DOI: http://dx.doi.org/10.26483/ijarcs.v8i8.4708

ISSN No. 0976-5697

Volume 8, No. 8, September-October 2017

International Journal of Advanced Research in Computer Science

RESEARCH PAPER

Available Online at www.ijarcs.info

OPTIMIZING THE STORAGE SPACE AND COST WITH RELIABILITY ASSURANCE BY REPLICA REDUCTION ON CLOUD STORAGE SYSTEM

Mrs. S.Annal Ezhil Selvi
Assistant Professor
Department of Computer Science
Bishop Heber College
Trichy,TamilNadu, India- 620017.

Dr. R. Anbuselvi
Associate Professor
Department of Computer Science
Bishop Heber College
Trichy,TamilNadu,India- 620017.

Abstract: Content Distribution Networks have been attracted a great deal of attraction in recent years on cloud computing. Replica placement problems (RPPs) as one of the key technologies in the Content Distribution Networks have been widely studied. The internet services are hosted by multiple geographically distributed datacenters. For the increasingly expanded utility of Cloud storage, the improvement of resources management and reduces the storage space, cost is complicated issues on data replication. So in order to reduce the data replication, this paper, proposed the concept of Reliability Assurance algorithm (RA). The RA is reducing the file replication from the server and improves the reliability of the server. The RA algorithm identified the data replication on un-accessing files. The unpredicted files called as replicated files, these files occupy a more space on cloud server. The low ranking prediction algorithm identified the unpredicted files on cloud server based on file accessing. Reduced data replication on cloud server it's allows optimizing the storage space and cost. Using the RA and Ranking algorithm combined to reduce the data replication and storage space.

Keywords: Cloud Computing, Data Replication, Popularity Degree, **Distribution** Networks, Reliability assurance algorithm.

1. INTRODUCTION

Cloud storage is a representation of data storage in which the digital records are stored in logical collection. The physical storage data stored on multiple servers (and frequent locations), manage by a hosting company [1 and 2]. These cloud storage sources are responsible for assuring the records available and accessible. Peoples and organizations buy or let storage capacity beginning the providers to store user. Today, popular Internet companies, such as Google, Yahoo, and Microsoft offers more services for millions of users every day. These services are hosted in datacenters that contain thousands of servers, as well as power delivery (and backup) and networking infrastructures. Because users demand high availability and low response times, each service is mirrored by multiple datacenters that are geographically distributed [3]. Each datacenter is supposed to serve the requests of the users that are closest (in terms of network latency) to them. If this datacenter becomes unavailable or unreachable, these requests are forwarded to a mirror datacenter.

A Cloud storage data replication service is a managed service in which stored or archived records is duplicated in real time over a storage area network. Further terms for this type of service consist of file replication, data replication, and remote storage replication. The appearance can also refer to a program or grouping that facilitates such duplication. Cloud Storage replication services provide an extra determine of redundancy that can be invaluable if the main storage backup system fails. The instant of that the cloud user can access to the replicated data to minimize downtime and its associated costs [4]. The services, if accurately implement, can clustering based make more efficient disaster recovery process by generating a replica copy of the entire backed-up files on a continuous basis [5].

Thus, in this paper to study the selection process, while fully characterizing the different parts of the potential locations for datacenters. First, propose a framework for selection that includes parameters representing all aspects of datacenter costs, response times, data consistency, and availability. The framework allows us to overcome the selection process as a non-linear Storage problem with response time, consistency, and availability as constraints [6 and 7].

Second, to propose the approaches for solving the storage problem efficiently by the process of rating of the file. The bunch of files are going to be hosed in the cloud servers and the users going to access the hosted file. If the user wants to access the particular file they accessed the cloud server and get that file. After the file is accessed by one user the rating of the particular file is increased. Using this rating process, it can easily identify the treading data in cloud servers. If the rating of the file is faded out the particular file or the data is considered as a waste data or unwanted data. For improving the cloud storage to remove that kind of unwanted data from the cloud servers.

The overall process is in Section 2, introduce the concept of replicated data, and then Section 3 focuses on how to improve the storage capacity of the server. Section 4 presents our experimental evaluation, and finally, Section 5 includes conclusions and future work.

