

A Survey on Regenerating Codes for Distributed Cloud Storage

Sophia S
Mtech(qip), student
Computer Science and engineering
RVCE, Bangalore -59
India

Dr. Sharvani G S
Associate Professor
Department of CSE
RVCE, Bangalore-59
India

Abstract - Cloud computing is a large group of interconnected remote system via the internet. Cloud computing provide services like data storage, software and infrastructure. When we outsource storage as a service there is possibility of data loss. To provide fault tolerant in cloud storage it is necessary to strip data in multiple cloud server and need to reconstruct the lost data from other existing cloud server to preserve data redundancy is known as repair operation. The repair is performed using the regenerating codes. Main objective is to minimize the repair traffic, since the users will be charge for the outbound data from cloud. In this paper we discuss about the various regenerating codes in order to improve storage efficiency and low repair traffic compare with traditional erasure code (eg. Reed Solomon codes).

Keywords: Multiple cloud storage; Erasure coding; regenerating code

INTRODUCTION

A cloud is large group of interconnected personal computers or network servers which can be public or private. The use of cloud computing in many organizations as increased rapidly. Use of Cloud computing provides low cost and accessibility of data [1]. Cloud Computing Technologies are categorized into four different services as shown in fig 1. They are DSaaS (Data Storage as a Service), SaaS (Software as a Service), IaaS (Infrastructure as a Service) and PaaS (Platform as a Service). [1]

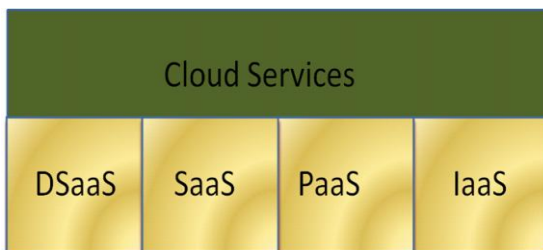


Fig 1. Services provided by Cloud Computing

Cloud storage means massive clusters of server in large data centers for processing and storing data. Service Providers, such as Amazon and Google create a virtual server by maintain and running the software and database for clients to process and store their data.

The Cloud storage can also be considered as the virtual storage area that combines many different physical storage

devices. While using cloud storage, some of the files may be on a physical server in china while other files are on a physical server in India. Therefore many users may not be aware of where their physical files are stored, using the cloud storage can be consider as a vague, thing which are not touchable – most like a cloud itself! [1].

Most of the data is accessed through the internet when it becomes the part of cloud is it is not stored on the personal machine. Example the people can access their email anywhere via internet connection via email services like Google, yahoo etc, where the data is stored on server owned by email providers ie, mail is in cloud but not on local machine. The **data centers** are that store the physical equipment used by the cloud. It can be anywhere in the world [1]. Cloud storage has three operation Upload – Files is taken from a machine which is local and sending it to cloud storage. Download – Files is taken from cloud storage and bring it to your local machine. Server- is a computer that provides service to requests of user via a computer network. [1].The important use of cloud storage is online backup and drawback is bandwidth cost.

The “single cloud” storage providers are become less popular among customers due to risks of failure of service availability and loss of data and also single point of failures [2] and vender lock-ins [3]. To overcome the problem of single cloud provider, the data is striped across the multiple cloud providers as in [2][3][4][5]. Distributed cloud storage improves the fault tolerant of cloud storage with sufficient redundancy as shown in fig 2. The normally redundancy is just the replication of data is stored in order to have data integrity but this reduces the storage efficiency. But in case of erasure code Reed Solomon has less storage overhead than the replication with same faults tolerant. [6]

The striping of data in distributed cloud storage using the conventional erasure coding can still leads to two types of failures like permanent and transient failure, the permanent failure where the data stored in failed storage node is permanently lost or unavailable[14]. Therefore it is necessary to perform the repair operation to maintain the data redundancy, where repair is to reconstruct the lost data in failed node from other surviving

nodes. With convention erasure code repair operation reconstruct the original lost data from failed node and stores in new node but it does not consider the bandwidth and its cost. The amount of data moved between the cloud matters and it should be less. To reduces the repair bandwidth, the proposed regenerating codes [7] that stores the encoded chunk of linearly combined data in distributed cloud storage. In this paper the variety of regenerating code are discussed to minimize the repair traffic.

