OPEN ACCESS

International Journal Of Modern Engineering Research (IJMER)

Performance Evaluation and Enhancement of Distributed Storage Area Network using Network Coding

Priyanka Ramesh Nanaware[1], Prof. S. K. Srivastava[2]

[1]*Department of Electronics Engineering, Pillai Institute of Information Technology, Engineering, Media Studies & Research New Panvel, India, ME(Electronics Engineering) Final Year*

[2]*Associate Professor in Department of Electronics and Telecommunication Engineering, Pillai Institute of Information Technology, Engineering, Media Studies & Research New Panvel, India*

ABSTRACT:- *In this paper, it is evaluated the performance of distributed storage area network based on the evaluation of different performance parameters such as bandwidth consumption, throughput, packet delivery ratio, latency and recovery speed using network coding such as erasure code, regenerating code and self-repairing code. The results show that Self-repairing code provides better throughput and fast recovery as compared to erasure and regenerating code. At the same time, it also gives low delay, less bandwidth consumption and high packet delivery ratio. Therefore it concludes that the performance of self-repairing code over erasure code and regenerating code is better in this scenario.*

Keywords:- *Distributed storage area Network, Network Coding, Ns3 Simulator, Performance parameters*

I. INTRODUCTION

Distributed storage area network play a very important role in the field of Networking and in Advanced Computing. Such a distributed storage area network is formed using cluster. There are number of Types and ways to form a cluster such as Hadoop Cluster, High performance cluster, High availability cluster, Ceph cluster etc. In a recent scenario a wide variety of applications, relies on distributed environments to process and analyse large amounts of data it is necessary to improve the performance of such Distributed storage network. One way to achieve this is by network coding.[1][2][9][10]

In traditional network data is stored in a single disk, suppose that disk will failed then there is completely loss of data. After that In RAID there is mirroring technique, it will store the same copy of original data in another disk. So here if original disk gets failed then another disk (which have the same copy of original data) have same data, hence there is no loss of data. But in case of RAID due to mirrored technique the Bandwidth required is more.[3]

The summarization of this system configuration is as follows:

- Distributed storage area network is formed using ceph cluster.
- In a distributed storage area network the data is distributed in number of disks.
- Here if one or two disks get failed then also we can recover that using the data in the other disks.
- So the main advantage of distributed storage area is the data is stored is secured form and there is no any kind of loss of data.

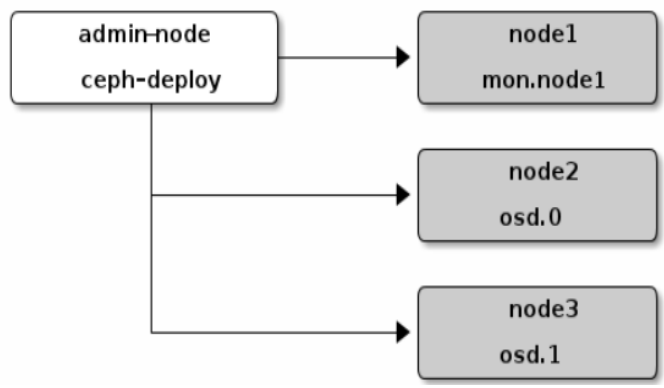

Fig 1 Ceph Cluster

- Cluster monitors (ceph-mon) that keep track of active and failed cluster nodes.
- Metadata servers (ceph-mds) that store the metadata of all nodes and directories
- Object storage devices (ceph-osd) that actually store the content of files. Ideally, OSDs stores the data which is distributed by the admin node of the ceph cluster.
- Ceph admin node responsible for in which way data should distribute.[4]

II. NETWORK CODING IN DISTRIBUTED STORAGE AREA NETWORK

Network coding is a method of optimizing the flow of digital data in a network by transmitting digital evidence about messages. The "digital evidence" is, itself, a composite of two or more messages. When the bits of digital evidence arrive at the destination, the transmitted message is deduced rather than directly reassembled.[5][6]

III. ERASURE CODE

Erasure code is also called Maximum Distance Separable (MDS) code. This code improves the storage efficiency. Here the original data is divided into M fragments which are stored in n nodes. Each node stores M/k data fragments and k nodes are required to recreate the original data. MDS code achieves the optimum in the redundancy-reliability trade-off.[7]

When a storage node is repaired in an (n,k) MDS system, the new node connects to k surviving nodes and download M, the original file. The new node first recreates the original data and then creates the fragments that are to be stored.

