JACET Journal of Advances in Computer Engineering and Technology, 2(3) 2016

E2DR: Energy Efficient Data Replication in Data Grid

Kobra Bagheri[1], Mehran Mohsenzadeh[2]

Received (2016-07-14)
Accepted (2016-11-08)

Abstract — **Data grids are an important branch of gird computing which provide mechanisms for the management of large volumes of distributed data. Energy efficiency has recently emerged as a hot topic in large distributed systems. The development of computing systems is traditionally focused on performance improvements driven by the demand of client's applications in scientific and business domains. High energy consumption in computer systems leads to their limited performance because of the increased consumption of carbon dioxide and amount of electricity bills. Thus, the goal of design of computer systems has been shifted to power and energy efficiency. Data grids can solve large scale applications that require a large amount of data. Data replication is a common solution to improve availability and file access time in such environments. This solution replicates the data file in many different sites. In this paper, a new data replication method is proposed that is not only data aware, but also is energy efficient. Simulation results with CLOUDSIM show that the proposed method gives better energy consumption, average response time, and network usage than other algorithms and prevents the unnecessary creation of replica, which leads to efficient storage usage.**

Index Terms — **data replication, data grid, energy efficient, CLOUDSIM**

1- Department of Computer, Science and Research Branch Islamic Azad University, Tehran , Iran.
(Bagheri128@gmail.com)
2- Department of Computer, Science and Research Branch Islamic Azad University, Tehran, Iran.

I. INTRODUCTION

Large scale geographically distributed systems are becoming popular for data sharing in data intensive scientific applications [1]. Data grid systems allow for sharing data or sources in dynamic and multi-institutional virtual organizations [1,2]. Data grids are among the components of grid computing that manages and processes a large amount of distributed data [1]. Biomedical Information Research Network (BIRN) [4] and Large Harbon Caller (LHC) [5] are two grid examples. These scientific experiments have generated millions of files and thousands of clients around the world that will access these data files [3,4]. In the near future, the volume of data that need to be accessed on data grids may be up to terabyte [3]. Data intensive applications are one of the major applications that run in data grids, and data replication methods are one of the most important research fields [4,6]. Data replication is an important technique to manage a large number of data in a distributed manner and places replicas of data in various locations [5,7]. Replica placement, replica management, and replica selection are three key issues in all the data replication algorithms [5]. If replicas are placed in the appropriate site, bandwidth consumption and average response time will be reduced [5]. Unnecessary data replication and job scheduling in data grids and poor quality and inefficient use of resources cause high energy consumption. They also depend on resource management and efficiency of applications running in the systems [8]. Fan [9] found a strong relationship between CPU utilization and total power consumption by a server [9]. When components of servers have inefficient power in the idle state, overall narrow dynamic power range of 30% occurs, which shows that, even if a server is completely idle, it will still consume more than 70% of its peak power [8]. The main aim of this paper is to use the idle state of server efficiently and reduce energy consumption in data grids. Free storage of each site of grid is limited and inefficient replica of data in site wastes storage; but, proposed method that called Energy Efficient Data Replication (E2DR) replicates data based on the storage of sites.

II. RELATED WORKS

Energy is a key challenge in large scale distributed systems such as grids [10,11,12] and clouds [13,14]. Energy resources are frequently consumed and requests for replacement energy resources are growing [15]. This issue has been discussed at different levels and some studies are focused on specific components, e.g. network interface cards [16] and CPU [17,18]. In [19], the authors introduced a model of energy consumption based on CPU activity. In [20,21], energy consumption is decreased using event–monitoring counters. ON/OFF algorithms [22], load balancing [21], task scheduling [22,23,24], or thermal management [9,20] is some of more general studies. In [25], a method was proposed that considered energy efficiency and bandwidth consumption of the system.

