一种容三盘失效纠删码的单数据盘失效快速重建方法

邱丽娜 王 芳 李 楚
(华中科技大学计算机学院 武汉 430074)
(武汉光电国家实验室 武汉 430074)
(教育部信息存储及应用实验室 武汉 430074)

摘 要 现代存储系统采用纠删码避免因磁盘故障导致的数据丢失,提高系统的可靠性和可用性.因容三盘失效纠删码,如TP编码和STAR编码,可同时容忍系统中任意3个磁盘损坏,可靠性超过RAID6编码,故而受到越来越多的关注.针对发生频率最高的单盘故障实现快速重建恢复数据服务,尚未得到有效实现.重建方案有多种,选择何种重建方案会影响重建性能甚至影响前端服务的响应时间.传统的单盘重建方法不仅耗时良久而且会造成带宽的浪费.针对单盘重建中传统方法的缺点,提出一种均分机制方法,利用处于"均分状态"的重建校验集合推理出单盘重建时所使用的数据量的最优解从而找出最佳重建方案,减少磁盘I/O时间,以加速重建过程.测试结果表明与传统方法相比,均分机制方法减少TP编码25%~30.6%和STAR编码30%~33.64%的磁盘读数据量,使重建时间显著减少;且在不同的数据块和不同的磁盘个数时,均分机制方法的重建性能均优于传统方法.

关键词 磁盘阵列;可靠性;磁盘损坏;重建算法
中图法分类号 TP302 **DOI号** 10.3724/SP.J.1016.2013.02041

EDS: A Novel Scheme for Boosting Single-Disk Failure Recovery of Triple-Erasure-Correcting Code Storage Systems

QIU Li-Na WANG Fang LI Chu
(*School of Computer, Huazhong University of Science and Technology, Wuhan* 430074)
(*Wuhan National Laboratory for Optoelectronics, Wuhan* 430074)
(*Key Laboratory of Data Storage Systems of Ministry of Education, Wuhan* 430074)

Abstract Modern storage systems apply erasure codes to protect against disk failures and improve system reliability and availability. MDS codes such as STAR code, Triple Parity (TP) have attracted to more and more attention because they are triple-erasure-correcting and offering higher reliability than RAID6 codes. It has not been solved effectively yet about how to complete fast recovery from single failure in systems using triple-erasure-correcting codes. There exist many recovery solutions and it makes great influence on recovery performance and response time of requests from foreground. Conventional scheme of recovering system from single failure is time consuming and bandwidth wasting. To address this problem, we propose an Equal Division Scheme (EDS) for triple-erasure-correcting codes to realize fast single failure recovery. Our scheme deduced the amount of data transmitted for single failure recovery through using recovery parity collection in Equal Division Status and then found out an optimal recovery collection to reduce disk I/O time so that recovery process is boosted. Experiment results show that EDS

收稿日期:2013-06-26;最终修改稿收到日期:2013-09-01.本课题得到国家"九七三"重点基础研究发展规划项目基金(2011CB302301)、国家"八六三"高技术研究发展计划项目基金(2013AA013203)、国家自然科学基金(61025008,60933002,61232004)资助.**邱丽娜**,女,1989年生,硕士研究生,主要研究方向为RAID编码、海量网络存储系统和并行存储系统.E-mail:linaQ@hust.edu.cn.**王 芳**,女,1972年生,教授,博士生导师,中国计算机学会(CCF)会员,主要研究领域为海量网络存储系统、并行存储系统和存储系统能耗.**李 楚**,男,1989年生,博士研究生,主要研究方向为海量网络存储系统、并行存储系统、存储可靠性研究.

consumes 25%—30.6% less data transmission approximately for TP and 30%—33.64% less for STAR than the conventional strategy and reduces recovery time observably. And with different chuck sizes and different disk numbers, EDS outperforms conventional remarkably.

Keywords RAID; reliability; disk failure; recovery algorithm

1 引 言

现代存储系统在存储数据时不仅会将数据条带化处理以提高访问性能,还往往存储冗余信息以保证系统可靠性和数据的可用性.备份和纠删码是构成冗余信息的两种方式,因备份会造成存储空间的极度浪费,纠删码成为存储冗余信息的优选,相关研究也层出不穷.长期以来,针对容两盘错的 RAID6 编码的研究占据了主流.

但随着分布式文件系统的广泛应用和存储规模的扩大,传统 RAID6 方案已满足不了需求.容三盘错的纠删码,如 STAR 编码[1]、Triple Parity[2](TP)编码和扩展的 EVEVODD 编码[3],可以向系统提供高于 RAID6 方案的可靠性,使系统即便在三个存储节点的数据同时失效时也能利用幸存节点上的数据恢复所有失效数据.由于 Chen 等人[4]指出磁盘失效的发生不是独立存在的,一旦出现第 1 个失效磁盘,其它磁盘失效的概率会大大增加,更多磁盘失效会随后发生,这极大地提高了失效磁盘的总数超出系统容错能力的概率,一旦系统中失效磁盘的个数高于它们的容错级别时,系统将不可修复,数据会永久丢失.因此,当系统中出现第 1 个磁盘失效时,需要尽快修复至正常状态.现阶段关于容三盘失效纠删码的研究[1-3]大多关注于创建新的编码、解码规则来降低计算复杂度和更新数据时的复杂度,很少关注单盘失效时的快速重建.

另外,尽管单节点失效的快速重建已经得到充分的研究,但大部分是基于 RAID6 编码方法的,很少有针对容三盘失效纠删码的重建研究.例如,RDOR[5]方法是为 RDP[6]的单节点失效所设计,PDRS[7]是为 P-Code[8]和 X-Code[9]所设计等.Wang 等人[10]提出了类似的方法.Zhu 等人[11]解决使用 RAID6 编码的异构环境下的单点故障快速重建问题.NCCloud[12]中的 F-MSR(Functional Minimum-Storage Regenerating Code)编码能够快速重建网络存储中的单点故障,但它的容错能力和存储性能仍是 RAID6 级别的.面对容三盘失效的编码在单盘出错的情况,也有一些相关方法,但这些方法均不是最佳的.采用传统方法重建时会从磁盘上反复读取已读取过的数据,数据有效利用率低,重建时间长.Khan 等人[13-14]提出枚举所有重建方案以得到最佳重建方案(即从磁盘读取数据量最少的方案),但求取最佳方案的过程耗时巨大,尤其是当存储系统中节点数目较大时.

单盘失效时系统需要以最快的速度重建的另一个重要原因是多数情况下,重建时,系统仍需要响应来自用户的请求.用户读/写请求会与重建进程 I/O 请求竞争磁盘 I/O,不仅导致重建过程延长,而且用户请求相应时间也延长,降低了服务质量.因此,为了尽快向用户提供高质量的服务减少用户的不满意度,单节点失效时需要尽可能地减少重建时间,使系统尽快恢复至正常状态.

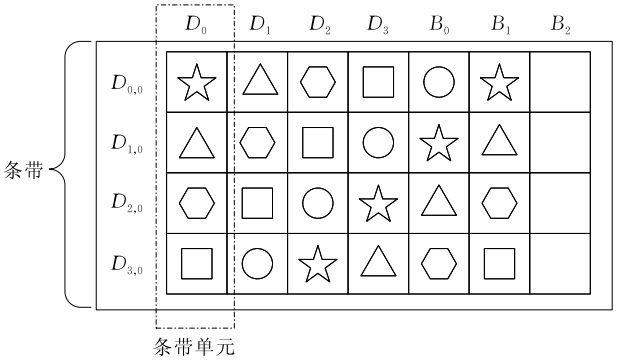
容三盘失效的编码系统中有 3 种校验方式,传统方法只使用其中的一种校验方式进行重建计算.事实上,同时使用 3 种校验方式重建会找到最佳重建方案,提高数据有效利用率.最佳重建方案需要从磁盘读取的数据量(即最小数据量)不能直接求得,但可以推导出范围.基于以上发现,我们提出均分机制方法,经过数学分析和推理定位最小数据量的取值范围,依次经过启发式搜索快速返回最佳重建方案,从而 TP 编码在重建时从磁盘读取的数据量减少大约 25%～30.6%,减少磁盘 I/O 时间,使系统快速重建至正常状态.该方法同样适用于 STAR 编码、通用 EVENODD 编码等常用的容三盘错的编码系统.

