

自适应可分解部分重复码的扩展构造

王甜甜¹, 王汗青¹, 孟洁¹, 余春雷², 王晓峰¹

(1. 海军航空大学 航空基础学院, 山东 烟台 264000;

2. 四川文理学院 智能制造学院, 四川 达州 635000)

摘要:部分重复(Fractional Repetition, FR)码能够实现精确无编码修复,修复复杂度低且修复带宽成本小。在动态分布式存储系统中,要求FR码的节点存储开销和数据块重复度会随机动态变化。为了使FR码更灵活地适应动态分布式存储系统,该文提出利用超图实现自适应可分解FR码的扩展构造方法。具体地,建立超图中边和顶点与FR码中节点和数据块的对应关系,通过增加或删除超图中对应边和顶点,实现超图的扩展构造,进而得到存储系统规模和存储文件规模变化时自适应可分解FR码的扩展构造。基于这种方法,能够扩展构造出给定参数范围内所有自适应可分解FR码,列举了存储节点数20以内的所有参数。自适应可分解FR码与常见的简单再生码(Simple Regenerating Codes, SRC)和RS(Reed-Solomon)码相比,在修复局部性和修复带宽开销方面具有一定优势。

关键词:部分重复码;分布式存储系统;超图;自适应可分解;扩展构造

中图分类号:TP391.9

文献标识码:A

文章编号:1673-629X(2023)11-0014-06

doi:10.3969/j.issn.1673-629X.2023.11.003

Extended Construction of Adaptive-and-resolvable Fractional Repetition Codes

WANG Tian-tian¹, WANG Han-qing¹, MENG Jie¹, YU Chun-lei², WANG Xiao-feng¹

(1. Aviation Foundation College, Naval Aviation University, Yantai 264000, China;

2. School of Intelligent Manufacturing, Sichuan University of Arts and Science, Dazhou 635000, China)

Abstract: Fractional repetition (FR) codes can achieve exact uncoded repair for failed nodes, with lower repair computational complexity and repair bandwidth overhead. In dynamic distributed storage system, node storage overhead and coded packets repetition degree of FR codes will change randomly and dynamically. In order to make FR codes more flexible to adapt to dynamic distributed storage system, we propose the extended construction method of adaptive-and-resolvable FR codes by hypergraph. Specifically, the corresponding relationship between edges and vertices in hypergraph and nodes and data blocks in FR codes is established. By adding or deleting corresponding edges and vertices in hypergraph, the extended construction of hypergraph and adaptive-and-resolvable FR code are realized when file size and scale of distributed storage system change. Based on the scheme, all adaptive-and-resolvable FR codes within given parameter ranges can be constructed by extension, and all the parameter within 20 storage nodes are listed. Compared with Simple Regenerating Codes (SRC) and Reed-Solomon (RS) codes, the adaptive-and-resolvable FR codes have some advantages in repair locality and repair bandwidth overhead.

Key words: fractional repetition codes; distributed storage systems; hypergraph; adaptive-and-resolvable; extended construction

0 引言

现如今,海量数据仍迅速膨胀,针对大数据的可靠安全存储问题被广泛重视。大规模分布式存储系统成为当前存储海量数据的有效途径,因其具备可用性高、成本低、吞吐量高和可扩展等优势^[1]。由于系统中经常发生存储节点故障的情形,为了使分布式存储系统提供可靠的存储服务,需要通过连接其他存活节点对

故障节点进行修复。通常需要采取一定的冗余策略,以保证数据存储的有效性和可靠性。主流的冗余策略是复制(Replication)^[2]和纠删码(Erasure Codes)^[3]。复制策略的存储成本太大,在修复故障节点时纠删码的带宽开销较高。

针对复制策略和纠删码策略在数据存储和故障修复中存在的局限性,通过将“网络编码”的思路引入分

收稿日期:2022-12-31

修回日期:2023-05-04

基金项目:国家自然科学基金资助项目(62001059);陕西省重点研发计划项目(2021GY-019);海军航空大学科研自主立项青年基金项目(H3202201018)

