RAID-VCR：一种能够承受三个磁盘故障的 RAID 结构

董欢庆 李战怀 林 伟

（西北工业大学计算机学院 西安 710072）

摘 要 提出了一种新 RAID 结构——RAID-VCR. 这种结构仅需要 3 个额外的磁盘来保存校验信息，但是却能够承受任意模式的 3 个成员磁盘故障. 与现有的其它 RAID 结构相比，RAID-VCR 的容灾能力大幅提高，但是对磁盘空间利用率和系统吞吐量的影响却非常小. RAID-VCR 的编码和解码过程都是基于简单的 XOR 操作，并且以明文方式保存了用户数据，从而可以高效地执行读操作. 仿真实验结果表明，RAID-VCR 的编码和解码性能较好，具有很好的应用前景.

关键词 冗余存储；抗删除编码；RAID；容灾；VCR 码

中图法分类号 TP334

RAID-VCR: A New RAID Architecture for Tolerating Triple Disk Failures

DONG Huan-Qing LI Zhan-Huai LIN Wei

(*School of Computer Science, Northwestern Polytechnical University, Xi'an* 710072)

Abstract Although almost all vendors enhance the availability of their storage systems by levering RAID technology to achieve redundant storage, there is currently no feasible RAID architecture being able to tolerate simultaneous failures of three member disks. To improve the fault-tolerant capability of RAID structure, this paper introduces a novel RAID structure named RAID-VCR which taking advantage of CR coding techniques used in telecommunication while making some modification to adapt to the characteristics of storage systems. The encoding and decoding procedure of RAID-VCR are based on XOR operations, and the computing complexity is quadratic in the number of data disks. This makes it quite simple and feasible. According to mathematical proof and experiments, RAID-VCR could improve the fault-tolerant capability remarkably and tolerate triple simultaneous disk failures in any pattern with only three extra disks for parity information, while makes few negative influences on system's throughput and disk capacity utilization.

Keywords redundant storage; erasure resilient code, RAID; fault-tolerance; VCR code

1 引 言

虽然近年来在提高磁盘阵列的性能方面的研究取得了较大的进展. 但是，关于磁盘阵列可靠性的研究更加急迫，因为部分磁盘的失败可能导致整个磁盘阵列上的数据不可用，这是一种灾难性的后果[1]. 一般来说，可以通过增加冗余来提高存储系统的可

收稿日期：2004-09-07；修改稿收到日期：2005-11-24. 本课题得到国家自然科学基金（60373108）资助. 董欢庆，男，1976 年生，博士，研究方向为网络存储与数据容灾. E-mail：donghuanqing@ict.ac.cn. 李战怀，1961 年生，教授，博士生导师，主要研究领域为数据库理论与技术、网络存储等. 林 伟，博士研究生，研究方向为高可用系统和分布式存储系统.

靠性，RAID 磁盘阵列[2]就是一种广泛接受的冗余技术. 在 RAID 结构中，通过把用户数据进行编码产生冗余(校验)信息并和用户数据一起保存在磁盘阵列上，以便在故障发生后进行数据恢复. 随着成员磁盘数目的增加，RAID 磁盘阵列发生磁盘故障的可能性也会增加. 许多研究表明[2,3]，如果 RAID 系统中一个成员磁盘发生故障，那么在未来很短的时间内其它成员磁盘发生故障的可能性非常高. 还有许多其它原因[4]，例如分布式 RAID 环境等也非常需要能够承受多个磁盘故障的 RAID 结构. 目前已经有一些具有强容灾能力的 RAID 磁盘阵列系统面世，比如惠普的 EVA 系列高端磁盘阵列就采用了多层 RAID 来保证多个磁盘同时发生故障的情况下用户数据不会丢失.

RAID 结构的核心问题就是冗余信息的产生方法(即编码方法). 在数据通信领域关于编码和解码方法的研究已经很多，并且有许多成熟的编码方法，但是这些编码方法并不适合于直接在 RAID 阵列中使用. 这两个领域需要的编码方法存在以下的不同：(1)在通信编码领域，设计编码时，主要是考虑解码时的查错和纠错能力，考虑接收端接收到的每一个 bit 都有发生错误的可能，一般根据极大似然法纠错；而在 RAID 阵列中，可认为每个磁盘有没有发生故障是已知的、没有发生故障的磁盘上的数据都是正确的，主要考虑的是进行数据恢复而不是纠错或者查错. (2)在 RAID 结构中，考虑故障时总是认为整个磁盘发生故障，这样丢失的多个信息元之间有明确的位置关系，一般相当于编码矩阵中的一列或者一行，而信号传输过程中任意组合的错误都可能发生.

虽然 RAID 磁盘阵列领域的编码和数据通信领域的编码在应用目标上有所不同，但是编码方法是有共性的. 因此，我们希望能够借鉴通信编码领域的成果，找到一种比较简单的通信编码方法，在对其进行改造和优化后能够适应 RAID 领域的应用. 基于这种思想，我们最终选择了 CR 码[5]作为基础进行改造，这主要是因为 CR 码具有以下特征：(1)编码过程中把信息元按照列组织，而每一列正好可以对应 RAID 编码中 1 个磁盘上的多个单元；(2)编码过程简单，计算复杂度低，并且各列校验元的计算之间是独立的. 在对 CR 码进行改造后，我们得出一种新的适合 RAID 磁盘阵列的编码，称为 VCR 码，而基于 VCR 码的 RAID 结构称为 RAID-VCR，它包含 $q+3$ 个成员磁盘，其中 q 个成员磁盘用来保存原始的用户数据，另外 3 个成员磁盘用来保存根据用户数据编码生成的校验信息. 这种结构能够承受任意组合的 3 个成员磁盘故障. 与其它 RAID 结构相比，RAID-VCR 在增加少量开销的情况下，大大提高了系统的容灾能力.

2　相关工作

能够承受 2 个磁盘故障的 RAID 结构已经被提出并获得应用，这主要包括 RAID-6 和 EVENODD[3]. 而能够承受多个磁盘故障的 RAID 结构目前并不成熟. DATUM[4]是一种能够承受多个磁盘故障的 RAID 结构，并且容灾能力随着冗余磁盘数的增加而增加. 这种结构的主要问题在于阵列中没有保存原始的用户数据，即磁盘上保存的所有数据都是经过编码得到的，编码和解码复杂度高，并且读小块数据的过程中也牵涉到多个磁盘操作和解码操作，因此其系统资源占用比例比较高，工作效率低.

