International Journal of Applied Engineering Research
ISSN 0973-4562 Volume 10, Number 8 (2015) pp. 19559-19574

http://www.ripublication.com

Dynamic Server Replication Using Sophisticated Precedence Based Optimization Algorithm

[1]Chenthil Kumaran, N,[2]Dr.Y.Jacob Vetha Raj, [3]Dr.P.Arockia Jansi Rani
[1]Assistant Professor Department of computer Science and Engineering
Lourdes Mount College of Engineering and Technology,
Nattalam, Kanyakumari-629165, India.
[2]Assistant Professor, Department of computer Science and Engineering,
Nesamony Memorial Christian College, Marthandam-629 165, India.
[3]Assistant.Professor, Department of computer Science and Engineering,
ManonmaniamSundaranar University, Thirunelveli, India.
E-mail:[1]chenthil696@gmail.com

Abstract

Server replication is an approach that often used to ameliorate the scalability of the service and an efficient factor in the utilization of replicated servers. It has the ability to direct client request to the best servers andcomplete according to some optimality criteria. Moreover, it amends the performance and send data more proximity to the users. The conventional method for server replication occur through Genetic Replication Algorithm. GRA is much slower and the performance of this algorithm is most horrible. In this proposed research, an effective dynamic server replication has proposed with the sophisticated precedence optimization algorithm. In this system, the nodes are randomly grouped towards the servers and the servers are then filtered. The servers with higher priority send to the optimization process, but some optimization factors are used to optimize the best servers and these optimized servers are replicated dynamically. As a result the proposed system has been validated with the help of the optimization algorithm. Hence, the experimental results demonstrated that the performance of the system improved significantly.

Keywords: Server replication, Sophisticated Precedence,Optimization algorithm, Grouping, Prioritizing, Priority queue.

Introduction

A distributed system is a collection of independent computers that appears to its users as a single coherent system and it connects the users as well as IT resources in a

transparent, open, cost-effective, reliable and scalable way [1]. The network impact of emerging large-scale distributed systems, where traffic flows and what it costs must encompass users' behavior, the traffic they generate and the topology over which that traffic flows [2]. Distributed systems are complex, being usually composed of several subsystems running in parallel [3]. The computers in the distributed system do not share a memory instead they pass messages asynchronously or synchronously between them. This is a type of segmented or parallel computing that runs on a heterogeneous system [4]. To improve the reliability, fault tolerance and availability in distributed systems, Replication was used [5].

The main idea in replication is to keep several copies or replicas of the same resources on different servers [6] [7]. Two replication strategies have been used in distributed systems: Active and Passive replication [8] [9]. In active replication, each client request is processed by all the servers. This requires that the process hosted by the servers is deterministic [10] [11]. Deterministic means that, given the same initial state and a request sequence, all processes will produce the same response sequence and end up in the same final state.In order to make all the servers receive the same sequence of operations, an atomic broadcast protocol must be used [12]. An atomic broadcast protocol guarantees that either all the servers receive a message or none, plus that they all receive messages in the same order. In passive replication there is only one server (called primary) that processes client requests [13]. After processing a request, the primary server updates the state of other (backup) servers and sends back the response to the client [14]. If the primary server fails, one of the backup servers takes its place.

In the data network of the servicing of client file the access can be switched over from the source file server to the destination file server [15] and completed the request of the system administrator at any time during a replication process. Initiating the replication process creates an empty local repository and then populates it with data mirroring in the source repository of cloud servers. The services that are provided by the cloud servers can be accessed from anywhere and data flows from one place to another [16]. Since data are moving via network, there are chances of data loss. So we need to keep multiple copies of data [17]. The multiple copies of data should be stored in different locations by the replication process [18], so that data can be easily recovered if a copy at one location is lost or unavailable due to large-scale data grid. In a large-scale data grid, replication provides a suitable solution for managing data files also the replication Server will maintain copies of the data [19] [20].

A Replication Server could be used to manage one or more databases and it is adequate for some replication systems for servicing the client requests [21]. While servicing the client request, the client file can be switched over automatically from the source file server to the destination file server to provide the clients with access to a destination file server which enhanced storage, performance and additional service capabilities [22]. When any detection of failure in the source file server for servicing the file access requests from the client's and when there has been an ongoing replication of the client's file systems from the source file server to the destination file server then servicing of client file access can also be switched over automatically from the source file server to the destination file server [23].