作者简介:王甜甜(1994-),女,助教,硕士,通信作者,研究方向为分布式存储、网络编码、再生码。

布式存储系统中, Dimakis 等人提出了再生码 (Regenerating Codes, RC)^[4], 降低了修复带宽开销和节点存储开销^[5]。根据存储—带宽开销权衡曲线上的最佳极值点, Rashmi 等人提出了最小带宽再生 (Minimum Bandwidth Regeneration, MBR) 码和最小存储再生 (Minimum Storage Regeneration, MSR) 码^[6]。但再生码需要连接较多存活节点来修复故障节点。为了降低修复故障节点时的修复局部性, Papailiopoulos 等人提出了局部性修复编码 (Locally Repairable Codes, LRC)^[7], 在局部修复组内对故障节点进行修复。结合 RS 码与简单的异或运算, Papailiopoulos 提出了简单再生码 (Simple Regenerating Codes, SRC)^[8]。再生码和 LRC 需要通过大量运算来修复故障节点, 其修复过程中的计算复杂度较高, 导致修复时间较长^[9]。

为了保证修复故障节点时具有较低修复带宽开销, 同时进一步提高修复效率, Ramchandran 等人提出了一种实现对故障节点进行精确无编码修复的 MBR 码——部分重复 (Fractional Repetition, FR) 码^[10]。实际的分布式存储系统是一种动态的随机变化的环境, 也就是说, 存储节点故障和部分数据丢失等情况随时可能发生^[11]。针对这一问题, 朱兵提出了自适应 FR 码^[11], 以使 FR 码适用于动态分布式存储系统。采用自适应 FR 码的分布式存储系统中, 在节点中部分数据丢失并无法修复的情况下, 通过简单调整系统结构以满足 FR 码特性, 无需重新配置存储系统^[12]。随后, Oktay Olmez 提出了可分解 FR 码^[13-15], 并提出了基于组合设计的构造方法。可分解 FR 码中每个平行类均存储原文件大小数据量, 在部分节点故障的情况下, 剩余平行类中存活节点仍满足 FR 码特性。Su Yisheng 结合自适应 FR 码与可分解 FR 码的特性, 提出了自适应可分解 FR 码, 并提出了基于仿射置换矩阵 (Affine Permutation Matrices, APMs) 与循环置换矩阵 (Circulant Permutation Matrices, CPMs) 的构造方法^[16], 有效适应了节点存储容量和数据块重复度在动态存储系统中的随机动态变化。为了使系统中存储节点数和节点存储容量易于扩展, 朱兵提出了可扩展 FR 码的显式构造方法^[17]。Krishna 提出了具有非对称参数的广义部分重复 (Generalized Fractional Repetition, GFR) 码, 并建立了 GFR 码与超图的对应关系^[18]。

基于上述研究基础, 该文在自适应可分解 FR 码原始构造方案的基础上, 提出了利用超图实现自适应可分解 FR 码的扩展构造方法, 实现 FR 码在动态分布式存储系统中的灵活扩展。当文件规模或分布式存储系统规模发生变化时, 根据超图中边和顶点对应于 FR 码中节点和数据块的关系, 通过增加或删除超图中对应的边和顶点, 实现动态分布式存储系统中自适应可

分解 FR 码的扩展构造。通过这种基于超图的扩展构造方法, 能够扩展出给定参数范围内的所有自适应可分解 FR 码, 该文列举出了存储节点数 20 以内自适应可分解 FR 码的所有参数。自适应可分解 FR 码与 SRC 和 RS 码相比, 在修复局部性和修复带宽开销方面具有一定优势。

1 基础知识

1.1 自适应可分解 FR 码

在 (n, k, d) 分布式存储系统中, 假定节点存储容量为 α , 从 n 个存储节点中至少连接任意 k 个存活节点即可重构原文件, 单节点故障需要至少连接 d 个存活节点实现修复, 则满足条件 $k \leq d \leq n-1$, 且 $d = \alpha$ 。

FR 码^[10]: 在 (n, k, d) 分布式存储系统中, 设定 FR 码 $C = (\Omega, N)$ 由 n 个子集组成的集合 $N = \{N_1, \dots, N_n\}$ 表示, 其中, 每个子集均由符号集 $\Omega = \{1, \dots, \theta\}$ 中元素描述, 且符号集 Ω 中元素的重复度均为 ρ , 该 FR 码亦可描述为 (n, d, θ, ρ) FR 码^[19]。该 (n, d, θ, ρ) FR 码满足以下条件:

- (1) 子集 $N_i (i = 1, \dots, n)$ 的大小均为 d ;
- (2) 符号集 Ω 中每个元素均属于集合 N 中的 ρ 个子集;
- (3) 子集 N_i 和 $N_j (i \neq j)$ 最多包含符号集 Ω 中的一个元素;
- (4) $\theta\rho = nd$ 。

自适应 FR 码^[11]: 在 FR 码 $C = (\Omega, N)$ 中, 如果 $\exists S \subset \Omega$, 且由非空子集 $N_0 \setminus S, N_1 \setminus S, \dots, N_{n-1} \setminus S$ 组成的集合也是一个 FR 码 C' , 则 C 为自适应 FR 码。

当存储系统中部分数据永久性故障而无法恢复时, 原来的自适应 FR 码 C 通过简单调整得到新的 FR 码 C' , 以适应新的分布式存储系统。

可分解 FR 码^[13]: 在 FR 码 $C = (\Omega, N)$ 中, 对于子集 $P \subset N$, 当 $N_i \in P$ 且 $N_j \in P (i \neq j)$ 时, 若同时满足 $\bigcup_{i: N_i \in P} N_i = \Omega$ 和 $N_i \cap N_j = \emptyset$, 则 P 是 FR 码的平行类。如果包含多个平行类 P 的至少一个划分, 则称为可分解 FR 码。

通过改变可分解 FR 码中的平行类数, 即可改变数据块的重复度。由于每个平行类中的节点均存储全部数据块, 因此重构原始文件可以通过下载任一平行类中的数据块实现, 进而修复故障节点。当 FR 码中包含多个平行类时, 即使删除该故障节点所在的平行类, 其余节点仍满足 FR 码特性。

自适应可分解 FR 码同时满足自适应 FR 码和可分解 FR 码的特性, 能够有效适应于动态分布式存储系统^[15]。

1.2 超图

超图^[20]: 对于离散集合 $H = (V, E)$, 其中 $V = (v_1, \dots, v_n)$ 是离散元素的有限集合, $E = (E_1, \dots, E_m)$ 是 V 中各非空子集的集合, 则超图就是离散集合 $H = (V, E)$ 。在超图中, 顶点的集合为 V , 边的集合为 E , 任意非零数量的顶点连接成超图的边。

简单超图: 超图的边集 $E = (E_1, \dots, E_m)$ 中, 如果满足 $E_i \subset E_j$, 则 $i = j$, 即任意两条边 E_i, E_j 没有包含关系, 则该超图为简单超图。

线性超图: 超图的边集 $E = (E_1, \dots, E_m)$ 中, 当 $i \neq j$ 时, $|E_i \cap E_j| \leq 1$, 即任意两条边 E_i, E_j 连接不超过一个共同顶点, 则该超图为线性超图。

r -一致超图: 超图的秩记为 $r(H) = \max_j |E_j|$, 超图的下秩记为 $s(H) = \min_j |E_j|$, 当满足 $r(H) = s(H)$ 时, 超图 H 称为一致超图。每条边均连接 r 个顶点的超图, 简记为 r -一致超图。

t -正则超图: 超图 H 中, 连接顶点 $v_i (i = 1, \dots, n)$ 的边数为顶点 v_i 的度, 凡所有顶点的度均相同的超图称为正则超图^[21]。每个顶点均由 t 条边连接的超图, 简记为 t -正则超图。

(r, t) -超图: 对于线性超图, 如果所有的边均包含 r 个顶点, 且所有顶点均存在于 t 条边中, 则称为线性 r -一致 t -正则超图, 简记为 (r, t) -超图。

子超图: 对于超图 $H = (V, E)$, 顶点子集 $A \subseteq V$, 其子超图为 $H_A = (A, \{e \cap A \mid e \in E \wedge e \cap A \neq \emptyset\})$ 。在原超图的基础上去掉某些顶点后的超图, 就是子超图。

部分超图: 超图 $H = (V, E)$ 中, 边集 $E = \{e_i \mid i \in I_e \wedge e_i \subseteq V \wedge e_i \neq \emptyset\}$ 的索引集为 I_e , 对于 $I \subset I_e$, 该超图的部分超图为 $H_I = (V, \{e_i \mid i \in I\})$ 。