文献[6]提出的 RAID 结构用明文方式保存了原始的用户数据，并且有多个磁盘用于保存冗余(校验)信息，但是由于创建冗余信息的编码是不对称的，即用户数据在冗余信息中的分布不对称，这种结构虽然能够在大部分模式的多磁盘故障发生后恢复用户数据，但是对于某些模式的多磁盘故障则无能为力. 例如，部分数据单元只在 2 个校验磁盘上出现，如果该单元所在磁盘及该单元对应的 2 个校验磁盘出现故障(即发生 3 个磁盘故障)，则该数据单元根本不可能被恢复；而另外一些数据单元则只在 1 个校验磁盘上出现，如果该单元所在的磁盘及该单元对应的那个校验磁盘出现故障(即只发生 2 个磁盘故障)，则显然该单元也无法被恢复.

文献[7]也提出了一种用于承受 3 个磁盘故障的 RAID 结构，这种结构要求成员磁盘的个数为 $N+1$(其中 N 为素数)，虽然文献中给出了在发生 3 个磁盘故障后恢复用户数据的例子(例子中 $N=5$)，但是文中利用汉明(Hamming)距离进行的正确性证明并不充分. 该文中认为提出的编码方式得到的码字之间的汉明距离为 4，根据编码理论，这只能证明这种编码能够纠正 3 个单元错，而作者要证明的应该是这种编码能够在任意 3 个磁盘发生故障后恢复数据，即要恢复 $3N$ 个单元上的数据，而不是纠正 3 个单元上发生的错误.

另外,文献[8]提出了一种结合文件系统的语义提高系统容灾能力的 RAID 结构,但是由于依赖于特定的文件系统实现,不是一种通用的方法.

3 RAID-VCR 的编码

RAID-VCR 采用的编码(即 VCR 码)是我们在 CR 码[5]的基础上改进得到的,为此,我们将先介绍 CR 码的编码,然后介绍 VCR 码.

在本文后面的部分中,我们都是利用校验信息的构造矩阵来讨论 RAID 结构,构造矩阵中的一列表示来自同一磁盘的多个连续单元,在实现中,一个单元可以对应一个字节、扇区或者其它单位. 我们用大写的字母表示矩阵,例如 $\boldsymbol{D}$ 和 $\boldsymbol{P}$,其中 $\boldsymbol{D}$ 表示用户数据部分构成的矩阵,$\boldsymbol{P}$ 表示校验信息构成的矩阵. 每个矩阵中的一列用矩阵标志符加上下标表示,例如 $\boldsymbol{D}_j$ 表示 $\boldsymbol{D}$ 中的一列,而小写的字母加上下标表示矩阵中的 1 个单元或者元素,例如 $d_{i,j}$ 表示 $\boldsymbol{D}$ 中的 1 个单元,因此,$\boldsymbol{D}_j=(d_{0,j},d_{1,j},d_{2,j},\cdots)^{\mathrm{T}}$,$\boldsymbol{P}_j=(p_{0,j},p_{1,j},p_{2,j},\cdots)^{\mathrm{T}}$. 在考虑 RAID 阵列的磁盘故障时,我们总是考虑整个磁盘故障,因此上述矩阵中每一列中各个单元的状态总是相同的,即处于故障(丢失)或者可用(完好)状态. 在发生磁盘故障的情况下,重建用户数据的问题可以归结为利用完好的列来重建丢失的列.

3.1 CR 码介绍

CR 码是一种具有强纠错能力的编码方法,主要用于不可靠线路上的数据通信. 设 q 是一个素数,用矩阵 $\boldsymbol{D}_{q\times q}$ 代表用户数据,检验矩阵 $\boldsymbol{P}_{q\times(q+1)}$ 的构造方法是:

$$p_{i,j}=\begin{cases}\bigoplus_{l=0}^{q-1} d_{\langle i-jl\rangle_q,l}, & 0\leqslant j\leqslant q-1\\ \bigoplus_{l=0}^{q-1} d_{l,i}, & j=q\end{cases} \tag{1}$$

其中,$\langle x\rangle_q$ 表示 x 对 q 取模. 图 1 给出了 CR 码的一个例子,其中用户数据矩阵 $\boldsymbol{D}$ 的各个单元用 1 个字母表示,而校验矩阵的单元则为 $\boldsymbol{D}$ 中多个单元的异或.

$\boldsymbol{D}_0$	$\boldsymbol{D}_1$	$\boldsymbol{D}_2$	$\boldsymbol{P}_0$	$\boldsymbol{P}_1$	$\boldsymbol{P}_2$	$\boldsymbol{P}_3$
a	b	c	$a\oplus b\oplus c$	$a\oplus e\oplus i$	$a\oplus h\oplus f$	$a\oplus d\oplus g$
d	e	f	$d\oplus e\oplus f$	$d\oplus h\oplus c$	$d\oplus b\oplus i$	$d\oplus e\oplus c$
g	h	i	$g\oplus h\oplus i$	$g\oplus b\oplus f$	$g\oplus e\oplus c$	$c\oplus f\oplus i$

图 1 CR 码的编码示意图($q=3$)

CR 码是一种具有强纠错能力的编码方法,在通信领域有很好的用途. 但是由于在 RAID 阵列中磁盘故障模式的特殊性,即总是认为整列处于相同的状态,基于 CR 码的 RAID 结构的容灾能力并不是最优的,假设其中几列发生故障,则很容易证明需要求解的线性方程组存在线性相关. 为此,我们把 CR 码进行了改造,以使之适合于 RAID 阵列的编码.

3.2 VCR 码的几何表示

VCR 码可以看成是 CR 码的一个变体. VCR 码的编码矩阵由 $q+3$ 列构成(q 是一个大于 2 的素数),每一列有 $q-1$ 行,其中 q 列以明文方式存放用户数据,其余 3 列为校验信息. 我们用矩阵 $\boldsymbol{D}_{(q-1)\times q}$(或者 $\boldsymbol{D}$)表示 q 列用户数据构成的矩阵,用 $\boldsymbol{P}_{(q-1)\times 3}$(或者 $\boldsymbol{P}$)表示 3 列校验信息构成的矩阵. $\boldsymbol{P}$ 中的各列是通过式(2)产生的. 为了便于用方程表述,在编码时我们假想矩阵 $\boldsymbol{D}$ 中有一个额外的全 0 行(第 $q-1$ 行)存在.

$$p_{i,j}=\left(\bigoplus_{l=0}^{q-1} d_{\langle i-jl\rangle_q,l}\right)\oplus t_j,\ 0<j<3,\ 0<i<q-1 \tag{2}$$

其中,$t_j=\bigoplus_{l=0}^{q-1} d_{\langle q-1-jl\rangle_q,l}$,$0<j<3$,特别地,$t_0=0$.