Recently, data replication in data grids has been an important research topic in data grid [26]. In [27], centralized and decentralized two data replication algorithms were introduced. In the centralized method, a table was used by replica master that ranked each file access in a descending order. If average of file access were bigger than a file access, it would be removed from the table and all of files that were at the top of table would be deleted and replicated using a replacement algorithm. In the decentralized method, each site saved file access in its table and updated this table with neighbors. Since every site knew the average number of access for each file, it removed the files whose number of access was less than the average number of access and replicated them locally. In [28], BHR method was proposed that was used for a 2-level hierarchical structure based on the Internet hierarchy and only considered dynamic data replication without taking scheduling into account. Rangathan and Foster [29,30] proposed six different data replication strategies: (1) No Replication: This method replicates data in a central site at the first of the scenario and is the base state for comparison with other methods; (2) Best Client: A replica is created at the best client that has the largest number of requests for the file; (3) Cascading: A replica whose popularity exceeds the threshold is replicated at the next level which is on the path to the best client. The threshold in this method is the average of file access; (4) Plain Caching: The client that requests the data file stores a copy of this file locally; (5) Caching and Cascading: Attributes of caching and cascading methods are merged; and (6) Fast Spread: Copies of files at each node are replicated on the path to the best client.

In [31], an efficient two-level job scheduling algorithm and two-phase dynamic replication strategy, named JS2DR2, were proposed for data grid. In this method, a two-phase dynamic replication strategy coupled with two-level job scheduling was proposed to provide efficient data access and job scheduling. In the first phase, replicas were created and, in the second phase, the best network link from local node to the nodes containing other replicas of some files was selected.

Dynamic replication algorithm (DRA) [32] is another method that was proposed by Sachi et al. (2010) for European Data grid. This strategy considers network topology in different clusters. Sites that are close to each other are in a cluster. DRA improves file access in the cluster by data replication. The data are initially produced in cluster master and then distributed to all cluster heads. So, access frequency of all files is determined and the most popular files to the site that have the highest requests for these files are replicated considering the geographical and temporal localities.

Khanli et al. (2011) proposed a hierarchical fast spread (PHFS) algorithm based on fast spread in multi–tier data grid. PHFS tries to predict user's subsequence component to adapt the replication configuration with the available condition and increase locality in access. One of the main results is that PHFS is suitable for applications that are not random. Results of simulation show that PHFS has better performance and less latency than fast spread [33].

In [34], two methods, called Simple Bottom Up (SBU) and Aggregate Bottom Up (ABU), were proposed for the multi-tier data grid. SBU replicates data close to the clients requesting the data files with high rates which exceed the pre-defined threshold. If the number of request for file “k” is bigger than the default threshold and “k” exists in the parent node of the client which has the highest request rate, there is no need for replication. In contrast, if “k” does not exist in the parent node of the client and the parent node has enough available space for it, then replication occurs. SBU does not consider the relations among historical records and process the records individually, which are its disadvantages. But, ABU aggregates the historical records to the upper tier until reaching the root. ABU adds to the number of access for the records that access the same file and have the same parent. After the aggregation, the node id of the same parent replaces the parent id in the records.

Shorfazman et al. (2010) proposed Popularity-based replica placement (PBRP) method [35] for hierarchical data grids forwarded by file popularity. File access rate by the clients determined the popularity of files. This method replicated copies of file in nearby clients to decrease the number of file access. The threshold of file popularity was the most important in PBRP method. They also proposed Adaptive-PBRP (APBRP) method that calculated this threshold dynamically based on data request arrival rates. Simulation results showed that PBRP had better average response time and average bandwidth consumption than other data replication strategies.

In File Reunion (FIRE) algorithm which was proposed in 2010, each site saved access history of all the local files and number of requested to remote file and then removed old files to replace new files. FIRE replicates files that are requested by a group of jobs in data grid [36].

Data replication strategies do not consider energy parameter; but, E2DR considers to energy for data replication in data grids.

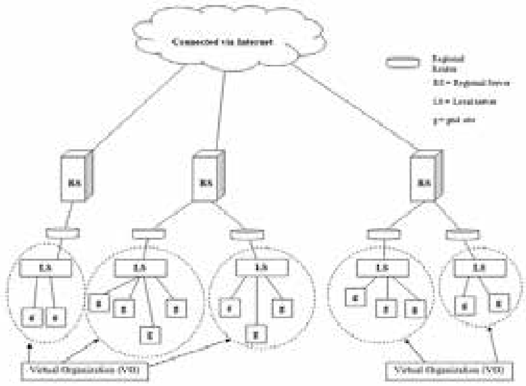

Fig.1. Structure of data grid for E2DR

III. PROPOSED METHOD

In this section, first, the network structure is described and then E2DR algorithm is proposed.