2 相关研究

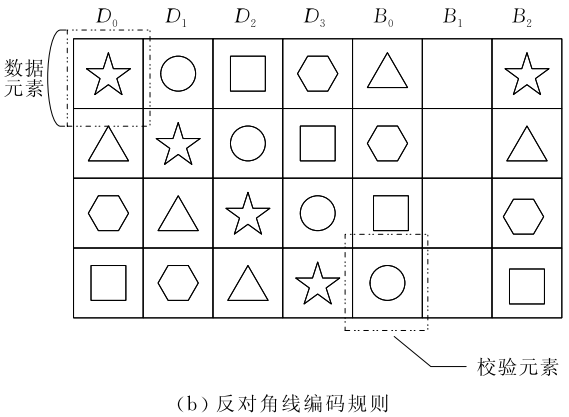
2.1 定 义

本文中我们按照 Plank 使用的编码术语来描述[15].假设一个由 n 个磁盘构成的存储系统,其中 k 个磁盘存储原始数据,剩余的 $n-k$ 个磁盘存储校验数据.每个磁盘容量相同,划分成多个由 w 个元素组成的条带单元,每个元素代表一定数量的数据块.在该磁盘阵列中,所有磁盘上位于相同偏移的不同

条带单元构成一个条带，如图 1 所示. 在一个条带内部，数据元素根据一定的编码规则生成相应的校验元素，它们共同组成一个校验组. 数据的编解码是以条带为单元进行的，不同条带间编解码相互独立. 不失一般性地，在讨论最佳重建方案时我们假设磁盘阵列只包含一个条带，用一个 R 行 n 列的阵列，即 $R\times n$ 阵列描述此单一条带内部结构.



(a) 对角线编码规则



(b) 反对角线编码规则

图 1　TP 的对角线校验和反对角线校验的编码规则

2.2　容三节点失效纠删码

本文主要研究 TP 编码和 STAR 编码的单点失效快速重建，因为相比其它纠删码，TP 和 STAR 的编码和解码复杂度非常低，而这是影响人们选择同一级别的纠删码的重要因素.

TP 编码是由 RDP 编码衍生出来的，它包含 3 种斜率的校验：行校验、对角线校验和反对角线校验. 在采用 TP 编码的系统中，一个条带可以用 $(p-1)\times(p+2)$ 阵列来描述，p 是大于 2 的素数. 图 1 描述的是在 $p=5$ 时，由 7 个磁盘构成的系统中单一条带的示意图，前 4 列代表数据盘，后 3 列分别代表行校验、对角线校验和反对角线校验的校验盘. 为方便表述，用 $D_{i,j}$ 表示位于第 i 行，第 j 列的数据元素，其中 $0\le i\le p-2$，$0\le j\le p-2$；$B_{i,v}$ 表示位于第 i 行，第 $(p-1+v)$ 列的校验元素，其中 $0\le i\le p-2$，$0\le v\le 2$. 实际上，当 $v=0$ 或 1 时，$B_{i,v}$ 所在的校验组的斜率是 v，$B_{i,v}$ 代表行校验元素或对角线校验元素；当 $v=2$ 时，$B_{i,v}$ 所在的校验组的斜率是 -1，$B_{i,v}$ 代表反对角线校验元素. 我们将 $B_{i,v}$ 所在的校验组称为 $B_{i,v}$ 校验组. 行校验的编码规则是由同一行的数据元素异或计算出行校验元素，如 $D_{0,0}+D_{0,1}+D_{0,2}+D_{0,3}$ 求得 $B_{0,0}$. 对角线和反对角线校验的编码规则如图 1 所示.

STAR 纠删码的编码规则与 TP 类似，但 STAR 中的条带是 $(p-1)\times(p+3)$ 阵列，且 STAR 的行校验元素不像 TP 编码中那样参与计算对角线/反对角线校验元素.

2.3　相关研究

当磁盘阵列中的失效磁盘为校验盘时，校验盘上的校验信息可以按照最初计算的过程再次求解，这是唯一的方法，没有优化的空间. 本文关注的是失效数据盘的单盘快速重建方法.

(1) 传统重建方法. 传统方法读取行校验元素和同一校验组下有效的数据元素，经过异或运算，求得失效的数据元素，失效条带单元上的所有数据元素都根据此过程重建. 以图 1 为例，假设数据盘 D_0 失效，4 个数据元素 $D_{0,0}$，$D_{1,0}$，$D_{2,0}$，$D_{3,0}$ 的数据失效. 恢复 $D_{0,0}$ 的数据，传统方法从磁盘上读取 $D_{0,1}$，$D_{0,2}$，$D_{0,3}$ 和 $B_{0,0}$ 进行异或运算；类似地，恢复 $D_{1,0}$ 的数据，传统方法从磁盘上读取 $D_{1,1}$，$D_{1,2}$，$D_{1,3}$ 和 $B_{1,0}$ 进行异或运算；恢复 $D_{2,0}$ 的数据，传统方法从磁盘上读取 $D_{2,1}$，$D_{2,2}$，$D_{2,3}$ 和 $B_{2,0}$ 进行异或运算；恢复 $D_{3,0}$ 的数据，传统方法从磁盘上读取 $D_{3,1}$，$D_{3,2}$，$D_{3,3}$ 和 $B_{3,0}$ 进行异或运算. 重建整个条带，需要从磁盘读取的数据量是 16 个元素(包含数据元素和校验元素).

传统重建方法只使用行校验一种校验方式，忽略了在容三盘出错纠删码中一个数据元素受到多个不同的校验组保护的现象，不是一种高效快速的重建方法. 当采用多种校验方式来重建时，不同校验方式的校验组之间共同包含的数据元素只需读一次，可有效地减少磁盘数据读取量.

(2) 枚举法. 枚举法基于编码的产生矩阵列出所有可能重建方案，从中取出最佳的方案. 这种方法对任何级别的基于异或运算的纠删码均适用.

以图 2 为例说明枚举法的详细步骤. 图 2 描述了 TP($p=5$) 通过产生矩阵生成校验的规则. 产生矩阵和数据元素表示的列向量相乘求得编码向量. 阴影小方格表示取值 1，白色小方格表示取值 0. 假设磁盘 D_0 失效，需要重建其上的 4 个数据元素

$D_{0,0}$,$D_{1,0}$,$D_{2,0}$,$D_{3,0}$. 枚举法使用重建方程减少重建所需元素的数目. 重建方程由一组元素组成,这些元素对应于产生矩阵中的行向量的异或值为 0. 例如 $D_{0,0}$,$D_{0,1}$,$D_{0,2}$,$D_{0,3}$ 和 $B_{0,0}$ 所在的行向量的异或值为 0,故它们组成一个重建方程. 我们可以使用重建方程中其余幸存的元素来恢复任意一个失效元素. 例如,当 $D_{0,0}$ 失效时,$D_{0,1}$,$D_{0,2}$,$D_{0,3}$ 和 $B_{0,0}$ 通过异或运算可以恢复 $D_{0,0}$. 需要注意的是,$D_{0,0}$ 可与其它元素组成重建方程,故也可通过其它重建方程重建. 当重建失效磁盘 D_0 时,首先根据产生矩阵,枚举出所有的重建方程,然后按以下步骤恢复 D_0 上失效元素:给出 4 个集合 E_0,E_1,E_2 和 E_3,E_i 是失效元素 $D_{i,0}(0\leq i\leq 3)$ 所在的所有重建方程组成的集合;从每个 E_i 中选取一个重建方程 e_i 使得所有 e_i 组成的并集中元素数目最少. 本例中,通过枚举法获得的一个最佳重建方案如下:

