Review of Replication Techniques for Distributed Systems

Ahmad Shukri Mohd Noor[1(✉)], Nur Farhah Mat Zian[1], Mustafa Mat Deris[2], and Tutut Herawan[3]

[1] School of Informatics and Applied Mathematics, Universiti Malaysia Terengganu, Kuala Terengganu, Terengganu, Malaysia
ashukri@umt.edu.my, farhah.matzian@outlook.com
[2] Faculty of Information and Technology Multimedia, University Tun Hussein Onn Malaysia, Batu Pahat, Johor, Malaysia
mmustafa@uthm.edu.my
[3] Department of Information System, University of Malaya, Pantai Valley, 50603 Kuala Lumpur, Malaysia
tutut@um.edu.my

Abstract. Distributed systems primarily provide the access to data intensive computation through a wide range of interfaces. Due to the advances of the systems, the scales and complexity of the system have increased, causing faults are likely bound to happen leading into diverse faults and failure conditions. Therefore, fault tolerance has become a crucial property for distributed system in order to preserve its function correctly and available in the presence of faults. Replication techniques particularly concentrates on two fault tolerance manners; masking the failures on the fly as well as reconfiguring the systems in response. This paper presents a brief reviews on different replication techniques, such as Grid Configuration (GC), Box-Shaped Grid (BSG) and Neighbor Replication on Grid (NRG) by comparing and formalizing its communication costs and availabilities analysis based on k-out-of-n model. Each of these techniques presents their own merits and demerits which form the subject matter of this review.

Keywords: Distributed systems · Fault tolerance · High availability · Replication techniques · Communication cost · Availability analysis

1 Introduction

Distributed systems consist of a determinate set of sites that are connected to each other via communication links on various hardware hosts at different physical locations where remotely located users sharing the data to produce some invaluable results. The increase in complexity of distributed systems causes most application unable to survive infrastructure outages in the event of failure. A fault, a defect that come about the lowest level of abstraction may cause

G. Di Fatta et al. (Eds.): IDCS 2015, LNCS 9258, pp. 169–176, 2015.
DOI: 10.1007/978-3-319-23237-9_15

an internal data state, error which may later lead to failure, the external visible deviation from the system correctness behavior [1]. However, fault does not inevitably result in an error as well as an error in failure. Fault tolerance is an added property to ensure the availability and reliability of a distributed systems. There are two ways to implements fault tolerance in distributed systems. The first one is reconfiguring the service to utilize the components that replaced the failed one and the second one is by using the masking approach that is placing redundant components that able to operate under partial failures providing enough functionality to the users.

1.1 Replication Definition

Replication is a process of maintaining different copies of replicated data or objects as well as the synchronization of updating the data in its replica. The data is not automatically overwritten whenever any changes occurred in the original data that resulted in immediate lost at any historical state. Replication is divided to two types of solution that is synchronous or asynchronous solutions. The asynchronous solution manages the data by immediately capture any changes on the primary replica and will be timely propagated to its replica. This solution presents a low cost as well as flexible solution nevertheless have to deal with network bandwidth and data lost. Synchronous solution on the other hand based on quorum in updating its replicas at the same time and will roll back if one fails. It breaks into several schemes that are full replication (all-data-to-all-sites) and partial replication (all-data-to-some-sites). This solution offers high availability of data, auto fail-over and minimal data loss. However, the drawbacks of this solution are one has to deal with network efficiency, less flexible, scalability issues and cost.

Replication has been successfully implemented for distributed systems by allowing data to remain distribute and systems able to operate in the presence of fault. Considering a large number of different replication techniques available, choosing the best approach under different levels of faulty systems with different systems performance requirements is not an easy task. In this paper, we will present a brief review some of the replication techniques that has been implemented by addressing its own merits and demerits. The performance of each technique is analyze based on the communication costs and availability using k-out-of-n model. This comparison is focusing only on existing synchronous replication. The remainder of this paper is organized as follows. Section 2 will discuss briefly on the k-out-of-n model and Sect. 3 will review each of the replication techniques by presenting its communication costs and availability analysis. The conclusion of our work will be presented in Sect. 4.

2 System Availability Evaluation Using K-out-of-n Model

In this study, the k-out-of-n model is used to estimate the operation availability. This model refers to the independent nodes that have some identical data or services [12]. In other words, this configuration will not affect the remaining nodes and all nodes have the same failure distribution. The assumption of k-out-of-n model could be evaluated using the binomial distribution or as below:

1. The data item and its copies are in one of the two states: accessible or inaccessible.
2. The states of the copies are changed independently.
3. If at least k of its n copies is accessible, then the data item is available for an operation.

