ISSN No. 0976-5697
Volume 5, No. 8, Nov-Dec 2014
International Journal of Advanced Research in Computer Science
RESEARCH PAPER
Available Online at www.ijarcs.info

Replica Location Cost Estimation for Replica Performance in Grid Systems

P.Sunil Gavaskar
Department of Computer Science and Engineering
Sri Venkateswara University
Tirupathi,AndhraPradesh,India

Ch.D.V. Subbaro
Department of Computer Science and Engineering
Sri Venkateswara University
Tirupathi,AndhraPradesh,India

Abstract: The objective challenge in High performance computing is fault tolerance and avoidance. In widely used Existing check pointing schemes provides a way of fault detection and recovery. Fault tolerance, communication, efficiency and reliability are important requirements in grid environment. In general replicas are used as proactive (i.e. failure considered before scheduling of a job) and post active handles the job failure after it has occurred. In our proposed model applications job schedule is periodically uses replicas to maximize the success rate of long execution jobs. In order to make the application execution more reliable we use fault tolerant agents replica as centralized and local replicas within agents. Generally static fault tolerance strategies in job scheduling may use additional resources when compare to dynamic strategy. In this paper we plan to introduce the dynamic fault tolerance architecture model for agents oriented replica to overcome the existing fault tolerance uncertainties that occurs due to the parameters like ability of CPU processing, replicas updating cost, replication read cost, replicas search rate. The proposed fault tolerance architecture model is able to setup some parameters of replicas such as centralized and local replicas with varying mean time failures. This model uses number of replicas, replication objects with different states, requirement of task such as replicas availability, processing capability, and mapping of agents replica to tasks. Finally we observed that the replicas utilization and their reliable activation of replication cost are the major activities to provide qualitative fault tolerance model. The proposed model is evaluated using replicas access rate with different mean failure rates.

Keywords: Read cost, Search cost, Agent replica, Grid information service, Mean time to failure, Resource fault occurrence history.

I. INTRODUCTION

Grid [2] is a basic infrastructure of national high performance computing and Information services, it achieves the integration and interconnection of many types of high performance Computers, Data servers, large-scale storage systems which are Distributed and heterogeneous, and the important application research queries which are lack of effective research approach. Distributed hash tables (DHTs) and other structured overlay networks were developed to give an efficient key based addressing of nodes in volatile environment [6](i.e. distributed nodes that may join, leave or crash at any time).Due to these, two challenges arise in such environment:(a).When a node crashes , all data stored on this node is lost, But it can be addressed by data replication.(b).The node is suspected to be crashed, lookup inconsistencies may occur as a result of that wrong query results or loss of update requests may occur, the second issue can only be relieved but not overcome: It was shown in asynchronous network[14].The task execution requirements are dynamic and effective on the basis of RFOH (Resource Fault Occurrence History) while processing[5].

Data grid [4] provides services for supporting the discovery of resources while computation and enables computing in heterogeneous storage resource by storage of resource agent. Data consistency cannot be achieved if responsibility consistency violated [14] (i.e. when a node is suspected to be crashed, which lead to loss of update requests or wrong query results).As shown in[13]the probability of inconsistent data accesses can be reduced by increasing the replication degree and performing reads on a majority of replicas. In order to ensure data availability when nodes join, leave or crash at any time, the Data consistency is enforced by performing all data operations on a majority of replicas [8].Apart from gathering grid job statistics, the grid job allows for monitoring internal state of objects within an object oriented environment. Taxonomy of grid monitoring tools has been provided in grid [9]. A Priori Replica strategy optimizes distance between the data hosted on the grid as result of that gain in execution time due to the improvement in the file transfer time [15]. In order to identify and handle failures then support for reliable execution, and divided the workflow failure handling into two levels namely task level, workflow-level and provides user-defined exception to specify treatment for certain failures which occurs in data movement[10]. Reliability and Availability are two qualitative measures in Grid computing, which depends on Mean time to Failure, Mean time between failures and Mean time to Repair.