① e_0: $D_{0,0}$, $D_{0,1}$, $D_{0,2}$, $D_{0,3}$, $B_{0,0}$

② e_1: $D_{1,0}$, $D_{1,1}$, $D_{1,2}$, $D_{1,3}$, $B_{1,0}$

③ e_2: $D_{2,0}$, $D_{1,1}$, $D_{0,2}$, $B_{3,0}$, $B_{2,1}$

④ e_3: $D_{3,0}$, $D_{2,1}$, $D_{1,2}$, $D_{0,3}$, $B_{3,1}$

e_0,e_1,e_2 和 e_3 的并集除去 4 个失效元素 $D_{0,0}$,$D_{1,0}$,$D_{2,0}$,$D_{3,0}$ 和 4 个公共元素 $D_{1,1}$,$D_{0,2}$,$D_{1,2}$,$D_{0,3}$ 共有 12 个元素,使用这 12 个元素可以完成磁盘 D_0 的重建. 然而,枚举法需要列出每个失效元素的所有重建方程,之后通过枚举求出使得恢复各个失效元

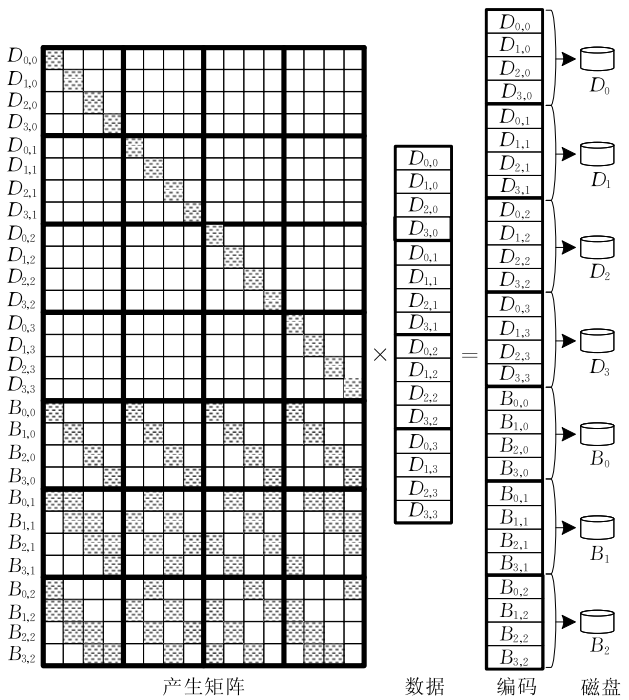


图 2 通过产生矩阵计算编码

素的重建方程的并集中元素个数最小的组合. 当磁盘个数增加时,重建方程的个数指数增长,耗时巨大. 通常情况下,使用枚举法找最佳重建方案的问题是 NP 问题.

3 均分机制方法

TP 编码规则的图解如图 1 所示,求解校验的数学公式表述如下(〈〉代表模 p 运算):

$$B_{i,0}=\sum_{j=0}^{p-2}D_{i,j} \tag{1}$$

$$B_{i,v}=\sum_{j=0}^{p-2}D_{\langle i+(-1)^v\times j\rangle,j}+B_{\langle i-(-1)^v\rangle,0}(v=1\text{ 或 }2) \tag{2}$$

$B_{i,v}$ 代表位于第 i 行,第 $(p-1+v)$ 列的校验元素,"+"和"$\sum$"表示异或运算.

单个磁盘失效时,表现为存储在其上的条带单元上的数据不可用,重建需要恢复该条带单元上 $(p-1)$ 个数据元素. 每个失效的数据元素可以通过将其所在的校验组中其余幸存的元素进行异或运算而恢复. 因此,我们的目标是选取 $(p-1)$ 个合适的校验组,既可以将失效元素全部重建,又可以使重建成本最小,重建速度最快. 传统方法选取的是 $(p-1)$ 个行校验组,根据式(1)可知,$(p-1)$ 个行校验组包含的数据元素和校验元素的个数之和(除去失效元素)为 $N=(p-1)\times(p-1)$,且这 $(p-1)$ 个行校验组彼此不相交,彼此没有公用的数据元素. 然而某些校验组之间拥有公用的数据元素,若选取合适的满足重建要求的 $(p-1)$ 个校验组(不仅限于行校验组),使这些校验组之间的交点最多,则从磁盘上需要读取的元素的总数 R 会达到最小,数值为 $(N-\text{交点总数}\sum)$. 下面,我们分析影响交点个数的因素.

从式(1)和式(2)得出,对所有的 $j=0,1,\cdots,p-2$ 来说,都满足如下现象:每个行校验组 $B_{i,0}$ 包含一个数据元素 $D_{i,\langle j-i\rangle}\in B_{j,1}$ 对角线校验组,每个对角线校验组 $B_{i,1}$ 包含一个数据元素 $D_{j,\langle i-j\rangle}\in B_{j,0}$ 行校验组. 因此,每个对角线校验组 $B_{i,1}(i=0,1,\cdots,p-2)$ 与每个行校验组 $B_{j,0}(j=0,1,\cdots,p-2)$ 均相交于一个数据元素,交点有且仅有一个,我们把相交的元素称为公共块. 同理,每个反对角线校验组 $B_{i,2}(i=0,1,\cdots,p-2)$ 与每个行校验组 $B_{j,0}(j=0,1,\cdots,p-2)$ 也均有一个公共块. 而并非每个对角线

校验组 $B_{i,1}$ 与每个反对角线校验组 $B_{j,2}$ 均相交，当 i 和 j 的关系满足公式

$$\langle i+j\rangle=p-2 \text{ and } 0\leq i,j\leq p-2 \tag{3}$$

时，两者相交于编码结构的虚拟行第 $p-1$ 行，产生的交点称作虚拟公共块. 由于虚拟公共块不是真实存在的，求交点总数时应减去它们的个数. 同一类型的不同校验组之间彼此不相交，例如所有的对角线校验组彼此之间不存在公共块.

假设我们选取的用以重建的校验集合为包含 $(p-1)$ 个校验组的集合，其中有 α 个行校验组，β 个对角线校验组，γ 个反对角线校验组，且有 $\alpha+\beta+\gamma=p-1$. 由于任意一个行校验组同任意一个对角线校验组产生一个公共块，则 α 个行校验组与 β 个对角线校验组产生 $\alpha\times\beta$ 个公共块；由于任意一个行校验组同任意一个反对角线校验组产生一个公共块，则 α 个行校验组与 γ 个反对角线校验组产生 $\alpha\times\gamma$ 个公共块. 由于对角线校验组和反对角线校验组并不总是产生有效的公共块（即会产生虚拟公共块），产生虚拟公共块的对角线校验组和反对角线校验组的组合个数为 η，且 $0\leq\eta\leq\dfrac{(p-1)}{2}$，因此 β 个对角线校验组和 γ 个反对角线校验组产生的公共块总数至少为 $\beta\times\gamma-\eta$. 值得注意的是，有些公共块是 3 个校验组的交点，由集合理论知应从中减去这类公共块的个数 ϕ 才得到正确的交点总数. 因此我们得出式(4)，R 表示重建单条带所需的元素总数（也可称为数据量）.