2 自适应可分解 FR 码的扩展构造

建立 (d, ρ) -超图与 (n, d, θ, ρ) 自适应可分解 FR 码的对应关系, 超图中边和顶点分别对应 FR 码中存储节点和数据块。通过对 (d, ρ) -超图进行扩展, 实现自适应可分解 FR 码在动态分布式存储系统中的扩展, 而无需对系统进行重新配置。本节基于 (d, ρ) -超图, 从存储系统规模和存储文件规模变化这两个方面, 对自适应可分解 FR 码进行扩展构造, 使扩展后的 FR 码仍满足自适应和可分解特性。

2.1 存储系统规模变化

当存储系统规模发生变化时, 通过改变存储节点数和数据块重复度, 即可对原自适应可分解 FR 码进行扩展构造。

如果存储系统规模减小, 即存储节点数 n 减小, 假定节点存储容量 d 和存储文件规模 (即数据块数 θ) 均

不变, 为满足 FR 码的存在条件 (即 $\theta\rho = nd$), 则需要减小数据块重复度 ρ 。因为 FR 码中平行类的存储节点与超图的染色边相对应, 删除 (d, ρ) -超图中对应染色边, 就能实现向 $(d, \rho - 1)$ -部分超图的扩展。根据 $(d, \rho - 1)$ -部分超图中边和顶点对应于 FR 码中节点和数据块的关系, 实现分布式存储系统规模变化时的扩展。

图 1 为 $(4, 4)$ -超图, 图 2 为对应的 $(16, 4, 16, 4)$ 自适应可分解 FR 码。当存储节点数减小为 $n = 12$ 时, 为满足 $\theta\rho = nd$, 则减小数据块重复度为 $\rho = 3$ 。在 $(4, 4)$ -超图基础上, 删除染色边 $\{e_{13}, e_{14}, e_{15}, e_{16}\}$, 得到图 3 所示 $(4, 3)$ -部分超图, 对应得到图 4 所示 $(12, 4, 16, 3)$ 自适应可分解 FR 码。

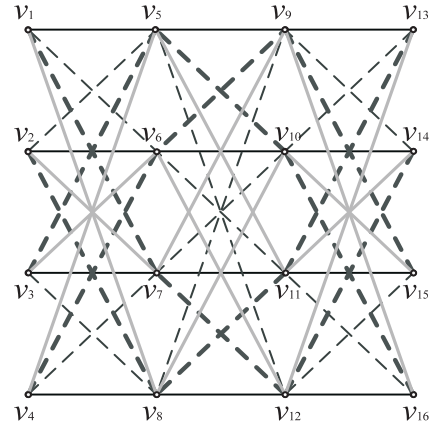


图 1 $(4, 4)$ -超图

N_1	1 5 9 13	N_5	1 6 11 16	N_9	1 7 12 14	N_{13}	1 8 10 15
N_2	2 6 10 14	N_6	2 5 12 15	N_{10}	2 8 11 13	N_{14}	2 7 9 16
N_3	3 7 11 15	N_7	3 8 9 14	N_{11}	3 5 10 16	N_{15}	3 6 12 13
N_4	4 8 12 16	N_8	4 7 10 13	N_{12}	4 6 9 15	N_{16}	4 5 11 14

图 2 $(16, 4, 16, 4)$ 自适应可分解 FR 码

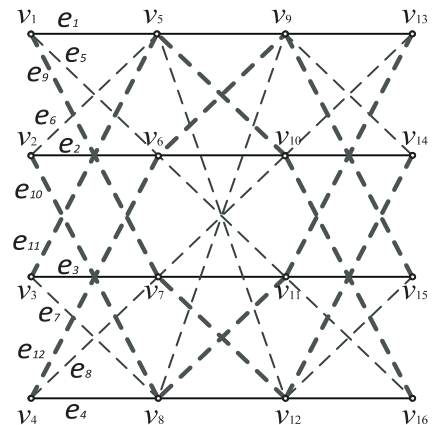


图 3 $(4, 3)$ -部分超图

N_1	1 5 9 13	N_5	1 6 11 16	N_9	1 7 12 14
N_2	2 6 10 14	N_6	2 5 12 15	N_{10}	2 8 11 13
N_3	3 7 11 15	N_7	3 8 9 14	N_{11}	3 5 10 16
N_4	4 8 12 16	N_8	4 7 10 13	N_{12}	4 6 9 15

