Minimum Bandwidth Regenerating Codes Based on Cyclic VFR Codes

Jing Wang*, Shuxia Wang, Tiantian Wang, Xuefei Zhang
School of Information Engineering, Chang'an University
Xi'an, Shaanxi 710064 - China
[e-mail: jingwang@chd.edu.cn, 2016124007@chd.edu.cn, 2016124016@chd.edu.cn, 2016124002@chd.edu.cn]
*Corresponding author: Jing Wang

Received June 13, 2018; revised August 11, 2018; revised November 21, 2018; accepted February 14, 2019; published July 31, 2019

Abstract

In order to improve the reliability and repair efficiency of distributed storage systems, minimum bandwidth regenerating (MBR) codes based on cyclic variable fractional repetition (VFR) codes are constructed in this thesis, which can repair failed nodes accurately. Specifically, in order to consider the imbalance of data accessed by the users, cyclic VFR codes are constructed according to that data with different heat degrees are copied in different repetition degrees. Moreover, we divide the storage nodes into groups, and construct MBR codes based on cyclic VFR codes to improve the file download speed. Performance analysis and simulation results show that, the repair locality of a single node failure is always 2 when MBR codes based on cyclic VFR codes are adopted in distributed storage systems, which is obviously superior to the traditional MBR codes. Compared with RS codes and simple regenerating codes, the proposed MBR codes based on cyclic VFR codes have lower repair locality, repair complexity and bandwidth overhead, as well as higher repair efficiency. Moreover, relative to FR codes, the MBR codes based on cyclic VFR codes can be applicable to more storage systems.

Keywords: Distributed storage systems, minimum bandwidth regenerating codes, variable fractional repetition codes, repair locality

This research was supported by the Special Fund for Basic Scientific Research of Central Colleges, Chang'an University (300102248104, 300102248201, 300102248401), the Natural Science Foundation of Shaanxi Province (2018JM6081).

http://doi.org/10.3837/tiis.2019.07.014 **ISSN : 1976-7277**

1. Introduction

In the era of digital information, massive data storage and its reliability are the key problem that we need to solve urgently. Distributed storage, which has been increasingly deployed and gradually replaced centralized storage, shares the storage load effectively, and has lower cost and good scalability by scattering huge amounts of data on multiple physical storage devices [1, 2]. Distributed storage systems (DSSs) are composed of many cheap storage devices, which inevitably lead to node failure and result in data loss. In order to ensure the reliability and availability of data storage, replication strategy and erasure codes have been more widely adopted in many current DSSs [3-6]. For example, Google File System (GFS) and Hadoop Distributed File System (HDFS) adopt multi-replication [7, 8]. However, since multi-replication needs to store a large number of data to ensure high reliability, its storage cost is high. Erasure codes need to be encoded and decoded, and their calculation complexity is high. In addition, adopting erasure codes, the entire file needs to be downloaded during the repair process of failed nodes, and their repair bandwidth overhead is also large.

Towards the problems above, Dimakis et al. used network flow graph to represent DSSs [9], and further proposed the concept of regenerating codes based on network coding [10, 11]. Rashmi et al. gave the storage-bandwidth overhead curve, as well as the regenerating codes that reach two optimal limit points, i.e., minimum storage regenerating (MSR) codes and minimum bandwidth regenerating (MBR) codes. Regenerating codes reduce the bandwidth overhead by transmitting linear combinations of multiple data [12-14]. Dimakis et al. have proved that the receiver can recover the original file when the maximum-flow minimum-cut of the network flow graph is greater than or equal to the original file size. Regenerating codes significantly reduce the bandwidth overhead of the failed node repair, but can't achieve low disk I/O overhead, and have higher computational complexity. Papailiopoulos et al. proposed simple regenerating codes (SRC) by combining erasure codes with XOR operation [15], which can quickly repair a single failed node by accessing a small number of nodes. The number of surviving nodes connected in the repair process is small as using locally repairable codes (LRC) [16], that is, LRC has small repair locality and lower repair bandwidth, but its repair complexity is high.

While ensuring low disk I/O overhead, Rouayheb and Ramchandran proposed fractional repetition (FR) codes to reduce the computational complexity of repairing failed nodes [17]. Then, a series of FR codes for heterogeneous distributed storage systems are proposed, such as FR codes for different node capacities [18, 19]. However, the number of nodes connected to the surviving node when repairing a single failed node is determined by the number of data blocks stored in the failed node. For improving the problem above, Nam et al. proposed locally FR codes [20], that is, the number of nodes connected when repairing a failed node is smaller than the number of data blocks stored in the node. Locally FR codes have lower disk I/O overhead compared with FR codes, but are not applicable for actual DSSs. Considering that

the access of user to data is often unbalanced, that is, “hot” data is often accessed, “cold” data is rarely accessed [21, 22], Li et al. proposed variable fractional repetition (VFR) codes [23]. Although VFR codes take into account the imbalance of user access to data, the number of surviving nodes that need to be connected when repairing a failed node is equal to the number of data blocks in the failed node, not making the disk I/O overhead optimal. Moreover, the VFR codes have certain limits in applicable occasions, and the number of nodes in the storage system is limited by parameters, must be some specific even.

In order to further reduce the disk I/O overhead in the process of repairing failed nodes, and improve the flexibility of VFR codes, a class of MBR codes based on cyclic VFR codes is constructed by adopting the idea that data with different heat degrees are copied in different repetition degrees. Specifically, through making repetition degree of the original data blocks larger than the repetition degrees of the parity blocks in locally FR codes, thereby cyclic VFR codes are obtained. On the basis of the constructed cyclic VFR codes, MBR codes are designed based on MDS property to improve the download speed of files by user. The MBR codes proposed have the same performance in terms of bandwidth overhead as the traditional MBR codes in the literature [14], but the advantage of repair locality is obvious, which improves the situation that the repair localities of traditional MBR codes are higher than that of the existing RS codes and SRC. Theoretical analysis shows that, the proposed codes ensure that the locally repair degree of a single node failure is always 2 while considering the imbalance of accessing data by user. Compared with SRC and RS codes, the proposed MBR codes based on cyclic VFR codes have advantages in repair bandwidth and repair speed, moreover improving the download speed of the original file.