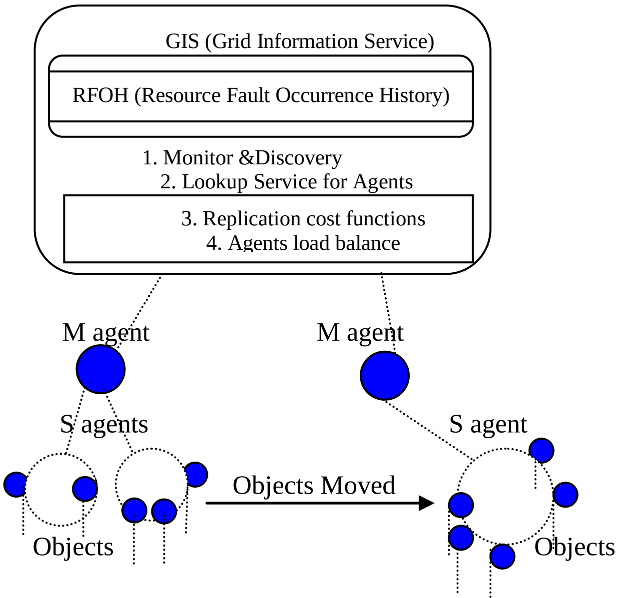
Replication in distributed is a practical and effective method to achieve efficient and fault tolerant data access in grids [1718, 19, 7, 16]. Replica utilization in fault tolerance strategy is challenging research in data grid. By placing multiple replicas at different locations, replica management process can reduces network delay and bandwidth consuming of remote data access [12]. Issues with Replication are widely studied in Grid. Our Proposed Agents replica model mainly shows that the optimized Fault tolerance model.

The rest of the paper is organized as follows. In section 2, we present proposed model related to replication in grid environment, The Section 3 describes the analysis of fault tolerance reliability model with Load in replicas, Section 4 are results finally, we conclude our strategy in section 5.

II. SYSTEM MODEL

A. Fault tolerance replicas architecture

In proposed model each agent is associated with group of objects. A group of tasks are associated with a resource broker (RB) with duration of time t units for execution in computing grid with agents as replicas, here resource broker is responsible for resource discovery, selection, dispatching tasks for execution, tasks execution tracking and delivering of results. Resource broker discovers that a group of n agents to associate with the service task and splits the job into various tasks then distributes to several agents according to user requirements and availability of agents. The Grid Information Service (GIS) maintains information of all agents and helps the broker for scheduling. If resource broker schedules then GIS maintains Resource Fault Occurrence History(RFOH)(i.e. agents status),based on this information different intensity of replicas and objects are discovered while scheduling the jobs on agents to increase the performance of jobs executed within specified time[18,19].The following fault tolerance Replica model plays major role in read cost, update cost and search cost estimation which helps to handle faults.



. Figure.1.Replicas Initial Stage Figure.2.Replicas when S agent fails

The fault tolerance system proposed in this section has been providing reliable model for replication in grid environments. If Agents become failed then adjacent agent performs its fault recovery operations with help of the RFOH maintained by resource broker.

S agents: The role of agents is storing local replicas of the currently registered distributed objects.

M agents: They also act as replicas and contain local copies, there by subset of Agents becomes under the control of M agents, in the proposed model we consider that we have numbers of M agents are equivalent of classes. S agents are having length of class i.e. Number of objects supporting in it.

The agents are considered as replicas, it contains objects to identify particular replica or same replica with different request will be treated as object with different state clients. In order to identify particular object failed or not it requires log C steps, where C represents number of objects currently possessed by an agents.The client initiates the update or search operations. (1).The M agent discover that particular S agent is not active in the system(i.e. agent failed), then the M agent which is in charge of its S agent performs object reallocation to adjacent S agent using inheritance technique.(2).If M agent fails, one of the M agent connected S agent converted into M agent.

B. Cost function of fault tolerance replicas

In general current grid systems are probably hierarchical grids [11, 1], With reference to figure1 and Figure2 the kind of hierarchical data grid is modeled as an undirected tree [3].

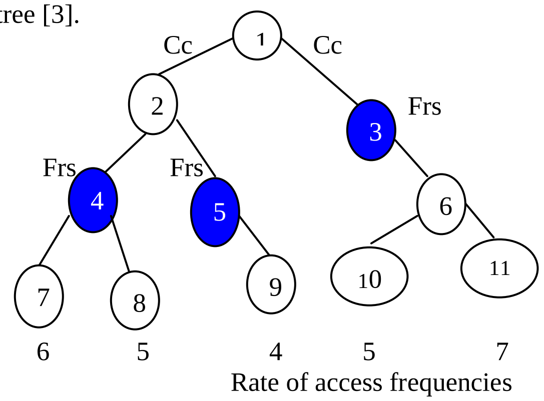


Figure. 3. Formation of Model (M) Agents oriented replica objects with access frequencies.