The motivation of this paper is to direct the clients to the best server by using optimization criteria of replicated server. Optimization is based on the performance as supported by the clients (e.g. Response time) or by the application provider (e.g. Global throughput) and second resource usage (e.g. Processor occupation).Replication Server uses data packets to share information these data packets are transmitted and returned back to its source, the total time for the round trip is known as latency. To improve the network performance different techniques used in previous years. In this paper, an algorithm proposed for optimizing the latency in replicated server and this algorithm proposed, because of its robustness and service rates. It can identify the best server by its bandwidth, power, and successful completion of the task. So that, the workload completion and response time become faster.

Related Work

Data replication is used to replicate data belonging to multiple nodes from one server to another server, so that if the main source server to which data is being backed-up goes down, the clients can recover their data from the replication site. A computer program product for replicating objects from a source storage managed by a source server to a target storage managed by a target server.

Qi Zhang *et.al* [24] this paper presented a survey of cloud computing, highlighting its key concepts, architectural principles, state-of-the-art implementation as well as research challenges. The paperhad focused to provide a better understanding of the design challenges of cloud computing and identify important research directions in this increasingly important area.

G. Ananthanarayanan*et.al* [25] proposed the technique to improve data locality in cluster computing systems using adaptive replication with low system overhead. It was suggested in a recent study that the benefit of data locality might disappear as bandwidth in datacenters increases.

Diwaker Gupta *et.al* [26] these paper proposed a large-scale network services can consist of tens of thousands of machines running thousands of unique software configurations spread across hundreds of physical networks. Testing such services for complex performance problems and configuration errors remains a difficult problem. Existing testing techniques, such as simulation or running smaller instances of a service, have limitations in predicting overall service behavior at such scales.

Matthew *et.al* [27] presented the work for a storage-management server, such as Tivoli® Storage Manager (TSM) stores data objects in one or more storage pools and uses a database for tracking metadata about the stored objects. The storage management server may replicate the data objects to a remote location for disaster recovery purposes. Some of the methods used to transfer data to a remote location include physically transporting tapes containing copies of the data from the source site to the disaster recovery site, electronically transmitting the data (TSM export/import) or using hardware replication of the source site disk storage to create a mirror of the data.

BrankoMilosavljevic*et.al* [28] presented their work in a software package for transparent communication of client and server side of the library circulation system. Database operations are executed through this package. The package can execute the

operations under different protocols, which enables the work of the library circulation system in the intranet or internet environments.

Dongmei Zhou*et.al* [29] presented their work for the invention generally relates to a system and method for network file system server replication, and in particular, to using reverse path lookup to map file handles that represent file objects in a network file system with full path names associated with the file objects, detect and distinguish hard links between different file objects that have the same identifiers with different parents or different names within the network file system and cache results from mapping the file handles with the full path names and distinguishing the hard links to enable replicating changes to the network file system.

Dinuzzo*et.al* [30] presented an architecture to perform information fusion from multiple datasets while preserving privacy of individual data. The role of the server is to collect data in real time from the clients and codify the information in a common database. Such information can be used by all the clients to solve their individual learning task, so that each client can exploit the information content of all the datasets without actually having access to private data of others.

Sophisticated Precedence Based Optimization

Replication is a technology for copying and distributing data and database objects from one database to another and then synchronizing between databases, it involves sharing information so as to ensure consistency between redundant resources to improve reliability, fault-tolerance, or accessibility,it is an approach in which several copies of some data are stored at various sites. By using replication, distribute data to different locations and to remote or mobile users over local and wide area networks. Server replication is an approach that frequently used to upgrade the scalability of the service. One of the efficient factors in the efficient utilization of replicated servers is the ability to direct client request to the best servers according to some optimality criteria. Server replication moves transactions (insert, updates and deletes) from a source data server to destination data servers. The important factor in the utilization of replicated server is the ability to direct clients to the best server, according to some optimization criteria. The main mechanism used in the server replication is for reducing user waiting time, reduction in latency, increasing data availability, reducing the communication cost, integrating data from multiple sites and minimizing bandwidth consumption by offering the user different replicas with a coherent state of the same service.