I. RELATED WORK

A. MULTIPLE CLOUD STORAGE

The Single cloud service provider is a cloud computing implementations that deliver services through a website Amazon Web Services such as storage and computation services is provided. Cloud service providers online data sharing through network facilities can leads to loss of privacy[8][9][10]. According to a [11] Protecting important private information from attackers and malicious insiders is of critical task. Moving database to a large data centre involves many tasks such as Virtual, accessibility, privacy and data accessed from a third party may lead to loss of data integrity, confidentiality of data, and loss of data. Single cloud storage always leads to many problems [8][9][11]. Hence instead of using single cloud, the data can be striped across the multiple clouds [2][3][4]. Thus improve the fault tolerant of cloud storage. The migration of cloud computing from single toward multi-clouds to ensure the security of user's data and to prevent loss of data has been extremely important. The term "multi-clouds" is similar to the terms "intercloud" or "cloud-of-clouds".

Multiple-cloud storage has been proposed by several systems. In paper [4] the integrity and availability for stored data is guarantee by HAIL. In paper [2] the erasure coding is used solve vendor lock-ins problems when switching cloud vendors by RACS, it retrieves data from the cloud failed node and the data is moved to the new cloud. In paper[12] the design for intercloud storage (ICStore), is proposed which is a similar than RACS and dependable service are in multiple clouds in HAIL and the theories and protocols are developed to address the confidentiality, integrity, reliability and consistency of the data stored in clouds .But in NCCloud, the failed node is excluded in repair. In paper [13] the byzantine fault tolerance is provided by vukoli uses multiple independent clouds. In paper [5], the encryption and erasure coding are combined to provide Byzantine fault tolerance for the stored data (DEPSKY). Nccloud goes one step ahead by using the regenerating code to provide fault tolerant and repair as in fig2.

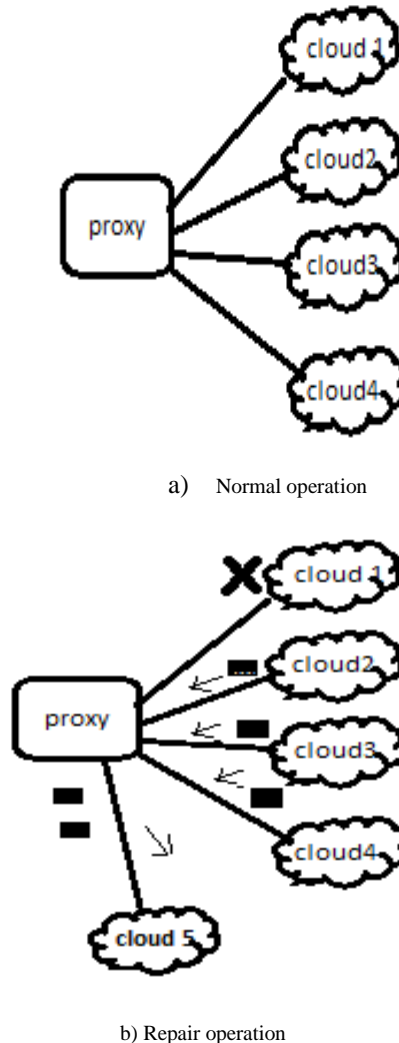


Fig 2.The multiple-cloud storage design that uses proxy server: (a) normal operation (b) Repair operation during cloud 1, the data is regenerated by proxy server and stores in new cloud. [14]

B. Erasure Code In Cloud Storage

In a storage system the data are stored on a large number of commodity disks and there is a possibility of disk failure. Therefore, it is necessary to store redundant data in storage system in such a way that when a data is lost in certain disk, still the data can be accessed from other disks. For example, the N -way replication, where N replicas are stored in N different disks, can tolerate at most N - 1 disk failures [15]. The simplest form of data redundancy is replication, with one or more copies of an "original" data and available if that original gets lost. When the algorithms that is used for RAID has been applied cloud storage: they are called as erasure codes