$$\begin{aligned}R&\geq(p-1)\times(p-1)-(\alpha\beta+\alpha\gamma+\beta\gamma-\eta-\phi)\\&\geq(p-1)^2+\underbrace{(\alpha+\beta)^2-\alpha\beta}_{4\alpha\beta\leq(\alpha+\beta)^2}-(p-1)(\alpha+\beta)\\&\geq(p-1)^2+\frac{3}{4}(\alpha+\beta)^2-(p-1)(\alpha+\beta)\\&=\frac{2}{3}(p-1)^2+\frac{3}{4}\left[(\alpha+\beta)-\frac{2}{3}(p-1)\right]^2\\&\geq\frac{2}{3}(p-1)^2\end{aligned}\tag{4}$$

式(4)推算出单条带的单盘重建时，需要从磁盘上读取的元素总数 R 的下界. 当 α,β,γ 的值彼此接近时，R 的取值会无限逼近下界 $\frac{2}{3}(p-1)^2$. 因为 $\alpha+\beta+\gamma=p-1$，故 $\left\lfloor\dfrac{p-1}{3}\right\rfloor\leq\alpha,\beta,\gamma\leq\left\lceil\dfrac{p-1}{3}\right\rceil$. 因此我们重建策略选取 3 类校验组个数 α、β 和 γ 的值彼此接近的策略，我们定义同时包含 3 类校验组且每种类型的校验组个数彼此接近的重建校验集合为处于“均分”状态下的重建校验集合.

单盘失效情况下，为使从磁盘读取的用以重建的数据量 R 尽可能最小，重建时采用的校验集合是 $B_{3n+v,v}$，其中 $v=0$、1 或 2，$0\leq3n+v\leq p-2$. 我们选取的校验集合中，任意一个对角线校验组和任意一个反对角线校验组均不满足式(3)，所以不存在虚拟公共块. 接下来分析采用 3 类校验集合重建时，R 的值是否可以确定下来.

若 3 个校验集合 $B_{3n,0}$，$B_{3m+1,1}$，$B_{3l+2,2}$ 在第 y 列有个公共块，由式(1)和(2)得出，$3n\equiv3m+1-y\equiv3l+2+y\pmod p$. 因此，$2y\equiv6(m-n)+2\equiv3(m-l)-1\pmod p$，继续推导出 $n-l\equiv m-n+1\pmod p$. 由于 $0\leq m,n,l<p/3\Rightarrow-p/3<n-l,m-n<p$，则 $n-l=m-n+1\Rightarrow2n=m+l+1$，且 $m+l$ 的和必须是奇数. 若 m 的值固定，令 $K=\left\lfloor\dfrac{p-1}{3}\right\rfloor$. 则 (m,n,l) 的取值有两种情况：(1) $0\leq m<n\leq l\leq K$，(2) $0\leq l<n\leq m\leq K$. 若 $0\leq m<n\leq l\leq K$，满足 $2n=m+l+1$ 的 (n,l) 对的个数不大于 $(K-m-1)/2+1$；若 $0\leq l<n\leq m\leq K$，满足 $2n=m+l+1$ 的 (n,l) 对的个数不大于 $(m+1)/2$. 因此，满足等式 $2n=m+l+1$ 的 (n,m,l) 取值的个数为

$$\phi\leq\sum_{m=0}^{K}\left(\frac{K-m-1}{2}+1+\frac{m+1}{2}\right)=\frac{K^2}{2}+\frac{3}{2}K+1 \tag{5}$$

由式(5)知，ϕ 的取值只跟 p 的取值有关. 且有单节点故障情况下，重建单条带需从磁盘读取的数据量 R 满足以下公式：

$$\begin{aligned}R&=(p-1)(p-1)-(\alpha\beta+\alpha\gamma+\beta\gamma-\phi)\\&<(p-1)^2-(K^2+K^2+K^2)+\frac{1}{2}K^2+\frac{3}{2}K+1\\&<\frac{13}{18}p^2+\frac{31}{18}p-\frac{76}{9}\end{aligned}\tag{6}$$

其中 $(p-4)/3<K\leq(p-1)/3$. 传统方法中，仅仅使用一类校验组来重建单条带中的失效的数据条带单元时，从磁盘读取的数据量是 $N=(p-1)^2$ 个元素；由式(4)和(6)知，若使用 3 类校验组且每种类型的校验组个数彼此接近时，R 的取值介于 $R_{\text{low}}=2N/3$ 和 $R_{\text{high}}=13N/18$ 之间，与传统方法相比降低了 27.8%～33.3%. 事实上，虽然从磁盘读取的数据量的最小值 R 不能直接确定，但由于

$$\Delta=\frac{R_{\text{high}}-R_{\text{low}}}{R_{\text{low}}}\times100\%=8.33\% \tag{7}$$

R_{high} 相对 R_{low} 增长了 8.33%，取增长率的平均值 $w=4.2\%$ 来确定 R 的平均值 R_{avg}，以此作为在单条带的单节点快速重建情况下，从磁盘读取的数据量的最优解. 在容三盘纠删码存储系统中单盘失效的

情况下,同时使用行校验组,对角线校验组和反对角线校验组3类校验组进行重建,且3类校验组的个数彼此接近,这是均分机制方法的基础,也是核心.

$$R_{avg}=R_{low}(1+w) \tag{8}$$

由于重建校验集合处于"均分"状态是该重建校验集合是最佳重建校验集合的必要不充分条件,因为有些均分校验集合不能完全覆盖所有失效的数据元素而不足以用来重建.在提出均分机制方法的具体算法之前,首先探讨失效的数据元素被哪几种校验组覆盖.

在一个$(p-1)\times(p+2)$的TP编码的条带中,假设第$k(0\leq k\leq p-2)$列的条带单元失效,其包含的$p-1$个数据元素$D_{i,k}(0\leq i\leq p-2)$需要恢复.定义重建校验集合$\pi=\{x_i\mid 0\leq i\leq p-2, x_i=0,1 \text{ or } 2\}$描述重建方案,$x_i$表示在重建失效的条带单元所包含的第$i$行数据元素时,需要使用该数据元素所在的何种类型的校验组.若x_i分别取值0、1和2,则分别代表需使用该数据元素所在的行校验组,对角线校验组和反对角线校验组恢复该数据元素的数据.

引理1.

(1)若$i=p-1-k$,失效的数据元素$D_{i,k}$可以使用其所在的行校验组$B_{i,0}$和其所在的反对角线校验组$B_{\langle i-k\rangle,2}$恢复失效数据.

(2)若$i=\langle k-1\rangle$,失效的数据元素$D_{i,k}$可以使用其所在的行校验组$B_{i,0}$和对角线校验组$B_{\langle i+k\rangle,1}$恢复失效数据.

(3)若$i\neq p-1-k$ and $i\neq\langle k-1\rangle$,失效的数据元素$D_{i,k}$可以使用其所在的行校验组$B_{i,0}$,对角线校验组$B_{\langle i+k\rangle,1}$和反对角线校验组$B_{\langle i-k\rangle,2}$任意一种恢复失效数据.

情况(1)和情况(2)的存在是由于这些失效数据元素正好位于不参与计算校验的缺失对角线或缺失反对角线上,这些数据元素只参与计算到两种类型的校验组中,受两个校验组的保护.情况(3)是大多数情况,这些数据元素参与计算3种类型的校验组而受3种校验组的保护.对单一条带的$(p-1)$个数据条带单元而言,除第0列的条带单元的所有数据元素均受3个校验组保护,其余列的条带单元均会有且仅有两个数据元素分别满足情况(1)和情况(2)的情况,其余$(p-3)$个数据元素满足情况(3)而受到3种校验组的保护.