2. RELATED WORKS

Data Replica placement is one of the significant issues for the high reliable distributed system. Dynamic Replica placement compact with how much divergence should be deployed and how to locate them. Replica placements extend into other crucial support of dynamic distributed system. This dynamic and adaptive replica placement policy must ensure routine over a long period of system operation [8].

In the last few years, very few proposals have looked at these problem but overlooked many important parameters. For instance, proposed a scheme that tries to find an optimal migration schedule for data in order to minimize the total migration time. By our algorithm, we improved the storage space and the processing time of the user request [5].
When a replica has to be created/migrated in a new location, it will not be available until all its content is copied from other existing replicas [10].
In this work, our research focuses on minimizing the Cloud storage consumption by minimizing data replication while meeting the data reliability requirement. Firstly, through analysis of existing studies, a generalized data reliability model for multiple replicas is proposed, in which the data reliability with variable disk failure rates is well investigated. Compared with much research that assumes a constant disk failure rate .It didn't focus on the replica placement, but rather on reducing the overhead of migrating from an original placement of replicas to the new one, which should take place right after the execution of the replica placement algorithm [9].

A. Background Process

The size of Cloud storage is expanding data impressive speed. It is estimated that by 2016 the data stored in the Cloud will reach 0.8 Zetta Bytes (i.e., 0.8*1021 Bytes or 800,000,000 TB), while even more data is "touched" by the Cloud within the data life cycle. Meanwhile, with the development of the Cloud computing paradigm, Cloud-based applications have put forward a higher demand for Cloud storage. While the requirement of data reliability should be met in the first place, data in the Cloud needs to be stored in a highly cost-effective manner [20 and 21].
In modern Clouds, data replication is the most commonly applied approach for providing data reliability assurance, which creates and stores multiple replicas of the data to reduce the probability of data loss. For example, storage systems such as Amazon S3, Google File System, and Hadoop Distributed File System all adopt similar data replication strategies that we call the conventional 3-replica strategy, in which three replicas, i.e. three data copies including the original data, are stored for all data [15].

B. Replication

Data is replicated down with multiple server environments that are able to handle with different data subsets. During replication data is obfuscated and deleted, depending on low ranking and security regulations. Low ranking or replicated data updates are adjusted automatically to cloud server the different data structures handled by environments. Content Distributed applications that are hosted in a Hybrid Cloud frequently need access to the same data from different application components [16]. If application components access the data is globally distributed, data access performance could be reduced significantly if data is only stored in one geographic location. Therefore, the data probably will have to be replicated. Due to laws and corporate regulations, some of these locations may possibly only handle a subset of the available data or data has to be obfuscated.

C. Storage

Data storage and data replication have received a lot of attention at the data management, distribution, and application level since the distribution of original data objects and their replicas are crucial to overall system performance, especially in the cloud environment where data are supposed to be protected and highly available in different data centers. The current literature concerns essentially the cloud storage problem in tandem with replication techniques to improve data availability [14]

D. Total cost

The solution approach reformulates the optimization problem to remove variables A_d and P_c^d, and use a linear version of max replicated [19]. The removal of A_d and P_c^d, requires stricter constraints on the placement of servers for each population center. We proportionally set the maximum number of servers at each datacenter [22], so that the sum of all maximum numbers of servers is equal to MaxA. With the maximum number of servers defined for each datacenter, we compute its ranking max using the proper costs per MW. Finally, we use the original function to compute the cost of the network of datacenters. Due to its simplifications and restrictiveness, this approach may produce higher total cost for a datacenter network than the other approaches.

$$TotalCost = \sum_{d \in D} AB_d . RDATA\_ind(d) + \sum_{d \in D} A_d . RANK\_max(d) + \sum_{c \in C} \sum_{d \in D} P_c^d . \left(ACC_{act} + OP_{act(d)} + OP\_util(Util, d) \right)$$

3. EXISTING SYSTEM

The existing work of this research was ranked the file which are stored in cloud storage based on their Popularity. Popularity is nothing but the number of accessing frequencies. The following figure shows process of existing work clearly [1].