The proposed sophisticated precedence based optimization algorithm comprises of two phases, selection process and optimization process. In selection process phase clustering and prioritization process are takes place. In clustering method the nodes are clustered into single hop distance and in prioritization phase filtering the servers based on their previous task completion time can be done. In the prioritization process priority servers are preferred based on two priority queues, such as high priority queue and low priority queue. The higher priority queue encloses the server, which completes the task rapidly where the lower priority queue encloses the server which have unfinished task. The servers in the highest priority queue are to be optimized to pick out the best server on the basis of some factors like storage capacity, bandwidth,

power and past history of the server. The prohibitive preference algorithm is used to pick the optimum server, subsequently the replication process taking place dynamically.

Fig. 1 contains the collection of nodes indicates $\{L_1, L_2 \ldots L_m\}$ where m is the number of nodes retrieving the servers denotes $\{K_1, K_2 \ldots K_n\}$ where n is the number of servers. The server can be filtered based on their previous task completion time, by means of the task completion time the server can be filtered and kept in higher priority queue and a lower priority queue. The server, which entails a smaller amount time to complete the unit task is in the higher priority queue and the server which have fragmentary task is in the lower priority queue. The servers inhigher priority queue are employed in the optimization process based on the factors such as storage capacity $\hat{c}$, bandwidth $\hat{w}$, power $\hat{e}$ and past history $\hat{h}$ of server, subsequently the best server will be optimized by means of a prohibitive preference algorithm, consequently the best server will be replicated.

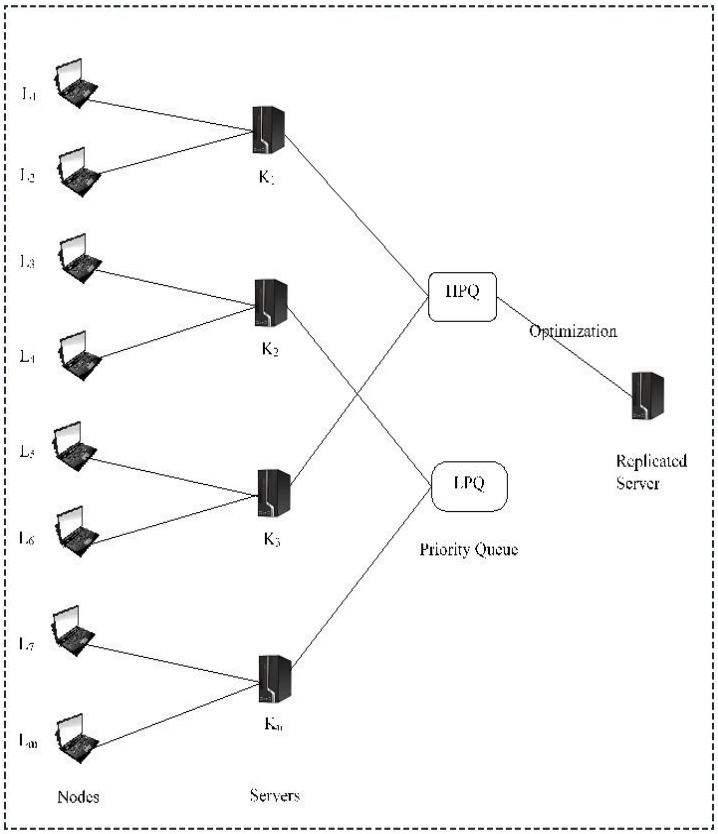

Figure1: Network Diagram forSophisticated Precedence Based Optimization

Selection Process

Selection process phase in sophisticated precedence based optimization algorithm shelters clustering and prioritizing the servers. In clustering process the servers are grouped into single hop distance and in prioritization process the servers are filtered

based on their preceding task completion time and threshold values can be deliberate in this ranking process.

Clustering of Nodes to the Server

Clustering the servers are explicitly intended for high availability solutions, scalable solutions and for easier maintenance. The cluster is two or more interconnected nodes that harvest a solution to afford high availability, high scalability or both, the clustering technique is used for high performance, large capacity and the incremental growth.A node groups composed of group of client system accessing the numerous server which works in an organized manner to provide enhanced scalability and reliability and in node/server computing, nodes request for server services and servers run their own services. When the server uniformly scattered to the node groups, optimal placement can attain good load balancing in directing the nodes and the efficient server can achieve high throughput to the node groups.