In the above Tree representation all the leaf nodes(L) are local replica where local site clients can issue requests to access their required data objects(O).The total replication cost function of the specified model (RCF) =Replication read cost(RRC)+Replicas update cost(RUC).

In order to estimate agents replicas read cost, given that the replicas access frequency of user L for file or object O is mentioned as U(L,O),the read cost as follows.

$$U(L, O) .Cc(L, Frs(L,R)) \qquad (1)$$

Now, Given that Cc is communication cost of links (i.e summation of costs of the links along the path).Frs is first agent's replica that meets while going from user L to Root node r.The following formula give all users read cost.

$$RRC=\sum_{L\in M} U(L, O) .Cc(L, Frs(L,R)) \qquad (2)$$

From the figure.3, the agents replicas read cost of (3,4,5) for the object replicas is 6+5+4+(2×5)+(2×7)=39.By assuming the same tree representation Replicas update cost as follows.

$$RUC=\sum_{M_L\in R\neq 0} \mu(O) .Cc(L, Root(L)) \qquad (3)$$

$$\text{Replication cost (RCF)} = \sum_{L \in M} U(L, O) \cdot Cc(L, Frs(L,R)) + \sum_{M L \in R \neq 0} \mu(O) \cdot Cc(L, Root(L)) \quad (4)$$

Where μ(O) is update frequency of replicas or objects, By considering above tree number of updates required for root at most is 3 in an given time period, then replicas update cost of(3,4,5) is Objects(6+6+3).The number of objects increases update cost also increases but search cost is always same for all update frequency. This kind of scalable cost function provides good reliability if fault occurs.

C. Search cost of fault tolerance replicas

In proposed execution model, value 'x' denotes centralized replicas are allocated to the application as Minimum and Maximum number of objects and agents. The search costs of replicas are as follows.

$$x=M*K \quad (5)$$

The Local replicas used with agents and objects, where search costs are calculated using the following equation.

$$y=M*K*(4O+LogC) \quad (6)$$

Where K is the object/agents keys in GIS, M is root agents or M agents, O is the number of objects matching search criteria, C is the objects currently posses by agents.

In dynamic grid environment, there is some fuzziness in the factor of Replicas used as centralized, value 'x' denotes the centralized replicas search cost, the service reliability of centralized replicas are

$$X(t)=1-(1-e^{-\lambda t})^{x} \quad (7)$$

For Local replicas service reliability of maximum and minimum agents/object is

$$Y(t)=1-(1-e^{-\lambda t})^{y} \quad (8)$$

Where Y(t) represents the service reliability of the agents groups. Moreover 'y' denotes Local replicas search rate calculated from equation2,'t' is the time period of a task.

D. Change of replicas location in Grid

Data grid hierarchy G with it roots m. If 'n' becomes user node in G, we use G_n notation to show sub tree rooted at n then $G'_{n} = G_n - n$ is the descendant of n. Further we take p (n, k) mentioning that k-th ancestor of n when moving towards root of G.

Update cost: This process start at Root node and moves towards leaf nodes then ends with some update frequency μ(O) issued by root leaf nodes. Simply it is computed using following Replica updating cost (RUC) = (Number of roots – 1) × frequency of updates issued by root.

Read cost: This process start at non-leaf node and moves towards root node then ends with some access frequency rate 'L'. For replicas read cost (RRC).For read cost estimation refer to equation.2.

Here the cost function of C_f (n, K) gives the replication cost of G_n. if replica used at p(n, Cc), with $0 \leq Cc \leq k$. C_f (n,0) indicates replication cost of sub tree G_n.. if replica used at node n (i.e replication cost is the cost applied for read of all the descendants of n and Update cost of replicas at n. Second case $1 \leq Cc \leq k$ while replicas used at any ancestors of n, C_f (n, Cc) indicates the replication cost of sub tree G_n, which contains read cost alone for the all nodes in G_n..By assuming that one bandwidth unit per data object file of movement per hop of the data grid and update frequencies issued by the root is μ(O) in specific period of time.

Figure 3 shows that illustrative example for replicas placed at nodes 3,4,5 with G_n and references from equation.4 the C_f(n, Cc) is the replicas read cost is 6+5+4+(2×5)+(2×7)=39 and then the μ(O) issued at root is 3 the update cost is (6+6+3)=15,The total cost C_f (n,1) becomes 54.The replica placed on 3,4,6 then G_n , From C_f (n,0) the replicas read cost becomes 6+5+4+5+7=27.If the replica is placed at root 'm' then the replicas present in DHT right side cost function C_f (Nr, Cc) becomes -1 instead of 1.where Nr indicates nodes in right side of root 'm' the Cc at root also becomes -1.