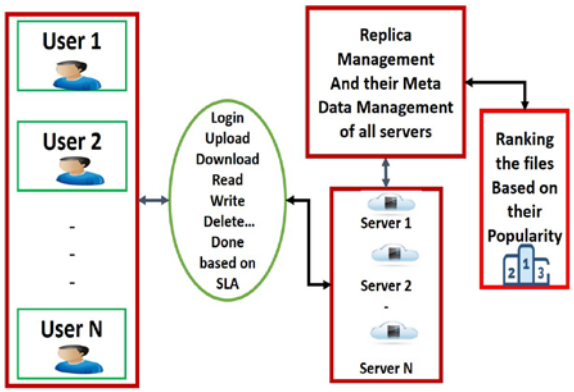

Figure 1 Existing System

Ranking Algorithm: in [1]

File ranks (k-Rank)

Input: N (S), N(F), i=0, q, k=0, ∅, t=currentTime

Output: Rank { q's result set}

1. While t >=0
2. If file upload
3. i=1
4. End if

5. While N(S)= no. of iteration i ∈ M(F)
6. If file access
7. Ø = i+1;
8. End if
9. End while
10. for each N(S) do
11. for each N(S) do
12. for each N(F) do
13. K=k+ Ø
14. End for
15. End for
16. Rank. insert (k)
17. End for
18. If Rank =q
19. Return Rank
20. End if
21. End while

Where S is server in cloud storage system N(S) is number of servers. M (F) is file's Meta data (log file). 'i' is the number access frequency of a particular file, q query result, k sum of access frequency of all servers. 'Ø' is access frequency of a particular file in all servers individually.
This algorithm works 24/7 in replication system and deals with Meta data. Initially the Ø value is 0 then its value changed based on the operation. If the file initiated to store on cloud storage the Ø value is 1. After that the Ø value may incremented based on the number of access frequency. Finally all Ø values are summed together from all servers for each file which is k value.

Table 1 File Ranking based on their Popularity

S.No	File Name	File Type	Frequencies of access per week						$\sum_{i=1}^{n} Si$
			Server 1	Server 2	Server 3	Server 4	-	Server N	Frequencies /week
1	Array_Java	docx	0	4	5	2	-	-	11
2	Tree_ds	mp4	0	0	0	0	-	-	0
3	HelloEnglish	mp3	4	8	2	10	-	-	24
4	CS_C	pdf	2	1	1	1	-	-	5
5	Img_001	jpeg	1	3	0	2	-	-	6

The above table (table 1) denotes the N number of user can avail the Cloud storage services. S_1, S_2…S_n are the N number of data Centre (Servers). The users can store their files (any type) F_n on any data Centre (Servers) of cloud storage based on their SLA. The value 0 denoted that file is not in that server. The ranking done based on the files popularities from the log file (Meta data) which means how many number of times that files are accessed. That frequencies are calculated individually. Finally, in order to rank that files all individual frequencies are needed to sum by using the following equation.

$$\sum_{i=1}^{n} S\,i$$

4. PROPOSED SYSTEM

Here, the research proposed the Reliability Assurance (RA) algorithm which is used to reduce the number of replications without affecting the reliability concerns. Based on the existing work's result the less frequently accessed which low ranking files replicas are going to be reduced. The following diagram describes the process of proposed system.

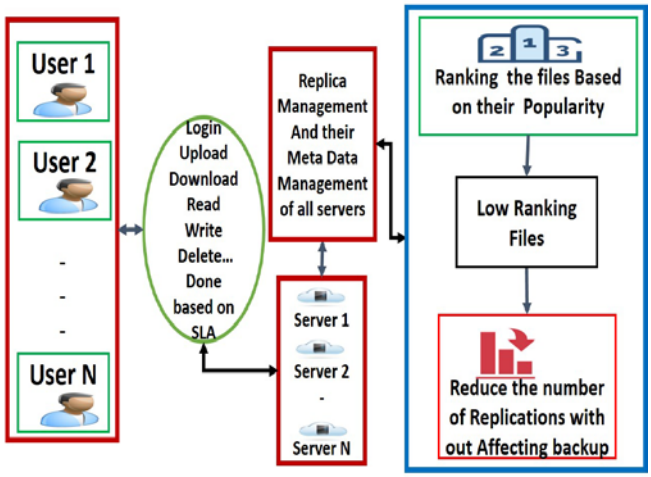

Figure 2 Proposed Replica management Architecture

5. RELIABILITY ASSURANCE (RA) ALGORITHM

Here show a dynamic data replication strategy to enhance the performance of software system. To identify the suitable file to replicate and to decide the respective number of replicas, we calculate popularity degree and replica factor. We use the fuzzy logic system to identify the system to place the replicas and we use the round robin method to place the replicas in the identified systems