1. Network Structure

Figure 1 shows the structure of the proposed method that could support data replication and energy consumption. Cluster or virtual organization unit is a group of sites that are geographically close to each other; each virtual organization consists of the computers that are connected by a high bandwidth. Each cluster has a header or local server. Grid sites are at the lowest level of this structure. Each site has a computation and storage element. Local server saves file access sequence and list of replicas in each cluster. Speed of access to replicas in each cluster is faster than the one across another cluster. Cluster servers are at the upper level that manages one or more clusters. Speed of data transmission in a cluster is bigger than the one across another cluster in a hierarchical structure. At first, end user submits jobs at resource broker and the broker schedules the jobs on local server according to the available scheduling strategies. After cluster servers transmit jobs to their local servers, finally, local servers assign jobs to the grid sites.

2. Data Replication Phase

When a job is allocated to a site of grid and files of job are not locally required, these files should be transferred to the site by a replica manager. Therefore, the site sends a request to local server and gathers all of the cluster’s access in one place. There is also a table that sorts file name, location of sites, file access, and access time for each file in each row. The access time is the most important element, because information in each time interval is twice as much as the previous period. Therefore, time is divided into different time intervals and assigned to the table information according to Equation 1.

$$Time\ Stamp = 2^{-(N_T - t)} \quad (1)$$

N_T: Number of time intervals.
t: t th time interval.

If the number of file access is larger than the average number the access, the file is popular and local server replicates it in the best location by Equation 2.

$$BRS = Number\ of\ Request * Time\ Stamp \frac{Free\ Memory\ of\ Site}{File\ Size} \quad (2)$$

Number of request = Number of request of this site for data file

Time stamp= we use Equation 1 for this parameter to give lower weight to the previous access

This equation is calculated for the sites that request popular files and is replicated in the site that has maximum BRS. If the site does not have enough storage for the requested file, some files of the site should be deleted and new files are saved. Least Recently Used (LRU) and Least Frequency Used (LFU) are two strategies for putting new copies. LRU deletes copies that have not been recently used and LFU deletes the copies that are not frequently used.

3. Energy Consumption Phase

When jobs are scheduled and allocated to the sites, the workload of site is increased and energy consumption is increased in a non-linear way, which is inefficient for the grid. It may overload one or more sites and double energy consumption.

In this method, local servers in each cluster must do something in a special time period to reduce energy consumption (the time period is 300 Ms). Local server sorts the workload of their sites in a descending order. In this method, two MIN and MAX thresholds are defined that are analyzed in the next section. If a site is overloaded, i.e. the site workload is bigger than the max threshold, then the workload should be reduced, and load of the site transmit to a suitable site for energy consumption reaches an acceptable level, and the site load becomes lower than the max threshold.

For this purpose, the specific amount of workload is transmitted to the site so that its total and transmitted workload would be less than the max threshold. If there is more than one site with this condition, the site which has the workload with maximum distance to the max threshold is selected.

If several sites are in the same situation, one site is randomly selected. In some cases, sites have a very low workload and their energy consumption is very high. In this case, workload is transferred to another site that has maximum distance to the max threshold and the site is hibernated.

When workload is transmitted to another site and all sites are overloaded in a cluster, the hibernated site that consumes less energy than other sites in the past should be woken up and workload should be transmitted to it. In all of the above conditions, tasks are transmitted to other sites if the target sites have the required CPU and RAM.

IV: Evaluation

Table 1 shown comparison between replication methods and E2DR.

In order to evaluate the performance of E2DR, CLOUDSIM tool is used for data grid topology shown in Figure 1. In past, authors have used GRIDSIM and OPTORSIM simulators for the simulation of data replication in grids; but, E2DR merges data replication method and energy. Therefore, we use CLOUDSIM simulator that supports energy consumption in distributed systems. We compare E2DR, FIRE, and JS2DR2, because these methods have better response time and network usage than other methods. Table 2 demonstrates the parameters used in the simulation experiment.