2. MDS codes and regenerating codes in DSSs

2.1 MDS codes in DSSs

If (n, k) MDS codes are used in DSSs, a file of size M is split into k data blocks with the identical size of M/k, and encoded into n data blocks of the same size, stored in n nodes. The original file can be constructed by downloading the minimal data of size M, from any k out of n data blocks, which is the MDS property. The codes are systematic MDS codes if the generated n data blocks contain k original data blocks. Systematic (n, k) MDS codes can be expressed as

$$\mathbf{c} = \mathbf{m} \cdot \mathbf{G} \tag{1}$$

Here $\mathbf{m} = (m_0, m_1, \ldots, m_{k-1})$ is the k original data blocks contained in the original file. $\mathbf{G} = [\mathbf{I} | \mathbf{P}]_{k \times n}$ is the generator matrix of the systematic MDS codes, where $\mathbf{I}$ is a $k \times k$ identity matrix, and $\mathbf{P}$ is a $k \times (n-k)$ submatrix. $\mathbf{c} = (m_0, \ldots, m_{k-1}, p_k, \ldots, p_n)$ is the n coded data blocks generated by adopting systematic (n, k) MDS codes.

2.2 Regenerating codes in DSSs

Adopting network flow graph to represent DSSs, and further introducing the idea of network coding, Dimakis et al. proposed the concept of regenerating codes [10, 11]. Assuming that the DSS contains n nodes, it is necessary to store the original file of size M, and the storage overhead of each node is α. The new node downloads data from $d \geq k$ surviving nodes to recover the failed node, and each surviving node participating in the repair process transmits data of size $\beta \leq \alpha$ to the new born node. So the bandwidth overhead is $\gamma = d\beta$. According to the maximum-flow minimum-cut theorem, the network flow graph must satisfy the equation (2) to recover the original file, and the codes are optimal when the equal is established [11].

$$M \leq \sum_{l=1}^{k} \min\{\alpha, (d-l+1)\beta\} \tag{2}$$

When the parameters M, k, d are determined, the trade-off curve of storage-bandwidth and two limit points on the curve, i.e., MSR point and MBR point, can be obtained.

The regenerating codes, corresponding to the MSR point, are termed as MSR codes, satisfying

$$\alpha_{MSR} = \frac{M}{k}, \ \beta_{MSR} = \frac{M}{k(d-k+1)} \tag{3}$$

The regenerating codes, corresponding to the MBR point, are called MBR codes, satisfying

$$\alpha_{MBR} = \frac{2Md}{2kd-k^2+k}, \ \beta_{MBR} = \frac{2M}{2kd-k^2+k} \tag{4}$$

3. Explicit construction of MBR codes based on cyclic VFR codes

In the real system, the user only needs the original data blocks, i.e., the original file. The parity data blocks need to be downloaded only if the original data blocks are not available. Therefore, the original data blocks are hot data, and the parity data blocks are cold data. In this section, we have designed a code to consider this feature. We first construct cyclic (n, k, ρ_1, ρ_2) VFR codes with repetition degrees ρ_1 and ρ_2 respectively, where ρ_1 and ρ_2 are positive integers and $\rho_1 \neq \rho_2$. We further divide the storage nodes into groups, and design MBR codes based on cyclic VFR codes.

3.1 Construction of cyclic VFR codes

In a DSS, a file of size M is split into k data blocks with the identical size of M/k, and generates $k+3$ encoded data blocks through the systematic $(k+3, k)$ MDS codes. The systematic MDS codes can also be expressed as Eq. (1). Let α_1, α_2, α_3 be three non-zero elements in *GF(q)*. The generator matrix of systematic $(k+3, k)$ MDS code is

$$\mathbf{G}=[\mathbf{I}|\mathbf{P}]_{k\times(k+3)}=\begin{bmatrix}1&0&\dots&0&0&1&1&1\\0&1&\dots&0&0&\alpha_1&\alpha_2&\alpha_3\\\vdots&\vdots&\ddots&\vdots&\vdots&\vdots&\vdots&\vdots\\0&0&\dots&1&0&\alpha_1^{k-2}&\alpha_2^{k-2}&\alpha_3^{k-2}\\0&0&\dots&0&1&\alpha_1^{k-1}&\alpha_2^{k-1}&\alpha_3^{k-1}\end{bmatrix} \tag{5}$$

The cyclic VFR codes with different repetition degrees are constructed based on the systematic $(k+3,\ k)$ MDS codes. Firstly, the original data blocks $m_0, m_1, \dots, m_{k-1}$ are copied two times, and a total of three copies of $m_0, m_1, \dots, m_{k-1}$ are stored. The parity data blocks p_k, p_{k+1}, p_{k+2} are copied only one time, and a total of two copies of p_k, p_{k+1}, p_{k+2} are stored in the DSS. The storage structure of the corresponding cyclic VFR codes is shown in **Fig. 1**.

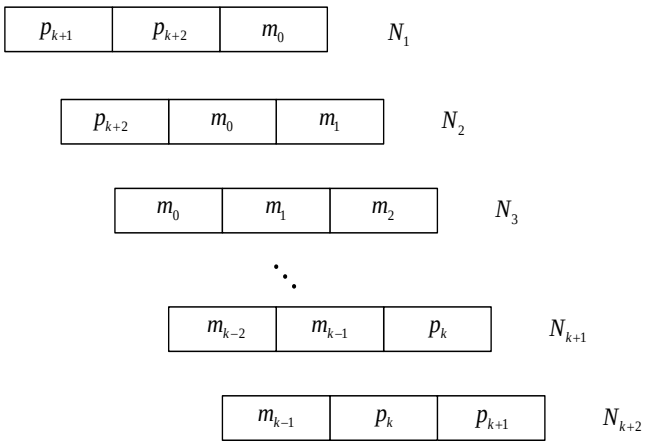


Fig. 1. Storage structure of cyclic VFR codes

From **Fig. 1**, there are three copies of the original data blocks and two copies of the parity data blocks. Each node stores 3 data blocks, and node N_i stores data blocks with indices of $i-1$, $i-2 \bmod (k+2)$, $i-3 \bmod (k+2)$ $(i \in \{1,\dots,k+2\})$. Here, adjacent nodes contain two identical data blocks.