Performance Parameters	Simulation Time (ms)		
	100	150	200
Bandwidth Consumption(Hz)	93.4933	93.4917	128.261
Throughput(Mbps)	1914.74	2872.07	5253.57
Packet Delivery Ratio	0.449333	0.449635	0.450003
Latency(ms)	0.800182	0.800622	0.801754
Recovery Speed(Mbps)	1.44933	1.44964	1.45000

Table 1 Performance Parameters of Erasure Code

IV. REGENERATING CODE

This is another variation of the erasure code. The regenerating code provides an improved repair function. This is made possible because of the fact that the regenerating code allows a new node to connect all remaining storage nodes after a node failure. This means $k \leq d \leq n - 1$ for the regenerating code, compared to $d = k$ for the MDS code.

There is a storage/bandwidth trade-off that is of great interest. It describes the relationship between the amount of stored data and the amount of data that necessary to transfer for a repair. There are two extreme points on this curve called Minimum Storage Regenerating (MSR) and Minimum Bandwidth Regenerating (MBR). They are what they sound like; MSR stores the smallest amount of data possible in each node and MBR requires the least amount of data to be transferred to make a repair.[7]

Performance Parameters	Simulation Time (ms)		
	100	150	200
Bandwidth Consumption(Hz)	74.7946	74.7954	102.609
Throughput(Mbps)	15812.1	23717.7	43384.3
Packet Delivery Ratio	0.549339	0.549641	0.550009
Latency(ms)	0.359548	0.359988	0.360792
Recovery Speed(Mbps)	1.50456	1.50487	1.50523

Table 2 Performance Parameters of Regenerative Code

V. SELF-REPAIRING CODE

The concept of self-repairing codes as (n, k) codes designed to suit networked storage systems, that encode k fragments of an object into n encoded fragments to be stored at n nodes, with the properties that:

(a) Encoded fragments can be repaired directly from other subsets of encoded fragments without having to reconstruct first the original data. More precisely, based on the analogy with the error correction capability of erasure codes, which is of any $n - k$ losses independently of which losses,

(b) a fragment can be repaired from a fixed number of encoded fragments, the number depending only on how many encoded blocks are missing and independent of which specific blocks are missing.[8]

In Self-repairing code as the simulation time increases the latency will decreases. It have a advantage as high throughput and faster recovery.

Performance Parameters	Simulation Time (ms)		
	100	150	200
Bandwidth Consumption(Hz)	56.096	56.098	76.9565
Throughput(Mbps)	18852.8	28278.8	51727.4
Packet Delivery Ratio	0.84935	0.84966	0.85602
Latency(ms)	0.29160	0.17496	0.11664
Recovery Speed(Mbps)	1.50566	1.50597	1.50633

Table 3 Performance Parameters of Self-Repairing Code

VI. RESULTS AND CONCLUSION

For the analyzing of evaluation of different performance parameters, a simulated model is developed with ceph cluster for distributed storage area network based on NS-3 simulator.[11][12][13]

Parameter Assumptions:

1. Simulation Tool : NS3
2. Graph evaluation : Xgraph
3. Area of Experiment: 5000x5000
4. No of Nodes: 20
5. Packet Size: 400 bytes
6. Simulation Time: 100 ms, 150 ms, 200 ms

Performance Parameters	**Network Codes**		
	Erasure Code	**Regenerative Code**	**Self- Repairing Code**
Bandwidth Consumption	**93.4933**	**74.7946**	**56.096**
Throughput	**1914.74**	**15812.1**	**18852.8**
Packet Delivery Ratio	**0.449333**	**0.549339**	**0.849359**
Latency	**0.800182**	**0.359548**	**0.29160**
Recovery Speed	**1.44933**	**1.50456**	**1.50566**