图 4 $(12, 4, 16, 3)$ 自适应可分解 FR 码

2.2 存储文件规模变化

当存储文件规模发生变化时,通过改变数据块数和节点存储容量,即可对原自适应可分解 FR 码进行扩展构造。

如果存储文件规模减小,假定数据块大小不变,则数据块数 θ 减小,存储节点数 n 不变,为满足 FR 码的存在条件(即 $\theta\rho = nd$),保持数据块重复度 ρ 不变,则需要减小节点存储容量 d 。因为 FR 码的数据块与超图的顶点相对应,删除 (d, ρ) - 超图中对应顶点,就能实现向 $(d-1, \rho)$ - 子超图的扩展。根据 $(d-1, \rho)$ - 子超图中边和顶点对应于 FR 码中节点和数据块的关系,实现存储文件规模变化时的扩展。

在图 4 所示 $(12, 4, 16, 3)$ 自适应可分解 FR 码的基础上,当数据块数减小为 $\theta = 12$ 时,为满足 $\theta\rho = nd$,则减小节点存储容量为 $d = 3$ 。在图 3 所示 $(4, 3)$ - 超图基础上,删除顶点 $\{v_{13}, v_{14}, v_{15}, v_{16}\}$,得到图 5 所示 $(3, 3)$ - 子超图,对应得到图 6 所示 $(12, 3, 12, 3)$ 自适应可分解 FR 码。

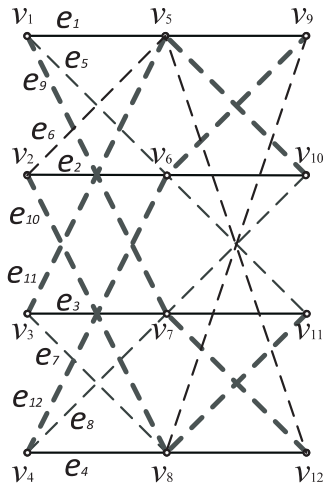


图 5 $(3, 3)$ - 子超图

N_1	1	5	9
N_2	2	6	10
N_3	3	7	11
N_4	4	8	12
N_5	1	6	11
N_6	2	5	12
N_7	3	8	9
N_8	4	7	10
N_9	1	7	12
N_{10}	2	8	11
N_{11}	3	5	10
N_{12}	4	6	9

图 6 $(12, 3, 12, 3)$ 自适应可分解 FR 码

2.3 自适应可分解 FR 码的参数范围

利用 (d, ρ) - 超图实现自适应可分解 FR 码的扩展,能够在给定参数范围内构造全部自适应可分解 FR 码。表 1 为自适应可分解 FR 码在存储节点数 20 个以内的全部 (n, d, θ, ρ) 参数。如表 1 所示,全部 (n, d, θ, ρ) FR 码均可通过基于 (d, ρ) - 超图的扩展实现。例如,针对存储节点数 $n = 8$,通过存储文件规模变化时的扩展方法,能够实现 $(8, 2, 8, 2)$ 自适应可分解 FR 码向 $(8, 3, 12, 2)$ 和 $(8, 4, 16, 2)$ 自适应可分解 FR 码的扩展。

表 1 (n, d, θ, ρ) 自适应可分解 FR 码的参数范围
($n \leq 20$)