Erasure coding is able to tolerate the disk failures as same replication, but with better storage efficiency. Erasure coding based on Maximum Distance Separable (MDS) codes (e.g., Reed-Solomon codes [16]) achieves optimal storage efficiency. In the storage system, the file or a fix-sized block distributed on different storage systems. Suppose if the data are stored onto n disks then (n, k) MDS codes can able to tolerate at most $n - k$ disk failures given any arbitrary number k where $k < n$, i.e., to access any part of the original data any k disks are enough, where the data is encoded into n coded chunk and are uniformly distributed into the n disks.

The access latency of the storage system can be improved by using a cache to store one replica of the original data [17]. Recovering the whole data for data access may be reasonable in many cases. But from the point of the storage system, it is not necessary to recover the whole data when we need to repair one only lost coded block ie small portion of data.

Before the regenerating codes appeared, all MDS codes require accessing at least k disks to repair one disk only, but under replication, to repair one replica we transfer one replica. This requirement can generally increase bandwidth in a data center, and the performance of the storage system is affected and other applications in the cloud.

The two main drawbacks erasure coding is to read or write data, the encoding and decoding of data, leads to high access latency and a low access throughput because of CPU limit. The erasure coding stores multiple coded chunks, when one coded chunk is lost, the system access multiple coded blocks that are enough to recover all the data. This makes the repair very expensive in terms of both bandwidth and disk I/O overhead with erasure coding. The applications hosted in the cloud are sensitive to bandwidth and always limits the resource inside the data center because most data centers introduce oversubscription link [18]. The code that has the min-cut in the information flow graph is called regenerating codes. The storage and bandwidth are important in Cloud Storage. They are two special types of regenerating codes are the storage nodes with minimum storage requirement and the minimum bandwidth usage during repair is called Minimum-Storage Regenerating (MSR) codes. Many paper extensively study on erasure codes in [19]-[21] sparse graph achieve optimal reliability and redundancy and low decoding and encoding operation. In [22]-[25] parity array codes based on XOR operation mainly have low encoding and decoding. The erasure codes that we consider provide a tradeoff between reliability, redundancy and repair bandwidth.

II. REGENERATING CODES

A. Information flow graph

To determine the bandwidth consumption with erasure coding during repair operation, the information flow graph was proposed by Demakis et al.[26], which is used as a tool in the analysis of network coding, as a model to characterize the storage and bandwidth tradeoff.

As shown in Fig. 3, in the information flow graph, all servers can be classified as the source, storage nodes,

and the data collector. The server where the original data is originated is called Source. If the size of the original data is M bits and the coded blocks of α bit after the encoding are distributed into n storage nodes. Each vertex in graph is represented as source and a storage node in graph. These two vertices are connected by edge with weight corresponds to the amount of data stored in the storage node. All n storage nodes stores α bit after dissemination recover the original data by connecting any k storage nodes. During the failure of storage node, a new node does not connect to k available storage nodes, also storage nodes d as providers ($d \leq k$).

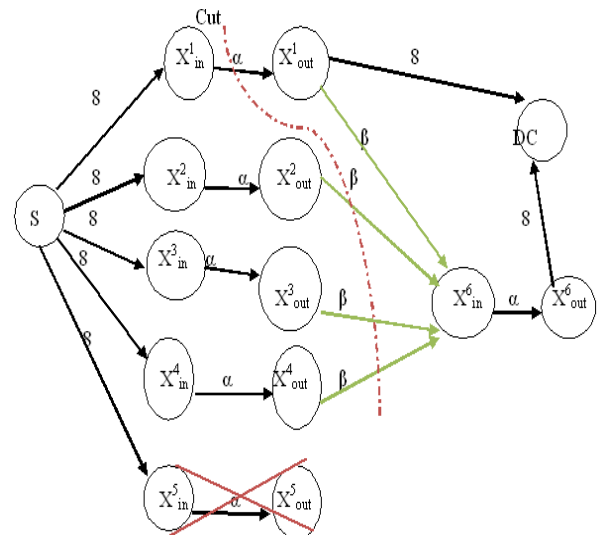


Fig.3 An illustration of the information flow graph that corresponds to $(4,2)$ MDS codes[28].