下面依据引理1举例验证校验组的可行性.假设图1表示的存储系统中编号为$k=1$的磁盘因软硬件故障而失效,反映到图1为编号为D_1的条带单元失效,有$D_{0,1}$、$D_{1,1}$、$D_{2,1}$、$D_{3,1}$ 4个数据元素需要恢复.$D_{0,1}$由于$i=0=\langle k-1\rangle$,根据情况(2)只能通过行校验组$B_{0,0}$和对角线校验组$B_{1,1}$恢复,在重建方案中$x_0=0$或$x_0=1$;$D_{1,1}$和$D_{2,1}$符合情况(3),均能通过3种校验组恢复,它们分别是$B_{1,0}$、$B_{2,1}$、$B_{0,2}$恢复$D_{1,1}$和$B_{2,0}$、$B_{3,1}$、$B_{1,2}$恢复$D_{2,1}$,在重建方案中x_1和x_2的值均可取0、1或2中任意一个;$D_{3,1}$由于$i=p-1-k$,根据情况(1)只能通过行校验组$B_{3,0}$和反对角线校验组$B_{2,2}$恢复,在重建方案中$x_3=0$或$x_3=2$.给出一条重建校验集合$\pi_0=\{2001\}$,尽管π_0处于"均分"状态,但基于引理的分析,π_0是非可行的,不足以重建失效条带单元中的所有元素,因此该集合绝非最佳重建方案.

若重建校验集合处于"均分"状态且可行并且依据该校验集合从磁盘读取的数据量等于R_{avg},则该校验集合是最佳重建校验集合.算法1给出求解最佳重建校验集合的算法.

算法1. 均分机制方法(EDS).

输入:素数p,失效节点的编号k

输出:最佳重建校验组

```
1. Initialize N=3^(p-1), sign=true
2. step:
3. FOR i=0 to N
4. /*计算重建校验组 S*/
5.    convert i into an integer sequence S
6. /*检验重建校验组 S是否处于"均分"状态且可行*/
7.  IF Filter (S,k)==false THEN
8.     GOTO step
9.   END IF
10. /*计算重建校验组 S需要从磁盘读取的元素的个
      数 R*/
11. compute R=number of element reads for S;
12. /*检验 R是否小于等于最优解 R_opt*/
13. IF R<=R_opt THEN
14. RETURN S;
15. ELSE
16.  GOTO step;
17. END FOR.
```

4 性能评估

本节首先实现了TP编码单盘失效重建的均分机制方法、枚举法和传统法,并对均分机制方法和枚举法在搜索到最优重建校验集合过程中所消耗的时间和各自最优重建校验集合在重建时从磁盘读取的元素数目进行对比.实验还对比了均分机制方法和

传统法从磁盘读取的元素数目的差异，并与理论分析的最小值进行对比.

为了证明均分机制方法的有效性，即它能通过减少重建过程中元素的读取数目，减少磁盘读数据量，而减少重建过程的总时间，实验对比测试了均分机制方法和传统方法在重建相同数量的数据情况下所花费的时间. 为了验证算法适用性，同时对 TP 编码和 STAR 编码进行测试，并测试在不同大小的条带单元和不同个数的磁盘数目时的重建时间. 另外，为了验证均分机制方法对在线重建的适用性，实验测试了 TP 编码和 STAR 编码在不同负载下的重建时间. 实验选取了来自多个国际企业数据中心的 5 种工作负载①：financial1 (fin1)，financial2 (fin2)，websearch1 (web1)，websearch2 (web2)，websearch3 (web3). 两个 financial 的 I/O trace 来自于两个大型财务公司的在线交易处理软件，3 个 websearch 的 I/O trace 来自于一个广泛应用的搜索引擎. 这些 trace 被广泛应用于各项研究[16-17].

4.1　实验设置及测试方法

搜索性能评估和重建数据量评估实验在配置为 Intel Xeon E5620 2.4GHz CPU 和 4GB RAM 的 Linux 服务器上进行. DiskSim[18] 是卡耐基梅隆大学研发的磁盘系统模拟器，广泛应用于存储系统结构的各项研究中. 本节中重建性能评估用 DiskSim 模拟磁盘访问，所模拟的磁盘规格为 15000-RPM 的 Seagate Cheetah，容量 146 GB. 测试 TP 编码时，使用 $(p+2)$ 个磁盘；测试 STAR 编码时，使用 $(p+3)$ 个磁盘. 重建过程是在离线模式下进行，即在重建过程中没有来自前端访问的请求. 磁盘上的所有数据根据相应的编码规则条带化存放于各个磁盘. 在衡量重建性能时，我们以重建每 MB 数据所消耗的时间作为标准. 测试方法如下：首先将 TP 编码和 STAR 编码的均分机制方法和传统方法均实现到 DiskSim 环境中；重建每个条带时，内存一旦计算完毕失效条带单元的数据，便立即将它们写入备份的数据盘. 为了确保实验结果的可信性，获得每个重建时间均执行 4 次程序，每次均在所有数据磁盘上执行一遍，即假定所有数据盘失效的概率是相等的. 例如 TP 编码 $(p=7)$，一共有 6 个数据磁盘，在求取单盘失效的重建时间时，一共执行了 4×6 次操作，然后在 24 次结果上取平均作为最终结果.

实验过程中，我们假定每个元素由一个数据块构成，且数据块的大小通常会较大. 这是由于现代的分布式系统中往往对大数据块进行操作，例如 GFS[19] 使用数据块的大小是 64 MB. 4.4 节将描述数据块的大小对重建性能的影响.

4.2　搜索性能评估

本文针对均分机制方法和枚举法返回最佳重建序列的搜索时间进行测试，结果如表 1 所示. 枚举法无法预先锁定最佳重建校验集合所处的状态，随着 p 的增长，重建方程的数目爆炸式增长且枚举法需要枚举出所有重建方程的组合才能确定下来最优的组合，导致搜索时间指数攀升. 均分机制方法预先将最佳重建数组锁定在处于均分状态且可行，且由于从磁盘读取的最佳数据量已确定，当符合以上条件的重建校验集合出现时立即停止搜索，作为最佳重建校验集合返回. 与枚举法相比，均分机制方法的搜索时间大大减少.

当 $p\leq17$ 时，枚举法的搜索时间较短；当 $p=19$ 时，枚举法需要花费超过 18 min 的时间找到最佳的重建数组；当 $p=23$ 时，搜索时间接近 2 天 3 个小时. 当 $p\leq23$ 时，搜索到最佳重建数组所花费的时间最多不超过 5 s，大部分是 ms 数量级.

表 1　搜索性能对比

p	时间 (Enumeration)	时间 (EDS)
5	51 μs	28 μs
7	421 μs	233 μs
11	59.9 ms	1.66 ms
13	724 ms	4.12 ms
17	98.7 s	79.5 ms
19	18 min 37 s	446.3 ms
23	2 d 2 h 47 min 12 s	4.96 s

4.3　重建数据量评估

首先使用仿真实验比较重建时分别使用传统方法，均分机制方法从磁盘读取数据量，并与数据量的下界对比，结果如图 3 所示. 当 $p<40$ 时，均分机制方法所需的数据量与下界十分逼近.