节点数 n	自适应可分解 FR 码的参数 (n, d, θ, ρ)
4	$(4, 2, 4, 2)$
6	$(6, 2, 6, 2), (6, 3, 9, 2)$
8	$(8, 2, 8, 2), (8, 3, 12, 2), (8, 4, 16, 2)$
9	$(9, 2, 6, 3), (9, 3, 9, 3)$
10	$(10, 2, 10, 2), (10, 3, 15, 2), (10, 4, 20, 2), (10, 5, 25, 2)$
12	$(12, 2, 12, 2), (12, 3, 18, 2), (12, 4, 24, 2), (12, 5, 30, 2), (12, 6, 36, 2), (12, 2, 8, 3), (12, 3, 12, 3), (12, 4, 16, 3)$
14	$(14, 2, 14, 2), (14, 3, 21, 2), (14, 4, 28, 2), (14, 5, 35, 2), (14, 6, 42, 2), (14, 7, 49, 2)$
15	$(15, 2, 10, 3), (15, 3, 15, 3), (15, 4, 20, 3), (15, 5, 25, 3)$
16	$(16, 2, 16, 2), (16, 3, 24, 2), (16, 4, 32, 2), (16, 5, 40, 2), (16, 6, 48, 2), (16, 7, 56, 2), (16, 8, 64, 2), (16, 2, 8, 4), (16, 3, 12, 4), (16, 4, 16, 4)$
18	$(18, 2, 18, 2), (18, 3, 27, 2), (18, 4, 36, 2), (18, 5, 45, 2), (18, 6, 54, 2), (18, 7, 63, 2), (18, 8, 72, 2), (18, 9, 81, 2), (18, 2, 12, 3), (18, 3, 18, 3), (18, 4, 24, 3), (18, 5, 30, 3), (18, 6, 36, 3)$
20	$(20, 2, 20, 2), (20, 3, 30, 2), (20, 4, 40, 2), (20, 5, 50, 2), (20, 6, 60, 2), (20, 7, 70, 2), (20, 8, 80, 2), (20, 9, 90, 2), (20, 2, 10, 4), (20, 3, 15, 4), (20, 4, 20, 4), (20, 5, 25, 4), (20, 10, 100, 2)$

3 性能分析

分布式存储系统在修复故障节点时的两个重要性能是修复局部性和修复带宽开销。本节从这两个方面分析自适应可分解 FR 码的性能,并对比最常见的 SRC 和 RS 码,通过 Matlab 仿真给出这三种编码方式的性能对比。

3.1 修复局部性

在分布式存储系统中,假设原文件大小为 M ,存储节点数为 n ,重构原文件需要至少连接任意 k 个存活节点。 (n, k) RS 码中,只要故障节点数小于等于 $n - k$,均需要连接 k 个存活节点的数据解码重构原文件,再编码恢复故障节点数据,则修复局部性为 k 。 (n, k, f) SRC 中,原文件由 f 个采用 (n, k) RS 编码的子文件组成,通过连接 $2f$ 个存活节点能够修复单节点故障,则单节点故障的修复局部性为 $2f$;当两节点同时故障时,需要连接 k 个存活节点解码重构原文件后进行故障修复,则两节点同时故障的修复局部性为 k 。 (n, d, θ, ρ) 自适应可分解 FR 码中,当单节点故障时,

需要连接任一平行类的 d 个存活节点下载相应数据块,则修复局部性为 d 。当两节点同时故障时,若重复度 $\rho > 2$,通过任一平行类的 $\min\{n/\rho, 2d\}$ 个存活节点下载相应数据块,则修复局部性为 $\min\{n/\rho, 2d\}$;若重复度 $\rho = 2$,且两个故障节点存储相同数据块,连接任意 k 个存活节点重构原文件进行故障修复,则修复局部性为 k ;若重复度 $\rho = 2$,且两个故障节点没有存储相同数据块,需要连接 n/ρ 或 $2d$ 个存活节点,则修复局部性为 n/ρ 或 $2d$ 。

设定 (n, k, f) SRC 和 (n, k) RS 码的存储节点数为 $n = 12$,重构原文件需要至少连接任意 $k = 9$ 个存活节点。SRC 中原文件由 $f = 3$ 个子文件组成,每个子文件均采用 $(12, 9)$ RS 编码。为便于性能分析,设定 (n, d, θ, ρ) 自适应可分解 FR 码外部采用 $(12, 9)$ MDS 编码,其编码数据块的数目 $\theta = 12$,数据块重复度 $\rho = 3$,因此节点存储容量为 $d = 3$ 。如图7所示,相比于 $(12, 9, 3)$ SRC 和 $(12, 9)$ RS 码,外部采用 $(12, 9)$ MDS 编码的 $(12, 3, 12, 3)$ 自适应可分解 FR 码在修复局部性上具有较大优势。