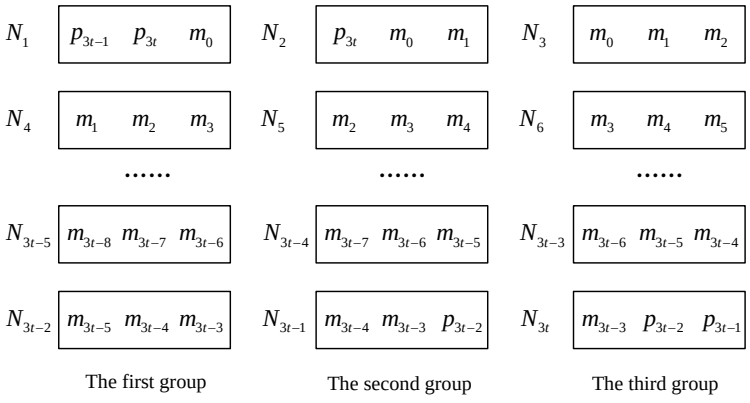
3.2 Construction of MBR codes based on cyclic VFR codes

Based on the constructed cyclic VFR codes, the construction of MBR codes based on cyclic VFR codes is studied in this section. In order to reduce the seek time for the user to download the original file, the nodes can be divided into three groups, so that the user can get the original file without accessing the redundant nodes directly in any one group of the three groups, which reduces the seek time of the nodes and realizes fast download of the files. Nodes are grouped according to the MDS characteristics of the cyclic VFR codes. Each group contains k different original data blocks. Suppose that the DSS has $n=k+2$ nodes, the specific situations are as follows:

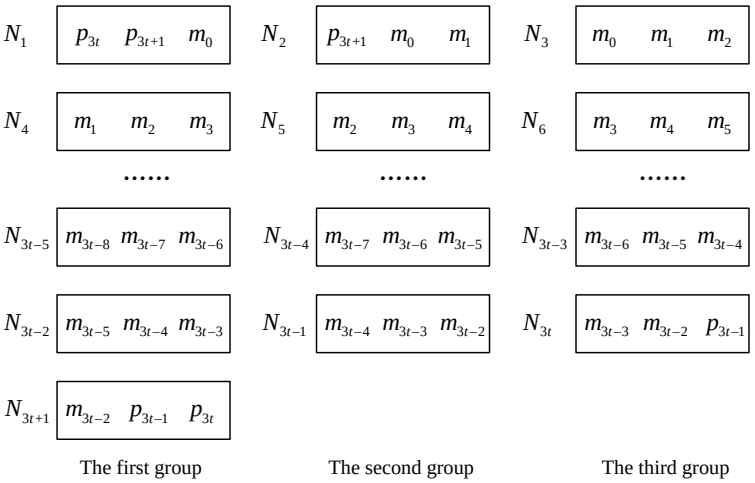
(i) $k+2=3t$ $(k\geq 2)$. The nodes of DSS are divided into three groups. The first group contains N_1, N_4, N_7,…, N_{3t-2}, abbreviated as $N_{1+3h}(0\leq h\leq t\text{-}1)$. The second group contains N_2, N_5, N_8,…, N_{3t-1}, abbreviated as N_{2+3h} $(0\leq h\leq t\text{-}1)$. The third group contains N_3, N_6, N_9,…, N_{3t}, abbreviated as $N_{3+3h}(0\leq h\leq t\text{-}1)$, as shown in **Fig. 2(a)**. Each group contains t storage nodes and stores a total of $3t$ different data blocks, in which $3t\text{-}2=k$ data blocks are the original data blocks. Therefore, users can download the original file by connecting any group of t nodes without any coding operation.

(ii) $k+2=3t+1$ $(k\geq 2)$. The nodes are divided into three groups. The first group contains N_1, N_4, N_7,…, N_{3t+1}, abbreviated as N_{1+3h} $(0\leq h\leq t)$. The second group contains N_2, N_5, N_8,…, N_{3t-1}, abbreviated as N_{2+3h} $(0\leq h\leq t\text{-}1)$. The third group contains N_3, N_6, N_9,…, N_{3t}, abbreviated as $N_{3+3h}(0\leq h\leq t\text{-}1)$, as shown in **Fig. 2(b)**. There are $3t+3$ data blocks in the first group, in which there are two data blocks with subscript of $3t$, containing $3t\text{-}1=k$ different original data blocks. Similarly, there are $3t$ data blocks in the second and third group, also containing $3t\text{-}1=k$ different original data blocks. Therefore, the users can download the original file by connecting all the nodes in any one group, that is, $t+1$ nodes are connected in the first group, and in the second and third group, t nodes are connected respectively to download the original file, and no coding operation is required.