Compare to conventional MDS codes, from each provider, the new node can receive β bits, represented weighted edge $\beta \leq \alpha$ between the provider and the newcomer. After when a total number of bits $r = d\beta$ received from providers, the bits are stores by new node that becomes a new storage node. The MDS property after every round of repair has to check that any k coded blocks enough to recover the original data. The data collector connects storage nodes where data directly received from the source, as well as the storage nodes repaired from newcomers in repairs. The weights of edges connected to the source or the data collector are all set to be infinity.

Therefore, in the storage system the repair problem can be consider as a multicast problem in the information flow graph, where data is multicast by the source to all possible data collectors. In the multicast problem the minimum cut that separates the source from any receivers equal to the maximum multicast rate and the network coding [27] is used to achieve this rate. Therefore, it the MDS property have been proved that its holds good after any rounds of repairs, the size of the original data is less than the min-cut in the information flow graph and can obtain tradeoff between α bit and bandwidth consumption.

B. Minimum-Storage And Minimum-Bandwidth Regenerating Codes

The regenerating codes are the code that achieves the min-cut in the information flow graph. There are two types of regenerating code based one minimum storage requirement space at storage nodes and the minimum bandwidth consumption during repair.

Regenerating codes are based on the concept of network coding [28] and tradeoff the repair traffic is reduced among storage nodes. The storage cost and repair traffic achieve optimal, and the optimal points are two, one optimal point to the minimum storage regenerating (MSR) codes, focus on minimize the repair bandwidth the condition that each node stores the minimum amount of data as in Reed-Solomon codes. Another optimal point is the minimum bandwidth regenerating (MBR) codes, which minimize the repair bandwidth further allowing each node to store more data. The MBR codes construction is discussed in [29], where as MSR codes based on interference alignment is discussed in [30], [31]. In this work, we focus on the MSR codes.

C. Exact Regenerating Code

The exact regenerating codes are constructed using important tool is interference alignment, which is proposed for wireless communication initially. The vectors which are undesired can be eliminated by aligning them onto the same linear subspace. In Fig. 3 MSR codes where data are encoded ($n=4, k=2, d=3$), i.e., data is stored in four cloud such that during failure any two of the four nodes can reconstruct the original file. In each of the storage node, contains two coded segments, such as (A1, A2). If this storage node fail then to reconstruct A1 and A2 from the failed node, the newcomer download half of coded block from 3 storage nodes as providers i.e., to achieve the bandwidth become more complicated by downloading a coded segment from each provider. The B1 and B2 provided by provider are unwanted and can be eliminated by aligning into same linear subspace, decoding of A1 and A2 can be done solving three equations.

The explicit construction of scalar exact MSR codes was proposed in Suh and Ramchandran[32] such as $d \geq 2k-1$, size at least $2(n-k)$ over a finite field .

Product-Matrix construction improves the choices of parameters such that $d \geq 2k-2$ by using the, with a larger finite field of size at least $n(d-k+1)$ in Rashmi et [9]al,

In Ref [34], there no scalar exact MSR codes exist is proved when $d < 2k-3$. In ref [32]and [35] showed that is it not possible to achieved asymptotically when constructing vector exact MSR codes for any choices of (n, k, d) as the field size goes to infinity. It is impractical to constructing regenerating codes on a very large finite field due to overhead of encoding/decoding. Since the exact repair makes much more suitable for the systematic part. The exact code repairs the systematic part.