图 2 给出了 $q=3$ 情况下 VCR 编码的示意图,注意最后一行是假想的全 0 行.

$\boldsymbol{D}_0$	$\boldsymbol{D}_1$	$\boldsymbol{D}_2$	$\boldsymbol{P}_0$	$\boldsymbol{P}_1$	$\boldsymbol{P}_2$
a	b	c	$a\oplus b\oplus c$	$a\oplus e\oplus b\oplus f$	$a\oplus f\oplus e\oplus c$
d	e	f	$d\oplus e\oplus f$	$d\oplus c\oplus b\oplus f$	$d\oplus b\oplus e\oplus c$
0	0	0	0	0	0

图 2 VCR 码的编码示意图($q=3$)

3.3 VCR 码的代数表示

显然,在发生部分成员磁盘故障时,我们需要求解一个线性方程组来恢复丢失的用户数据,而上一节给出的编码描述不便于证明方程组是否有唯一解. 为了便于证明 RAID-VCR 的容灾能力,我们需要引入与上述几何方式等价的代数表示方式,这样在证明 RAID-VCR 在发生磁盘故障后重建能力时可以用代数方法. 注意这些代数方法在编码和解码过程中并不需要.

为此,我们必须先构造一种代数表示,并证明这种代数表示与上一节给出的编码方式等价,这样才能在后面证明线性方程组可解时使用对应的代数方式. 我们参照文献[9]中的方法,构造一种基于有限域上多项式环的代数表示.

由于异或运算本身就和素数域 $GF(2)$ 上的加法运算(即模 2)等价,我们用 F 表示 $GF(2)$. 而矩阵

$\boldsymbol{D}$ 和 $\boldsymbol{P}$ 中的每一列可以看成一个多项式的系数，例如，D_2 可以看成 $\boldsymbol{D}_2(x)=d_{0,2}x^0+d_{1,2}x^1+\cdots+d_{q-2,2}x^{q-2}$ 的系数.

定义 $M_q(x)=\sum_{i=0}^{q-1}x^i$，考虑域 F 上阶数小于 $q-1$ 的多项式的集合，如果多项式的乘法对 $M_q(x)$ 取模，则当 q 是素数时，这个多项式集合构成一个环[9]，我们用 $R_q(2)$ 或 R_q 表示这个多项式环. 为了避免混淆，当我们表示 R_q 上的多项式时，将用 a 作为未知量；当我们表示多项式环 $F[x]$（即域 F 上所有多项式构成的环）上的多项式时，将使用未知量 x.

在上述定义的基础上，我们将要证明式(2)表示的编码和下式的表述是等价的：

$$\begin{bmatrix}1 & 1 & 1 & \cdots & 1\\ 1 & a & a^2 & \cdots & a^{q-1}\\ 1 & a^2 & a^4 & \cdots & a^{2(q-1)}\end{bmatrix}\begin{bmatrix}\boldsymbol{D}_0(a)\\ \boldsymbol{D}_1(a)\\ \vdots\\ \boldsymbol{D}_{q-1}(a)\end{bmatrix}=\begin{bmatrix}\boldsymbol{P}_0(a)\\ \boldsymbol{P}_1(a)\\ \boldsymbol{P}_2(a)\end{bmatrix} \tag{3}$$

证明.　上式可以写成

$$\sum_{j=0}^{q-1}a^{jl}\boldsymbol{D}_j(a)=\boldsymbol{P}_l(a),\ 0\le l\le 2 \tag{4}$$

写成多项式环 $F[x]$ 上的多项式的形式，即

$$\sum_{j=0}^{q-1}x^{jl}\boldsymbol{D}_j(x)=\boldsymbol{P}_l(x)\ (\mathrm{mod}\ M_q(x)),\ 0\le l\le 2 \tag{5}$$

展开后变成（注意 $\boldsymbol{P}_l(x)$ 的阶数也小于 $q-1$）：

$$\sum_{i=0}^{q-1}\sum_{j=0}^{q-1}d_{i,j}x^{i+jl}\equiv\sum_{i=0}^{q-2}p_{i,l}x^i\ (\mathrm{mod}\ M_q(x)),\ 0\le l\le 2 \tag{6}$$

由于 $(x^q-1)=(x-1)M_q(x)$，因此，$x^q-1=0(\mathrm{mod}\ M_q(x))$，即 $x^q=1(\mathrm{mod}\ M_q(x))$，据此把式(6)的左边进行变换，得到

$$\sum_{i=0}^{q-1}\sum_{j=0}^{q-1}d_{\langle i-jl\rangle_q,j}x^i\equiv\sum_{i=0}^{q-2}p_{i,l}x^i\ (\mathrm{mod}M_q(x)),\ 0\le l\le 2 \tag{7}$$

如前所述，$P_l(x)$ 的阶数小于 $q-1$，所以可以进行变换，使式(7)的左边也变成 $q-2$ 阶多项式. 考虑到

$$\sum_{i=0}^{q-1}m_ix^i=\sum_{i=0}^{q-2}(m_i+m_{q-1})x^i\ (\mathrm{mod}\ M_q(x)),$$

式(7)可以变成

$$\sum_{i=0}^{q-2}\Big(\sum_{j=0}^{q-1}d_{\langle i-jl\rangle_q,j}+\sum_{j=0}^{q-1}d_{\langle q-1-jl\rangle_q,j}\Big)x^i\equiv\sum_{i=0}^{q-2}p_{i,l}x^i\ (\mathrm{mod}M_q(x)),\ 0\le l\le 2,$$

即

$$\sum_{i=0}^{q-2}\Big(\sum_{j=0}^{q-1}d_{\langle i-jl\rangle_q,j}+\sum_{j=0}^{q-1}d_{\langle q-1-jl\rangle_q,j}\Big)a^i\equiv\sum_{i=0}^{q-2}p_{i,l}a^i,\quad 0\le l\le 2 \tag{8}$$

