International Journal of Latest Trends in Engineering and Technology
Special Issue SACAIM 2016, pp. 37-42
e-ISSN:2278-621X

OPTIMIZATION TECHNIQUES FOR INTEGRATED JOB SCHEDULING AND DATA REPLICATION IN GRID ENVIRONMENTS USING OPTORSIM SIMULATOR

Abdo H. Guroob[1], Manjaiah D.H.[2]

Abstract- Data Grid Environments (DGEs) are a geographically distributed system that are emerging as the solution of data-intensive applications in scientific and engineering enterprise projects, which require to sharing frequently used data. Thus,DGEs are need to minimize and optimize the job execution time very important for high performance and throughput in these environments. Job Scheduling and Data Replication have considered the major challenges in the Data Grid.Nevertheless,many researchers focuson scheduling or data replication separately. In this paper, we present the effects of combining the various job scheduling and data replication strategies together and evaluate several performance metrics using OptorSim simulator, which is original as part of the European Data Grid (EDG) project. The simulation result shows that combining the various job scheduling and data replication strategies can have more effect on both performance and throughput in minimizing and optimizing the job execution time.

Keywords: Data Grid, Job Scheduling, Data Replication, OptorSim simulator.

I. INTRODUCTION

Nowadays data-intensive the major challenge in Data Grid Environment, because dealing with huge data very difficult, in addition, these huge data need to process in the minimum time as possible. In the literature, many techniques are there, which trying to reduce the job execution time for high performance and good throughput. From these techniques job scheduling and data replication algorithms. When these techniques are implemented in Grid Environment, give different results, and many researchers day by day are creating new techniques for that. In contrast, most of these techniques are focusing on job scheduling or data replication separately. Consequently, the results of most of them lacking to an ideal outcome. OptorSim simulator was developed by European Data Grid project for studying the effect of different algorithms like scheduling and replication as a performance measure in Grid Environment.

The rest of this paper is organized as follows: section 2 discusses related work, section 3 describes the architecture OptorSim simulator, section 4 evaluates the performance of simulation experiments using OptorSim, and section 5 conclusion and suggestions for future work.

[1] *Department of Computer Science Mangalore University, Mangalore, Karnataka, India*
[2] *Department of Computer Science Mangalore University, Mangalore, Karnataka, India*

II. RELATED WORK

There are some works that address the challenges of scheduling and/or replication in Data Grid Environments as well as the combination between them. Ranganathan, K. and Foster, I. [22] proposed a data locality in job scheduling problem. The authors proposed their architecture based on 3 main components: External Scheduler (ES), Local Scheduler (LS) and Dataset Scheduler (DS). On this architecture, the authors developed and evaluated various data replication and scheduling strategy to study effect of the two systems. The result showed the importance of data locality in scheduling job. William H. Bell, et al [3]. proposed architecture is the combination and improvement of data replication and job scheduling in a two-stage optimization mechanism using OptorSim and the result showed the importance of data replication and scheduling in Data Grid. Chakrabati and Tang, M et al. [7] develop the older works by integrating the scheduling and replication strategy to increase the scheduling performance.

III. OPTORSIM ARCHITECTURE

OptorSim's architecture, whichis based on the European Data Grid CMS testbedarchitecture, was developed by the European Data Grid project. The main components of OptorSim are Computing Elements (CEs), Storage Elements (SEs), Resource Broker (RB), Replica Manager (RM), and Replica Catalog (RC) as shown in figure 1. CEs are responsible for running jobs of data files in the Grid site, in which data files are stored in the same SE. Initially, a user submits jobs to the Grid sites according to the access pattern, which the user can choose. In OptorSimthereare several types of access patterns such as:

- SequentialAccessGenerator - The access to files in an orderly manner.
- RandomAccessGenerator - The access to files using a flat random distribution.
- RandomWalkUnitaryAccessGenerator - The access to files using a unitary random walk.
- RandomWalkGaussianAccessGenerator - The access to files using a using a Gaussian random walk.
- RandomZipfAccessGenerator - The access to files using a Zipf distribution.