The server can group the nodes in single hop distance, in single hop networks, information can be transferred from one node to another only when one of the transmitters of the source node and one of the receivers of the destination node are rehabilitated to the identical wavelength. Information transmitted from one node to another without going throughintermediate nodes.

$$C = \frac{\sum_{L_m, K_n \in L}^{n} distance(L_m, K_n)}{\sum_{L_m, K_n \in L}^{n} hopcount(L_m, K_n)} \tag{1}$$

where,

L_m	–	Number of Nodes
K_n	–	Number of Servers
Distance (L_m, K_n)	–	Distance from Node to Server
Hop count (L_m, K_n)	–	Minimum number of hops from Node to Server
C	–	Mean cluster value

Filtering the Servers

The filtering is the process of finding the best among the group, in our method the filtering is used to find the best server among the group of servers. Once the clustering process is concluded, the servers are filtered and ranked dynamically through its estimated time to process a task of unit size. The threshold value can be considered in the process. The threshold value is,

$$T_h = \frac{H \times (1 - \alpha)}{T} \tag{2}$$

where, T_h– Threshold value

H	–	Average processing time
T	–	Total process in the server
α	–	Value from 0 to 1

The server, which has a smaller amount oftime desires to process the unit task, then that server is the higher ranked server at that moment threshold values can be deliberated for each server, this value can be considered based on the previous task completion time of the server. The server, which possess threshold value greater than the particular value send to the High priority Queue (HPQ) and the server which possess low values send to the Low Priority Queue (LPQ), that is represented in Fig. 2. The higher priority queue encompasses tasks that are prepared to be assigned and the servers with high priority are designated for the optimization process. The low priority queue contains tasks that have been assigned to one or more servers, but not finished and the servers with lower priority are worst server.