由于 $a^0,a^1,\cdots,a^{q-2}$ 线性无关，所以式(8)两边多项式中对应项的系数应该相等，即

$$\sum_{j=0}^{q-1}d_{\langle i-jl\rangle_q,j}+\sum_{j=0}^{q-1}d_{\langle q-1-jl\rangle_q,j}=p_{i,l},\ 0\le l\le 2 \tag{9}$$

把 $GF(2)$ 上的加法运算符替换成异或运算符，即得到

$$\Big(\bigoplus_{j=0}^{q-1}d_{\langle i-jl\rangle_q,j}\Big)\oplus\Big(\bigoplus_{j=0}^{q-1}d_{\langle q-1-jl\rangle_q,j}\Big)=p_{i,l},\ 0\le l\le 2 \tag{10}$$

显然，式(10)与式(2)是相同的. 因此，式(2)表示的编码与式(3)表示的编码等价.

3.4　VCR 码与 EVENODD 编码的关系

EVENODD[3]码的编码矩阵包含 m 个数据列和 2 个校验列，每一列有 $m-1$ 个单元，其中 m 为素数. 对 EVENODD 的编码方程与 VCR 码进行比较研究后，我们发现 EVENODD 码的两列校验信息的构造与 VCR 码的前两列实际上相同，但是 VCR 码多了一个校验列. 因此，EVENODD 的校验码是 VCR 中校验码的子集.

4　RAID-VCR 的容灾能力

在这一节，我们将证明 RAID-VCR 能够在发生任意组合的 3 个成员磁盘故障的情况下恢复用户数据. 所有的磁盘故障组合可以分为：3 个数据盘故障、2 个数据盘加 1 个校验盘故障、1 个数据盘加 2 个校验盘故障和 3 个校验盘故障等共四种模式. 由于 3 个校验盘故障的情况下没有造成用户数据丢失，以下分别针对前三种模式进行证明.

4.1　3 个数据盘故障

在发生 3 个数据盘故障的情况下，还有 $q-3$ 个数据盘和 3 个校验盘完好，我们用 μ_0，μ_1 和 μ_2（$0\le\mu_0<\mu_1<\mu_2\le q-1$）分别表示 3 个故障盘的编号，用 Ω 表示 μ_0，μ_1 和 μ_2 构成的集合. 为了便于说明，我们构造一个临时矩阵 $\boldsymbol{Y}=[\boldsymbol{Y}_0\boldsymbol{Y}_1\cdots\boldsymbol{Y}_{q-1}]$，$\boldsymbol{Y}_j$（$0\le j\le q-1$）的定义为

$$\boldsymbol{Y}_j=\begin{cases}\boldsymbol{D}_j, & j\notin\Omega\\ \boldsymbol{0}, & j\in\Omega\end{cases},$$

根据 $\boldsymbol{Y}$ 的定义，把式(4)进行替换，我们得到

$$\begin{cases}\sum_{j=0}^{q-1}\boldsymbol{Y}_j(a)+\boldsymbol{D}_{\mu_0}(a)+\boldsymbol{D}_{\mu_1}(a)+\boldsymbol{D}_{\mu_2}(a)=\boldsymbol{P}_0(a)\\ \sum_{j=0}^{q-1}a^j\boldsymbol{Y}_j(a)+a^{\mu_0}\boldsymbol{D}_{\mu_0}(a)+a^{\mu_1}\boldsymbol{D}_{\mu_1}(a)+\\ \quad a^{\mu_2}\boldsymbol{D}_{\mu_2}(a)=\boldsymbol{P}_1(a)\\ \sum_{j=0}^{q-1}a^{2j}\boldsymbol{Y}_j(a)+a^{2\mu_0}\boldsymbol{D}_{\mu_0}(a)+a^{2\mu_1}\boldsymbol{D}_{\mu_1}(a)+\\ \quad a^{2\mu_2}\boldsymbol{D}_{\mu_2}(a)=\boldsymbol{P}_2(a)\end{cases}\tag{11}$$

用 $\boldsymbol{E}_k$ 代表 $\boldsymbol{D}_{\mu_k}$，用 a_k 代替 a^{μ_k}，我们得到

$$\begin{cases}\boldsymbol{E}_0(a)+\boldsymbol{E}_1(a)+\boldsymbol{E}_2(a)=\boldsymbol{P}_0(a)+\sum_{j=0}^{q-1}\boldsymbol{Y}_j(a)\\ a_0\boldsymbol{E}_0(a)+a_1\boldsymbol{E}_1(a)+a_2\boldsymbol{E}_2(a)=\\ \quad \boldsymbol{P}_1(a)+\sum_{j=0}^{q-1}a^j\boldsymbol{Y}_j(a)\\ a_0^2\boldsymbol{E}_0(a)+a_1^2\boldsymbol{E}_1(a)+a_2^2\boldsymbol{E}_2(a)=\\ \quad \boldsymbol{P}_2(a)+\sum_{j=0}^{q-1}a^{2j}\boldsymbol{Y}_j(a)\end{cases}\tag{12}$$

把式子右边的值用 $\boldsymbol{S}_l(a)$ 表示，即令

$$\boldsymbol{S}_l(a)=\boldsymbol{P}_l(a)+\Big(\sum_{j=0}^{q-1}a^{jl}\boldsymbol{Y}_j(a)\Big),\quad 0\le l\le 2\tag{13}$$

式(12)可以写成

$$\begin{bmatrix}1&1&1\\ a_0&a_1&a_2\\ a_0^2&a_1^2&a_2^2\end{bmatrix}\begin{bmatrix}\boldsymbol{E}_0(a)\\ \boldsymbol{E}_1(a)\\ \boldsymbol{E}_2(a)\end{bmatrix}=\begin{bmatrix}\boldsymbol{S}_0(a)\\ \boldsymbol{S}_1(a)\\ \boldsymbol{S}_2(a)\end{bmatrix},$$

显然，我们只要根据上述方程组解出 $\boldsymbol{E}_0(a)$，$\boldsymbol{E}_1(a)$ 和 $\boldsymbol{E}_2(a)$ 即可以恢复失败的数据磁盘上的数据. 为了证明上述方程组可解，只要证明其系数行列式非 0 即可. 考虑其系数行列式：

$$\boldsymbol{V}=\begin{bmatrix}1&1&1\\ a_0&a_1&a_2\\ a_0^2&a_1^2&a_2^2\end{bmatrix},$$

显然，$\boldsymbol{V}$ 的行列式是一个范德蒙(Vandermonde)行列式，由于 $a_i(0\le i\le 2)$ 分别对应 a 的不同次幂，$\boldsymbol{V}$ 的行列式的值肯定不等于 0，并且 $\boldsymbol{V}$ 在 R_q 上存在一个逆矩阵 $\boldsymbol{V}^{-1}$(具体证明过程参见文献[9])，因此上述线性方程组一定存在唯一解. 我们将在第 5 节介绍如何进行求解.