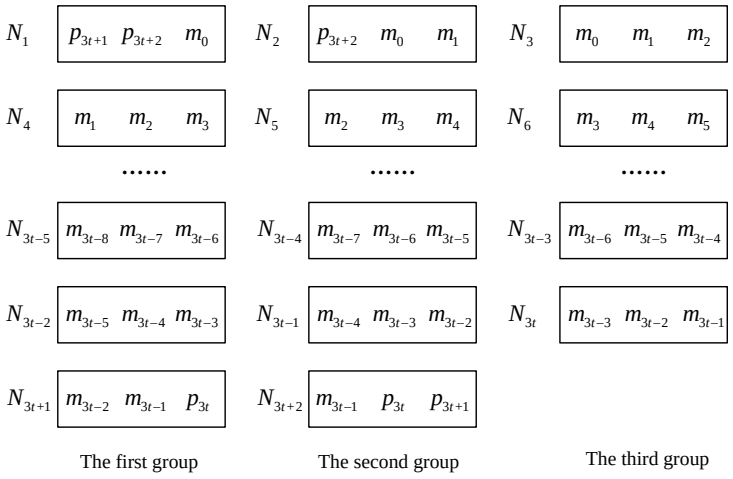
(iii) $k+2=3t+2$ $(k\geq 2)$. The nodes are also divided into three groups. The first group contains N_1, N_4, N_7,…, N_{3t+1}, abbreviated as $N_{1+3h}(0\leq h\leq t)$. The second group contains N_2, N_5, N_8,…, N_{3t+2}, abbreviated as N_{2+3h} $(0\leq h\leq t)$. The third group contains N_3, N_6, N_9,…, N_{3t}, abbreviated as $N_{3+3h}(0\leq h\leq t\text{-}1)$, as shown in **Fig. 2(c)**. The first two groups have $3t+3$ different data blocks respectively, containing $3t=k$ different data blocks. And the third group has $3t$ different data blocks in total, including $3t=k$ different original data blocks. Thus, in the first and second group, $t+1$ nodes are connected to download the original file, and in the third group, t nodes are connected without coding operation.



(a) $k+2$ can be divisible by 3



(b) $k+2$ divided by 3 more than 1



(c) $k+2$ divided by 3 more than 2

Fig. 2. Encoding scheme

Note that the minimum number of nodes connected to download the original file is $d_{\min}$, termed as disk I/O overhead. Then $d_{\min}=\left\lceil k/3 \right\rceil$ nodes of DSS need to be connected to recover the original file by adopting MBR codes based on cyclic VFR codes, while k nodes need to be connected to restore the original file through RS codes or SRC. Obviously, the proposed MBR codes based on cyclic VFR codes have much lower disk I/O overhead than RS codes and SRC.

When adopting traditional MBR codes in the DSS, $\beta_{MBR}=\frac{2M}{2kd-k^2+k}$ data will be downloaded from each survival node during repairing a failed node, and the node storage overhead is $\alpha_{MBR}=\frac{2Md}{2kd-k^2+k}=d\beta_{MBR}$. The cyclic VFR codes constructed in this paper need to connect two nodes to repair a failed node, i.e., $d=2$. One data block is downloaded from a surviving node, and two other data blocks are downloaded from another surviving node for a

single node repair, and the two cases are equal probability. The amount of data downloaded from the two surviving nodes are β_1=1 and β_2=2 respectively, and the node storage overhead in the cyclic VFR codes is 3, which is consistent with the result calculated by $\alpha_{MBR}=\frac{2Md}{2kd-k^2+k}=d\beta_{MBR}$, i.e., the storage overhead is $\alpha_{MBR}=\frac{1}{2}(d\beta_1+d\beta_2)=3$. Then, the cyclic VFR codes satisfy the MBR point condition under the accurate uncoded repair. Since a single failed node can be repaired only by downloading the lost data, cyclic VFR codes are MBR codes.

Fig. 3 shows a DSS in which cyclic (7, 4, 3, 2) VFR codes are adopted with repetition degrees $\rho_1=3$, $\rho_2=2$. $\mathbf{m}=(m_0,m_1,m_2,m_3)$ represents the original file stored in the DSS, encoded by systematic $(7, 4)$ MDS codes to obtain $\mathbf{c}=(m_0,m_1,m_2,m_3,p_4,p_5,p_6)$. Then according to **Fig. 1**, construct cyclic VFR codes and store the data blocks. Finally, the MBR codes based on cyclic VFR codes are constructed under the condition that $k+2$ can be divisible by 3 according to **Fig. 2(a)**. Users can download the original file by connecting all nodes in any one group of the three groups.