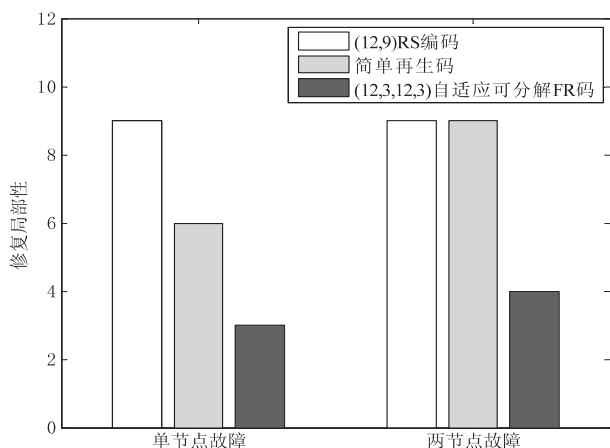


图7 修复局部性

3.2 修复带宽开销

(n, k) RS 码中,只要故障节点数小于等于 $n - k$,均需要连接 k 个存活节点并分别下载 M/k 的数据量,则修复带宽开销为 M 。 (n, k, f) SRC 中,当单节点故障时,下载 f 个数据块并进行异或运算能够修复一个故障数据块,由于每个节点存储 $f + 1$ 个数据量为 M/fk 的数据块,则单节点故障的修复带宽开销为 $(f + 1)M/k$;当两节点同时故障时,同 (n, k) RS 码一样,修复带宽开销为 $M^{[22]}$ 。采用 (θ, j) MDS 编码的 (n, d, θ, ρ) 自适应可分解 FR 码,每个数据块的数据量为 M/j ,当单节点故障时,通过 d 个存活节点下载相应故障数据块即可,则单节点故障的修复带宽开销为 Md/j 。当两节点同时故障时,若重复度 $\rho > 2$,这两个故障节点最多存储一个相同数据块,则修复带宽开销为 $M(2d - 1)/j$ 或 $2Md/j$;若重复度 $\rho = 2$,且两个故障

节点存储相同数据块,需要重构原文件进行故障修复,则修复带宽开销为 M ;若重复度 $\rho = 2$,且两个故障节点不包含相同数据块,只需要从存活节点中下载相应故障数据,则修复带宽开销为 $2Md/j$ 。

在3.1节 $(12, 9, 3)$ SRC、 $(12, 9)$ RS 码和 $(12, 3, 12, 3)$ 自适应可分解 FR 码的基础上,设定原始文件大小 $M = 1\,000$ Mb。如图8所示,相比于 $(12, 9, 3)$ SRC 和 $(12, 9)$ RS 码,外部采用 $(12, 9)$ MDS 编码的 $(12, 3, 12, 3)$ 自适应可分解 FR 码,在修复带宽开销上具有较大优势。

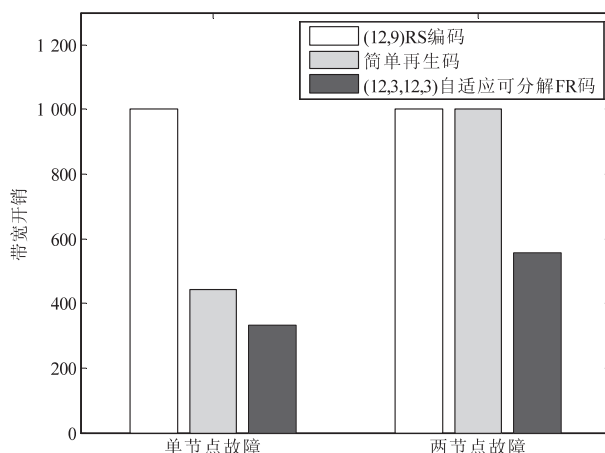


图8 修复带宽开销

4 结束语

针对动态分布式存储系统中 FR 码的灵活扩展问题,提出一种利用超图实现自适应可分解 FR 码的扩展构造方法,能够实现当文件规模变化和存储系统规模变化时的扩展构造。具体地,建立 (d, ρ) - 超图与 (n, d, θ, ρ) 自适应可分解 FR 码的对应关系,在原 FR 码构造和原超图结构的基础上,通过增加或删除超图中边和顶点,实现对动态分布式存储系统中自适应可分解 FR 码的扩展构造。基于这种方法,能够扩展构造出给定参数范围内所有自适应可分解 FR 码。自适应可分解 FR 码相比于 SRC 和 RS 码,具有较优的修复局部性和修复带宽开销。

参考文献:

- [1] CALDER B, WANG J, OGUS A, et al. Windows azure storage: a highly available cloud storage service with strong consistency[C]//Proceedings of the twenty-third ACM symposium on operating systems principles. Cascais: Association for Computing Machinery, 2011: 143-157.
- [2] LIU Y, VLASSOV V. Replication in distributed storage systems: state of the art, possible directions, and open issues [C]//2013 international conference on cyber-enabled distributed computing and knowledge discovery. Beijing: IEEE, 2013: 225-232.