RA Algorithm

1: t ←GET-TIME ()
2: A←PROC-HIST (H) //(Digitalize sequence)
3: for tier ←Client Tier -1 down to RootTier+1 do
4: A← AGGREGATE (A) //Ranking
5: for all record r Є A do
6: if r, numOfAccesses≥ thresholds [tier] then // Less than 2 server set
7: if EXIST-IN (r.fileID, r.nodeID) then
8: UPDATE-CTIME(r. fileID, r.nodeID,t)
9. GETREPLICATE (r.fileID, r.nodeID, t)
10: REMOVE (A, r)
11: end if
12: end if
13: end for
14: end for

In RA algorithm line 4 the access records in *A*are aggregated to the current tier. The details of function Aggregate will be analyzed average dataset level. After the aggregation, all records' *nodeID*s are in the current processing tier. For every record *r* in *A*, if the *r.numofaccess*exceeds the threshold for the current tier, it will be process further (line 6). If the replica of file *r.fileID* exist in the node of *r.nodeID*, then its creation time is updated to the current replication session time and *r* is removed from *A* (lines 7–9).

Or else, if node *r.nodeID* has sufficient space for file *r.fileID*, replicate the file to the node and eliminate record *r* from *A*(lines 10–12). After the inner **for** loop is done, the remaining records in *A* will be aggregated to the next higher tier in line 4, and the updated array *A* will be process again as stated above.

Ranking the best file is done by calculating the ratio between the unfrequented searched file from the server and the total number of mostly searched files. The frequent search is calculated and compared with the rest of the other server. The maximum frequent files are ranked in order. The files are ranking in frequently access for the user as the all the results are stored in the main cloud. The ranking is done on the server side, which may bring in huge computation and post processing overhead.

Moreover, the Ranking is known that a cloud computing system usually has a large number of servers. The RA based ranking process is user access the data in cloud server based on domain ranking. The minimum number of predicted files on each server calculated based accessing the files. Low ranking files are calculated in *rnk* based on prediction on cloud server. The minimum rank files are calculated *App* is not equal to number of iteration. So calculated the low ranking chooses the minimum selection of file first identified. The second to choose a low ranking of services in '*i*' that agrees with the pair wise preferences as much as possible ranking structure identified. Selected *app* and stored only two servers remaining replicated data's are removed from other server based on *threshold*. The ranking values can be calculated from the multiple servers and removed the files based on server memory.

6. RESULT AND DISCUSSION

As the primary objective of this paper is to increase the free available space of server and reduce storage cost without affecting the reliability concerns. For that, the following graphs and tables clearly explains the changes happened in the cloud storage by the proposed algorithm.

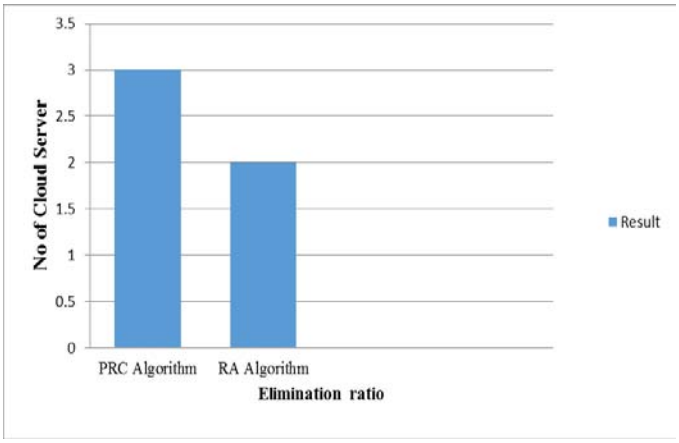

Figure (1)

The graph in Figure (1) shows the existing replication algorithm and proposed RA algorithm compare the replication ratio based on unpredicted file on Low ranking process. These graph existing replication technique is show on figure (1) take more memory on cloud server. The RA techniques based on low ranking process on unpredicted files show on figure (1) in graph less memory and better performance using RA.