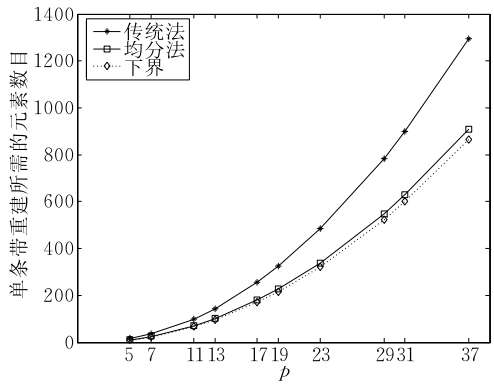


图 3　TP 编码单条带重建时所需从磁盘读取的元素的数目

① http://traces.cs.umass.edu/index.php/Storage/Storage

为了验证均分机制方法的有效性，我们也评估了枚举法与传统方法相比的节省量，实验结果如图4所示.当 $p=5$ 和 $p=11$ 时，均分机制方法的表现和枚举法一样优，均能达到最佳重建数组.当 p 取其它值时，均分机制方法达到次优解，但与枚举法的最优解的差异仅在0.7%～2.8%之间.与传统方法相比时，均分机制方法节省了25%～30.6%的元素读取量；枚举法节省了25%～31.5%的元素读取量.若系统中单个节点的容量是1 TB，共有13个磁盘（$p=11$），其中有10个数据盘.单节点失效重建时，采用传统方法从磁盘读取数据共10 TB，而采用均分机制方法从磁盘读取数据只需6.9 TB.目前磁盘I/O速度为100 MB/s，为了不影响前台服务，重建速度配置为30 MB/s时，采用均分机制方法重建可因少读取3.1 TB的数据使得重建时间减少2.87 h（$3.1\times10^6/(10\times30\times3600)$），这里假设读磁盘的过程为并发多线程读，且所有磁盘读的时间相等.均分机制方法使得从磁盘读取的数据量大幅减少，加速了重建过程，大大提高了重建性能.

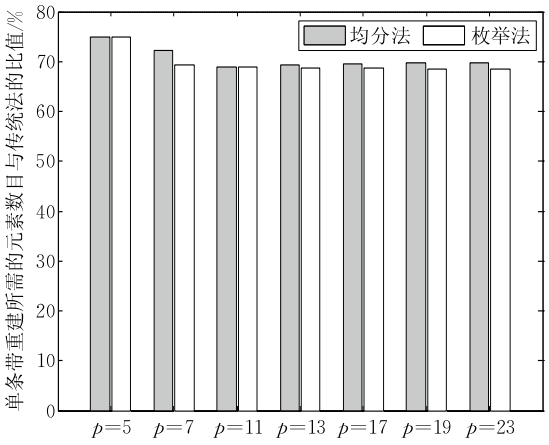


图4 TP枚举法和均分法重建时所需元素的数目

使用STAR编码的存储系统在单节点失效时，也可采用均分机制方法进行最优重建校验集合的查找.我们也评估了STAR编码分别使用均分机制方法和枚举法的重建性能，并与传统法进行比较，结果如图5所示.总体来看，均分机制方法和枚举法的表现非常接近，按照两者寻找到的最佳重建数组重建时从磁盘读取的数据量（每条带）与传统法比分别节约30%～33.64%和33.33%～35.71%.如图5，$p=7$ 时，虽然均分机制方法节省的百分比相较于枚举法少5%，但事实上两者的最优重建数组所需的数据量仅仅相差1.当 p 越大时，均分机制方法的表现越趋于枚举法，能找到全局最优的重建校验集合.

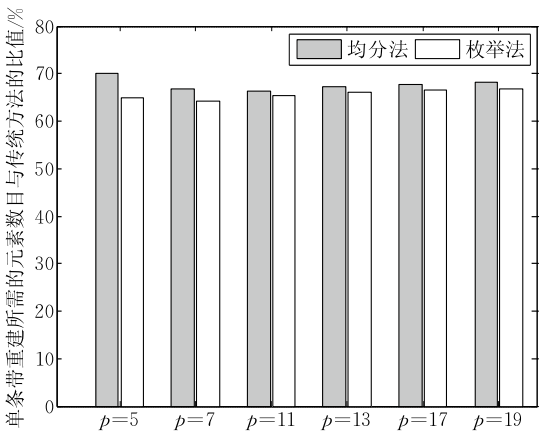


图5 STAR枚举法和均分法重建时所需元素的数目

4.4 重建性能评估

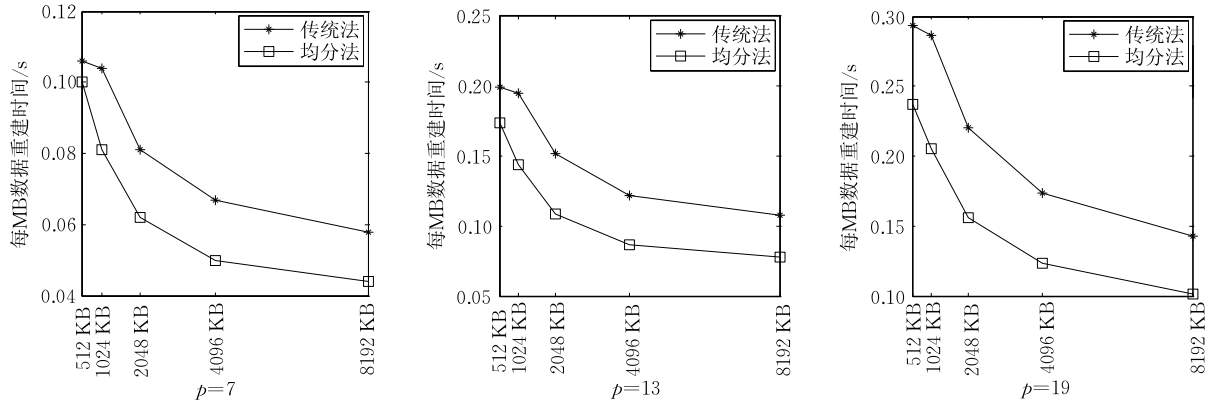
实验1. 数据块的大小对性能的影响.实验时我们固定 p 的值，假定 p 取值分别为7、13和19，每个 p 对应一定的磁盘数目，选取不同大小的数据块，从512 KB变化到8196 KB，来评估数据块的大小对重建性能的影响.

图6(a)和(b)分别表示使用传统方法和均分机制方法重建每MB数据所消耗的时间，实验结果表明数据块大小在512 KB到8192 KB间变化时，均分机制方法均优于传统方法，即重建相同的数据耗时更短.当数据块的大小增加时，每MB数据的重建时间会减少，并且减少的速率随着数据块的增加而变缓.因此我们预测当数据块的大小增加到一定数值时，每MB数据的重建时间趋于某一稳定值.如图6所示，TP和STAR的传统方法和均分机制方法均符合上述趋势.

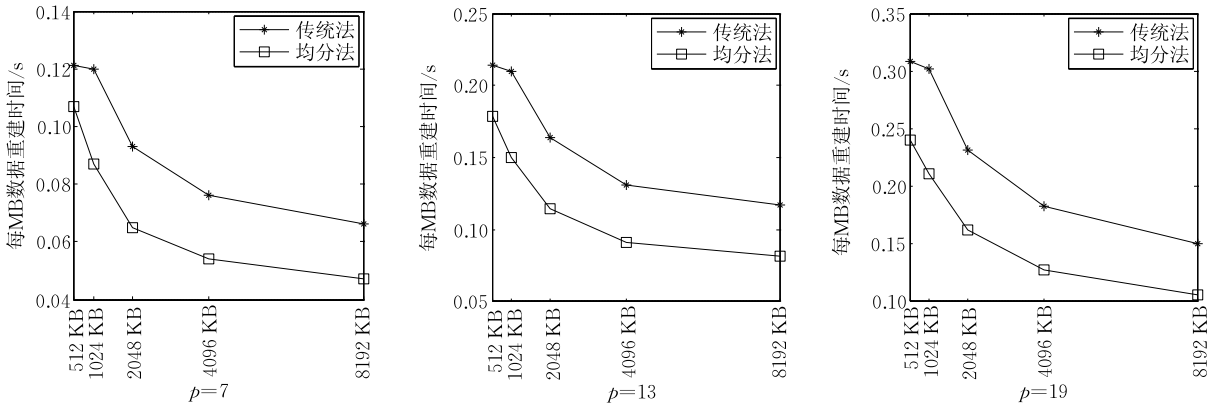
实验2. 磁盘个数对重建性能的影响.我们分别测试了TP和STAR在不同 p 对应的磁盘个数下的重建性能.实验时我们固定数据块的大小，假定数据块大小取值分别为512 KB、1024 KB和2048 KB，选取不同的素数 p，依次为7、11、13、17和19，来评估磁盘个数对重建性能的影响，实验结果如图7所示.

