the same file will be slow. To avoid this problem, proposed work implemented to reduce the loading time based on harmonization and clustering.

From Figure 4.1, illustrate the role of the proposed work. All the media file will act as the shareable object. Initially find the size of the object and memory utilization of the object using java programming. Estimate grid process running time by using GridSim 5.0, where the size of the file acts as the x-axis and memory utilization acts as the y-axis. Next, each shareable object is harmonized based on the file size and memory utilization.

After harmonization, each file is clustered based on instructional parameter and media parameter. The final stage is to optimize the load balancing using poison distribution are as follows.

$$\mathbf{P(\lambda, k) = (\lambda^{k}{*}e^{-\lambda})/k!}$$

Where λ is the value of an average number of occurrences of the event is known, ***k*** is the number of events in some given time interval. ***P(λ, k)*** is used to find the load balancing of the shareable object at the same time for the massive user in the open online courses.

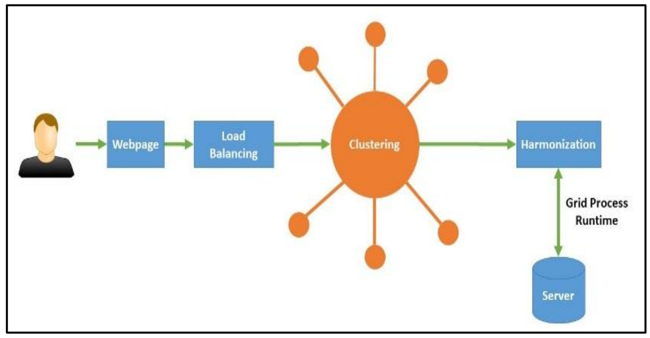

Figure 4.1 Process of Load Harmonization

V. METHODOLOGIES

Methodologies of the proposed work are classified into four categories of the proposed works.

a) **Grid Process Running time:** Let the given file size acts as the x-axis and memory utilization acts as y-axis to determine the grid process running time dynamically.
b) **Harmonization:** Based on the file size and memory utilization the file will be classified based on their Grid process running time.
c) **Clustering:** Clustering is classified into two categories such as instructional parameter i.e., definable, demonstrable, solvable, perceivable and media parameters such as text, graphics and animation.
d) **Load Balancing:** Number of events and the Average number of occurrences can be calculated based on poisson distribution, which is used to optimize the load harmonization.

VI. RESULT AND COMPARISON

The sample data for the proposed experiments contain e-content objects that are in split up form of an integrated lot (both are used in the experiments). Based on the file size and memory utilization of an object, files are clustered into two categories:

- Instructional Parameter
- Media Parameter

The following shows the instructional objects based on the file size and the memory utilization and the processing times are calculated using GridSim 5.0. File size and Memory are measured in terms of Kilo Bytes. Grid processing time is measured in terms of milliseconds.

- The file size of the ‘definable’ instruction object in storage is 188KB; and its size in memory is 186.6KB; processing time by the GridSim 5.0 would be around 2300 ms (excluding user retention time).
- The file size of the ‘demonstrable’ instruction object in storage is 654KB; and its size in memory is 651KB; processing time by the GridSim 5.0 would be around 14300 ms.

- The file size of the 'solvable' instruction object in storage is 362KB; and its size in memory is 660.7KB; processing time by the GridSim 5.0 is around 8600 ms.
- The file size of 'perceivable' instruction object in storage is 160KB; and its size in memory is 158.8KB; processing time by the GridSim 5.0 would be around 1800 ms.
- The file size of the same content integrated into a single document is about 1346KB; size the same in memory is 1340KB; processing time by the GridSim 5.0 would be around 28350 ms (excluding user retention time).

With authorized research support [Kaladevi 2013], the average computational ratio of Definable , Demonstrable , Solvable , Perceivable has been empirically worked out to be about: 1.00 , 4.00,,3.00 , 0.75 (i.e 11.5% : 46% : 34.5% : 8%), which is more or less matching with size and computational time.

In a similar empirical study on media categories with authorized research support [Jagadeesan, B 2014], the average empirical computational ratio of Textual: Graphical: Animation is about 1.00 , 1.10 , 1.30 (i.e 28% : 33% : 39%) respectively.