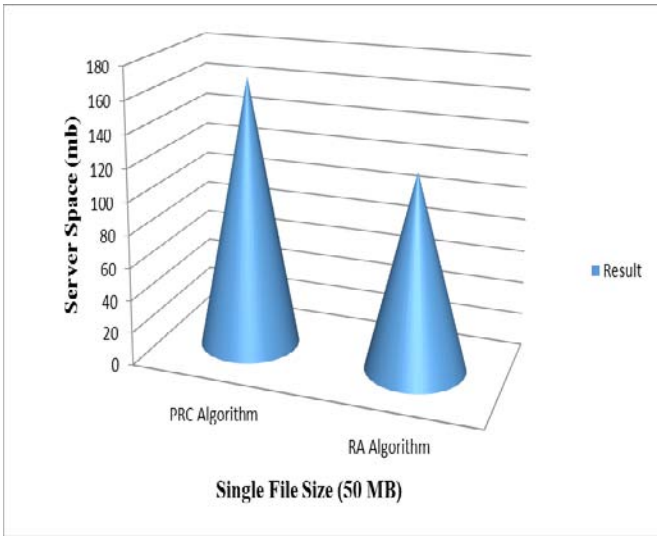

Figure (2)

The graph Fig (2) representation of cloud Server space analyzed. The X axis denotes the single file size and Y axis denotes the server space. In this graph implementing the algorithms are PRC and RA. The PRC Algorithm implementing in the existing process, it takes more memory (150 MB) on the cloud server. The RA algorithm classifies the unpredicted files (Low ranking files). The unpredicted files stored on any two servers based on server memory and the remaining servers its remove the files. The RA algorithm based on low ranking process on unpredicted files show on figure (2) less memory (100 MB) and better performance on cloud server.

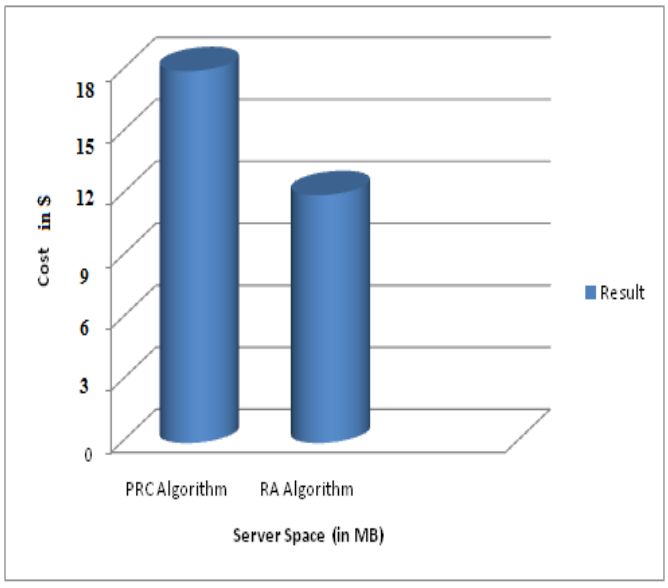

Figure (3)

The graph Fig (3) representation of Cost based on Server space. The X axis denotes the server space and Y axis denotes the Normalized Cost. In this graph implementing the algorithms are PRC and RA. The PRC Algorithm implementing in the existing process, it more replicas occur in cloud server takes high memory so the cost is very high (Single file storage & maintained cost is 18$). The RA algorithm classifies the unpredicted files on clouds server using Low ranking algorithm. The unpredicted files (Low ranking files) stored on any two servers and the remaining files its remove from the other server, so it's very

less memory and reduces the storage cost (Single file storage & maintained cost is 12$).