对TP编码和STAR编码来说，与传统方法相比，均分机制方法均减少了每MB数据的重建时间，这是由于均分机制方法能显著减少重建过程中从磁盘读取的数据量.以TP编码为例，数据块大小为512 KB时，每MB数据重建时间减少6.00%～19.3%；数据块大小为1024 KB时，每MB数据重建时间减少22.1%～27.3%；数据块大小为2048 KB时，每MB数据重建时间减少24.5%～29.0%.

图7(b)表示STAR编码的测试结果，使用均分机制方法重建时每MB数据的重建时间明显减少，说明均分机制方法对STAR编码同样适用.

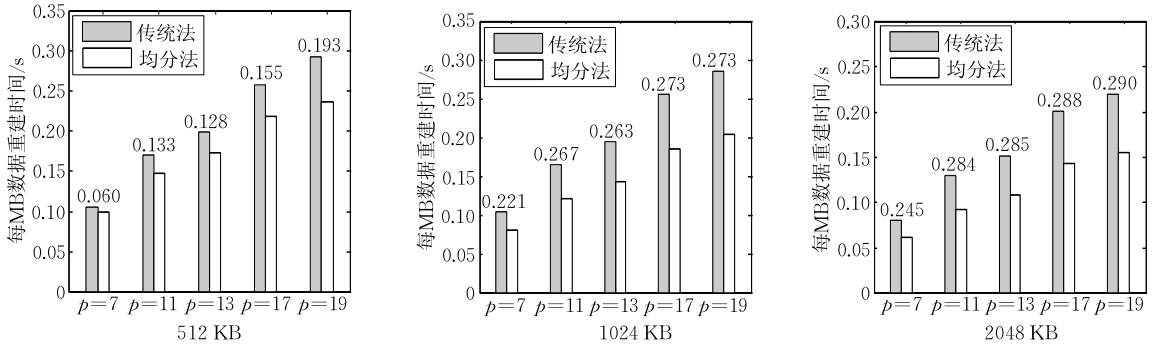


(a) TP

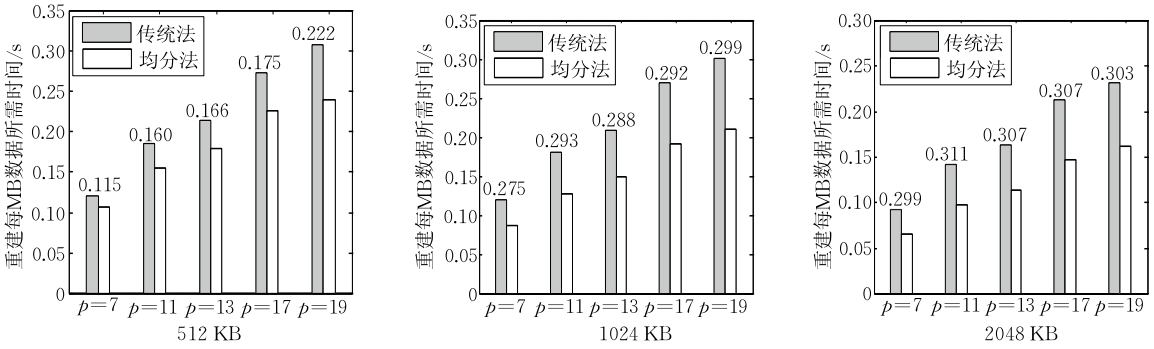


(b)STAR

图 6　数据块的大小对重建性能的影响



(a) TP

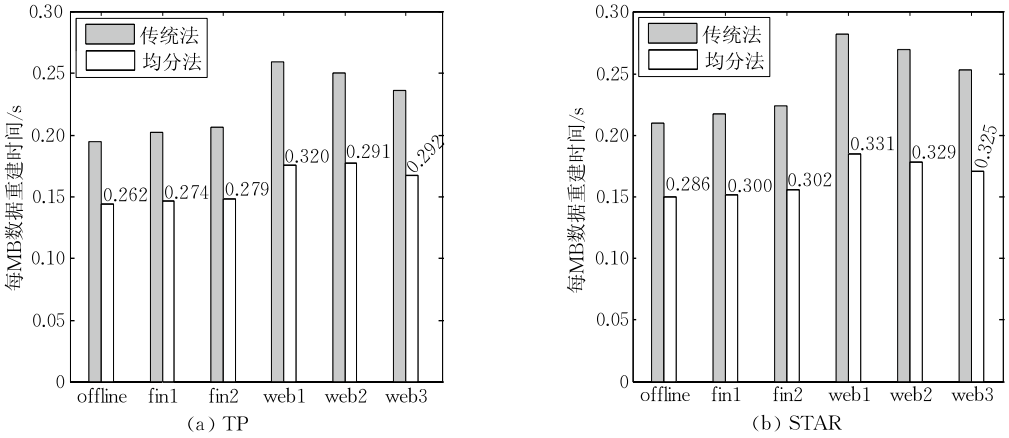


(b) STAR

图 7　磁盘个数对重建性能的影响

4.3 节显示 TP 编码减少 25%～30.6%的读数据量，STAR 编码减少 30%～33.64%的读数据量，但在 4.4 节的实验 1 和实验 2 中，读数据量减少的百分比没有完全转化为重建时间减少的百分比，这是由于均分法改变了磁盘顺序读的特性，使得磁头在寻道和旋转上有一定的延迟.但这种影响细微，尤其数据块较大时.总之，重建性能评估实验说明对 TP 和 STAR 而言，不同的数据块和不同的磁盘个数，在重建性能上均分法均优于传统方法.

实验 3. 在线重建性能测试.实验时，我们选取元素大小为 1024 KB 的 TP($p=13$)和 STAR ($p=13$)做代表来测试每 MB 数据的重建时间，以此衡量在线重建的性能，实验结果如图 8 所示.



(a) TP (b) STAR

图 8 不同工作负载下的重建性能

图 8(a)表示 TP 编码的在线重建性能.离线重建时，重建每 MB 数据均分法比传统法节省了 26.2%的时间；在线重建时，对所有的 trace 而言，重建每 MB 数据均分法仍比传统法的重建速度快，节省 27.4%～32.0%的时间.从图 8(a)看出，当采取传统法重建时，在线重建速度比离线重建速度要慢，这是由于在线重建时，重建 I/O 和前台应用程序的 I/O 共同竞争磁盘带宽，传统方法重建时从磁盘读取的数据量大，因此强烈的竞争导致重建速度明显放缓.而采取均分法重建时，虽然重建 I/O 和前台应用程序的 I/O 也会竞争，但由于均分法大大减少了重建 I/O 使得竞争激烈程度降低，因此在线重建的速度减缓程度较小.

图 8(b)表示 STAR 编码的测试结果，使用均分法在线重建仍能明显加速重建速度，说明均分法对 STAR 编码在线重建同样适用.