D. Repair-By-Transfer Regenerating Codes

The above regenerating codes require encoding their data by the provider to align vectors in a specific linear subspace and to eliminate undesired components, the newcomer has to encode received data and if the field size is larger then it is expensive to use the arithmetic operations performed on a finite field. Thus, the repair-by-transfer property can be used in the cloud storage systems. In repair by transfer regenerating code require no arithmetic operations at providers. With the repair-by-transfer property, the only data is read from provider that is needed by the newcomer. To construct the corresponding repair-by-transfer regenerating codes there is some choices of parameters. The exact MBR codes can be constructed over a finite field of size 2 and has a problem with the choices of parameters. The linear exact MBR codes cannot achieve the repair-by-transfer property [28]. Nevertheless, Rashmi et al.[29] proposed the construction of repair-by-transfer exact MBR codes based on an intuitive graph, where repairing any missing block from its direct neighbors in the graph. Fractional repetition codes are used in the construction graph where the direct neighbors share the same segment and the newcomer repair itself by only retrieving replicas of the segments from its neighbors, where no arithmetic operations are required at the provider and newcomer. With a wide range of parameter choice the MDS and storage efficiency can be sacrificed.

E. Functional Minimum Storage Regenerating Code

FMSR codes based on type of maximum distance separable (MDS) codes with the parameter (n,k), where $k < n$. An (n,k) MDS code property is defined as the original file can be reconstructed from any k out of n . An additional feature of FMSR codes is that a particular piece of file can be reconstructed from data of size less than M . FMSR codes is a regenerating code based on concept of network coding [28], minimizing the repair traffic and preserves the MDS property.

The distributed storage setting is considered as in fig 2 where a file is striped over n cloud server using an FMSR code. Each server can be a cloud storage provider or storage site, and is it independent of each other. An (n,k)-FMSR code splits a file of size evenly into $k(n-k)$ native chunks F_i , where $i=1,2,\dots,k(n-k)$ and these native chunk are linearly combined to form $n(n-k)$ coded chunk P_i where $i=1,2,\dots,n(n-k)$ and encodes them using Galois Field(2^8) in ref[29], where each native and code chunk has size $|M| / k(n-k)$. The ($n-k$) code is stored in each cloud server and $n(n-k)$ coded chunk is stored in n cloud server. The original file can be constructed by downloading $k(n-k)$ chunk from any k cloud server Decoding can be done by inverting the encoding matrix [14]. FMSR codes are nonsystematic codes that does not keep native chunks instead coded chunk is stored. FMSR codes are mainly used for long-term applications, where there is low read frequency and in order to have backup we need to retrieve entire file rather than chunk [14].

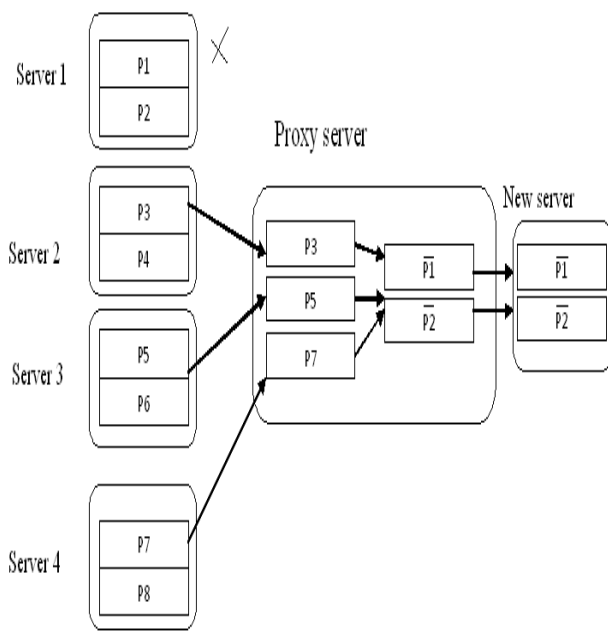


Fig 4 Example of how a file is repaired in (4,2)- FMSR codes. Each of the code chunks [14[]] P1,.....,P8 is a random linear combination of the native chunks 1 and are distinct random linear combinations of P3, P5 and P7

III. CONCLUSION

In this paper, the various regenerating codes for cloud storage systems overview is given. The exact regenerating codes is systematic erasure codes in the storage system because it stores the data as it is such that original file can recovered with no decoding operations. The repair-by-transfer regenerating code is used with no arithmetic operation is performed on both the newcomer and the providers during repair operation. The FMSR codes which regenerate the data across a multi-server and two-phase checking are performed on the new code chunks generated in the repair operation. These codes reduce the bandwidth consumption of the repair operations.