To predict optimal replication cost C_f and estimation of replica location in grid for nodes the n is considered for each distance possibility Cc. At first we do calculation of sum for cost of all children's Chil(C_f) of n when Cc=0,which is formulated with cost function at starting stage.

$$Chil(C_f(n,0)) = \begin{cases} \sum_{q \in Chil(n), chil(n) \neq \emptyset} Cf(q,0) & n \text{ is not } belongs\ to\ client \\ SCf(n,0) & n \text{ belongs to client} \end{cases} \quad (9)$$

If Cc=0 then the optimality in replication cost becomes G_n and locates the replicas using following formulation

$$Cf(n,Cc) = \begin{cases} Chil(Cf(n,0)), & \text{when } SCf(n,Cc) > Chil(Cf(n,0)) \\ SCf(n,Cc) & otherwise \end{cases} \quad (10)$$

For the same set of replicas available at right side of representation, this shows location of replicas such as

$$Nr(n,Cc) = \begin{cases} -1, & when\ SCf(n,Cc) > Chil(Cf(n,0)) \\ n, & otherwise \end{cases} \quad (11)$$

Which gives that children's cost function summation is for all children's of n, and becomes less than that of start stage cost. it is best fit cost to use replicas in children's of node n and also those nodes belongs to right hand side of DHT structure which is set to -1, otherwise node replication position belongs to n. Where Nr is not true (ie which may left side or right side of Tree structure).

When $-1 \leq Cc \leq k\ and\ p(n,k) = m, then$ location changes as follows at the node n.

$$Cf\ (n, Cc) = \begin{cases} Cf\ (n, 0) & when\ SCf\ (n, Cc) > \ \mathrm{Cf}\ (\mathrm{n}, 0) \\ SCf\ (n, Cc) & otherwise \end{cases} \tag{12}$$

$$\mathrm{Nr}(\mathrm{n}, \mathrm{Cc}) = \begin{cases} \mathrm{Nr}(\mathrm{n}, 0), \ when\ SCf\ (n, Cc) > \ \mathrm{Cf}\ (\mathrm{n}, 0) \\ p(n, Cc) \qquad otherwise \end{cases} \tag{13}$$

Communication cost (i.e. distance) Cc becomes -1then it is improved by applying rotation and then perform comparisons on replicas location.

From the observation of Tables.1, 2, 3, 4 the model with agents 2,3,5,7 are considered to express the performance of 25 tasks with an estimated execution time (t) duration of 1800sec and different mean failure rates.

TABLE I
RELIABILITY WITH MEAN FAILURE RATE=0.001

Agents size	Centralized (x)		Local (y)	
	Min.	Max.	Min.	Max.
2	0.4995	0.9685	0.9372	1.0000
3	0.7495	0.9990	0.9994	1.0000
5	0.9842	0.9999	1.0000	1.0000
7	0.9997	1.0000	1.0000	1.0000

TABLE II
NUMBER OF REPLICATIONS SEARCH RATE

Agents size	Centralized(x)		Local(y)	
	Min.	Max.	Min.	Max.
2	1	5	4	103.49
3	2	10	16.602	205
5	6	30	74.86	1835.28
7	12	60	199.22	3678.06

TABLE III
RELIABILITY WITH MEAN FAILURE RATE=0.004

Agents size	Centralized (x)		Local (y)	
	Min.	Max.	Min.	Max.
2	0.4980	0.9681	0.9364	1.0000
3	0.7480	0.9989	0.9999	1.0000
5	0.9839	0.9999	1.0000	1.0000
7	0.9997	1.0000	1.0000	1.0000

TABLE IV
RELIABILITY WITH MEAN FAILURE RATE=0.008

Agents size	Centralized (x)		Local (y)	
	Min.	Max.	Min.	Max.
2	0.4960	0.9674	0.9354	1.0000
3	0.7460	0.9989	0.9999	1.0000
5	0.9836	0.9999	1.0000	1.0000
7	0.9997	0.9999	1.0000	1.0000

From the analysis of Table.2 We have observed that search rate values growth rate is depends on number object replica states and agents replicas.