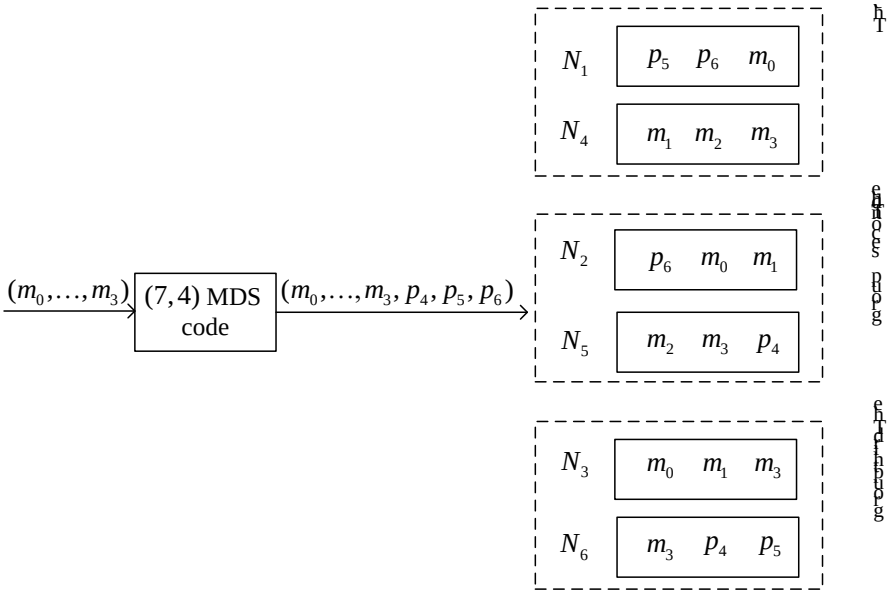


Fig. 3. (7, 4, 3, 2) VFR codes

4. Repair of failed nodes

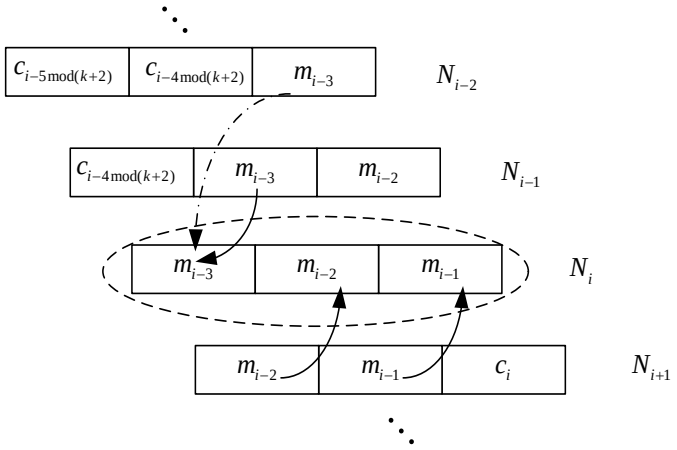
If a given node fails, the set of nodes that are connected to recover the failed node is called a failure repair group of the failed node. From **Fig. 2**, when a single node fails, only two surviving nodes need to be connected to recover the failed node. Only one repair group exists for N_1, N_{k+2}, and meanwhile N_2, N_{k+1} have two failure repair groups. Furthermore, the other nodes have three failure repair groups, and each failure repair group contains two nodes. For example, if node N_1 fails, only surviving nodes N_2, N_{k+2} need to be connected for

repairing N_1, and the failure repair group is termed as (N_2, N_{k+2}). As summary, if node N_i $(1 \le i \le k+2)$ fails, the failure repair group is as follows

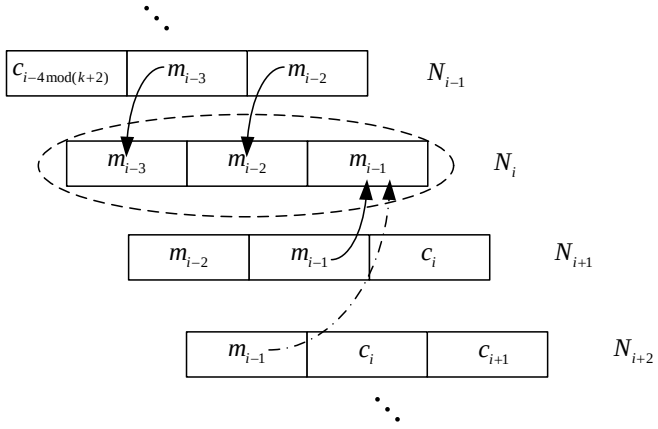
$$\begin{cases} (N_{i-1 \bmod (k+2)}, N_{i+1 \bmod (k+2)}) & i \in \{1, k+2\} \\ (N_1, N_3) \text{ or } (N_1, N_4) & i = 2 \\ (N_{i-1}, N_{i+1}) \text{ or } (N_{i-1}, N_{i+2}) \text{ or } (N_{i-2}, N_{i+1}) & i \in \{3, \ldots, k\} \\ (N_k, N_{k+2}) \text{ or } (N_{k-1}, N_{k+2}) & i = k+1 \end{cases} \tag{6}$$

Two nodes in the failure repair group can be connected to accurately recover the data of failed node without coding if a node fails. If node N_i $(i \in \{3, \ldots, k\})$ fails, several repair methods for the failed node are shown in **Fig. 4**, where c may be either an original data block or a parity block.

When two nodes fail, the original file should be restored to recover the data blocks if the two failed nodes contain the same parity blocks. Otherwise, the data blocks of the failed nodes can only be restored directly by copying.



(a)



(b)

Fig. 4. Repair of a single failed node

5. Performance analysis

This section mainly analyzes the repair bandwidth overhead, the repair locality and the repair complexity of MBR codes based on cyclic VFR codes, and compares the performances of MBR codes based on cyclic VFR codes with SRC, RS codes and FR codes (ρ=2, ρ=3). Because VFR codes are greatly affected by parameter constraints, and the number of nodes in the storage system is limited to be some specific even, the proposed codes are compared with FR codes.

Table 1 shows the storage overhead, repair bandwidth overhead and repair locality of the five codes above. For comparison in this section, here fix file size M=1000Mb, the subfile number in SRC $f=4$. The first layers of the last three codes all adopt $(k+3,\ k)$ MDS codes.

Table 1. Performance analysis of several coding schemes

		(n,k,f) SRC $2f \le n$	(n,k) RS Codes	FR Codes① ($\rho=2$)	FR Codes② ($\rho=3$)	MBR Codes Based on Cyclic VFR Codes
Node storage overhead		$\frac{(f+1)M}{fk}$	$\frac{M}{k}$	$\frac{(\sqrt{2k+6.25}-0.5)M}{k}$	$\frac{(\sqrt{1.5k+4.5625}-0.25)M}{k}$	$\frac{3M}{k}$
Repair bandwidth overhead	Single node failure	$\frac{(f+1)M}{k}$	M	$\frac{(\sqrt{2k+6.25}-0.5)M}{k}$	$\frac{(\sqrt{1.5k+4.5625}-0.25)M}{k}$	$\frac{3M}{k}$
	Two nodes failure	$\frac{2(f+1)M}{k}$ or M	M		$\frac{(2\sqrt{1.5k+4.5625}-1.5)M}{k}$	$\frac{6M}{k}$ or M
Repair locality	Single node failure	$2f$	k	$\sqrt{2k+6.25}-0.5$	$\sqrt{1.5k+4.5625}-0.25$	2
	Two nodes failure	k	k		$\sqrt{1.5k+4.5625}-0.25$	$\left\lceil \frac{k}{3} \right\rceil +1$

Note: ① FR codes based on complete graph (ρ=2)

② FR codes based on Steiner system (ρ=3)

5.1 Bandwidth overhead

The repair bandwidth overhead is the amount of data that needs to be downloaded when repairing the failed nodes. In case of a single node failure, as adopting (n,k,f) SRC, each node stores $f+1$ data blocks, and needs to download f data blocks to repair a failed node. The size of each data block is M/fk, and the bandwidth overhead of repairing a failed node is $(f+1)M/k$. For $(n,\ k)$ RS codes, the entire original file needs to be downloaded to repair a failed node, and its bandwidth overhead is M. For MBR codes based on cyclic VFR codes, since each node stores 3 data blocks, and the size of each data block is M/k, the bandwidth overhead of repairing a failed node is $3M/k$.