Table1. Comparison of replication method and E2DR

	ABU, SBU	PBRP	BHR	PHFS	DRA	JS2DR2	E2DR
Replication Decision	Centralized	Centralized	Decentralized	Decentralized	Decentralized	Decentralized	Centralized
Architecture	Multi-Tier	Multi-Tier	General	Multi-Tier	Graph	Hierarchical	Hierarchical
Improved Availability	YES	YES	YES	YES	YES	YES	YES
Reduced Response Time	YES	YES	YES	YES	YES	YES	YES
Scalability	NO	NO	NO	NO	NO	YES	YES
Reliability	NO	NO	NO	NO	NO	NO	YES
Bandwidth Consumption Considered	YES	NO	YES	YES	YES	YES	YES
Load Balancing Considered	NO	NO	NO	NO	NO	NO	YES
Fault Tolerant Consideration	NO	NO	NO	NO	NO	NO	YES
Storage Assumption	Limited	Limited	Limited	Limited	Limited	Limited	Limited
Storage Utilization	Improved	Improved	Improved	Average	Optimal	Optimal	Optimal
Reduced Access Cost	NO	YES	YES	YES	NO	YES	YES
Threshold Based	NO	NO	NO	NO	YES	YES	YES
Optimal Number of Replica	NO	NO	NO	NO	YES	YES	YES
Top to Down/Down to Top	Down to Top	Top to Down/ Down to Top	Down to Top	Top to Down	Top to Down	Top to Down	Top to Down
Energy Considered	NO	NO	NO	NO	NO	NO	YES

Table2. Grid and Job Configuration

Value	Topology Parameters
10	**Number of clusters**
50	**Number of sites**
100	**Number of computing elements**
50	**Number of virtual machines**
Value	**Job Parameters**
500-3000	**Number of jobs**
6	**Number of job types**
1000-4000	**Length of jobs**
6 min- 1 hour	**Simulation time**

Before the simulation results of E2DR are shown, we evaluate the threshold values. If max threshold is high, it means that workload is high and increases energy consumption; when max threshold is low, CPU is overloaded frequently and local server should change the state of CPU (hibernate or wakeup) constantly, which leads to increased energy consumption and response time. Energy parameter is calculated for each server in data center and found to be equal to the total amount of server's energy consumption in simulation process. Power consumption of servers which is used to calculate energy consumption is calculated by the workload of each physical machine. There are several models for energy consumption in CLOUDSIM simulator; HP and IBM models are used, as shown in Table 3 .Values in the table are energy consumption values based on the CPU utilization of servers.

Table3. Properties of servers in simulator

CPU utilization	**HPproliantG4**	**IBMX3250**
0%	86(w)	42.3(w)
10%	89.3(w)	46.7(w)
20%	92.6(w)	49.7(w)
30%	96(w)	55.4(w)
40%	99.5(w)	61.8(w)
50%	102(w)	69.3(w)
60%	106(w)	76.1(w)
70%	108(w)	87(w)
80%	112(w)	96.1(w)
90%	113(w)	106(w)
100%	117(w)	113(w)

Figure 2 displays energy consumption based on varying minimum thresholds. The best state for energy consumption is max threshold = 80% of CPU utilization and min threshold = 20% of CPU utilization. We evaluated and compare the performance of E2DR with that of JS2DR2 and FIRE as two data replication strategies.

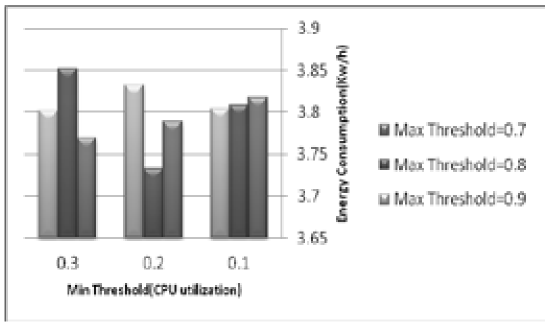

Fig2. Energy consumption of the proposal method based on different min and max thresholds

As shown in Figure 3, average response time of E2DR is better than that of other methods, because E2DR considers bandwidth, time stamp, and free memory of sites. Average response time is equal to the sum of average response time of jobs. Response time means the time between arrival time of job to grid and finish time of job. Response time changes according to size of job, its allocated resource, and job scheduling methods. Equation 4 calculates average response time in simulation.

$$Average\ Response\ Time = \frac{\sum_{i=1}^{job\_num} job(i).ArrivalTime - job(i).FinishTime}{job\_num} \quad (4)$$

Job_num is number of jobs.