From the Figure.3.agents size increases which lead to read cost, update cost of replicas will be high. As is observation of Tables.5, 6, 7, the model with agents 2, 3, 5 are considered to express the performance of 25 tasks with an estimated execution time(t) duration of 3600sec and different mean failure rates. It requires only 3 agents and limited number of objects so that loads can be handled sharply which leads to few faults. This kind techniques of reducing faults with will influence on CPU processing capability, when we use less agents CPU processing capability shared among few agents and also hierarchy of data grid from figure 3 can be few hierarchy (i.e. few levels of tree representation) as result of that read ,update and search costs less with respect to distance travel between leaves to root(i.e. cost of Cc (L, Root(L).

TABLE V
RELIABILITY WITH MEAN FAILURE RATE=0.001

Agents size	Centralized (x)		Local (y)	
	Min.	Max.	Min.	Max.
2	0.9990	1.0000	1.0000	1.0000
3	0.9999	1.0000	1.0000	1.0000
5	1.0000	1.0000	1.0000	1.0000

TABLE VI
RELIABILITY WITH MEAN FAILURE RATE=0.004

Agents size	Centralized (x)		Local (y)	
	Min.	Max.	Min.	Max.
2	0.9960	1.0000	0.9999	1.0000
3	0.9999	1.0000	1.0000	1.0000
5	1.0000	1.0000	1.0000	1.0000

TABLE VII
RELIABILITY WITH MEAN FAILURE RATE=0.008

Agents size	Centralized (x)		Local (y)	
	Min.	Max.	Min.	Max.
2	0.9920	1.0000	0.9999	1.0000
3	0.9999	1.0000	1.0000	1.0000
5	1.0000	1.0000	1.0000	1.0000

III. LOAD ON AGENT REPLICAS MODEL

According to the hierarchical data grids replication, the load processing capability depends on number of agents and objects used as replicas, failures that occur between agents replication, unlike some core load influencing capabilities like CPU capability, free disk space and I/O speed.The following formula show that loads on agent's model.

$$\text{Load on Agents model} = \left[\sum_{j=1}^{m} (\mathbf{O_j\ Sagents_j}^{2}) \right]^{1/2}$$

In the above formula O_j indicates numbers of objects in agents .The computation of load values happens periodically due to dramatically changes in the agent objects and S agents size, the average load value shows that real load on 'm' master agents. In general search for replica access depends on number objects , number of agents present, Task execution and Mean failures rates (i.e. CPU processing capability rate).The agents band width can be computed using disk I/O ,CPU capability is computed using frequency of read and updates is multiple with Usage as follows.

(a).CPU Processing = Frequency of read/Update/Search*(1-Usage of CPU).which include mean failure rates also.

(b).Agents model bandwidth is = total number of agents bandwidth / number of links connecting replicas.

IV. Evaluation

In globally dispersed structured systems, latency is an impressive issue. In order to access replicas with reliable search rate we provided agents replica. To evaluate our proposed system the model with agent nodes 2, 3, 5, 7 is considered to express the performance 40 to 1000 records of 25 tasks with job data size is from 30 to 180MB.For reliability performance we considered local replicas, centralized replica, mean failures that occurs, task execution time in second's in order to maintain account for the reliability and availability of agents when failure occurs. Moreover, we introduced load factor of agents while failure occurs at run time. The following are two situations that cause dynamic changes (a).When the number of objects present in agent is maximum than its capability (b). Number of S agents connected to M agents becomes too high. Besides the overhead introduced by the process of new agents comes in or objects moving from heavily loaded agents to lightly loaded agents, due to above said alternation the read cost, update cost of replicas can also affects the reliability, this shows marginal changes in fault tolerance agent's availability.

The results shows that reliability service depends on the probability of all agents are operational and can properly communicate with each other in entire time interval (0,t).The probabilistic values reliability comprises of the simple grid job submission with an estimation of mean failure rate of agents and tasks duration. Reliability performance of agent's model calculated upon read cost, search cost of local and centralized replicas. Unlike centralized replicas where as in local replicas model almost we have achieved 1.0000 reliability irrespective of agents size as result of that number of agents failures are less and also scalability is effective when compare to centralized replicas. The Table.2 shows that number of replicas is applied as centralized, local, with their minimum and maximum capabilities and its access rates.