4.2 2 个数据盘加 1 个校验盘故障

在 2 个数据盘加 1 个校验盘故障的情况下，有 2 个发生故障的数据盘和2个完好的校验盘. 我们用 μ_0 和 $\mu_1(0\le\mu_0<\mu_1\le q-1)$ 分别表示 2 个发生故障的数据盘的编号，用 λ_0 和 $\lambda_1(0\le\lambda_0<\lambda_1\le 2)$ 分别表示 2 个完好的校验盘的编号. 通过和 4.1 节类似的方法，我们可以得到以下的线性方程组：

$$\begin{bmatrix}a^{\mu_0\lambda_0}&a^{\mu_1\lambda_0}\\ a^{\mu_0\lambda_1}&a^{\mu_1\lambda_1}\end{bmatrix}\begin{bmatrix}\boldsymbol{D}_{\mu_0}(a)\\ \boldsymbol{D}_{\mu_1}(a)\end{bmatrix}=\begin{bmatrix}\boldsymbol{S}_{\lambda_0}(a)\\ \boldsymbol{S}_{\lambda_1}(a)\end{bmatrix}\tag{14}$$

其中，$\boldsymbol{S}_{\lambda_0}(a)$ 和 $\boldsymbol{S}_{\lambda_1}(a)$ 可以用类似于式(13)描述的方法得到，此处不再详述.

用 $\boldsymbol{V}$ 表示式(14)的左边的系数矩阵，其行列式的值为 $|\boldsymbol{V}|=a^{\mu_0\lambda_0+\mu_1\lambda_1}-a^{\mu_1\lambda_0+\mu_0\lambda_1}$. 由于 q 是素数，对于任意两个小于 q 的正整数 m 和 n，有 $mn\neq 0\pmod q$，而根据上述对 μ_0，μ_1，λ_0 和 λ_1 的定义，我们得到 $(\mu_0-\mu_1)(\lambda_0-\lambda_1)\neq 0\pmod q$，即

$$\mu_0\lambda_0+\mu_1\lambda_1\neq\mu_1\lambda_0+\mu_0\lambda_1\pmod q,$$

由于 $(x^q-1)=(x-1)M_q(x)$，因此，$x^q-1=0\pmod{M_q(x)}$，即 $a^q=1$.

令 $\rho=\langle\mu_0\lambda_0+\mu_1\lambda_1\rangle_q$，$\zeta=\langle\mu_1\lambda_0+\mu_0\lambda_1\rangle_q$，则 $|\boldsymbol{V}|=a^\rho-a^\zeta$，由于 $\rho\neq\zeta$，且 $0<\rho,\zeta<q$，因此 a^ρ 与 a^ζ 是线性无关的，故 $|\boldsymbol{V}|\neq 0$. 因此，上述方程组存在唯一解.

4.3 1 个数据盘加 2 个校验盘故障

在 1 个数据盘加 2 个校验盘故障的情况下，有 1 个发生故障的数据盘和 1 个完好的校验盘. 我们用 $\mu(0\le\mu\le q-1)$ 表示发生故障的数据盘的编号，用 $\lambda(0\le\lambda\le 2)$ 表示完好的校验盘的编号，则通过与上一节类似的方法，可以得到方程：$a^{\mu\lambda}\boldsymbol{D}_\mu(a)=\boldsymbol{S}_\lambda(a)$，显然可以通过把 $\boldsymbol{S}_\lambda(a)$ 做简单的旋转和降阶(相当于对 $M_q(x)$ 取模)得到 $\boldsymbol{D}_\mu(a)$.

5 恢复故障磁盘上数据的方法

本文的前面部分已经证明了 RAID-VCR 在发生任意 3 个磁盘故障的情况下都能够恢复用户数据. 但是由于基于多项式的运算和方程求解比较复杂，这一节我们将根据几何编码过程给出如何构造方程组和求解.

假设有 $k(1\le k\le 3)$ 个数据盘故障，我们也有 k 个完好的校验盘. 用 $\mu_0\cdots\mu_{k-1}$，$0\le\mu_0<\cdots<\mu_{k-1}\le q-1$ 表示各个故障磁盘的位置，而 $\lambda_0\cdots\lambda_{k-1}$，$0\le\lambda_0<\cdots<\lambda_{k-1}\le 2$ 表示完好的校验盘的位置，根据式(2)可以构造如式(15)所示的方程组.

$$\begin{bmatrix} c_{0,0} & c_{0,1} & \cdots & c_{0,(q-2)\times(k-1)} \\ c_{1,0} & c_{1,1} & \cdots & c_{1,(q-2)\times(k-1)} \\ \vdots & \vdots & \ddots & \vdots \\ c_{(q-2)\times(k-1),0} & c_{(q-2)\times(k-1),1} & \cdots & c_{(q-2)\times(k-1),(q-2)\times(k-1)} \end{bmatrix} \begin{bmatrix} d_{0,\mu_0} \\ d_{1,\mu_0} \\ \vdots \\ d_{q-2,\mu_0} \\ \vdots \\ d_{0,\mu_{k-1}} \\ d_{1,\mu_{k-1}} \\ \vdots \\ d_{q-2,\mu_{k-1}} \end{bmatrix} = \begin{bmatrix} s_{0,\lambda_0} \\ s_{1,\lambda_0} \\ \vdots \\ s_{q-2,\lambda_0} \\ \vdots \\ s_{0,\lambda_{k-1}} \\ s_{1,\lambda_{k-1}} \\ \vdots \\ s_{q-2,\lambda_{k-1}} \end{bmatrix} \tag{15}$$

式(15)中，$c_{m,n}$的定义为

$$c_{m,n}=T(h-\mu_j\lambda_l)\oplus T(q-1-\mu_j\lambda_l),$$

其中，$i=n \bmod (q-1)$，$j=\lfloor n/(q-1)\rfloor$，$l=\lfloor m/(q-1)\rfloor$，$h=m \bmod (q-1)$，函数 $T(x)$的定义为

$$T(x)=\begin{cases}1, & x \bmod q=n \\ 0, & \text{其它}\end{cases}.$$

上述的方程只需要做普通的线性变换即可求解出 $\boldsymbol{D}_{\mu_0}\cdots\boldsymbol{D}_{\mu_{k-1}}$.