Thus, the k-out-of-n model can be formulated as:

$$\Sigma_{i=k}^{n} \binom{n}{i} p^{i}(1-p)^{n-1}, k \geq 1 \tag{1}$$

where the notation is given as:

- n: the total number of components in the systems
- k: the minimum number of components required for the system to success
- p: the availability of each data item.

3 Replication Techniques

3.1 Grid Configuration (GC) Protocol

This protocol is introduced by Maekawa [5] where all quorums are of equals size in order to obtain a distributed mutual exclusion algorithm which later extended by Cheung et al. [6] for replicated data objects. This protocol introduced n copies of data objects are logically organized in the form of a $\sqrt{}$ n × $\sqrt{}$ n as depicted in Fig. 1. Read quorum consists of a copy from each column in the grid will be acquire in order to perform read on the data items. While to perform the write operation, write quorum consists of all copies in one column and a copy from each of the remaining columns will be needed. This protocol introduced the read and write operation in the size of O ($\sqrt{}$ n).

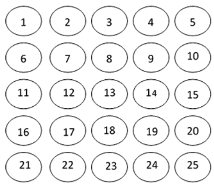

Fig. 1. Grid Configuration with 25 copies of data object

The communication cost for read operation of GC protocol, $C_{GC,R}$ can be formalized as:

$$C_{GC,R} = \sqrt{n} \tag{2}$$

Meanwhile, the communication cost for write operation of GC protocol, $C_{GC,W}$ can be formalized as:

$$C_{GC,W} = 2\sqrt{n} - 1 \tag{3}$$

In the case of quorum techniques, read quorums can be constructed as long as a copy from each column is available. Under this technique, as given in [3], read availability of GC protocol, $A_{GC,R}$ is:

$$A_{GC,R} = [1 - (1 - p)^{\sqrt{n}}]^{\sqrt{n}} \tag{4}$$

On the other hand, write availability of GC technique $A_{GC,W}$ can be write as:

$$A_{GC,W} = [1 - (1 - p)^{\sqrt{n}}]^{\sqrt{n}} - [1 - (1 - p)^{\sqrt{n}} - p^{\sqrt{n}}]^{\sqrt{n}} \tag{5}$$

The drawbacks of this technique is this structure degrades the communication cost and the availability of data as the number of copies for both read and write quorum is big as well as prone to failure of the entire row and column in the grid.

3.2 Box-Shaped Grid (BSG) Protocol

Box-Shaped Grid (BSG) was introduced by Deris, M.M et al. [7]. In this technique, all copies are logically organized in a box-shape structure with four planes. The box-shaped structure is classified as a perfect square if the number of copies in each plane is equal. From Fig. 2, BSG model consists of four planes (1, 2, 3, 4) and small circles symbolizing a copy of data object with location denoted in A, B, C, D, ... , and X. By assuming that each plane has a length (column), l and width (row) w, then l = w. Therefore, the number of copies in the structure can be calculated as l=$\frac{[1+\sqrt{(l+n)}]}{2}$.

Read transactions on a data object are executed by obtaining a read quorum that comprises of any hypotenuse copies, the pair of copies that located at hypotenuse edge. From the Fig. 2, hypotenuse copies are {X, A}, {V, C}, {G, R} or {I, P}. While for write transactions, a write quorum is formed from any planes that contain hypotenuse copies and all copies which are vertices where one of which is a hypotenuse copy. For an instance, if the hypotenuse copies, {I, P} are required to perform a read transaction, then the possible copies for write quorum are {I, P, V, R, X} that adequate enough to perform write operations. The size of a read quorum in BSG is hypotenuse copies, i.e., 2. Thus, the cost of a read operation, $C_{BSG,R}$ = 2, and the cost of write operation C_BSG, w can be represented as $C_{BSG,W}$ = hypotenuse copies + (all copies of vertices in a plane - hypotenuse copy in the same plane) = 2 + (4 - 1) = 5.