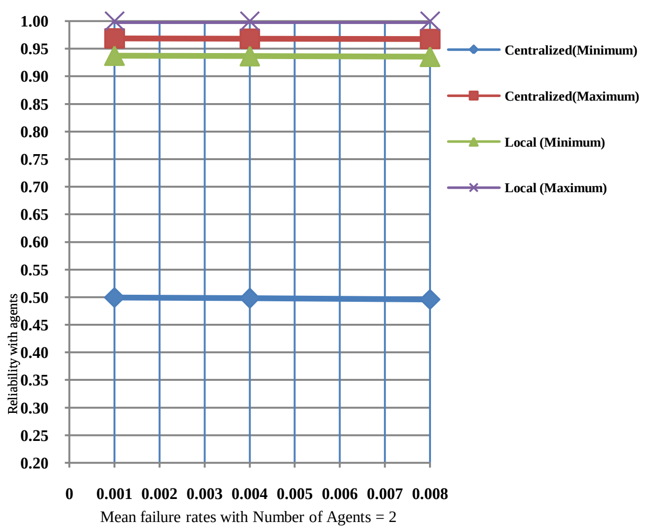


Figure.4. Relibility of agents with task execution 1800 sec.

In the above Graph representation, we have brought mean failure rates of 0.001,0.004,0.008 into agents model to attain reliability of 1.0000.In this section local (maximum) strategy alone have achieved the reliability of 1.0000 with task execution time 1800 second along with the CPU processing capability and load changes of agents. The search rate of local (Maximum) strategy assesses the constant impact on reliability of model irrespective of mean failure rates. From the table values we determine that increasing mean failure rates are carried out as result of that change in reliability due to impact of search rate, read cost, update cost and we have achieved failure –free performance of the system.

It could be argued that this measure will provides connection between the structural representations to cost reduction in replicas.

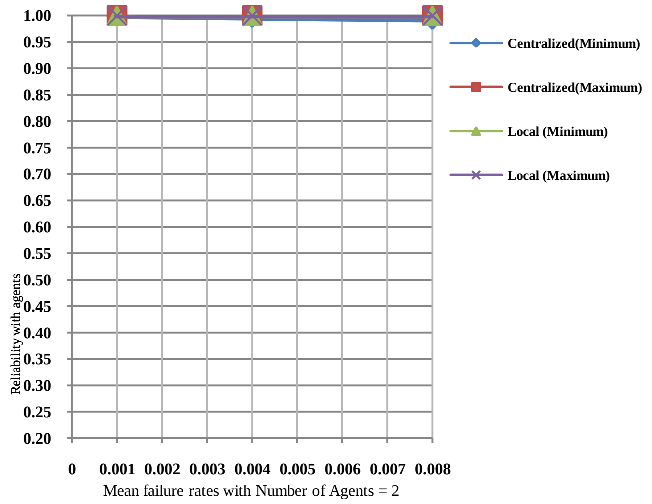


Figure.5. Relibility of agents with task execution 3600 sec.

Figure5 comprises of same mean failure rates which had been considered in previous case but difference exists only in the task execution time, Here the replicas with centralized and local strategies determines that agents 2,3,5 are enough to obtain proper reliability as result of that few agents in failure and it is useful when we apply tightly coupled replicas. As for working group of agent's replica have been number consistent initiatives to integrate with fault tolerance model.

Performance Studies:

The simulation configuration consisting of about 20 nodes which connected in cluster of sites with heterogeneous in nature, each having storage element and computing element with capabilities from 512 MB to 1GB.The number object lookup considered at the maximum is 400 objects are considered .A minimum number of storage element per site is 1.

In the Figure6 and Figure7, Each stage of nodes are considered as 10 percentile of response time, overheads respectively it means that at 2 nodes there is presence of 10 percentile of response time, overheads then it could be treated as distributed shared storage space up to 20 nodes with 90.The Figure6 shows percentile of response time comparisons are carried out with both DHT and Object space strategies. The totals of 10 clusters are used with each cluster having the minimum two clients and one master node per site. We use response times given in the existing DHT for our analysis.

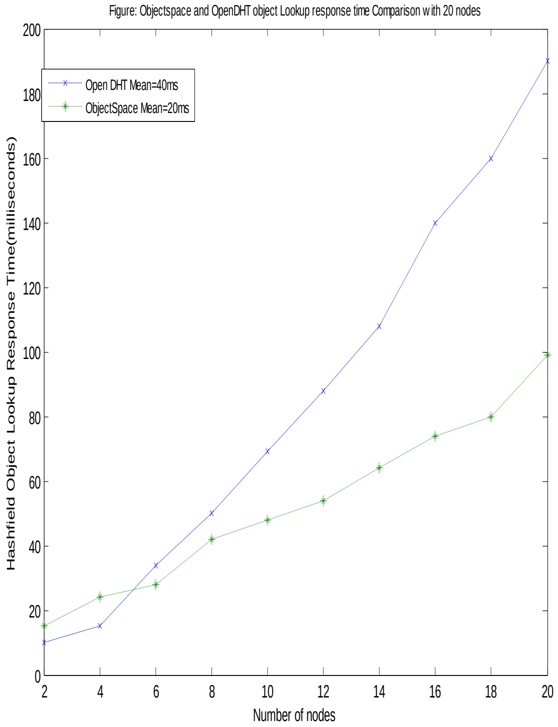


Figure.6.ObjectSpace Vs Open DHT response time with Proposed Object pace cost estimation.