Table (2) Storage space comparison

S.no	Single Data File Size	Existing (3 Replica placement algorithm)	Proposed (RA algorithm)
1	50 MB	150 MB	100 MB
2	100 MB	300 MB	200 MB
3	300 MB	900 MB	600 MB
4	500 MB	1500 MB	1000 MB
5	1000 MB	3000 MB	2000 MB

The table (2) shows the space comparison of existing and proposed system. The existing system is taking more space. So the server performance is affected but the proposed system is less storage space.

Table (3) Storage cost comparison

S.no	Cost For Single Data File in $		Existing		Proposed	
			3-replica Placement Algorithm		2-replica Placement Algorithm	
	Storage Cost	Maintenance Cost	Storage Cost	Maintenance Cost	Storage Cost	Maintenance Cost
1	2 $	4 $	6 $	12 $	4 $	8 $
2	4 $	10 $	12 $	30 $	8 $	20 $
3	10 $	12 $	30 $	36 $	20 $	24 $
4	21 $	24 $	63 $	72 $	42 $	48 $
5	100 $	150 $	300 $	450 $	200 $	300 $

The table (3) shows the cost comparison on existing and proposed. The existing method taking more cost compare with proposed.

7. CONCLUSION

In this paper, investigated the data replication and the data reliability of cloud storage in different perceptions. From the result and discussion section the research concludes this proposed work obtain the primary objective. This paper identified some issues due to replication system which are more utilization of storage space, maximized cost cannot be minimized in existing system. This proposed system solved above mentioned problems. First, the optimally solve this replication problem on clouds server using a Reliability Assurance algorithm (RA) combined with ranking algorithm. It applies an innovative proactive replication files checking approach to ensure the data reliability on cloud server. The graphs and tables in above section shows the RA algorithm reduced the number of replica which files are identified by ranking algorithm from the cloud servers. However, this work achieves the optimized server performance, storage space and cost. And ultimately the RA algorithm increases the Cloud server's free available spaces. Finally, this research work need to concentrate the file availability concerns and response time in future.

REFERENCE

1. S.Annal Ezhil Selvi and Dr. R. Anbuselvi, "Ranking Algorithm Based on File's Accessing Frequency for Cloud Storage System", International Journal of Advanced Research Trends in Engineering and Technology (IJARTET) Vol. 4, Issue 9, Sep 2017.
2. Jonathan L. Krein, Lutz Prechelt "Multi-Site Joint Replication of a Design Patterns Experiment using Moderator Variables to Generalize across Contexts" IEEE Transactions On Software Engineering, Vol. X, No. X, Month 2015
3. Wenhao Li, Yun Yang, Dong Yuan, "Ensuring Cloud Data Reliability with Minimum Replication by Proactive Replica Checking", IEEE Trans. Computers 65(5): 1494-1506 (2016)
4. YaserMansouri, Adel NadjaranToosi, and Rajkumar Buyya "Cost Optimization for Dynamic Replication and Migration of Data in Cloud Data Centers" IEEE Transactions On Cloud Computing, Vol. pp, No. 99, January 2017
5. Runhui Li, Yuchong Hu, and Patrick P. C. Lee "Enabling Efficient and Reliable Transition from Replication to Erasure Coding for Clustered File Systems" IEEE Transactions On Parallel And Distributed Systems, Vol. pp, No. 99, March 2017.
6. Jerry Chou, Ting-Hsuan Lai "Exploiting Replication for Energy-Aware Scheduling in Disk Storage Systems" IEEE Transaction On Parallel And Distributed Systems, Volume 26,No 10,Oct 2015.
7. Guoxin Liu, HaiyingShen, Harrison Chandler "Selective Data replication for Online Social Networks with Distributed Datacenters" IEEE Transactions On Parallel And Distributed Systems, Vol. 24, No. 8, August 2016
8. Amina Mseddi, Mohammad Ali Salahuddin "On Optimizing Replica Migration in Distributed Cloud Storage Systems" 4th IEEE International Conference on Cloud Networking (IEEE CloudNet 2015)
9. HaiyingShen, Guoxin Liu, "Swarm Intelligence based File Replication and Consistency Maintenance in Structured P2P File Sharing Systems" IEEE Transactions On Computers, Vol. 64, No. 10, Oct 2015.
10. SameeUllah Khan, Ishfaq Ahmad "Comparison and analysis of ten static heuristics-based Internet data replication techniques" Parallel Distrib. Comput. 68 (2008)
11. Zheng Yan, Lifang Zhang, Wenxiu Ding, and QinghuaZheng, "Heterogeneous Data Storage Management with Deduplication in Cloud Computing" IEEE Transactions On Big Data, Vol. pp, No.99, May 2017
12. Jing Zhao,XuejunZhuo, "Contact Duration Aware Data Replication in DTNs with Licensed and Unlicensed Spectrum" IEEE Transactions On Mobile Computing, Vol. 15, No. 4, April 2016
13. Jenn-Wei Lin, Chien-Hung Chen "QoS-Aware Data Replication for Data Intensive Applications in Cloud Computing Systems" IEEE Transactions On Cloud Computing May 2014
14. Rodrigo N. Calheiros, Rajkumar Buyya "Meeting Deadlines of Scientific Workflows in Public Clouds with Tasks Replication" IEEE Transactions On Parallel And Distributed Systems, Vol. 25, No. 7, July 2014
15. Han Hu, Yonggang Wen, Tat-Seng Chua, Jian Huang, Wenwu Zhu and Xuelong Li "Joint Content Replication and Request Routing for Social Video Distribution over Cloud CDN: A Community Clustering Method" IEEE Transactions on Circuits and Systems for Video Technology, Vol. 26, No. 7, July 2016
16. Mazhar Ali, Kashif Bilal, Samee U. Khan, Bharadwaj Veeravalli, Keqin Li, and Albert Y. Zomaya "DROPS: Division and Replication of Data in the Cloud for Optimal Performance and Security" IEEE Transactions on Cloud Computing, Vol. PP, No. 99, February 2015