After choosing one of the types of the access patterns, a single Resource Broker (RB) in the Grid Environment schedules the jobs for submission to the sites that have the data files, which are related to the required job according to specific scheduling algorithm to accelerate the job execution time for all work. If the data files are not present on the implementation site, so it therefore needs to be either replicated or read remotely. In each site in the Grid Environment there Replica Manager (RM), is responsible for handling own content data. Inside every RMthere exists a Replica Optimizer (RO), which is responsible for optimization performance according to data replication algorithms that lead to reducing the job execution time.

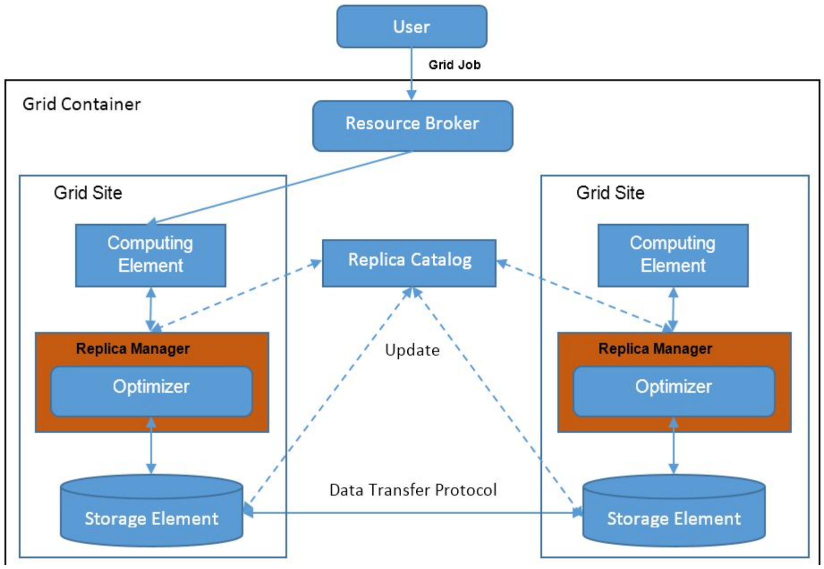


Figure 1: OptorSim architecture based on the EDG data management components.

A. Optimization Algorithms

There are two different types of optimization in OptorSim: The Job Scheduling Algorithm used by the Resource Broker to allocate jobs, and the Data Replication Algorithm used by the Replica Manager at each site to decide when to replicate a file, or which file to replicate and which to delete.

The main goal for using these algorithms is to reduce the job execution time, and use the Grid resources ideally. Currently,the optimization algorithms implemented in OptorSim are:

Scheduling algorithms

- Random - site is accessed randomly
- Access Cost - site has shortest time to access to all files required by job
- Queue Size - access to site, which has job queue, is the shortest
- Queue Access Cost - access cost to site in which all job in queue is shortest

Data Replication algorithms

- No replication
- Least Recently Used (LRU) - always data are replicated, delete the file which has least recently used
- Least Frequently Used (LFU) - always data are replicated, delete the file which has least frequently used
- Economic model (Binomial) - replicate data if economically useful, using binomial prediction function for file values
- Economic model (Zipf) - replicate data if economically useful, using Zipf-based prediction function.

B. Simulation Setup and Metrics

This paper uses job execution time as the evaluation metrics. Job Execution Time is defined as the total time to execute all jobs, divided by the number of jobs completed.

$$JobExecutionTime = \frac{TotalTime\ of\ All\ Job}{Number\ of\ Jobs}$$

OptorSimcan be run from the command line or from a GUI, and the output of simulator in statistics form are:

- Number of Jobs Remaining
- Mean Job Time of all Jobs on Grid
- Total Number of Replications
- Total Number of Local File Accesses
- Total Number of Remote File Accesses

• Percentage of CEs in Use
• Percentage of Storage Filled/Available
• Effective Network Usage.

IV. SIMULATIOM RESULTS

The topology is used for Data Grid in this simulator (CMS testbed), consists of 20 sites, every site has one computing element and storage of capacity 50 GB such as in figure 2. The simulation results obviously show that Queue Access Cost is the best scheduler when our interest in job times and computing element usage. The Least Frequently Used replication algorithm is faster when only 1000 jobs are submitted, but the economic models perform better when there are more jobs, especially with the Zipf-based prediction function. In contrast, when we combine the Queue Access Cost scheduling algorithm and LFU data replication algorithm as one integrated performance, then we get the best result of simulation in various metrics situations.