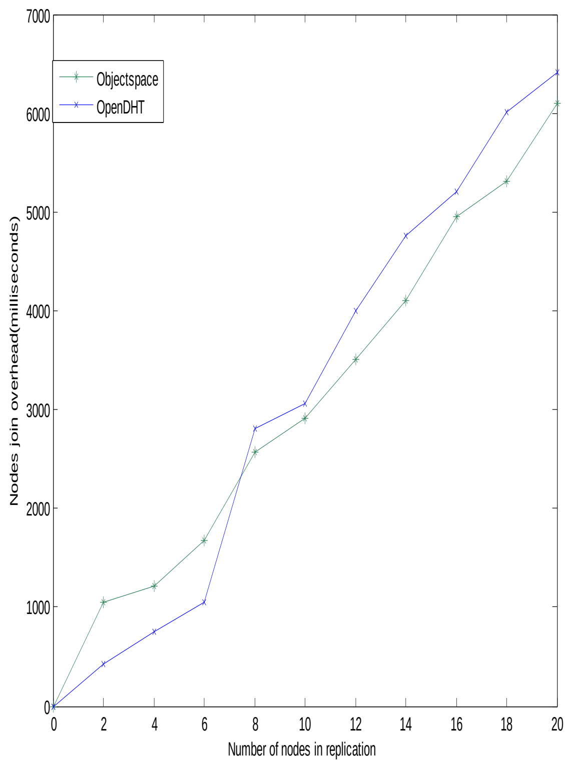


Figure.7. ObjectSpace Overhead Vs Open DHT Overhead Comparison.

We use nodes range from 2 to 20 for response time measure each stage of nodes is considered as 10 percentile in overall measurement (i.e 2 nodes belongs to 10 percentile, 20 node belongs to 90 to 99 percentile), until the 20th percentile the response times for DHT are lower when compare to our proposed object space repository maintenance of replica location, migration with read and update cost replication schemes. However 80 to 90 percentile response times for DHT are very high compare to proposed scheme. The response time of DHT at 90 percentile goes up to 190 ms where as in case of Object Space response time is 90ms. The objective reason for the worst case poor performance of DHT is the presence of fault over resources (i.e. slow nodes, Mean failure rate, mapping of replicas location or heavily loaded nodes with object of replicas).The figure7shows node join overhead and object replicas overhead with our proposed scheme. Therefore our proposed scheme provides Overhead best case performance when compare to existing scheme.

The reason for the performance improvement at 90th percentile of Object space scheme is the way we consider the read and update of Object space in metadata catalogue.

V. CONCLUSION

In this work, we have proposed dynamic fault tolerance strategy model which overcomes drawbacks of the core fault tolerance replication techniques, mechanism of scalable architecture techniques. Generally the local disk replicas can tolerate multiple node failures, Therefore Proposed strategies will helpful when long run of jobs and it covers drawbacks of centralized replica utilization techniques .In existing schemes the centralized replicas increases the replication time and hence degrades performance. Due to the reliable distance of replicas proposed in our scheme the replication will be reliable irrespective of replicas count because access rate can increase the response time.

In addition, we have illustrated the procedure to integrate the agents oriented fault recovery job scheduling system in hierarchical data Grid. Several issues like load, read cost, update cost, search costs are considered to account for replication fault tolerance. In future we plan to implement middleware approach into our proposed strategies.

ACKNOWLEDGMENT

We deeply thank Dr. Ch D.V. Subbarao for his valuable help. The authors would like to thank the anonymous peer reviewers for their suggestions that aided in improving this paper.

REFERENCES

[1] K. Ranganathan and I. Foster, "Identifying dynamic replication Strategies for a High-Performance Data Grid", in Proc. Intl. Wkshp. on Grid computing, 2001, pp 75-86.

[2] Foster I, Kesselman C.The grid: blue print for a new computing infrastructure [M]. San Francisco, USA: Morgan Kaufman Publishers, 1999.