Table 4 Comparison of results of Network codes

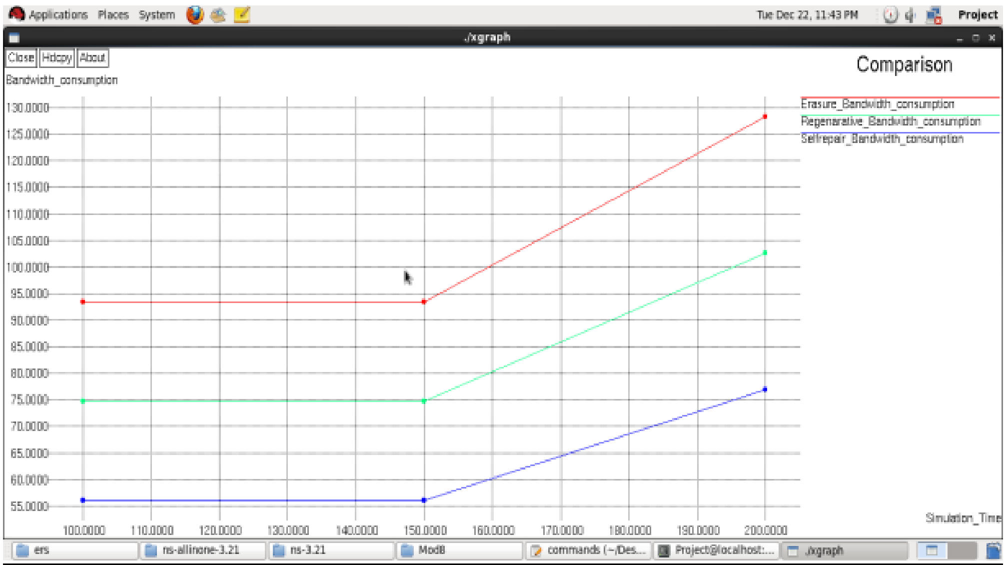

Fig 2 Comparison of Bandwidth Consumption of Erasure Code, Regenerative code and Self-Repairing code Self-repairing codes have the least bandwidth need for storage and also for repairs, regenerative code requires more bandwidth than self-repairing code but less than Erasure code. As the simulation time increases the bandwidth consumption is more.

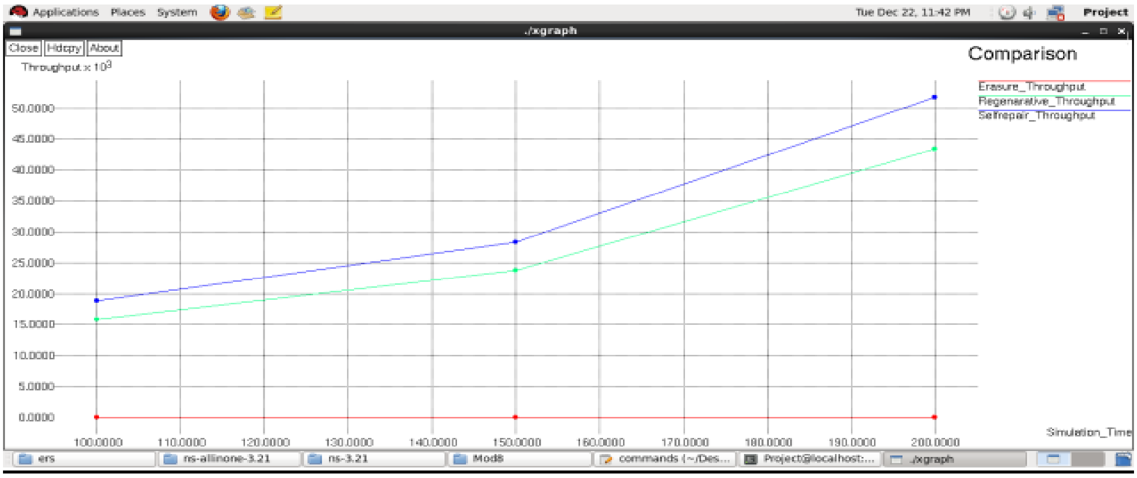

Fig 3 Comparison of Throughput of Erasure Code, Regenerative code and Self-Repairing code

Actual rate that information is transferred is more in self-repairing code. Very lower throughput in erasure code.