$$T = \begin{cases} HPQ, C \geq T_h \\ LPQ, C \leq T_h \end{cases} \tag{3}$$

where,

HPQ – High Priority Queue
LPQ – Low Priority Queue
T_h–Threshold Value

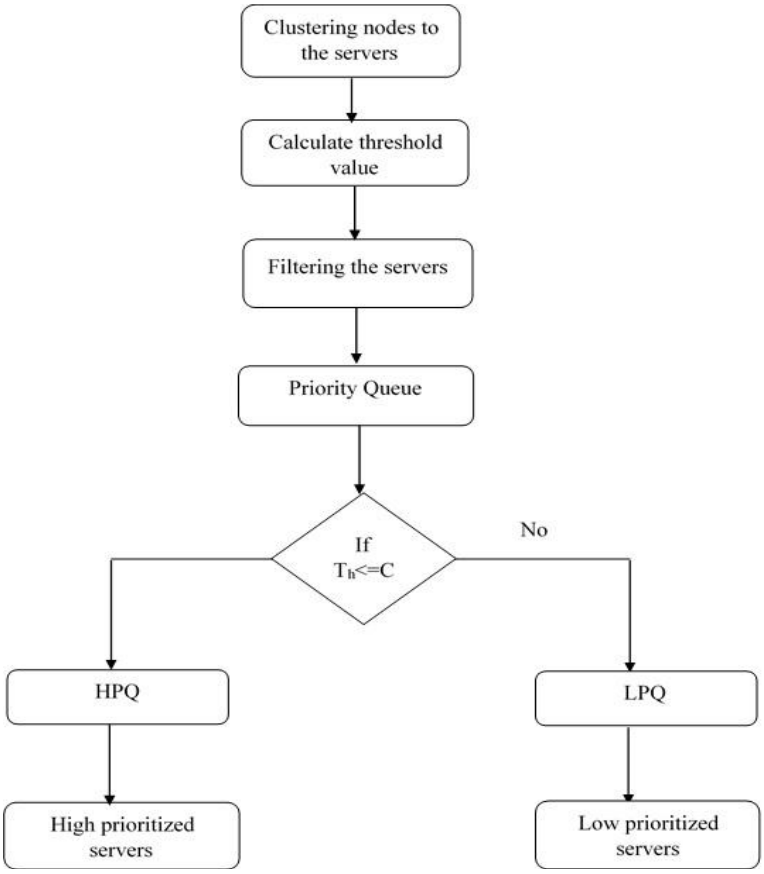

Figure2: Flow Diagram for Selection Process

Prohibitive Preference Algorithm

The servers in the highest priority queue are forward to the optimization process and in optimization phase the prohibitive preference algorithm is utilized to optimize the best server. From the optimized servers, the best server can replicate dynamically, the foremost benefit of prohibitive preference algorithm is the ease of its application since it has less parameters that essential to be modified before the inception of the search especially when associated with other techniques.

In this prohibitive preference algorithm entire number of filtered servers are used i.e., the high priority servers are send to the optimization process and estimate its factor value fitness function using the factors. Then initially populate the servers (generate new servers randomly) and compute factor values for new servers, based upon these factor values specific servers are prohibited and servers with high value are preferred as the best servers. Calculate the factor value randomly to the best servers, in this continuous process the best server will be elected.

The algorithm in Fig. 3 signifies the prohibitive preference algorithm in sophisticated precedence based optimization. In this algorithm there are total number of servers K_n are clustered and filtered then estimate the threshold value for the filtered servers. The value better than this threshold are send to the high priority queue and the further servers are send to the low priority queue. The servers in high priority queue are only to be initially populated and acquire the factor values for each servers. Assess fitness for storage capacity, bandwidth, power and past history for the servers formerly picked the server among n randomly and relate the fitness values for the servers then replace the low fitness server. This process can be finished when all the servers fitness are to be compared.At that time the worst servers are abolished and the new servers are to be built then preserve the best server i.e. the servers with high fitness are to be selected as the best server.

```
Begin
    Total number servers Kn
    Clustering the servers (C)
    Filtered servers (T)
    Calculate Threshold value (Th)
 if (Th<=C)
     T = HPQ
 else
     Send to LPQ
 else if
    Generate initial population of filtered servers Ti (i=1, 2, 3……, n)
 while (l < Maxgeneration) or (Stop criterion)
       Get a server randomly by factor values
       Evaluate its fitness (fi) for storage capacity, band width, and power, past history
       Choose a server among n (say, j) randomly
   if (fi > fj),
        Replace j by the new server;
   end if
      A function (w) of worse servers are abandoned and new ones are built;
      Keep the best server (or servers with high fitness);
      Rank the servers and find the current best
 end while
    Post process results and visualization
end
```

Figure3: Prohibitive Preference Algorithm

In optimization algorithm, arbitrarily elected servers are then evaluated by means of subsequent factors.

- Storage Capacity (ĉ)
- Bandwidth (ŵ)
- Power (ê)
- Past history (ĥ)

Storage capacity of the server is the supreme data's can be kept in the server, it measures how much data a server system may encompass. Bandwidth describes the maximum data transfer rateof servers, it measures how much data can be sent over a specific connection in a given amount of time and is used to describe server speeds, it does not measure how fast bits of data move from one location to another. The priority of the task have to be finished by a fast server in terms of its power and available bandwidth.The past history of the server defines the performance of the server which done to the nodes before. Based on the above factors the factor value can be calculated as,

$$f_i = \frac{(\hat{c} \times \hat{w} \times \hat{e} \times \hat{h})}{\vartheta} \quad (4)$$

where,

f_i – Factor value
$\hat{c}$ – Capacity
$\hat{w}$ – Bandwidth
$\hat{e}$ – Power
$\hat{h}$ – Past history
ϑ – Mean value of the factor

Based on the above factor value the server can be selected and the least factor value can be calculated as follows,

$$B_s = min\{f_1, f_2, f_3, \ldots, f_n\} \quad (5)$$

where, B_s – Best server
$f_1, f_2, f_3, \ldots, f_n$ – Factor value

The least factor value can be selected as the best server and the replication process can be done to this server.

Hence the sophisticated precedence based optimization was completed in two phases. They are selection process and optimization process. These two phase's filter the total servers finally present in the K_n, then the optimization phase filters the servers and select the best server in the final phase. By this innovative server replication strategy the process completion time should be reduced by allocating resources in the best server. In this sophisticated precedence based optimization the best server B_s will be selected and that server can be replicated dynamically.