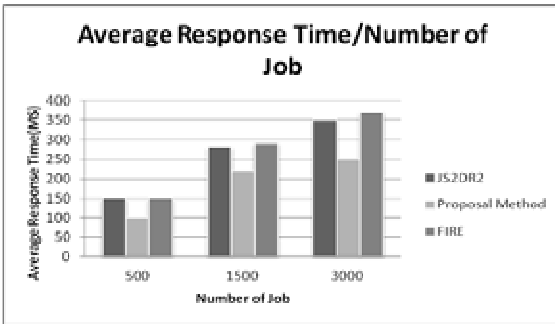

Figure 3. Average response time based on varying numbers of jobs

Figure 4 shows the energy consumption of E2DR was smaller than other strategies, because E2DR prevented energy and power consumption of hibernated sites and developed energy consumption of overloaded sites and E2DR uses energy parameter for selecting locations to replicate data file and defining thresholds to decrease energy consumption. Equation 5 shows energy consumption in the simulation. In different periods, power consumption is calculated by servers and placed in power variables in data center class during the simulation. Unit of energy consumption is watt that is divided by 1000*3600 and becomes KWH.

$$Energy(KWH) = \frac{Energy\ Consumption(W)}{3600 * 1000} \quad (5)$$

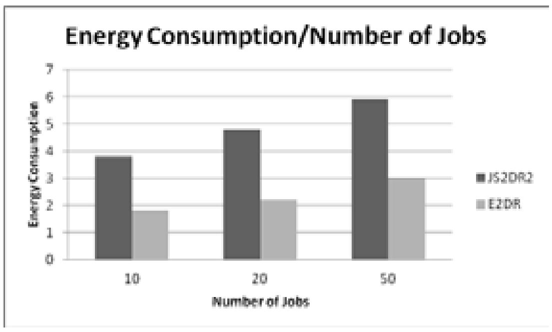

Figure 4. Energy consumption based on varying numbers of jobs

Efficient network usage is an important parameter and has been evaluated by most of the replication strategies. Means of network usage are storage memory usage. When storage memory is high, it is better to use and replicate copies are more applied. The following equation calculates efficient network usage. Figure 5 displays the efficient network usage based on the changing number of jobs for 3 algorithms. As shown, E2DR has better ENU, because E2DR transmit requires data files for the sites; then, the number of local access is increased and that of replica is decreased. Then, storage memory is efficiently used.

$$ENU = \frac{N\ Remote\ File\ Access + N\ File\ Replication}{N\ file\ Access} \quad (6)$$

Nremote file access= number of remote file access

Nfile replication= number of file replication

Nfile access= number of local and remote file access

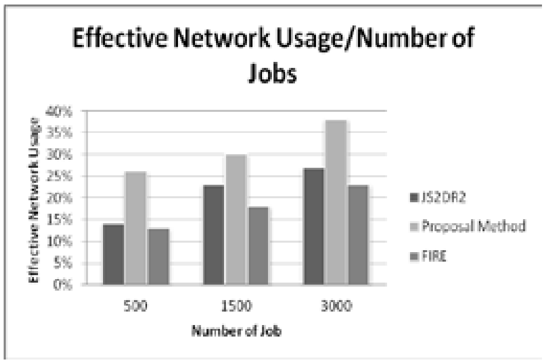

Figure5 . Efficient network usage based on varyig numbers of jobs

V:CALCULATION AND FUTURE WORKS

Data grids are highlighted in the development of grid technology, which can be treated as a suitable solution for high performance and data-intensive computing applications. Improvement of data access efficiency is a main issue, since the number and size of storage devices available in grids are limited while large size of data files is produced. In order to solve the problem, it is a good idea to create replicas of the files in appropriate locations. In this paper, a new replication strategy was proposed for the hierarchical structure network. The goal was to reduce effectively the response time and energy consumption. In this strategy, the replicas were stored in the best site, in which the file was accessed most, instead of storing files in many sites. Also, a new method was presented for reducing energy consumption. To evaluate the efficiency of the proposed replica strategy, cloud simulator CLOUDSIM was configured to test a real-world data grid. The simulation results showed it had less job execution time and energy consumption and more efficient network usage than other strategies. In future works, E2DR can be combined with a proper scheduling to improve performance. We also plan to investigate more replica replacement strategies to further improve the overall system performance. Replica selection can also be extended by considering additional parameters such as security. We can also define a dynamic threshold for energy consumption phase.