REFERENCES

- (1) Konwinski, G. Lee, D. Patterson, A. Rabkin, I. Stoica, and M.Zaharia, "A View of Cloud Computing" Comm. the ACM, vol. 53,no. 4, pp. 50-58, 2010.
- (2) Abu-Libdeh, L. Princehouse, and H. Weatherspoon, "RACS: A Case for Cloud Storage diversity," Proc. ACM First ACM Symp. Cloud Computing (SoCC '10), 2010.
- (3) A.bessani ,M.correia,B. Quaresma ,F.Andre, and P.sou sa "DEPSKY: Dependable and Secure Storage in a Cloud-of-Clouds," Proc. ACM European Conf. Computer Systems (EuroSys '11), 2011.
- (4) K.D. Bowers, A. Juels, and A. Oprea, "HAIL: A High-Availability and Integrity Layer for Cloud Storage," V Proc. 16th ACM Conf. Computer and Comm. Security (CCS '09), 2009.
- (5) M. Vukoli_, "The Byzantine Empire in the Intercloud," ACMc SIGACT News, vol. 41, pp. 105-111, Sept. 2010.
- (6) H. Weatherspoon and J.D Kubiawicz, "Erasure coding versus replication : A Quantitative Comparison, " in Proc . Int. Workshop peer to peer syst., 2002
- (7) K. Ramchandran, Y. Wu, and C. Suh, "A Survey on Network Codes for distributed Storage" , Proc. IEEE, vol. 99, no. 3, pp. 476-489, Mar. 2011.
- (8) H. Blodget, "Amazon's Cloud Crash Disaster Permanently Destroyed Many Customers' ,Data,," <http://www.businessinsider.com/amazon-lost-data -2011-4/>, Apr. 2011.
- (9) E. Naone, "Are We Safeguarding Social data?" <http://www.technologyreview.com/blog/Editor/22924/feb.2009>
- (10) R. Wauters,, "Online Backup Company Carbonite Loses Customers' Data, Blames and Sues Suppliers,," <http://techcrunch.com/2009/03/23/online -backup -company- carbonite- loses-customers- data-blames-and-sues-suppliers/>, Mar. 2009.
- (11) C. Preimesberger, "Many Data Centers Unprepared for Disasters: Industry Group," - <http://www.eweek.com/c/a/IT-Management/Many-Data-Centers-Unprepared-for- Disasters Industry-Group-772367/>, Mar. 2011.
- (12) R.Rhea ,P.Eaton, D.Geels, H.Weatherspoon ,B. Zhao and J.Kubiawicz, "pond: "The OceanStore prototype," presented at the USENIX File and Storage echnologies(FAST) ,2003.
- (13) M. Vukoli_, "The Byzantine Empire in the Intercloud," ACMc SIGACT News, vol. 41, pp. 105 File and s-111, Sept. 2010.
- (14) Y. Hu, H. Chen, P. Lee, and Y. Tang, —NCCloud: Applying Network Coding for the Storage Repair in a Cloud-of-Clouds,IProc. 10th USENIX Conf. File and Storage Technologies (FAST '12), 2014.
- (15) V. Anto Vin , S.Umamageshwari ,P.Saranya, " A Survey on regenerating codes " , volume 4,issue 11, november 2014.
- (16) J.S. Plank, "A Tutorial on Reed-Solomon Coding for Fault-Tolerance in RAID-Like Systems," Software—Practice & Experience, vol. 27, no. 9, pp. 995-1012, Sept. 1997.
- (17) C. Huang, H. Simitci, Y. Xu, A. Oguş, B. Calder,P. Gopalan, J. Li, and S. Yekhanin, Erasure coding in windows azure storage, in Proc. USENIX Annual Technical Conference (USENIX ATC), 2012.
- (18) M. Al-Fares, A. Loukissas, and A. Vahdat, A scalable,commodity data center network Architecture, in Proc. ACM SIGCOMM Conference on Data Communication IGCMM), 2008.
- (19) M. LUBY ,M .mitzenmacher, M. A. shokrollahi, and D. spielman, "Im-proved low-density Parity check codes using irregular graphs," IEEE trans, inf .theory,vol.47, pp.585-598, Feb.2001
- (20) M. ludy, "LT codes, "presented at the proc. IEEE foundations of computer science (FOCS), 2002.
- (21) A. shokrollahi, "Raptor codes," IEEE Trans, inf. Theory, vol. 52, no.6, pp. 2551-2561, Jun, 2006.
- (22) M. blaum, J. brady, and J. menon, "EVENODD: An Efficient scheme for tolerating double Disk failures," IEEE trans. Comput., vol. 44, pp. 192-202, feb. 1995.
- (23) L. Xu and J. Bruck, "X-code:MDS array codes with optimal encoding," IEEE Trans. Inf. Theory, vol. 45, no. 1, pp. 272-276, jan. 1999.
- (24) C. Huang and L. Xu,"START: An Efficient coding scheme for correcting triple storage Node failures," presented at the FAST-2005;4th usenix conf. file and storage technologies, san Francisco,, CA, Dec. 2005.
- (25) J. L. Hafner, "WEAVER codes: highly fault tolerant erasure codes for storage systems," presented at the FAST-2005;4th usenix conf. file and storage technologies, san Francisco, ca, dec. 2005.
- (26) A. G. Dimakis, P. B. Godfrey, M. J. W. Y. Wu, and K.