- [3] LI J, LI B. Erasure coding for cloud storage systems; a survey [J]. Tsinghua Science and Technology, 2013, 18(3): 259–272.
- [4] DIMAKIS A G, GODFREY P B, WU Y, et al. Network coding for distributed storage systems[J]. IEEE Transactions on Information Theory, 2010, 56(9): 4539–4551.
- [5] 马景学. 信息安全与大数据存储中的几个关键问题[D]. 杭州: 浙江大学, 2018.
- [6] RASHMI K V, SHAH N B, KUMAR P V. Optimal exact-regenerating codes for distributed storage at the MSR and MBR points via a product-matrix construction[J]. IEEE Transactions on Information Theory, 2011, 57(8): 5227–5239.
- [7] PAPAILIOPOULOS D S, DIMAKIS A G. Locally repairable codes[J]. IEEE Transactions on Information Theory, 2014, 60(10): 5843–5855.
- [8] PAPAILIOPOULOS D S, LUO J, DIMAKIS A G, et al. Simple regenerating codes; network coding for cloud storage [C]//2012 proceedings IEEE INFOCOM. Orlando: IEEE, 2012: 2801–2805.
- [9] 孙 伟, 沈克勤, 张鑫楠, 等. 异构部分重复码的构造[J]. 计算机系统应用, 2021, 30(2): 226–230.
- [10] EL ROUAYHEB S, RAMCHANDRAN K. Fractional repetition codes for repair in distributed storage systems [C]//2010 48th annual Allerton conference on communication, control, and computing (Allerton). Monticello: IEEE, 2010: 1510–1517.
- [11] ZHU B, LI H. Adaptive fractional repetition codes for dynamic storage systems[J]. IEEE Communications Letters, 2015, 19(12): 2078–2081.
- [12] RAI B K, DHOORJATI V, SAINI L, et al. On adaptive distributed storage systems[C]//2015 IEEE international symposium on information theory (ISIT). Hong Kong: IEEE, 2015: 1482–1486.
- [13] OLMEZ O, RAMAMOORTHY A. Repairable replication-based storage systems using resolvable designs [C]//2012 50th annual Allerton conference on communication, control, and computing (Allerton). Monticello: IEEE, 2012: 1174–1181.
- [14] OLMEZ O, RAMAMOORTHY A. Constructions of fractional repetition codes from combinatorial designs [C]//2013 Asilomar conference on signals, systems and computers. Pacific Grove: IEEE, 2013: 647–651.
- [15] OLMEZ O, RAMAMOORTHY A. Fractional repetition codes with flexible repair from combinatorial designs [J]. IEEE Transactions on Information Theory, 2016, 62(4): 1565–1591.
- [16] SU Y S. Constructions of fractional repetition codes with flexible per-node storage and repetition degree [C]//GLOBECOM 2017 – 2017 IEEE global communications conference. Singapore: IEEE, 2017: 1–6.
- [17] ZHU B, ZHANG S, WANG W. Expandable fractional repetition codes for distributed storage systems [C]//2021 IEEE information theory workshop (ITW). Kanazawa: IEEE, 2021: 1–5.
- [18] GOPAL K, GUPTA M K. Bounds on generalized FR codes using hypergraphs [J]. Journal of Applied Mathematics and Computing, 2021, 65: 771–792.
- [19] ZHU B. A study on universally good fractional repetition codes [J]. IEEE Communications Letters, 2018, 22(5): 890–893.
- [20] DIESTEL R. Graph theory [M]. 3rd ed. Heidelberg: Springer-Verlag, 2005.
- [21] 许小满, 孙雨耕, 杨 山, 等. 超图理论及其应用 [J]. 电子学报, 1994, 22(8): 65–72.
- [22] WANG J, SHEN K, LIU X, et al. Construction of binary locally repairable codes with optimal distance and code rate [J]. IEEE Communications Letters, 2021, 25(7): 2109–2113.