Experimental Results

The proposed server replication system is implemented in JAVA platform and it is evaluated using proposed based optimization algorithm. It has been used comprehensively in estimating the performance of the optimization algorithm. Comparing the performance of our proposed algorithm with Genetic Replication Algorithm and Cuckoo search optimization algorithm obtained by exhaustive search, would have been the best way to illustrate the qualities of our approach. The filtering strategies are given in the Table 1.

Table 1: Filtering Strategies

Server Replication Strategies	**Number of servers**
Total no of Servers (K_n)	50
Number of Fitness Servers (T_h)	25
Number of High Priority Servers (T)	10
Number of Optimized Server (B_s)	1

Comparative Analysis

The performance of the proposed system is evaluated by comparing its result with Genetic Replication Algorithm and Cuckoo search optimization algorithm. The Fig. 7 represents the comparison graph of the performance metric results of the proposed technique with GRA and CS based server replication system. From the performance graph we came to know that the performance of the proposed technique is higher than the GRA and CS based technique. The performance of the proposed system is evaluated by the performance metrics such as Execution time, Replicas and Network Transfer Cost (NTC).

Replicas

The Replica is an exact reproduction executed by the original or a copy or reproduction, especially one on a scale smaller than the original. The Fig. 4 represents that our proposed method overtakes GRA and CS in terms of solution quality in all cases, and GRA and CS produces the lowest quality replication schemes when compared to our proposed technique.

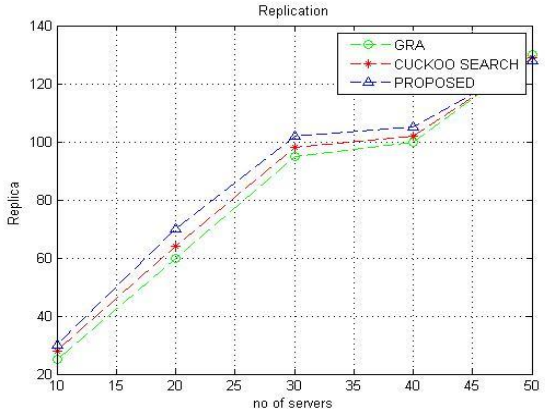

Figure4: Replicas for Proposed, GRA and CS

Execution Time

The time in which a single instruction is executed, it makes up the last half of the instruction cycle. The Fig. 5 denotes the execution time of our proposed method is much faster when compared to GRA and CS. The observation should be attributed to the fact that, our proposed method explores and balances these diverse effects better than the GRA and CS.

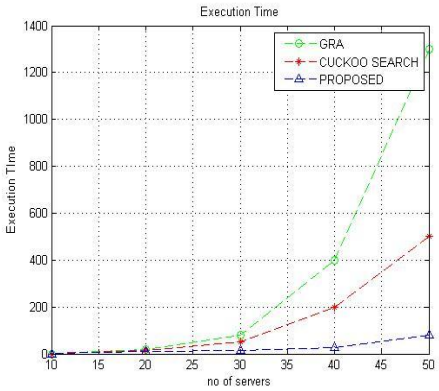

Figure5: Execution Time for Proposed, GRA and CS

NTC Savings

The Fig. 6 illustrates graph denotes Network Transfer Cost savings of GRA and CS is lower than our proposed optimization method.

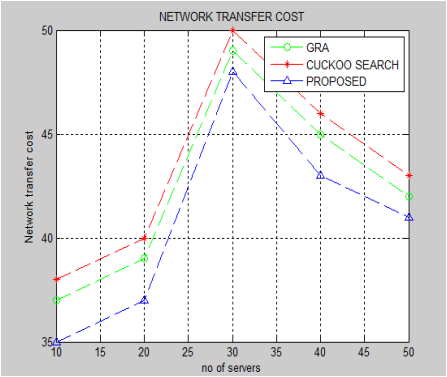

Figure6: NTC Savings for Proposed, GRA and CS

The aforesaid graphs represents quality and performance of our proposed method is higher than the Genetic Replication Algorithm and Cuckoo Search optimization Algorithm. The Fig. 7 shows that the comparison graph of our proposed method with GRA and CS.