17. Sebastiano Peluso, Virginia Tech, Pedro Ruivo, Paolo Romano and Lu´ıs Rodrigues "GMU: Genuine Multiversion Update-Serializable Partial Data Replication" IEEE Transactions on Parallel and Distributed Systems, Vol. 27, No. 10, October 2016
18. Hiroki Nishiyama, Asato Takahashi, Nei Kato, Katsuya Nakahira, and Takatoshi Sugiyama "Dynamic Replication and Forwarding Control Based on Node Surroundings in Cooperative Delay-Tolerant Networks" IEEE Transactions on Parallel and Distributed Systems, Vol. 26, No. 10, October 2015
19. Cijo George, Sathish Vadhiyar "Fault Tolerance on Large Scale Systems using Adaptive Process Replication" IEEE Transactions on Computers, Vol. 64, No. 8, August 2015.
20. Abdullah Gharaibeh, Samer Al-Kiswany, and Matei Ripeanu "ThriftStore: Finessing Reliability Trade-Offs in Replicated Storage Systems" IEEE Transactions on parallel and Distributed Systems, Vol. 22, No. 6, June 2011
21. BaharehAlamiMilani, NimaJafariNavimipour, "A Comprehensive review of the data replication techniques in the cloud Environments: Major trends and future directions" Research Gate, February 2016.
22. Bakhta Meroufela *, Ghalem Belalemb "Managing Data Replication and Placement Based on Availability" 2013 AASRI Conference on Parallel and Distributed Computing Systems

Mrs. S. Annal Ezhilselvi received her M.C.A and M.Phil degree in Computer Science from Bharathidasan University, Trichy, Tamilnadu,India in 2006 and 2011 respectively. And she cleared SET and NET exams in 2016. She was published 1 book and now she is working as an Assistant Professor in Bishop Heber College (Autonomous),Trichy, Tamilnadu, India. She has 9 years of teaching experience. Her research interest is Cloud Storage. She is now pursuing her PhD under the guidance of Dr. R. Anbuselvi.

Dr. R. Anbuselvi received her PhD degree in Computer Science from Mother Teresa University, Kodaikanal, Tamilnadu,India in 2013. She is now working as a Assistant Professor in Bishop Heber College (Autonomous),Trichy, Tamilnadu, India. She has 19 years of teaching experience. Her research interests are Artificial Intelligence, Cloud Computing and Data Mining. She is now guiding more than 8 PhD research scholars.