- Ramchandran,” Network coding for distributed storage system “, IEEE Trans. Inform. Theory, vol. 56, no. 9, pp.4539-4551, 2010.
- (27) S.-Y. R. Li, R. W. Yeung, and N. Cai, “Linear network coding “, IEEE Trans. on Inform. Theory, vol. 49, no. 2, pp. 371-381, 2003.
- (28) R. Ahlswede, N. Cai, S.-Y.R. Li, and R.W. Yeung, “Network Information Flow,” IEEE Trans. Information Theory, vol. 46, no. 4, pp. 1204-1216, July 2000.
- (29) K.V. Rashmi, N.B. Shah, P.V. Kumar, and K. Ramchandran, “Explicit Construction of Optimal Exact Regenerating Codes for distributed Storage”, Proc. Allerton Conf., 2009.
- (30) K. Rashmi, N. Shah, and P. Kumar, “Optimal Exact-Regenerating Codes for Distributed Storage at the MSR and MBR Points via a Product-Matrix Construction,” IEEE Trans. information Theory, vol. 57, no. 8, pp. 5227-5239, Aug. 2011.
- (31) C. Suh and K. Ramchandran, “Exact-Repair MDS Code Construction Using Interference Alignment”, IEEE Trans. Information Theory, vol. 57, no. 3, pp. 1425-1442, Mar. 2011.
- (32) D. S. Papailiopoulos and A. G. Dimakis, “Distributed storage codes through hadamard designs”, in Proc. IEEE International Symposium on Information Theory (ISIT), 2011, pp. 1230-1234.
- (33) N. B. Shah, K. V. Rashmi, P. V. Kumar, and K. Ramchandran, “Interference alignment in regenerating codes for distributed storage: Necessity and code constructions “, IEEE Trans. on Inform. Theory, vol. 58, no. 4, pp. 2134-2158, 2012.
- (34) V. R. Cadambe, S. A. Jafar, and H. Maleki, “Distributed data storage with minimum storage regenerating codes – Exact and functional repair are asymptotically equally efficient”, <http://arxiv.org/abs/1004.4299>, 2010.
- (35) K. Rashmi, N. Shah, and P. Kumar, “Optimal exact regenerating codes for distributed storage at the MSR and MBR points via a product-matrix construction “, IEEE Trans. on Inform Theory, vol. 57, no. 8, pp. 5227-5239, 2011.