Table 6.1 Experimental Setup for Grid Resources

Specification	Trials & range
Tasks for massive users (Experiment 1).	10, 50, 100, 200 and 400
Clusters (resources) requested for creation of tasks for massive users (Experiment 2).	1000 – 10000 in steps of 100
Massive clusters that would be grouped into resources.	Varies and will be decided through first experiments
Parameters for selection of clusters (Experiment 1)	Trust, performance, redundancy removal

The above Table 6.1 shows the summarized the specification, trials and range for the samples and inputs for the proposed experimental setup. Tasks of massive users take 10, 50, 100, 200 and 400 experimental trials of tasks. Clusters or Resources for creation of task for massive users takes trial ranges from 1000 to 10000. Massive clusters that would be grouped into resources that can be varied different ranges and will be decided through first experiments. Parameters for selection of clusters for experimental trials are trust based performance based and to remove the redundancy.

As implementing an algorithm, A Sample document consists of ten experimental trials of the task with instructional parameters, the following table is an average number of resource processing time in milliseconds.

Table 6.2 Average Resource Processing Time - Instructional Parameter

TRIALS	DEFINABLE (CLUSTER 4)	DEMONSTRABLE (CLUSTER 8)	SOLVABLE (CLUSTER 5)	PERCEIVABLE (CLUSTER 7)
1000	2858	15865	8740	1867
2000	2846	15020	8904	1845
3000	2818	16980	8965	1898
4000	2840	17030	9083	1904
5000	2878	16820	9154	2065
6000	2903	16092	9258	2037
7000	2840	17896	9474	2176
8000	2880	17670	9787	2297
9000	2895	18005	9937	2303
10000	2816	18875	9956	2455

Resource Processing time is calculated by using the number clusters, a number of experimental trials of the task using GridSim5.0.

Based on the above table 6.2, the following graph, Figure 6.1 generated as follows to determine the grid processing time for the ten experimental trials of the task.

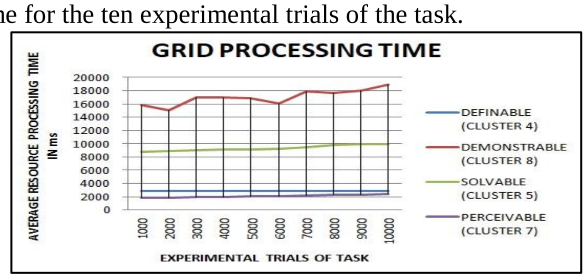

Figure 6.1 Comparison of Grid Processing Time – Instructional Parameter

As implementing an algorithm, A Sample document consists of ten experimental trials of the task with media parameters, the following table is an average number of resource processing time in milliseconds.

Table 6.3 Average Resource Processing Time – Media Parameter

TRIALS	TEXTUAL (CLUSTER 8)	GRAPHICAL (CLUSTER 9)	ANIMATION (CLUSTER 11)
1000	5002	5980	6379
2000	5976	6258	6904
3000	5620	6984	7408
4000	6278	7158	7923
5000	6128	8003	8509
6000	6503	8823	8165
7000	6400	9043	8964

8000	7078	8978	9472
9000	7587	9539	9946
10000	8198	9912	10027

Based on the above table 6.3, the following graph, Figure 6.2 generated as follows to determine the grid processing time for the ten experimental trials of the task.

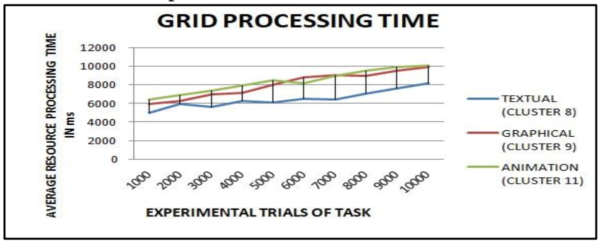

Figure 6.2 Comparison of Grid Processing Time for Media Parameter

VII. CONCLUSION AND FUTURE WORK

The proposed methodology which is more efficient to view the e-content based on the load harmonization techniques on media parameter in massive clusters of learning grids. This proposed methodology makes the user or customer more comfortable in learning and develop their domain knowledge, of particular media parameter such as file can be of Animation, Graphics or Video. The proposed methodologies are comparing with existing two algorithms such Recursive partition based K- mean algorithm and G/S/M algorithms. Determination of Load Harmonization on media parameters, based on the average number of success and actual number of success from the given experiments are optimized and compared two the exiting algorithms.