6　对 RAID-VCR 的相关分析

前一节分析了 RAID-VCR 的容灾能力，但是对于一种 RAID 结构来说，除了考虑容灾能力以外，还需要考虑性能、I/O 操作的复杂度、计算复杂度等问题. 以下将对 RAID-VCR 的相关特征进行分析.

6.1　编码过程的计算复杂度分析

为了简化分析过程，我们假定编码矩阵中每个单元都是一个 bit，我们将在这个前提下分析编码过程中异或操作的次数. 在下面的分析中，计算复杂度也都是以异或运算的次数来衡量的.

VCR 码有 3 个校验列. 第 1 个校验列的构造需要执行$(q-1)\times(q-1)$次异或操作，第 2，3 个校验列的构造分别需要执行$(q-1)+q\times(q-1)$次异或操作，因此，整个校验矩阵的构造过程需要执行 $3q^2-2q-1$ 次异或操作.

从理论上说，假设一个 RAID 结构的编码矩阵是一个$(q-1)\times(q+3)$矩阵，其中有 3 个校验列，如果要保证任意 3 列丢失的情况下都能够重建数据，则每一个校验列都应该包含对所有数据单元的校验. 由于总共有$(q-1)\times q$ 个数据单元，每一列检验元的构造过程中至少需要进行$(q-1)\times(q-1)$次运算，3 列校验元的生成过程中至少需要执行 $3q^2-6q+3$ 次异或操作. 我们在表 1 中列出了当 q 取不同的值时，VCR 的编码复杂度与能够承受任意 3 个磁盘故障的编码在理论上可能达到的最小值的比较，其中 C_{VCR}代表 VCR 码的编码计算复杂度，$C_{\min}$代表在理论上可能的达到的最小计算复杂度.

表 1　VCR 编码的计算复杂度与理论最小值的比较

q	C_{VCR}	$C_{\min}$	$\frac{C_{\text{VCR}}}{C_{\min}}$
5	64	48	1.33
7	132	108	1.22
11	340	300	1.13
13	480	432	1.11
17	832	768	1.08
19	1044	972	1.07
23	1540	1452	1.06
29	2464	2352	1.04
31	2820	2700	1.04
37	4032	3888	1.03
43	5460	5292	1.03
47	6532	6348	1.02
53	8320	8112	1.02
59	10324	10092	1.02

从表 1 可以看出，当 q 取值很小时，C_{VCR}与 $C_{\min}$的差别比较大，但是当 q 取值达到 17 以上时，C_{VCR}与 $C_{\min}$已经非常接近. 尽管 VCR 码的计算复杂度没有达到理论上的最低值，但是完全可以被接受. 另外，目前还没有一种计算复杂度比 RAID-VCR 更低的可行的编码.

6.2　解码过程的计算复杂度分析

假设有 k 个数据盘发生故障，解码计算过程可以分为包括以下几个步骤：

1. 通过求解式(15)得到发生故障的数据单元与特征值之间的关系表达式. 由于这种计算在整个数据恢复过程中只需要执行一次，与编码矩阵的个数无关，因此其计算量可以忽略不计.

2. 对于每一个编码矩阵，按照式(13)计算特征值. 从公式中可以看出，计算一个特征值需要执行$(q-k)$次异或操作. 在有 k 列故障的情况下，需要计算 $k\times(q-1)$个特征值. 因此，对于一个编码矩阵，这一步需要执行 $k\times(q-k)\times(q-1)$次异或运算.

3. 对于每一个编码矩阵，根据步 2 求得的特征值以及步 1 求得的求解表达式，计算出发生故障的磁盘上的数据元. 需要求解的单元个数为 $k\times(q-1)$，但是求解不同的单元时需要执行的异或运算的次数不是固定的，介于 0 和 $(k-1)\times(q-1)$之间. 因此，对于一个编码矩阵，这一步需要执行的异或操作次数小于$(k^2-k)\times(q-1)^2$.

因此，对于一个编码矩阵，解码过程中执行的异或操作的次数小于 $k^2\times q^2-(3k^2-k)\times q+2k^2-k$.

6.3 小 I/O 请求的过程分析

由于读操作并不牵涉到校验磁盘的 I/O 或者计算，因此，包含相同个数据盘的 RAID-5，EVENODD 和 RAID-VCR 三种结构的读性能相差很小. 而对于写操作，不同的写模式引起的 I/O 次数可能相差很大，首先考虑长度比较大的写操作(Large Writes)，对于 RAID-VCR 系统，如果一次更新某个数据列的全部$(q-1)$个单元，则除了需要对该数据列进行一次写操作外，还需要分别对 3 个校验列执行一次 Read-Modify-Write(RMW)，由于所有的磁盘都可以并行操作，所以上述 1 个写操作与 3 个 RMW 操作可以并行执行. 显然，从长度大的写操作的性能方面考虑，RAID-5，EVENODD 和 RAID-VCR 之间并没有很大的差别.

对于长度很小的写操作(Small Writes)，例如每次只写 1 个单元，同样需要进行多个磁盘的 I/O，而多个连续的 Small Write 虽然可能要更新同一个校验单元，却不能被合并，而上述三种 RAID 结构中校验元的个数是不同的. 在这种情况下，Small Write 所牵涉到的校验单元或校验磁盘的个数就会影响系统性能. 因此，我们将对 RAID-VCR 中的 Small Writes 的过程进行分析.