5 结论和进一步研究

本文提出了一种新型的均分机制方法，旨在解决容三盘失效纠删码在单盘失效时快速重建的问题.该方法能快速地为 TP 编码和 STAR 编码等容三盘出错纠删码寻找到最佳重建校验组，从而使得重建时从磁盘上读取的数据量大大降低.由于磁盘读在重建过程占据大部分的时间，最优重建校验组由于大大减少磁盘读的时间，会加速重建过程.本文首先给出满足最优重建校验组的充分必要条件，即处于均分状态且可行且从磁盘读取的数据量最少；在此基础上给出均分机制方法的算法步骤.实验表明，均分机制方法能返回与枚举法相当的最佳重建校验组，并且与枚举法相比，搜寻时间大幅度减少.理论分析表明，与传统方法相比，均分机制方法在重建读磁盘时减少了 27.8%～33.3%的数据量.实验表明与传统方法相比，均分机制方法在重建时间方面有明显的优势.

下一步工作主要包括对均分机制方法处于实际系统时进行重建时间的评估，并把它实现到分布式存储系统中观察性能.我们计划做更多实验，以便更加全面和合理地验证均分机制方法的有效性.

参 考 文 献

[1] Huang C, Xu L. STAR: An efficient coding scheme for correcting triple storage node failures. IEEE Transactions on Computers, 2008, 57(7): 889-901

[2] Corbett P, Goel A, et al. Triple parity technique for enabling efficient recovery from triple failures in a storage array. Network Appliance, INC, WO Patent WO/2007/078,803, 2007

[3] Blaum M, Brady J, Bruck J, et al. The EVENODD code and its generalization//Buyya R et al.. High Performance Mass Storage and Parallel I/O. Wiley-IEEE Press, 2002: 187-208

[4] Chen P, Lee E, Gibson G, et al. RAID: High-performance, reliable secondary storage. ACM Computing Surveys (CSUR), 1994, 26(2):145-185

[5] Xiang L, Xu Y, Lui J, Chang Q. Optimal recovery of single disk failure in RDP code storage systems. ACM SIGMETRICS Performance Evaluation Review, 2010, 38(1): 119-130

[6] Corbett P, English B, Goel A, et al. Row-diagonal parity for double disk failure correction//Proceedings of the 3rd USENIX Conference on File and Storage Technologies. San Francisco, USA, 2004: 1-14

[7] Li S, Cao Q, Huang J, et al. PDRS: A new recovery scheme application for vertical RAID-6 code//Proceedings of the Networking, Architecture and Storage (NAS). Dalian, China, 2011: 112-121

[8] Jin C, Jiang H, Feng D, Tian L. P-Code: A new RAID-6 code with optimal properties//Proceedings of the 23rd International Conference on Supercomputing. New York, USA, 2009: 360-369

[9] Xu L, Bruck J. X-Code: MDS array codes with optimal encoding. IEEE Transactions on Information Theory, 1999, 45(1): 272-276

[10] Wang Z, Dimakis A, Bruck J. Rebuilding for array codes in distributed storage systems//Proceedings of the GLOBECOM Workshops (GC Wkshps). Miami, FL, USA, 2010: 1905-1909

[11] Zhu Yunfeng, Lee Patrick P C, Xiang Liping, et al. A cost-based heterogeneous recovery scheme for distributed storage systems with RAID-6 codes//Proceedings of the 42nd Annual IEEE/IFIP International Conference on Dependable Systems and Networks. Boston, USA, 2012: 1-12

[12] Hu Yuchong, Chen Henry C H, Lee Patrick P C, Tang Yang. NCCloud: Applying network coding for the storage repair in a cloud-of-clouds//Proceedings of the 10th USENIX Conference on File and Storage Technologies. San Jose, CA, USA, 2012: 265-272

[13] Khan O, Burns R, Plank J, Huang C. In search of I/O-optimal recovery from disk failures//Proceedings of the Workshop on Hot Topics in Storage and File Systems. Portland, OR, USA, 2011: 1-5

[14] Khan O, Burns R, Plank J, et al. Rethinking erasure codes for cloud file systems: Minimizing I/O for recovery and degraded reads//Proceedings of the 10th USENIX Conference on File and Storage Technologies (FAST). San Jose, USA, 2012: 259-272

[15] Plank J, Luo J, Schuman C, et al. A performance evaluation and examination of open-source erasure coding libraries for storage//Proceedings of the 7th USENIX Conference on File and Storage Technologies (FAST). San Francisco, USA, 2009: 253-265

[16] Narayanan D, Donnelly A, Rowstron A. Write off-loading: Practical power management for enterprise storage//Proceedings of the File and Storage Technologies Conference. San Jose, CA, USA, 2008: 253-267

[17] Wu Suzhen, Jiang Hong, Feng Dan, et al. Workout: I/O workload outsourcing for boosting RAID reconstruction performance//Proceedings of the 7th Conference on File and Storage Technologies. San Francisco, California, USA, 2009: 239-252

[18] Bucy J, Schindler J, Schlosser S, Ganger G. The DiskSim Simulation Environment (v4.0). Version 4.0 Reference Manual. Carnegie Mellon University, Pittsburgh, USA: CMU-PDL-08-101, 2008

[19] Ghemawat S, Gobioff H, Leung S. The Google file system. ACM SIGOPS Operating Systems Review, 2003, 37(5): 29-43

QIU Li-Na, born in 1989, M. S. candidate. Her current research interests include RAID code, massive network storage system and parallel storage system.

WANG Fang, born in 1972, professor, Ph. D. supervisor. Her main research interests include massive network storage system, parallel storage system and power consumption of storage system.

LI Chu, born in 1989, Ph. D. candidate. His main research interests include reliability of storage system, massive network storage system and parallel storage system.

Background

Our research is aimed to address fast single failure recovery from triple-erasure-correcting codes in reliability field of storage systems. Even though single failure recovery has been researched adequately, most of those solutions are designed for RAID6 codes other than triple-erasure-correcting codes. Among those researches on triple-erasure-correcting

codes, we consider impressive ones of them: conventional recovery and enumeration recovery which was developed by Osama Khan. When reconstructing through conventional recovery, it will read the same data from surviving disks repeatedly, which lowers the effective utilization of data without any doubt and therefore extends reconstruction time. Khan et al. point that by enumerating all of the possible recovery solutions one can find out the optimal one that requires the minimum amount of data transmission. However, with the number of disks increasing, the time spent on enumerating the whole solutions is tremendous and intolerable.

To address those problems, we put forward a novel scheme for triple-erasure-correcting codes to boost single failure recovery and this method is suitable for those triple-erasure-correcting codes, such as TP, STAR, the generalized EVENODD and the like. There are three kinds of parity groups in codes that tolerate triple concurrent failures and conventional recovery merely makes use of one kind of them to retrieve the lost data. Actually, when we use three kinds of parity groups to form the recovery solution, it will improve the effective utilization of data during reconstruction and also help to find out the optimal solution. The amount of data that need to be read from surviving disks when recovering through the optimal path can not be confirmed directly, but the range of it can be deduced. Based on those discoveries mentioned above, we propose Equal Division Scheme (EDS) to realize fast recovery from single failure of triple-erasure-correcting codes. We locate the range of the minimum amount of data transmission for recovery by means of mathematics analysis and reasoning, and then find out the optimal recovery path through heuristic search. It turns out that EDS reduces the amount of data transmission being read from surviving disks about 25%—30.6% compared with conventional recovery for TP code and 30%—33.64% for STAR code. Empirical results show that EDS can return the near optimal solution as enumeration recovery while consuming much less time and the overall recovery time is greatly decreased for TP and STAR code storage systems.

This research is supported by the National Basic Research 973 Program of China under Grant No. 2011CB302301; 863 Project 2013AA013203; NSFC Nos. 61025008, 60933002, 61232004.