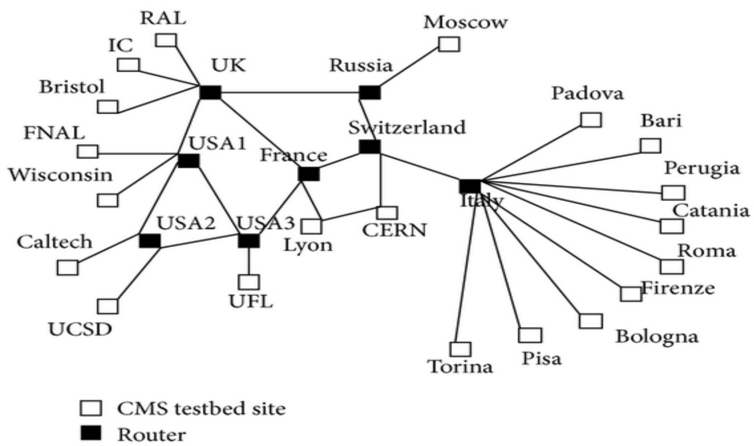


Figure 2: The grid topology (CMS testbed)

Table 1 illustrates the different parameters that set up in OptorSim simulator to evaluate the integrated job scheduling and data replication algorithms.

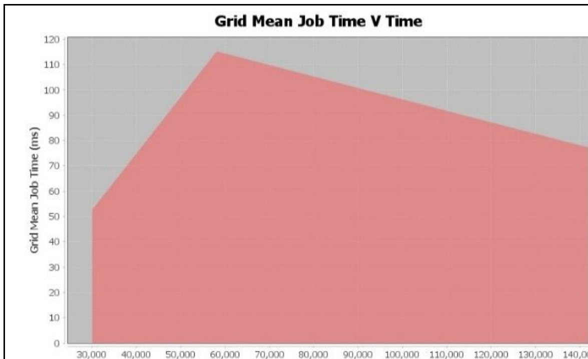
Table 1. Simulation Parameters

1	Number of jobs	1000
2	Access Pattern	Sequenti al
3	Access History Length (ms)	1000000
4	Zipf distribution shape	0.85
5	Initial File Distribution	8
6	Job Delay (ms)	2500
7	Max. Queue Size	200
8	Auction Initial Timeout (ms)	500
9	Start Time	0.0

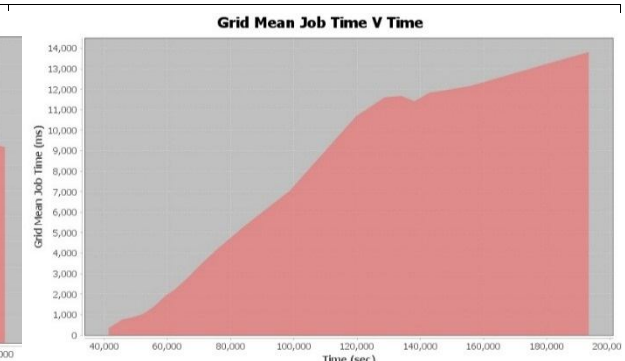
In table 2 below shows the results of simulating more than 60 times, each algorithm 3 times. Which reveals the various results of simulation from Scheduling Algorithms (SA) and Replication Algorithms (RA).According to the results obtained in table 2, we take snapshots from OptorSim's GUI, which are seenin figure 3, where the algorithms that have the longest time in implementing jobs and the algorithms that have the shortest time in the implementation of jobscan be seen. Where (a) Queue Access Cost, Least Recently Used (LRU) has the shortest job execution time and (b) Random, Economic model (Zipf) has thelong job execution time.