FR codes with $\rho=2$ can only solve the problem of a single node failure by duplication. Assuming that each node of the FR codes stores d data blocks, it can be known that there are a total of $d+1$ nodes by coding structure and the formula $d(d+1)=2(k+3)$ is satisfied [17], $d=\sqrt{2k+6.25}-0.5$ is further obtained. Meanwhile, any two nodes have at most one identical data block, it is necessary to connect d nodes when recovering lost data blocks by

replication, and each data block size is M/k, so the repair bandwidth of FR codes with $\rho=2$ is $(\sqrt{2k+6.25}-0.5)M/k$. For FR codes with $\rho=3$, it is known from [17] that the formula $2d(2d+1)=6(k+3)$ is satisfied, so $d=\sqrt{1.5k+4.5625}-0.25$. In the case of a single node failure, d surviving nodes need to be connected for repairing the failed node, similarly, the repair bandwidth overhead of the FR codes is $(\sqrt{1.5k+4.5625}-0.25)M/k$.

In case of two nodes failure, the repair bandwidth of RS codes is also M. For SRC, if the number of nodes between two failed nodes is greater than $f-1$, the two failed nodes can be repaired independently, and its repair bandwidth is $2(f+1)M/k$. Otherwise, the entire original file needs to be restored to repair the failed nodes, and the repair bandwidth overhead is also M. For MBR codes based on cyclic VFR codes, in addition to three cases of two-nodes failure, i.e., N_1 and N_2, N_1 and N_{k+2}, N_{k+1} and N_{k+2}, need to restore the original file to recover the data of the failed nodes, the remaining two-node failure case can also repair the failed nodes by copying, and then its repair bandwidth overhead is $6M/k$.

FR codes with $\rho=2$ cannot recover two node failures. FR codes with $\rho=3$ can tolerate two node failures and repair them by replication. It can be seen from [17] that $2d-1$ different data blocks will be lost if two nodes fail. Based on the above analysis of a single node failure, it can be concluded that the two-node repair bandwidth overhead of the FR codes is $(2\sqrt{1.5k+4.5625}-1.5)M/k$.

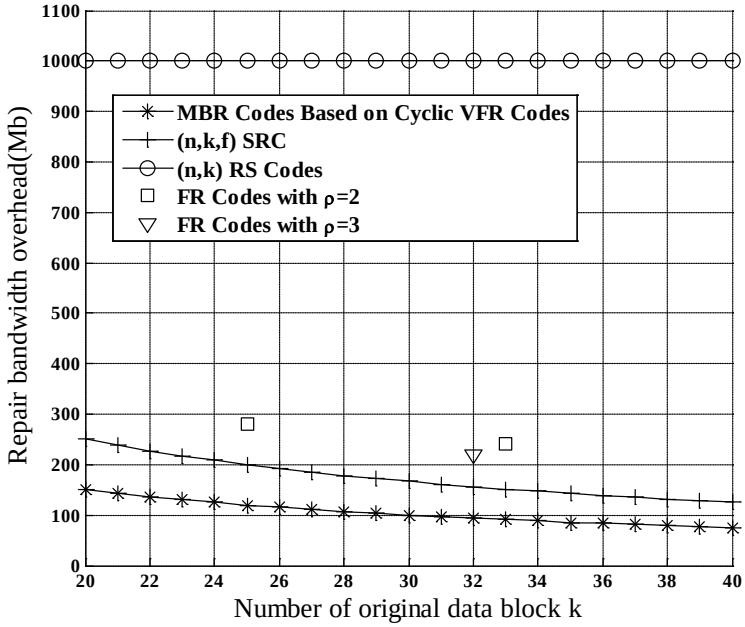


Fig. 5. Comparison of repair bandwidth overhead for a single node failure

Fig. 5 shows the repair bandwidth overhead of the five codes for the case of a single node failure. The repair bandwidth overhead of RS codes is constant, i.e., 1000Mb, but the

bandwidth overhead of SRC, MBR codes based on cyclic VFR codes decreases as k increases. As k is fixed, MBR codes based on cyclic VFR codes proposed in this paper have the smallest bandwidth overhead. For example, when k=30, the bandwidth overhead of MBR codes based on cyclic VFR codes is 100MB, far smaller than that of SRC, RS codes. The FR codes with $\rho=2$ and $\rho=3$ exist only when k takes some specific values, and the repair bandwidth overhead of the existing FR codes for a single node failure is significantly higher than the MBR codes proposed in this paper.

5.2 Repair locality

The repair locality is the number of nodes that need to be connected when repairing the failed nodes. If a single node fails, SRC needs to connect $2f$ $(2f \le n)$ nodes to repair the failed node, that is, the repair locality is $2f$. Since (n,k) RS codes need to connect k nodes to recover the original file for repairing the failed node, the repair locality is k. According to the analysis of repair bandwidth overhead above, we can obtain that the repair localities of the FR codes with $\rho=2$ and $\rho=3$ are $\sqrt{2k+6.25}-0.5$ and $\sqrt{1.5k+4.5625}-0.25$, respectively. MBR codes based on cyclic VFR codes need to connect 2 nodes to recover the lost data, and the repair locality is 2.