If the copy that initiate the write transaction is not a vertex-replica, then communication cost becomes 6. For BSG protocol, a read quorum can be constructed from any hypotenuse copies in the system. Thus, the read availability $A_{BSG,R}$ for a perfect square box where it has 4 hypotenuse copies is:

$$A_{BSG,R} = 1 - (1 - p^2)^4 \tag{6}$$

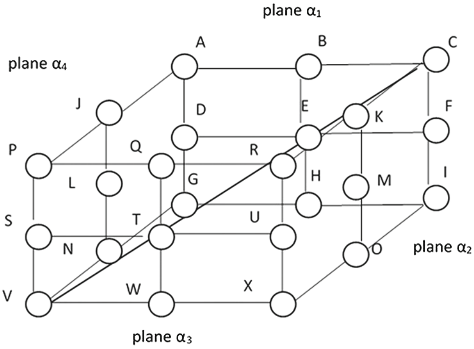

Fig. 2. A BSG organization with 24 copies of a data items

On the other hand, a write quorum can be constructed as: Let $\alpha_1, \alpha_2, \alpha_3, \alpha_4$ be a set of planes in the BSG protocol. Each of which consists of l $\times$ l copies. Let {V,C} be the hypotenuse copies, then write availability that consists of {V,C}, $W_{V,\ C}$, can be represented as:
Probability { V is available } * [ϕ available] + Probability { C is available } * [φ available] $-$ Probability { C and V are available } * [(ϕ and φ) are available] where,

$\phi = \Omega(\ \alpha_1\ ) + \Omega\ (\ \alpha_2\ )$ - $\Omega\ (\ \alpha_1 \cap \alpha_2\ )$
$\varphi = \Omega\ (\ \alpha_3\ ) + \Omega\ (\ \alpha_4\ )$ - $\Omega\ (\ \alpha_3 \cap \alpha_4\ )$,
and $\Omega\ (\ \alpha_1\ )$ = probability of plane α_1 available.

Without loss of generality, we assume that copy B $\in \alpha_1$ is a primary replica. The probability of α_1 is available, $\Omega\ (\alpha_1)$, can be presented as:
Probability { all copies of vertices from α_1 and primary copy are available } + Probability { (all copies of vertices and primary copy + 1 copy) from α_1 are available } + ... + Probability { all copies from α_1 are available }

and by calculating $\Omega\ (\alpha_1 \cap \alpha_2)$ using Venn Diagram and substituting into previous probability, then

$$\varphi = p^4(1 + p - p^2) \tag{7}$$

$$\phi = p^4(2 - p^2) \tag{8}$$

Since the probability {V is available} = probability {C is available} = p, then, by substituting Eqs. (7) and (8) into first probability, W_i is:

$$p_\varphi + p_\phi - p^2(\phi * \varphi) \tag{9}$$

By the same token, for the write availability that consist of other hypotenuse copies, $W_i = W_j = W_k = W_l$, where i $\in$ R.To compute the write availability, $A_{BSG,W}$, it is analogous to the read availability. Let $W_i = \beta$, i $\in$R, then $A_{BSG,W}$ is:

$$A_{BSG,W} = 1 - (1 - \beta)^4 \tag{10}$$

For this technique, the hypotenuse of the copies will not be accessible if one of the copies of each pair is not available, therefore, resulted in failure at the

Table 1. Comparison of the read and write communication costs for the three protocols

Number of copies in the system			
Protocol(X)	13	25	64
GC (R)	7	9	15
GC (W)	5	5	5
BSG (R)	2	2	2
BSG (W)	5	5	5
NRG (R)	3	3	3
NRG (W)	3	3	3

write quorum for executing write transaction. It also required both read and write quorum to intersect in order to access the read quorum hypotenuse. This resulted in the inability to update write quorum to the latest data object which affected the consistency of the data (Table 1).

3.3 Neighbor Replication on Grid (NRG)

Neighbor Replication of Grid has been introduced by N. Ahmad [3] by applying neighboring techniques in a two-dimensional grid structure. All the sites in this mechanism are logically organized in the form of n $\times$ n structure, by having a master data item stored in each site. The number of data replication from each site, d $\leq$ 5, that is the number of the sites neighbor sums up with a data from the site itself as in Fig. 3.

For simplicity, the read quorum equals to the write quorum is chosen in this analysis. Thus, the communication cost for read and write operations equals to total number of votes, υ over 2, that is:

$C_{NRG,R}$= $C_{NRG,W}$= $[\upsilon_x /2]$

As an instance, let say the primary site has four neighbors and assuming that each site has voted once. Thus, $C_{NRG,R} = C_{NRG,W} = [\frac{5}{2}] = 3$.

Let p_i denotes the availability of a data item at site i. For any assignment B and quorum q for the data x, define ϕ (B_x, q) to be the probability that at least q sites in $S(B_x)$ are available. Then

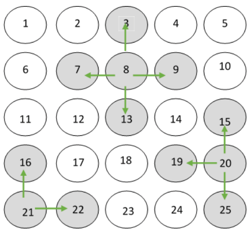

Fig. 3. Data Replication of 25 nodes in NRG

ϕ ($B_{x,q}$) = Pr{ at least q sites in S(B_x) are available}

$$\sum G \in Q(B_{x,q}(\prod_{j\in G}^{p_j} \prod j \in S(B_x) - G(1 - p^j)) \tag{11}$$

Thus, the availability of read, $A_{NR}G$, $_R$ and write operations $A_{NR}G$, $_W$ for the data item x for NRG, are $\phi(B_{x,r})$ and $\phi(B_{x,w})$ respectively.