Table 2. The different results of simulation for Scheduling Algorithms (SA) and Replication Algorithms (RA)

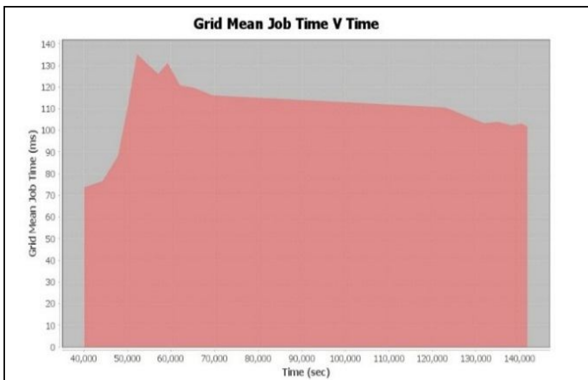
SA \ RA	No replication	Least Recently Used (LRU)	Least Frequently Used (LFU)	Economic model (Binomial)	Economic model (Zipf)
Random	9412, 8497, 8375	6759, 6859, 8374	7899, 8191, 6477	6206, 387, 9722	17144, 15731, 15421
Access Cost	416, 422, 484	447, 464, 487	461, 534, 433	496, 438, 502	395, 460, 460
Queue Size	221, 246, 297	234, 247, 296	142, 124, 126	166, 170, 189	400, 380, 412
Queue Access Cost	115, 119, 122	67, 73, 75	73, 68, 76	102, 87, 81	87, 76, 89



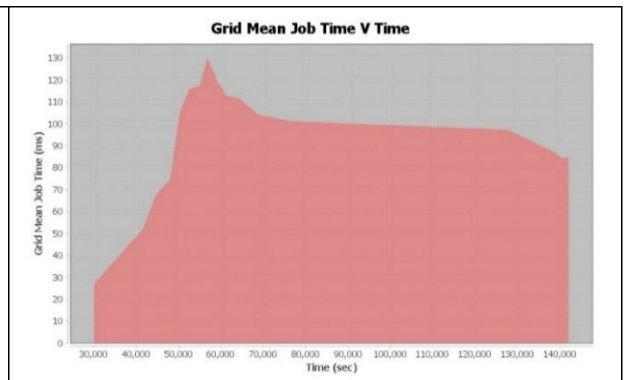
(a) Queue Access Cost, Least Recently Used (LRU)



(b) Random, Economic model (Zipf)



(c) Queue Access Cost, Economic model (Binomial)



(d) Queue Access Cost, Economic model (Zipf)

Figure 3: Mean job time for optimization algorithms with different schedulers and replication, 1000 jobs, CMS testbed.

V. CONCLUSION

The optimization of performance in the Grid Environments is very important for enhancing the throughput of the jobs. Job Scheduling and Data Replication Algorithms are the main methods of optimization in the OptorSim simulator. In contrast, simulate a job scheduling and data replication algorithms separately do not lead to better results. In contrast, if the user configures his work according to the effectiveness of the scheduling and replication as a single unit integrated metrics, the results give better performance according to the type of algorithm used in the Grid Environment, where not all integrated scheduling and replication algorithms give the better result. Consequently, in this paper presented the effect of scheduling and replication algorithm using different algorithms in the OptorSim. In future work, we plan to evaluate and develop integrated algorithms for scheduling and replication that have more effect and reduce the job execution time as possible.

REFERANCES

[1] William H. Bell, et al.,“OptorSim - A Grid Simulator forStudying Dynamic Data Replication Strategies,”International Journal of High Performance ComputingApplications,17(4), 2003.

[2] B. T. Huffman, R. McNulty, T. Shears, R. St. Denis and D.Waters “The CDF/D0 UK GridPP Project”, CDF Internal Note 5858, 2002.

[3] William H. Bell, et al,“Evaluation of an Economy-Based FileReplication Strategy for a Data Grid,”InternationalWorkshop on Agent based Cluster and Grid Computing atCCGrid 2003, Tokyo, Japan, May 2003. IEEE ComputerSociety Press.

[4] David G. Cameron, et al,“Evaluating Scheduling and ReplicaOptimisation Strategies in OptorSim,”4th InternationalWorkshop on Grid Computing (Grid2003), Phoenix, Arizona,November 17, 2003. IEEE Computer Society Press.