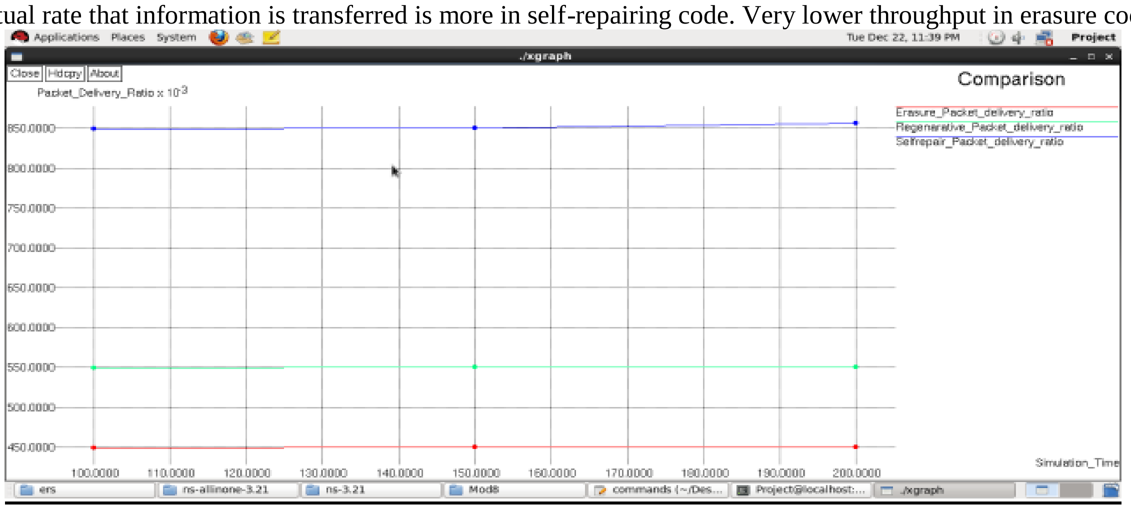

Fig 4 Comparison of Packet Delivery Ratio of Erasure Code, Regenerative code and Self-Repairing code

In self-repairing code, data packets received by the destinations to those generated by the sources are more.

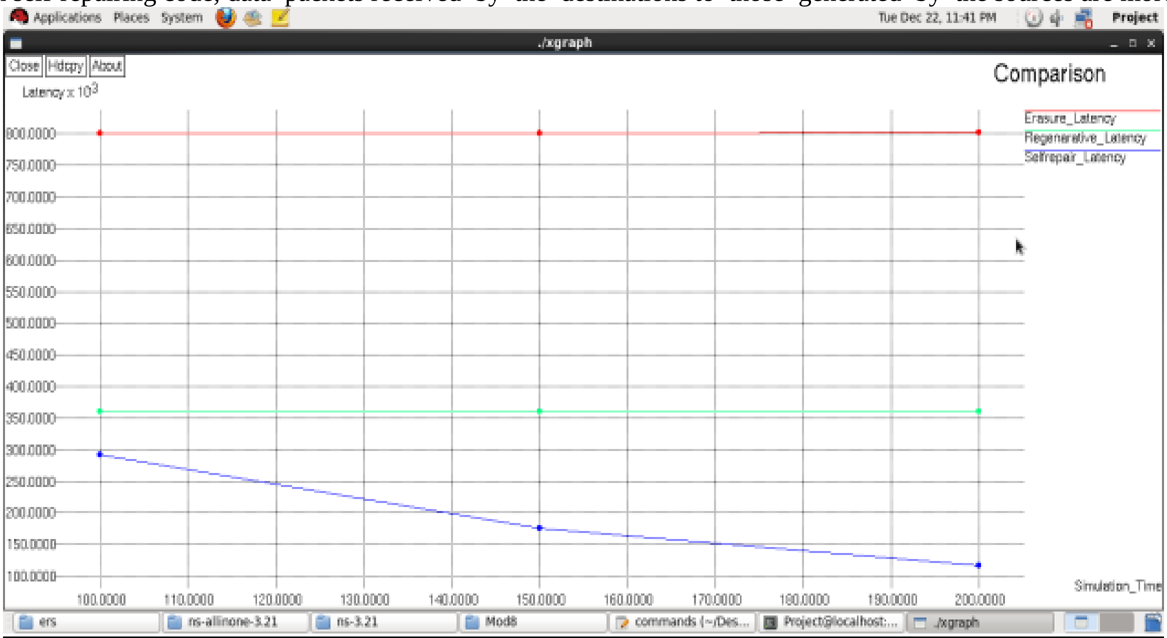

Fig 5 Comparison of Latency of Erasure Code, Regenerative code and Self-Repairing code

Erasure coding have a high delay. As the simulation time increases latency in self-repairing code decreases.