REFERENCE

[1] susan v. Vrbsky, ming lei, karl smith and jeff byrd ,data replication and power consumption in Data grids, 2nd ieee international conference on cloud computing technology and science,ieee(2010).

[2] Tarek Hamrouni,Sarra Slimani ,A critical survey of data grid replication strategies based on data mining techniques ,ICCS 2015 International Conference on Computational Sience volume 55, (2015), 2779 – 2788.

[3] somayeh abdi and somayeh mohamadi, two level job scheduling and data Replication in data grid, international journal of grid computing & applications (ijgca) vol.1, no.1(2010).

[4] ming tang, bu-sung lee, xueyan tang, chai-kiat yeo , The impact of data replication on Job scheduling performance in the data grid, future generation computer systems, volume 22, issue 3 (2006), 254-268.

[5] najme mansouri, gholam hosein dastghaibyfard, A dynamic replica management strategy in data grid, journal of network and computer applications 35 (2012),siencedirect,(2012), 1297–1303.

[6] JEMAL ABAWAJY ,DATA REPLICATION APPROACH WITH CONSISTENCY GUARANTEE FOR DATA GRID, IEEE TRANSACTION ON COMPUTERS , (2015), 1-17.

[7] Alireza Souri , Amir Masoud Rahmani, Survey for replica placement techniques in data grid environment , I.J.Modern Education and computer science,2014, (2014), 46-51.

[8] anton beloglazov, rajkumar buyya, young choon lee, albert zomaya,"a taxonomy and survey of Energy-efficient data centers And cloud computing systems,elsevier , (2011), 47-111.

[9] x. Fan, w.d. weber, l.a. barroso, power provisioning for a warehouse-sized computer, In: proceedings of the 34th annual international symposium on computer architecture (isca2007), acm new york, ny, usa, (2007), 13–23.

[10] m. Allalouf, y. Arbitman, m. Factor, r. I. Kat, k. Meth, and D. Naor. Storage modeling for power estimation. In systor '09: proceedings of systor 2009: the israeli experimental Systems conference, new york, ny, usa.Acm, (2009), 1-10.

[11] c. Patel, r. Sharma, c. Bash, and s. Graupner, Energy aware Grid: global workload placement based on energy efficiency. Technical report, hp laboratories,(2002).

[12] j. Torres, d. Carrera, k. Hogan, r. Gavalda, v. Beltran, and N. Poggi, Reducing wasted resources to help achieve green Data centers. In international symposium on parallel and Distributed processing (ipdps 2008)Ieee, (2008), 1-8.

[13] s. Srikantaiah, a. Kansal, and f. Zhao. Energy aware consolidation for cloud computing. In proceedings of hotpower '08 Workshop on power aware computing and systems(2008).

[14] a.-c. Orgerie and l. Laurent. When clouds become green: The green open cloud architecture. In international conference On parallel computing (parco 2009), lyon,

france(2009).

[15] Junaid Shuja, Kashif Bilal, Sajjad A. Madani, Mazliza Othman,Rajiv Ranjan, Pavan Balaji, and Samee U. Khan, Survey of Techniques and Architectures for Designing Energy-Efficient Data Centers, IEEE SYSTEMS JOURNAL, (2014), 1-13.

[16] c. Gunaratne, k. Christensen, and b. Nordman. Managing Energy consumption costs in desktop pcs and lan switches with Proxying, split tcp connections, and scaling of link speed. Int. J. Netw. Manag., 15(5), (2005), 297–310.

[17] d. C. Snowdon, s. Ruocco, and g. Heiser. Power management and dynamic voltage scaling: myths and facts. In Proceedings of the 2005 workshop on power aware real-time Computing, new jersey, usa(2005).