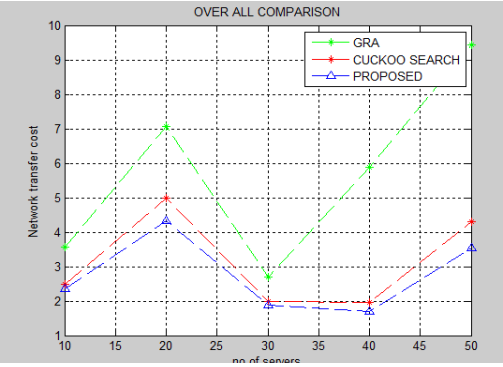

Figure7: Comparison Evaluation Graph of Proposed, GRA and CS

Result and Conclusion

In this paper, we addressed the server replication problem which is applicable to large distributed systems and distributed databases. We proposed a new optimization algorithm to solve the problem. This paper formulated dynamic server replication, for the above mentioned problem. The proposed server replication system uses clustering, filtering and optimization methods to replicate the server. The proposed server replication system is implemented in JAVA platform. In conclusion, the above experimental results demonstrate that the proposed dynamic server replication strategy effectively increases the server availability and reduces user waiting time by a very small number of replicas. The experimental analysis illustrated that our proposed optimization algorithm constantly outperforms Genetic Replication Algorithm and Cuckoo search optimization algorithm in terms of solution quality and execution time of our proposed method is fast when compared to the Genetic Replication Algorithm and Cuckoo Search optimization algorithm. The result of the comparison clearly explains that our proposed server replication method is better thanboth the GRA and CS.

Acknowledgment

The author thanks late Dr. T. Jebarajan, who had enraged in this work. Without his precious support it would not be possible to conduct this research.

References

[1] Garcia, M, Llewellyn-Jones, D, Ortin. F andMerabti. M, “Applying dynamic separation of aspects to distributed systems security: A case study, Software”, IET, vol.6, no.3, pp. 231 –248, 2012.

[2] John S. Otto,Mario A. Sánchez,David R. Choffnes,Fabián E. Bustamante,GeorgosSiganos, "On blind mice and the elephant: understanding the network impact of a large distributed system", Proceedings of the ACM SIGCOMM,vol.41,no.4,pp.110-121,2011.

[3] Leungwattanakit. W,Artho. C, Hagiya. M and Tanabe. Y, "Modular Software Model Checking for Distributed Systems", Software Engineering, IEEE Transactions, vol.40,no.5,pp. 483 –501, 2014.

[4] "Distributed systems and cloud computing" from http://www.resumegrace.appspot.com/pdfs/Distributed Systems and Cloud Computing.pdf.

[5] Wu Weijie and Lui John C.S, "Exploring the Optimal Replication Strategy in P2P-VoD Systems: Characterization and Evaluation", Parallel and Distributed Systems, IEEE Transactions, vol.23,no.8,pp. 1492 –1503, 2012.

[6] "Replication in Distributed File Systems" from http://crystal.uta.edu/~kumar/cse6306/papers/Smita_RepDFS.pdf.

[7] AmjadMahmood, "Replicating web contents using a hybrid particle swarm optimization", Journal of Information Processing & Management, vol.46,no.2,pp. 170 –179, 2010.

[8] "Active and Passive Replication in Distributed Systems" from http://jaksa.wordpress.com/2009/05/01/active-and-passive-replication-in-distributed-systems/

[9] S. U. Khan, A. A. Maciejewski, and H. J. Siegel, "Robust CDN Replica Placement Techniques", 23rd IEEE International Parallel and Distributed Processing Symposium (IPDPS), 2009.

[10] Sebastian Burckhardt, Alexey Gotsman, Hongseok Yang, and MarekZawirski., "Replicated Data Types:Specification, Verification, Optimality",ACM SIGPLAN,2014.

[11] S. U. Khan, and I. Ahmad, "Comparison and Analysis of Ten Static Heuristics-based Internet Data Replication Techniques", Journal of Parallel and Distributed Computing, vol.68, no.2, pp. 113-136, 2008.

[12] Denis McKeown, Jessica Holt, Jean-Francois Delvenne, Amy Smith, Benjamin Griffiths,"Active versus passive maintenance of visual nonverbal memory", Springer US, 2014.