This Proposed Application may have a future scope on Interactive media parameter such as online quiz , Cloud implementation for the data storage in the heterogeneous environments and Implementation of Mobile App.

REFERENCES

[1] Ng WaiKeat, Ang Tan Fong, Ling Teck Chaw and LiewChee Sun "Scheduling Framework for Bandwidth-Aware Job Grouping-Based Scheduling in Grid computing", Malaysian Journal of Computer Science, Vol.19 No.2, pp. 117-126, 2006.

[2] Thanasis G. Papaioannou and George D. Stamoulis, " Reputation-based estimation of individual performance in collaborative and competitive grids", Future Generation Computer Systems, Vol.26, Issue 8, pp. 1327-1335, 2010.

[3] RajkumarBuyya, David Abramson, Jonathan Giddy and Heinz Stockinger, "Economic models for resource management and scheduling in Grid Computing", Concurrency and Computation: Practice and Experience, Vol. 14, pp. 1507-1542, DOI: 10.1002/cpe.690. 2002.

[4] Yunnan Zhang, Wei Sun and YasushiInoguchi, "CPU Load Predictions on the Computational Grids", Proceedings of Sixth IEEE International Symposium on Cluster Computing and the Grid (CCGRID 2006). Singapore, May16-19.DOI.ieeecomputersociety.org/10.1109/CCGRID.2006.27, 2006.

[5] DeeptiMalhotra and Devanand, "Framework for Job Scheduling in Grid Environment", International Journal of Computer Applications, Volume 38, Number 7, pp. 42-44, DOI-10.5120/4703-6861,2012.

[6] ShaikNaseera, and Madhu Murthy K.V, "Trust-Based Load Distribution in a Data Grid Environment", International Journal on Computer Science and Engineering, Vol. 02, No.01S, pp. 24-29, 2010.

[7] Wen-Chung Shih, Chao-Tung Yang,andShian-Shyong Tseng,"Ontology-based Content Organization and Retrieval for SCORM-compliant Teaching Materials in Data Grids", Future Generation Computer Systems, No. 25, pp. 687-694,2009.

[8] Hsin-Chuan Ho, Chao-Tung Yang,and Chi-Chung Chang, "Building an E-Learning Platform by Access Grid and Data Grid Technologies", Proceedings of IEEE International Conference on e-Technology, e-Commerce and e-Service, Taipei, Taiwan, 2004, pp:452-455,2004.

[9] A.D. Mahajan1*, S. Chaudhary2-"Hybrid Features For Content Based Image Retrieval System", IJCSE, ,pp 11-15, Vol.-6, Issue-10, Oct 2018 E-ISSN: 2347-2693.

[10] S. Rana, "Analysis of Regular-Frequent Patterns in Large Transactional Databases", IJCSE pp 1-5, Vol.-6, Issue-7, July 2018 E-ISSN: 2347-2693

[11] Merrill M.D., "First Principles of Instruction", Englewood Cliffs, NJ: Educational Technology Publications,2002

[12] Ani Brown Mary, N and Saravanan K, "Performance Factors of Cloud Computing Data Centres Using M/G/1/GD Model Queuing Systems", International Journal of Grid Computing & Applications (IJGCA) Vol. 4, No. 1, March 2013.

[13]Selvi, K,"An Enhanced GridwayMetascheduler for Heterogeneous Grid Using Trust and Reputation", Ph.D. dissertation, Anna University, Chennai, India, 2012.

[15] Kaladevi, "Load Balance Optimization in Learning Grids Through Semantic Matching for E-Contents", Ph.D. dissertation, Anna University, Chennai, India, 2013.

[16] Jagadeesan,B,"Domain Dependent Vertical Scalability for Development Efforts on E-Contents", Ph.D. dissertation, Anna University, Chennai, India, 2014.

Authors Profile

Mr.G.Manoharan M.Sc,M.Phil, is pursuing P.hD in Manonmaniam Sundaranar University, Tirunelveli. He is working as Assistant Professor in computer science department for the past 18 years. He published five international journals and ten conference papers.

Dr.K.NIRMALA, M.C.A.,Ph.D., is Research supervisor in Manonmaniam Sundaranar University, & Associate professor, Department of computer science, Quaid-e-millath govt.college for women, Chennai-02.She published more than 25 international journals and organized five international conferences. She has been working in profession for the past 20 years. Dr. Nirmala produced more than 15 P.hD Scholars and 20 M.Phil students.