假设小的写操作在 q 个数据盘上的分布概率是相同的，如果需要被更新的单元既不属于 t_1 也不属于 t_2(参见式(2))，则只有 1 个数据单元和 3 个校验单元(每个校验列 1 个)需要修改，因此总共需要 1 个写操作和 3 个 RMW 操作，在不考虑其它 I/O 操作的情况下，这 4 个操作是可以并行执行的，其执行时间取决于一次 RMW 的时间，为了简单起见和便于表述，我们把上述的 1 个写操作和 3 个 RMW 操作用 4 个 RMW 操作来代替. 如果要更新的数据单元属于 t_1 或者 t_2(不可能同时属于两个)，例如属于 t_1，那么校验列 P_1 的所有单元都需要修改、更新，因此需要 $q+2$ 次 RMW 操作. 由于 $q>2$，所以需要的 RMW 操作次数大于 4.

由于每次磁盘 I/O 都以扇区为单位，为了避免小的写操作引起的性能瓶颈，我们可以采用类似于文献[10]中给出的方法，即通过选择合适的单元大小和素数 q 的值，使得每次 I/O 都是针对整个扇区，而不是一列中的某些单元，这样就可以保证每次写操作只需要做 4 次 RMW 操作. 例如，选择 $q=17$(在磁盘数目不足的情况下，可以假设存在一些全 0 的磁盘)，每个单元包含 32 个字节，这样一个 RAID 条带中的每一列正好包含 512 个字节，而目前磁盘的扇区大小基本上都是 512 个字节. 由于 I/O 以扇区为基本单位，因此每次写操作都正好引起 RAID-VCR 中的 1 个数据列和 3 个校验列的全部更新. 这样就可以保证每次写操作需要的 RMW 次数正好是 4.

6.4 I/O 性能分析

在分析写操作过程的基础上，我们还需要对 RAID-VCR 阵列的吞吐量进行分析. 为了简化性能分析和比较的过程，我们也假定每个单元仅由一个 bit 构成，这种假设并不影响分析和比较的结果.

显然，如果在一个 RAID 结构中使用专门的校验磁盘会使校验磁盘成为系统的 I/O 瓶颈. 虽然前面描述的 RAID-VCR 结构中有 3 个专门的校验磁盘，但是实际上很容易把校验信息分布到所有的 $q+3$ 个成员磁盘上，我们只要按照 RAID-5 中校验信息的分布方法，让 RAID-VCR 中 3 个校验列循环分布即可. 这样，可以认为所有的 I/O 操作在各个成员磁盘上平均分布.

我们用类似于文献[3]中的方法来比较几种不同的 RAID 结构的 I/O 吞吐量. 虽然近年来磁盘容量迅速增大，但是其旋转延迟和寻道延迟并没有多大变化，因此我们依然采用文献[3]中的基本性能数据，即认为寻道时间大约是 10ms，旋转一周的时间大约是 11ms，而一次 RMW 的时间大约是普通磁盘上一次 I/O 的 1.71 倍.

为了简化分析，我们把每个 RMW 操作看成是 1.71 次磁盘简单 I/O 操作. 考虑 1 个由 $q+3$ 个磁盘构成的 RAID-VCR 阵列、1 个由 $q+2$ 个磁盘构成的 EVENODD 阵列和 1 个由 $q+1$ 个磁盘构成的 RAID-5 阵列，每个阵列都有 q 个数据盘(虽然校验信息是循环分布的，但是这不影响我们的分析). 假设在所有的 I/O 请求中读请求占的比例为 r，写请求的比例为 $1-r$，那么对于每个 I/O 请求，上述三种磁盘阵列中每个磁盘平均需要执行的磁盘操作次数分别为 $r/(q+3)+(4\times1.71)(1-r)/(q+3)$，$r/(q+2)+(3\times1.71)(1-r)/(q+2)$和 $r/(q+1)+$

$(2\times1.71)(1-r)/(q+1)$. 假设 1 个磁盘在 1 秒钟内能处理 N(典型的数值是 60)个请求，读请求的比例 r 取比较典型的值 0.75，数据磁盘个数 $q=17$，在这种情况下，我们可以计算得到 RAID-VCR，EVENODD 和 RAID-5 阵列的吞吐量分别为 $8.13N$ 请求/s，$9.34N$ 请求/s 和 $11.2N$ 请求/s.

6.5　3 种 RAID 结构的纵向比较

为了便于全面了解 RAID-VCR 的特点，我们把 RAID-VCR，EVENODD 和 RAID-5 做了纵向比较，主要包括以下几个指标：(1)存储空间利用率(Disk Capacity Utilization，DCU)；(2)能承受的故障磁盘个数(Number of Disk Failures can be Tolerated，NDFT)；(3)计算复杂度；(4)吞吐量. 考虑包含 q 个数据磁盘的 RAID-VCR，EVENODD 和 RAID-5 阵列，比较结果见表 2.

表 2　3 种 RAID 结构的比较

	DCU	NDFT	计算复杂度	吞吐量($q=17$)
RAID-5	$q/(q+1)$	1	q^2-2q+1	$11.2N$
EVENODD	$q/(q+2)$	2	$2q^2-2q-1$	$9.34N$
RAID-VCR	$q/(q+3)$	3	$3q^2-2q-1$	$8.13N$

从表 2 可以看出，虽然 RAID-VCR 带来了一些额外的开销，比如磁盘空间利用率、计算复杂度和吞吐量等方面，但是这些开销并不大，而容灾能力却大幅提高. 尤其是在现代的磁盘阵列中，NVRAM(非易失 RAM)缓存被普遍采用，多个小的 I/O 请求可以被合并为一个大的请求，从而使上述三种阵列的吞吐量差别更小.

7　编码和解码过程模拟实验

从上一节的分析我们可以看出，RAID-VCR 在吞吐量以及小 I/O 请求的处理能力方面与 RAID5 和 EVENODD 的差别并不大，但是编码和解码过程的计算复杂度的差别相对较大. 为了考察 RAID-VCR 的实用性，我们对编码和解码过程的计算进行实验.

由于条件限制，我们只能进行模拟实验，实验的硬件环境为具有一个 Pentium M 1.6GHz CPU 和 512MB 内存的个人计算机，操作系统为 Windows XP，编译环境为 cygwin 2003 加 gcc 3.3.1.

7.1　编码计算模拟实验

在实验中，取素数 q 的值为 17，每个数据单元的大小为 32bits，每列正好为 512bits. 为了便于计算，数据列也取 16 列，这样每个 RAID-VCR 编码矩阵中的用户数据量为 8Kb.