[3] K. Ranganathan and I. Foster, "Design and Evaluation of dynamic Replication Strategies for a High-Performance Data Grid", in Proc. Intl. Conference. Computing in High Energy and Nuclear Physics (2001).

[4] Y. Wang, N. Xiao, R. Hao, et al. Research on key technology in data grid [J].Journal of Computer Research and Development, 2002, 39(8):943-947.

[5] Foster I, Kesselman C, Lee C, Lindell B, Nahrstedt K and Roy A,(1999)" A distributed resource management architecture that supports advance reservation and co-allocation ", Proceedings of the International Workshop on QoS,London,U.K.,27-36.

[6] L.Alima, S.El-Ansary, P.Brand and S.Haridi.DKS (N, k, f): A family of low-communication, scalable and fault-tolerant infrastructures for P2P applications. Workshop on Global and P2P Computing, CCGRID 2003, May 2003.

[7] Gao, M.Dahlin, A.Nayate, J.Zheng and A.Iyengar,"Improving Availability and Performance with Application-Specific Data Replication", IEEE Trans.Knowledge and Data Engineering, Vol.17, No.1, pp.106-200, 2005.

[8] A.Ghodsi, L.Alima, S.Haridi. Symmetric replication for structured Peer-to-Peer Systems.DBISP2P, Aug.2005

[9] S.Zanikolas and R.Sakellariou."A Taxonomy of grid monitoring system", In Future Generation Computer Systems, vol.21, issue.1, pp: 163-188, 2005, Elsevier Science Publishers.

[10] S.Hwang and C.Kesselman, (2003) "Grid Workflow: A Flexible failure handling framework for the Grid",In Proceedings of the 12th IEEE International Symposium on High Performance Distributed Computing,Seattle,Washington,USA.

[11] W.H.Bell,D.G.Cameron,R.CarvajalSchiaffino,A.P.Millar,K.Stockinger,and F.Zini, "Evaluation of an economy based file replication strategy for a data grid," in Proceedings of the 3rd IEEE/ACM International Symposium on Cluster Computing and the Grid,2003.

[12] H. Lamehamedi , B. Szymanski, Z. shentu, and E. Deelman , "Data replication strategies in grid environments, " in Proceedings of the Fifth International Conference on Algorithms and Architectures for Parallel Processing , 2002, pp.378-383.

[13] T.M.Shafsst, M.Moser, T.Schutt, A.Reinefeld, A.Ghodsi, S.Haridi,Key-based consistency and availability in structured overlay networks. Infoscale, June 2008.

[14] A.Ghodsi.Distributed k-ary system: Algorithms for distributed hash tables. PhD Thesis, Royal Institute of Technology, 2006.

[15] M.Garmehi and Y.Mansouri, "Optimal placement replication on data grid environments", in 10th International Conference on Information Technology (ICIT 2007), pp.190-195.

[16] M.Tang, B.S.Lee, X.Tang and C.K.Yeo"The impact on data replication on Job Scheduling Performance in the Data Grid" International Journal of Future Generation of Computer Systems, Elsevier (22), pp254-268, 2006.

[17] Jose M. Perez, Felix Gracia-Carballeira, Jesus Carretero, Alejandro Calderona and Javier Fernandeza, "Branch replication scheme: A new model for data replication in large scale data grids", Future Generation Computer Systems, Vol.26, No.1,,pg12-20,Jan2010.

[18] Leyli Mohammad Khanli, (2010)"Reliable Job Scheduler using RFOH in Grid Computing", Journal of Emerging Trends in Computing and Information Sciences,Vol.1,No.1.

[19] Leyli Mohammad Khanli, Maryam Etminan and Far Amir Masoud Rahmani,(2010) "RFOH: A New Fault Tolerant Job Scheduler in Grid Computing", In Second International Conference on Computer Engineering and Applications.

BIOGRAPHY

P. SunilGavaskar is a PhD candidate at the Department of Computer Science and Engineering, Sri Venkateswara University; Tirupathi.His research interests include distributed systems, Grid Computing. Contact him at sunil079@gmail.com.

Dr. Ch D.V. Subbarao is a professor and Head of the Department of Computer Science and Engineering at Sri Venkateswara University; Tirupathi. He is Fellow of IETE, Fellow of IE (India) , Senior Member of CSI, At present serving as the Honorary Treasurer, IETE, Tirupati Centre. He Received 'Rashtriya Gaurav Award' in 2011 and His research interests include distributed systems, Grid Computing. Advanced Operating Systems, Contact him at Subbarao_Chdv@hotmail.com.