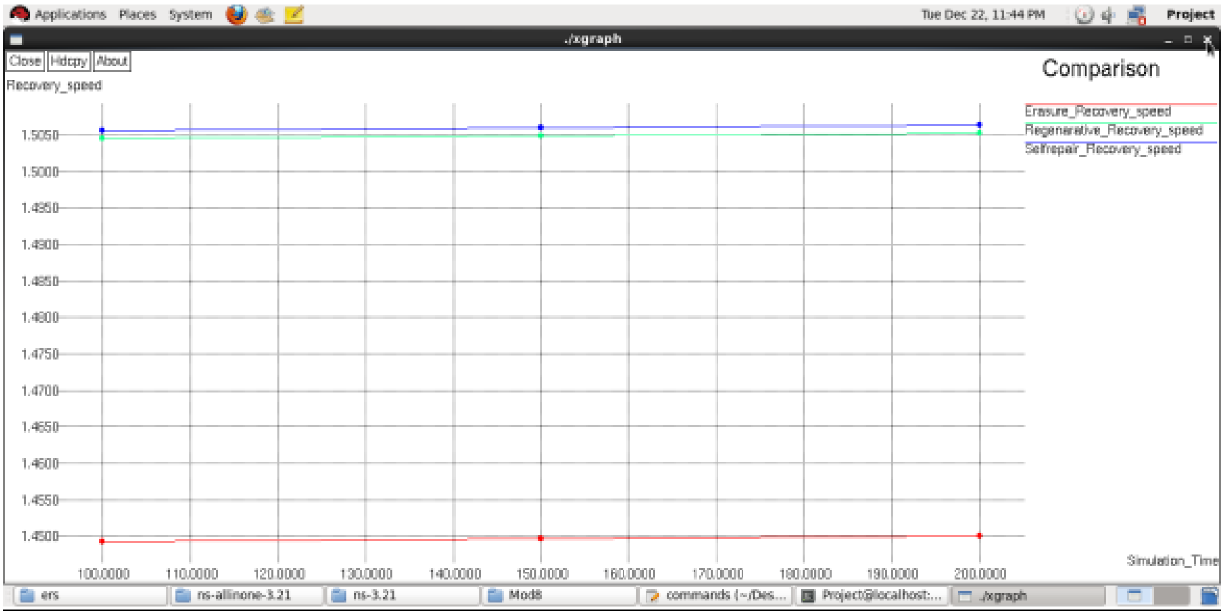

Fig 6 Comparison of Recovery Speed of Erasure Code, Regenerative code and Self-Repairing code Self-repairing codes can support fast and parallel repairs while dealing with a much larger number of simultaneous faults than regenerative code and erasure code.

Hence, it is observed that self-repairing code is the best network code amongst the other one in order to evaluate and then enhance the performance of distributed storage area network in the system.

REFERENCES

[1]. Muriel Medard and Alex Sprintson, *Network Coding Fundamentals and Applications*, Released: November 11, 2011, ISBN-10: 0123809185, ISBN-13: 978-0123809186

[2]. https://www.techopedia.com/definition/14833/network-coding as accessed on 8/8/2015.

[3]. R. W. Yeung, Z. Zhang, "Distributed Source Coding for Satellite Communications", IEEE TRANSACTIONS ON INFORMATION THEORY , VOL. 45, NO. 4, MAY 1999.

[4]. Ceph Documentation Installation http://docs.ceph.com/docs/master/start/quick-start-preflight/

[5]. Alexandros G. Dimakis, Kannan Ramchandran, Yunnan Wu, Changho Suh, "A Survey on Network Codes for Distributed Storage" Proceedings of the IEEE, Vol.99,No.3,March2011.

[6]. Alexandros G. Dimakis, P. Brighten Godfrey, Yunnan Wu, Martin J. Wainwrigh, Kannan Ramchandran, "Network Coding for Distributed Storage Systems", IEEE TRANSACTIONS ON INFORMATION THEORY, VOL. 56, NO. 9, SEPTEMBER 2010

[7]. RICKARD BROMAN, "A practical study of Network Coding in Distributed Storage Systems", Stockholm, Sweden 2013

[8]. Frederique Oggier, Anwitaman Datta, "Self-repairing Homomorphic Codes for Distributed Storage Systems" arXiv:1008.0064v1[cs.DC] 31 Jul 2010.

[9]. Bo Chen, Reza Curtmola, Giuseppe Ateniese, Randal Burns, "Remote Data Checking for Network Coding-based Distributed Storage Systems"

[10]. Benjamin Depardon, Gael Le Mahec, Cyril Seguin, "*Analysis of Six Distributed File Systems*", hal-00789086, 15 Feb 2013

[11]. Atta Rehman Khan, Sardar M. Bilal, Mazliza Othman "A Performance Comparison of Network Simulators for Wireless Networks", 30 E-28911 Leganés

[12]. Suraj G. Gupta, Mangesh M. Ghonge, Parag D. Thakare, Dr. P. M. Jawandhiya, "Open-Source Network Simulation Tools: An Overview" *International Journal of Advanced Research in Computer Engineering & Technology* (IJARCET) Volume 2, Issue 4, April 2013, ISSN: 2278 – 1323

[13]. https://en.wikipedia.org/wiki/Network_performance