实验程序的任务是模拟完成编码过程，即根据上述用户数据矩阵生成 3 列校验数据. 由于编码计算的执行时间较短，容易引起误差，在实验程序中我们把上述编码过程循环执行 100000 次，这样程序的每次运行都相当于对 800Mb 数据进行编码. 在统计实验结果数据时，我们运行实验程序 10 次，然后取平均值.

实验结果显示：对 800Mb 的数据进行编码，需要的时间为 9.6s，平均编码速度为 83Mb/s.

7.2　解码计算模拟实验

当只有 1 个数据盘故障时，数据恢复过程的计算复杂度与 RAID-5 的数据恢复过程接近或者相同，因此我们没有进行实验.

根据第 6.2 节的分析，在发生 2 个和 3 个磁盘故障的情况下，解码过程的计算复杂度分别不超过 $4q^2-10q+6$ 和 $9q^2-22q-3$.

我们对发生 2 个和 3 个数据盘故障后的解码过程进行模拟实验. 在实验中，我们分别随机选择 2 个或者 3 个数据盘故障，多次进行实验并记录所花费的时间. 实验结果显示，在发生 2 个数据磁盘故障的情况下，解码的性能大约是 80Mb/s；在发生 3 个数据磁盘故障的情况下，解码性能大约是 45Mb/s.

7.3　实验结果分析

目前主流的磁盘阵列系统都是基于光纤网络接口的，常见的光纤网络速度为 1Gb/s，磁盘阵列的吞吐量不超过 125Mb/s，其中还有相当一部分是不需要执行编码运算的读请求，因此我们认为 RAID-VCR 的编码性能(在 PC 上的实验性能是 83Mb/s)基本能够满足磁盘阵列系统的需求.

从实验结果看，RAID-VCR 的解码性能显得相对较低，但是上述实验是在 PC 上进行的. 从磁盘阵列厂商公布的技术指标来看，大部分中高端磁盘阵列都使用 64 位的基于 RISC 架构的 RAID 处理器，并且还有专门的异或运算协处理器(XOR Coprocessor)，这类处理器执行异或运算的性能应该远大于我们在实验中得到的性能. 例如，较早期的主频为 100MHz 的 Intel i960rn(64 位 RISC)处理器即可满足吞吐量为 80Mb/s 的 RAID5 阵列的需求，而当前的磁盘阵列采用的 64 位 RISC 处理器的主频已经接近 1GHz，根据前面的分析和模拟实验结果推算，RAID-VCR 的编码复杂度和解码计算复杂度(在发生 3 个磁盘故障的情况下)分别大约为 RAID5 的 3 倍和 6 倍，把 RAID-VCR 运用到当前的磁盘阵列

系统中是可行的.

8 结 论

由于性能等方面的原因,近年来包含大量磁盘的 RAID 系统应用越来越广泛,随着成员磁盘个数的增多,发生多个磁盘故障的可能性越来越大;另外,虽然磁盘容量越来越大,被访问的次数越来越多,但是磁盘的可靠性却没有提高.这些原因使得应用系统对能够承受多个磁盘故障的 RAID 结构的需求迅速扩大.本文提出的 RAID 大幅提高了 RAID 结构的容灾能力,而需要的额外开销却非常小,因此具有非常好的实用价值.

参 考 文 献

1 Schulze M., Gibson G. A., Katz R. H., Patterson D. A.. How reliable is a RAID. In: Proceedings of the IEEE COMPCON, San Francisco, CA, 1989, 118～123

2 Patterson D., Gibson G., Katz R.. A case for redundant arrays of inexpensive disks(RAID). In: Proceedings of the ACM SIGMOD International Conference on Management of Data, Chicago, 1988, 109～116

3 Blaum M., Brady J. *et al*. EVENODD: An optimal scheme for tolerating double disk failures in RAID architectures. In: Proceedings of the 21st Annual International Symposium on Computer Architecture, Chicago, 1994, 245～254

4 Alvarez G. A., Burkhard W. A., Cristian F.. Tolerating multiple failures in RAID architectures with optimal storage and uniform declustering. In: Proceedings of the 24th Annual International Symposium on Computer Architecture, Colorado, 1997, 62～72

5 Jin Fan. An investigation on new complex rotary codes. In: Proceedings of the IEEE ISIT85, Brighton, UK, 1985, 1～8

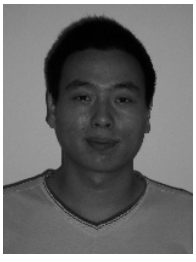
6 Park Chong-Won, Han Young-Year. A practical parity scheme for tolerating triple disk failures in RAID architectures. In: Proceedings of the International Conference on Parallel and Distributed Processing Techniques and Applications, Penang, Malaysia, 2000, 58～68

7 Tau Chih-Shing, Wang Tzone-I. Efficient parity placement schemes for tolerating triple disk failures in RAID architectures. In: Proceedings of the 17th International Conference on Advanced Information Networking and Applications, Xi′an, 2003, 132～139

8 Sivathanu M., Prabhakaran V.. Improving storage system availability with D-GRAID. In: Proceedings of the 3rd USENIX Conference on File and Storage Technologies, San Francisco, 2004, 15～30

9 Blaum M., Roth R. M.. New array codes for multiple phased burst correction. IEEE Transactions on Information Theory, 1993, 39(1): 66～77

10 Feng Dan, Jin Hai, Zhang Jiang-Ling. Improved EVENODD code. In: Proceedings of the IEEE International Symposium on Information Theory, Ulm Germany, 1997, 261～262



DONG Huan-Qing, born in 1976, Ph. D.. His research interests include network storage system and disaster tolerance.

LI Zhan-Huai, born in 1961, Ph. D., professor, Ph. D. supervisor. His research interests include database system and storage system.

LIN Wei, born in 1981, Ph. D. candidate. His research interests include high available system and distributed storage system.

Background

This research is partly sponsored by the National Natural Science Foundation of China under grant No. 60373108. The main purpose of the project focuses on high availability and high reliability of large scale storage systems, especially on Disaster Tolerance and Fault Tolerance of storage systems based on RAID technology. To improve the availability of user data, a novel RAID architecture called RAID-VCR is proposed in this paper. This technique provides a feasible solution for RAID-based storage systems to tolerate triple simultaneous disk failures in any pattern, which improves the availability and fault-tolerant capability of storage systems. The authors have designed and implemented Data Replication System on Linux platform, and studying the management and Quality of Service problem of large scale storage systems.