[13] Da-Wei Sun, Gui-Ran Chang, Shang Gao, Li-Zhong Jin, Xing-Wei Wang, "Modeling a Dynamic Data Replication Strategy to Increase System Availability in Cloud Computing Environments", Journal of Computer Science and Technology, vol.27, no.2, pp. 256-272, 2012.

[14] Divyakant, Agrawal, Amr El Abbadi, Shyam Antony, Sudipto Das, "Data Management Challenges in Cloud Computing Infrastructures", Databases in Networked Information Systems, vol.5999, pp. 1-10, 2010.

[15] AC Raman, "Big data computing and reference architecture", Igi global, 2014.

[16] Nirmal Singh, Sarbjeet Singh, "Design and Performance Analysis of File Replication Strategy on Distributed File System Using GridSim",

Advances in Intelligent Systems and Computing, vol.248,pp. 629-637, 2014.

[17] DhananjayaGupt, Mrs.AnjuBala, "Autonomic Data Replication in Cloud Environment", International Journal of Electronics and Computer Science Engineering, vol.2, no.2, pp.459-464, 2013.

[18] K. Sashi, Antony SelvadossThanamani, "Dynamic replication in a data grid using a Modified BHR Region Based Algorithm", Future Generation Computer Systems, vol. 27, no. 2, pp. 202–210, 2011.

[19] Thampi, Sabu M,Sekaran, Chandra K, "Differential Strategies for Replicating Web Documents", Network Protocols & Algorithms, vol. 2 no.1, pp.93-131, 2010.

[20] Tao Wu,Starobinski, D,"A Comparative Analysis of Server Selection in Content Replication Networks", Networking, IEEE/ACM Transactions, vol. 16, no.6, pp. 1461 – 1474, 2008.

[21] Samir Khuller, BarnaSaha, Kanthi K. Sarpatwar, "New Approximation Results for Resource Replication Problems", Springer Berlin Heidelberg, vol. 7408,pp. 218-230, 2012.

[22] Ramya Mohan, "Network Analysis and Application Control Software based on Client-Server Architecture", International Journal of computer application, vol.68, no.12, pp.34-39, 2013.

[23] Gaston Keller, HananLutfiyya, "Dynamic Resource Management in Virtualized Environments through Virtual Server Relocation", International Journal on Advances in Software, vol. 3, no.3, 2010.

[24] Huah-Yong Chan, Boon-YaikOoi,Yu-N. Cheah, "Dynamic service placement and replication framework to enhance service availability using team formation algorithm", Journal of Systems and Software, vol. 85, no. 9, pp. 2048–2062, 2012.

[25] Qi Zhang, Lu Cheng, RaoufBoutaba, "Cloud computing: state-of-the-art and research challenges", Journal of Internet Services and Applications, vol.1, no. 1, pp. 7-18, 2010.

[26] G. Ananthanarayanan, A. Ghodsi, S. Shenker, and I. Stoica, "Disklocality in datacenter computing considered irrelevant," in Proc. USENIX on Hot Topics in Operating Syst. (HotOS), 2011.

[27] Diwaker Gupta, KashiVenkateshVishwanath, Marvin McNett, Amin Vahdat, Ken Yocum, Alex Snoeren, Geoffrey M. Voelker "DieCast: Testing Distributed Systems with an Accurate Scale Model"Journal ACM Transactions on computer systems (TOCS), vol. 29, no. 2,2011.

[28] Matthew J. Anglin, David M. Cannon, Colin S. Dawson, Barry Fruchtman, Mark A. Haye,Howard N. Martin,"Replication of data objects from a source server to a target server," presented at International Business Machines Corporation,2013.

[29] BrankoMilosavljevic, DanijelaTešendic, "Software architecture of distributed client/server library circulation system", Electronic Library, vol. 28, no. 2, pp.286 – 299, 2010.

[30] Dongmei Zhou, Guoxian Shang, Baojian Chang, "System and method for network file system server replication using reverse path lookup", presented at Ca. Inc, 2013.

[31] Dinuzzo, F Pillonetto. G, De Nicolao. G, "Client–Server Multitask Learning from Distributed Datasets", Neural Networks, IEEE Transactions, vol. 22, no. 2, pp.290 – 303, 2011.