[18] h. Dietz and w. Dieter. Compiler and runtime support For predictive control of power and cooling. Parallel and Distributed processing symposium, international, (2006), 0-345.

[19] x. Fan, w.-d. Weber, and l. A. Barroso. Power provisioning For a warehouse-sized computer. In isca '07: proceedings Of the 34th annual international symposium on computer Architecture, new york, ny, usa.Acm (2007), 13–23.

[20] f. Bellosa, s. Kellner, m. Waitz, and a. Weissel. Event-driven Energy accounting for dynamic thermal management. In Proceedings of the workshop on compilers and operating Systems for low power (colp'03), (2003), 1–10.

[21] a. Merkel and f. Bellosa. Balancing power consumption in Multiprocessor systems. In sigops operating systems review, 40(4), (2006), 403–414.

[22] j. S. Chase, d. C. Anderson, p. N. Thakar, a. M. Vahdat, and R. P. Doyle. Managing energy and server resources in hosting Centers. In sosp '01: 18th acm symposium on operating Systems principles, new york, ny, usa,. Acm. , (2001), 103- 116.

[23] r. Jejurikar and r. Gupta. Energy aware task scheduling With task synchronization for embedded real-time systems. In Computer-aided design of integrated circuits and systems, Ieee transactions on. Ieee, (2006), 1024– 1037.

[24] g. Von laszewski, l. Wang, a. Younge, and x. He. Poweraware scheduling of virtual machines in dvfs-enabled clusters. In ieee international conference on cluster computing and Workshops (cluster '09),(2009), 1–10.

[25] Dejene Boru• Dzmitry Kliazovich• Fabrizio Granelli• Pascal Bouvry• Albert Y. Zomaya , "Energy-efficient data replication in cloud computing datacenters" , springer , (2015), 1-18.

[26] Jemal Abawajy , Data Replication Approach With Consistency Guarantee for Data Grid", IEEE TRANSACTIONS ON COMPUTERS DECEMBER 2014, (2014), 1-17.

[27] ali elghirani, riky subrata, albert y. Zomaya, and ali al mazari., performance enhancement Through hybrid replication and genetic algorithm co-scheduling in data grids, advanced Networks research group, school of information technologies, university of sydney, nsw Australia(2006).

[28] sang-min park, jai-hoon kim, young-bae ko: dynamicgrid replication strategy based on Internet hierarchy, book series lecture notes in computer science, grid and cooperative Omputing book,publisher springer, august 2005, volume 3033/2004, (2005), 838-846.

[29] k. Ranganathan and i. Foster, identifying dynamic replication strategies for a high Performance data grid. In proceedings of the international grid computing workshop, Denver, colorado, usa(2001).

[30] i. Foster, k. Ranganathan, design and evaluation of dynamic replication strategies for high Performance data grids, in: proceedings of international conference on computing in high Energy and nuclear physics, beijing, china, (2001).

[31] p. K. Suri, manpreet singh, js2dr2 : an effective two-level job scheduling algorithm and Two-phase dynamic replication strategy for data grid, 2009 international conference on advances in computing, control, and telecommunication technologies,ieee, (2009), 232-237.

[32] k. Sashi, a.s. thanamani, a new dynamic replication algorithm for European Data grid, in: proceedings of the third annual acm bangalore conference, 17(2010).

[33] Leyli mohammad khanli , ayaz isazadeh , tahmuras n. Shishavan, phfs: a dynamic replication method, to decrease access latency in the multi-tier Data grid, future generation computer systems 27,(2011), 233–244.

[34] m. Tang, b.-s. Lee, c.-k. Yeo, and x. Tang, dynamic Replication algorithms for the multi-tier data grid, future Generation computer systems, vol. 21, (2005), 775-790.

[35] Mohammad Shorfuzzaman, Peter Graham, Rasit Eskicioglu Distributed Popularity Based Replica Placement in Data Grid Environments, International Conference on Parallel and Distributed Computing, Applications and Technologies (PDCAT), 2010

[36] T.A.Abdurrab , FIRE: A File Reunion Based Data Replication Strategy for Data Grids.(2010).