[5] K. Ranganathan and I. Foster,“Decoupling computation anddata scheduling in distributed data-intensive applications,”InProceedings of the Eleventh IEEE Symposium on High

[6] PriyankaVashisht, Rajesh Kumar, Anju Sharma, “Efficient Dynamic Replication Algorithm Using Agent for Data Grid,” The Scientific World Journal, Volume 2014 (2014), Article ID 767016, 24 March 2014. http://dx.doi.org/10.1155/2014/767016.

[7] Tang, M., Lee, B., Tang, X., and Yeo, “The impact of data replication on job scheduling performance in the Data Grid,” Future Generation Computing System 22, 3 (Feb. 2006), 254-268. DOI= http://dx.doi.org/10.1016/j.future.2005.08.004

[8] XU PengZhi, WU YongWei∗, HUANG XiaoMeng, “Optimizing write operation on replica in data grid,” SP Science China Press (SCIS), Volume 54, Issue 1, pp 1-11, January 2011.

[9] Mansouri, N., Dastghaibyfard, G.H., Mansouri, E., “Combination of data replication and scheduling algorithm for improving data availability in Data Grids,” Journal of Network and Computer Applications, vol. 36, Pages 711–722, March 2013.

[10] J. M. Pereza, F. Garca-Carballeira, J. Carreteroa, A. Calderona, and J.Fernndeza. “Branch replication scheme: A new model for data replication in large scale data grids,” Future Generation Computer Systems, 26(1):12–20, 2010.

[11] D. Nukarapu, B. Tang, L. Wang, and S. Lu., “Data replication in data intensive scientific applications with performance guarantee,” IEEE Transactions on Parallel and Distributed Systems, 22(8):1299–1306, Aug. 2011.http://doi.acm.org/10.1145/2522548.2522605.

[12] Shvachko, K., Kuang, H., Radia, S., Chansler, R, “The hadoop distributed file system,” IEEE 26th Symposium on Mass Storage Systems and Technologies(MSST), pp. 1–10, May 2010.

[13] Abdo H. Guroob, Manjaiah D.H, “Data Replication in Data Grid Environment,” CSI Communications: Knowledge Digest for IT Community, Volume No.39, and Issue No.8 November 2015.

[14] Y. Lin, Y. Chen, G. Wang, B. Deng, “Rigel: A scalable and lightweight replica selection service for replicated distributed file system,” 10th IEEE/ACM International Conference on Cluster, Cloud and Grid Computing, CCGC, pp. 581–582, 2010.

[15] The European DataGrid Project, http://www.edg.org

[16] D. Cameron, R. Carvajal-Schiaffino, P. Millar, C. Nicholson, K. Stockinger and F. Zini, “ Evaluating Scheduling and Replica Optimisation Strategies in OptorSim”, Journal of Grid Computing (to appear)

[17] K. Ranganathan and I. Foster, “Identifying Dynamic Replication Strategies for a High Performance Data Grid”, Proc. of the Int. Grid Computing Workshop, Denver, Nov. 2001

[18] W. Bell, D. Cameron, L. Capozza, P. Millar, K. Stockinger and F. Zini, “OptorSim - A Grid Simulator for Studying Dynamic Data Replication Strategies”, Int. J. of High Performance Computing Applications 17 (2003) 4

[19] W. Bell, D. Cameron, R. Carvajal-Schiaffino, P. Millar, C. Nicholson, K. Stockinger and F. Zini, “ OptorSim v1.0 Installation and User Guide”, February 2004

[20] W. Bell, D. Cameron, R. Carvajal-Schiaffino, P. Millar, K. Stockinger and F. Zini“Evaluation of an Economy-Based Replication Strategy for a Data Grid”, Int. Workshop on Agent Based Cluster and Grid Computing, Tokyo, 2003

[21] GDB Resource Allocation and Planning, http://lcgcomputing-fabric.web.cern.ch/LCG-Computing- Fabric/GDB resource allocation planning.htm

[22] K. Ranganathan and I. Foster, “Simulation Studies of Computation and Data Scheduling Algorithms for Data Grids,” Journal of Grid Computing, Volume 1, Number 1, 2003, pp. 53-62(10).