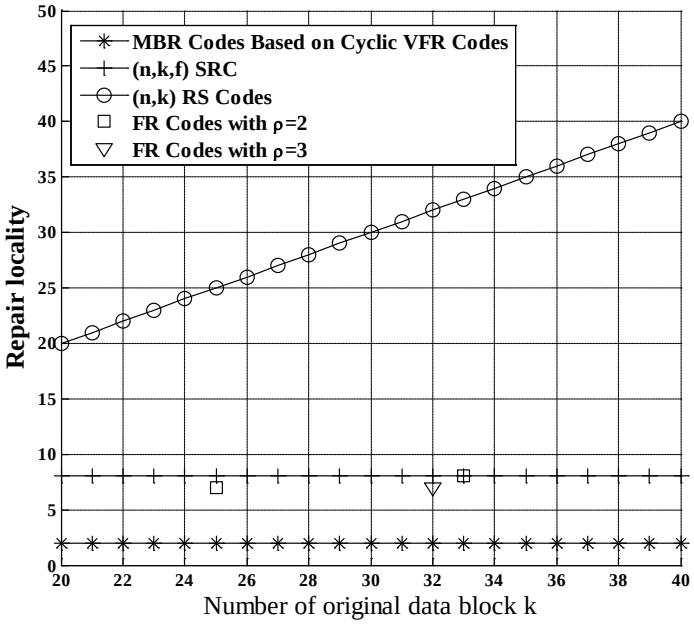


Fig. 6. Repair locality for a single node failure

Fig. 6 shows the repair locality for a single node failure. With the increasing of k, the repair locality of RS codes increases linearly, and the repair localities of SRC and MBR codes based on cyclic VFR codes are 8 and 2 respectively. As k is fixed, MBR codes based on cyclic VFR codes have the smallest repair locality.

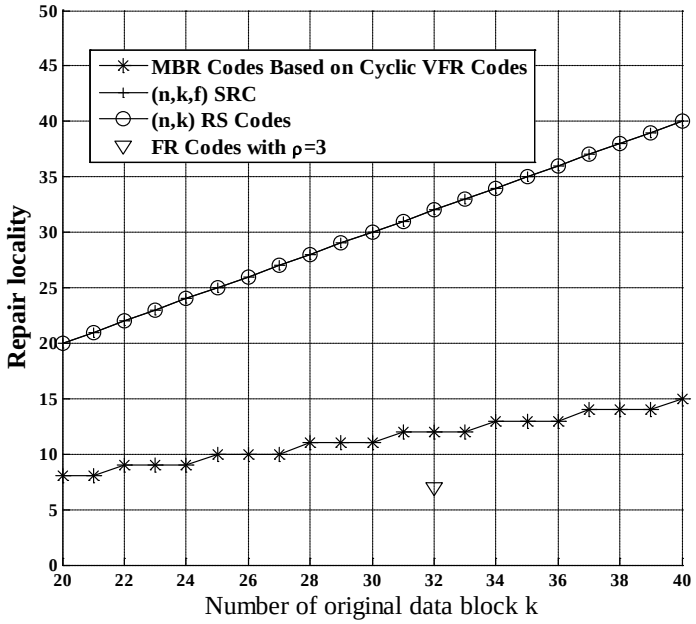


Fig. 7. Repair locality for two-node failures

When two nodes fail, SRC needs to recover the original file by connecting k nodes to repair the failed nodes, and the repair locality is k. Since (n,k) RS codes need to connect k nodes to repair the failed nodes, the repair locality is also k. For FR codes with $\rho=3$, the repair locality of the two-node failures is still $\sqrt{1.5k+4.5625}-0.25$. According to the group design of cyclic VFR codes above, the MBR codes proposed should connect $\lceil k/3 \rceil+1$ or $\lceil k/3 \rceil$ nodes to restore the original file and repair the failed nodes. Thus the repair locality is less than or equal to $\lceil k/3 \rceil+1$.

Fig. 7 shows the repair locality for repairing two failed nodes. It can be intuitively observed that, compared with SRC and RS codes, the repair locality of FR codes with $\rho=3$ is the smallest, MBR Codes based on cyclic VFR codes the second, and only one value of k satisfies the existence condition of the FR codes. Moreover, with the increasing of k, the superiority of the proposed MBR codes is becoming much obvious in the performance of repair locality.

In addition, the MBR codes proposed in this paper overcome the shortcoming that the repair locality of the traditional MBR codes is greater than RS codes and SRC, and further reduce the repair locality and repair bandwidth of the traditional FR codes in the case of a single node failure, with the performance reaching the equilibrium state.

5.3 Repair complexity

In SRC, $f-1$ XOR operations are required to recover one data block as a single node fails.

Since $f+1$ data blocks are stored in one node for SRC, $(f-1)(f+1)$ XOR operations are required in total for repairing a failed node. RS codes need to connect k nodes to recover the original file, and then recover the data blocks of the failed node. FR codes with $\rho=2$ and $\rho=3$ and MBR codes based on cyclic VFR codes don't require any encoding operation when a single node fails, and can directly recover the data of the failed node by copying. Thus the repair complexity of MBR codes proposed and FR codes is significantly lower than the other two codes, and they can repair the failed node at a very high speed.

6. Conclusion

In this paper, a new construction method of MBR codes based on cyclic VFR codes is proposed, which overcome the limitations of traditional VFR codes in applicable occasions. Taking into account the fact that the original data blocks in the actual DSS have higher access demand, we have improved FR codes and adopted the concept of hot and cold data, making the DSS store a total of three copies of the original data blocks and two copies of the parity blocks. The MBR codes constructed in this paper improve the defect that the repair locality of traditional MBR codes is larger than SRC and RS codes, having the smallest repair locality for a single node failure. Theoretical analysis shows that, the storage overhead of MBR codes based on cyclic VFR codes is greater than that of SRC and RS codes, but lower than that of the FR codes with $\rho=2$ and $\rho=3$. Moreover, the repair bandwidth overhead and repair locality are smaller than SRC and RS codes. Towards repair by transfer for a single node failure, the proposed MBR codes have lower repair complexity, as well as faster repair speed.

References

[1] C. Li, Z. Zhou and X. Zhai, "On a class of multi-source distributed storage with exact repair," *IEEE Access*, vol. 6, pp. 20704-20711, April, 2018. Article (CrossRef Link).

[2] N. Bardis, N. Doukas and O. P. Markovskyi, "Effective method to restore data in distributed data storage systems," in *Proc. of 2015 IEEE Military Communications Conference (MILCOM)*, pp. 1248-1253, October 26-28, 2015. Article (CrossRef Link).

[3] Z. Yuan and H. Liu, "Efficiently coding replicas to erasure coded blocks in distributed storage systems," *IEEE Communications Letters*, vol. 21, no. 9, pp. 1897-1900, September, 2017. Article (CrossRef Link).