The complexity of maintaining copies at different sites gave a challenged for NRG to handle its locking information especially in the current structure of distributed computing (Table 2).

Table 2. The read and write availabilities of each protocol when n = 13 and 0.1 ≤ p ≤ 0.9

Read (R) and Write (W) availability (A_{XR})									
Protocol (X)	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
GC (R)	0.014	0.124	0.333	0.57	0.773	0.902	0.968	0.994	0.999
GC (W)	2E-6	1E-4	0.14	0.65	0.186	0.392	0.646	0.873	0.986
BSG (R)	0.039	0.151	0.314	0.502	0.684	0.832	0.932	0.983	0.999
BSG (W)	1E-4	0.004	0.030	0.112	0.319	0.614	0.875	0.986	1.00
NRG (R)	0.009	0.058	0.163	0.317	0.500	0.683	0.837	0.942	0.991
NRG (W)	0.009	0.058	0.163	0.317	0.500	0.683	0.837	0.942	0.991

4 Conclusion

This paper briefly discussed existing replication techniques in distributed computing by addressing its communication cost and availabilities analysis based on k-out-of-n model and the importance of replication in the distributed computing. Each technique have been appropriately analyzed and presented under different headings by reviewing its concepts, communication costs, and availability analysis as well as providing illustration for each model.

References

1. Helal, A.A., Heddaya, A.A., Bhargava, B.B.: Replication Techniques in Distributed Systems. Kluwer Academic Publishers, United States of America (1996)
2. Kuo, W., Zuo, M.J.: Optimal Reliability Modelling: Principles and Applications. John Wiley and Sons Inc, United States of America (2003)
3. Noraziah, A.: Managing Replication and Transactions using Neighbor Replication on data grid Database design. Ph.D. Thesis. Universiti Malaysia Terengganu (2007)
4. Deris, M.M., Mamat, A., Seng, P.C., Saman, M.Y.: Three dimensional grid structure of efficient access of replicated data. Intll J. Interconnection Netw. World Sci. **2**(3), 317–329 (2001)

5. Maekawa, M.: A vn algorithm for mutual exclusion in decentralized systems. ACM Trans. Comput. Sys. **3**(2), 145–159 (1992)
6. Cheung, S.Y., Ammar, M.H., Ahmad, M.: The grid protocol: a high performance schema for maintaining replicated data. IEEE Trans. Knowl. Data Eng. **4**(6), 582–592 (1992)
7. Deris, M.M., Abawajy, J.H., Mamat, A.: An efficient replicated data access approach for large-scale distributed systems. Future Gener. Comput. Sys. **24**, 1–9 (2008)
8. Noraziah, A., Mat, D.M., Saman, M.Y.M., Norhayati, R., Rabiei, M., Shuhadah, W.N.W.: Managing transactions on grid-neighbour replication in distributed systems. Int. J. Comput. Math. **86**(9), 1624–1633 (2009)
9. Bansal, S., Sharma, S., Trivedi, I.: A detailed review of fault tolerance techniques in distributed system. Int. J. Internet Distrib. Comput. Sys. **1**(1), 33 (2012)
10. Shen, H.H., Chen, S.M., Shen, M.M. and Zheng, W.M.: Research on data replica distribution technique for server cluster. In: IEEE Proceedings of the 4th International Conference on Peformance Computing, pp. 966–968, Beijing (2000)
11. Natanzon, A., Bachmat, E.: Dynamic synchronous asynchronous replication. ACM Trans. Storage 9(3), 1–19 (2013). Article 8
12. Noor, A.S.M.: Neighbour replica affirmative adaptive failure detection and autonomous recovery. Ph.D. Thesis. Universiti Tun Hussein Onn Malaysia (2012)
13. Noor, A.S.M., Zian, N.F.M., Saman, M.Y.: Survey on replication techniques for distributed system. Sci. Int. Lahore **26**(4), 1523–1526 (2014)
14. Renesse, R.V., Guerraoui, R.: Replication Techniques for Availability. Theory and Practice. LNCS, pp. 19–40. Springer, Heidelberg (2010)
15. Deris, M.M., Evans, D.J., Saman, M.Y., Noraziah, A.: Binary vote assignment on a grid for efficient access of replicated data. Int. J. Comput. Math. **80**(12), 1489–1498 (2003)