[4] K. V. Rashmi, N. B. Shah, K. Ramchandran, et al., "Information-theoretically secure erasure codes for distributed storage," *IEEE Transactions on Information Theory*, vol. 64, no. 3, pp. 1621-1646, November, 2018. Article (CrossRef Link).

[5] R. Rodrigues and B. Liskov, "High availability in DHTs: erasure coding vs. replication," in *Proc. of 4th International Workshop on Peer-to-Peer Systems*, pp. 226-239, February 24-25, 2005. Article (CrossRef Link).

[6] E. Mugisha and G. Zhang, "A reliable secure storage cloud and data migration based on erasure code," *KSII Transactions on Internet and Information Systems*, vol. 12, no.1, pp. 436-453, January 31, 2018. Article (CrossRef Link).

[7] Z. Ullah, S. Jabbar, M. H. B. T. Alvi, et al., "Analytical study on performance, challenges and future considerations of google file system," *International Journal of Computer and Communication Engineering*, vol. 3, no. 4, pp. 279-284, July, 2014. Article (CrossRef Link).

[8] A. Higai, A. Takefusa, H. Nakada, et al., "A study of effective replica reconstruction schemes for the hadoop distributed file system," *IEICE Transactions on Information and Systems*, vol. E98D, no. 4, pp. 872-882, 2015.
[9] R. Ahlswede, N. Cai, S. Y. R. Li, et al., "Network information flow," *IEEE Transactions on Information Theory*, vol. 46, no. 4, pp. 1204-1216, 2000. Article (CrossRef Link).
[10] A. G. Dimakis, P. B. Godfrey, M. J. Wainwright, et al., "Network coding for distributed storage systems," in *Proc. of 26th IEEE International Conference on computer Communications (INFOCOM)*, pp. 2000-2008, May 6-12, September 4, 2007. Article (CrossRef Link).
[11] A. G. Dimakis, P. B. Godfrey, Y. Wu, et al., "Network coding for distributed storage systems," *IEEE Transactions on Information Theory*, vol. 56, no. 9, pp. 4539-4551, August 16, 2010. Article (CrossRef Link).
[12] K. W. Shum and Y. Hu, "Cooperative regenerating codes," *IEEE Transactions on Information Theory*, vol. 59, no. 11, pp. 7229-7258, July 22, 2013.
[13] K. W. Shum and Y. Hu, "Existence of minimum-repair-bandwidth cooperative regenerating codes," in *Proc. of 2011 International Symposium on Network Coding (NetCod)*, pp. 1-6, July 25-27, 2011.
[14] K. V. Rashmi, N. B. Shah and P. V. Kumar. "Optimal exact-regenerating codes for distributed storage at the MSR and MBR points via a product-matrix construction," *IEEE Transactions on Information Theory*, vol. 57, no. 8, pp. 5227-5239, January, 2011. Article (CrossRef Link).
[15] D. S. Papailiopoulos, J. Luo, A. G. Dimakis, et al., "Simple regenerating codes: network coding for cloud storage," in *Proc. of IEEE INFOCOM*, pp. 2801-2805, March 25-30, 2012. Article (CrossRef Link).
[16] D. S. Papailiopoulos and A. G. Dimakis, "Locally repairable codes," *IEEE Transaction on Information Theory*, vol. 60, no. 10, pp. 5843-5855, 2014. Article (CrossRef Link).
[17] S. E. Rouayheb and K. Ramchandran, "Fractional repetition codes for repair in distributed storage systems," in *Proc. of 2010 48th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, pp. 1510-1517, September 29-October 1, 2010. Article (CrossRef Link).
[18] B. Zhu, K. W. Shum, H. Li, et al., "General fractional repetition codes for distributed storage systems," *IEEE Communications Letters*, vol. 18, no. 4, pp. 660-663, March 11, 2014. Article (CrossRef Link).
[19] Q. Yu, C. W. Sung and T. H. Chan, "Irregular fractional repetition code optimization for heterogeneous cloud storage," *IEEE Journal on Selected Areas in Communications*, vol. 32, no. 5, pp. 1048-1060, April 24, 2014. Article (CrossRef Link).
[20] M. Y. Nam, J. H. Kim and H. Y. Song, "Locally repairable fractional repetition codes," in *Proc. of Seventh International Workshop on Signal Design and its Applications in Communications (IWSDA)*, pp. 128-132, September 14-18, 2015. Article (CrossRef Link).
[21] M. F. Aktas, E. Najm and E. Soljanin, "Simplex queues for hot-data download," *ACM SIGMETRICS Performance Evaluation Review*, vol. 45, no. 1, pp. 35-36, June, 2017. Article (CrossRef Link).
[22] B. Wei, L. M. Xiao, W. Wei, et al., "A new adaptive coding selection method for distributed storage systems," *IEEE Access*, vol. 6, pp. 13350-13357, February, 2018. Article (CrossRef Link).
[23] B. Zhu, H. Li, H. Hou, et al., "Replication-based distributed storage systems with variable repetition degrees," in *Proc. of Twentieth National Conference on Communications (NCC)*, pp. 1-5, February 28-March 2, 2014. Article (CrossRef Link).

Jing Wang is currently a professor with the School of Information Engineering, Chang'an University, China. She received her B.S., M.S. and Ph.D. degrees in the School of Communication Engineering from Xidian University, Shaanxi, China, in 2004, 2005, and 2009, respectively. Her research interests are in the area of network coding, distributed storage and regenerating codes.

Shuxia Wang received her B.S. degree in electronic information engineering from the School of Physics and Information Engineering, Shanxi Normal University in 2015. Her research interests include distributed storage, regenerating codes and locally repairable codes.

Tiantian Wang received her B.S. degree in communication engineering from the school of Optoelectronics Information Science and technology, Yantai University in 2016. Her research interests include distributed storage, regenerating codes and locally repairable codes.

Xuefei Zhang received her B.S. degree in communication engineering from the School of Computer and Information Engineering, Inner Mongolia Normal University in 2016. Her research interests include distributed storage, regenerating codes and locally repairable codes.