C-029

分散ファイル記憶システムのための低密度符号の構成法
A Class of Low Density Codes for Distributed Storage Systems

大出 弘之† Hiroyuki Ohde
藤原 英二† Eiji Fujiwara

1 まえがき

分散ファイル記憶システムにおいて，構成するサブシステムに故障が生じても，消失したデータを復元できる高信頼分散ファイル記憶システムが求められている．そのため，データのレプリカ（複製）を作成する手法が存在するが，このためには2倍の記憶容量が必要となり，効率的ではない．これに対し，誤り訂正符号を用いると消失データの復元を効率的に実現できる．例えば，RAID用に開発されたMDSアレー符号としてEVENODD[1]等が存在する．EVENODDは記憶容量の効率に優れた符号であるが，消失データの復元のためにシステムが含むほとんどすべてのデータを読み出す必要がある．そのため，データが広域に分散配置された分散ファイル記憶システムにEVENODDを適用すると，データ復元時のデータ読み出しに要する時間と他のサブシステムへの負荷が大きくなるという問題が生じる．一方，Steiner符号[2]，及びadditive-3符号[2]を分散ファイル記憶システムに適用すると，より少ない読み出しデータ量により消失データを復元できる．

本稿では，Steiner符号，及びadditive-3符号よりも更に容量効率に優れ，データ復元時に計算するデータ量が一般の誤り訂正符号やRAID用MDS符号よりも小さい低密度符号を提案する．

2 システムモデル

複数のサブシステム（ノードと呼ぶ）とそれらを接続するネットワークからなる分散ファイル記憶システムを考える．システムにはK個のシステム本来のノード（情報ノード），M個のデータ冗長化のためのノード（検査ノード），並びにt個のスペアノードが含まれる．これを図1に示す．各検査ノードに対して，複数の情報ノードからなるグループ（パリティグループ）を構成し，グループ毎に情報ノードのデータ（情報データ）の排他的論理和を計算したものを検査ノードに記録しておく．これにより，各パリティグループに含まれるノードのうち，1個までのノード故障に対して消失データを復元することができる．更に，各情報ノードが複数のグループに含まれるようにし，M個のパリティグループを適切に構成することにより，複数個の同時ノード故障に対しても消失データを復元できる．

情報データを更新する際には，関連する複数の検査ノードのデータも更新する必要がある．これを更新ペナルティと呼ぶ[2]．本稿では，ノードにおける記憶装置故障からデータ消失を防ぐ問題を考える．故障が検出されると，スペアノードは消失データの復元が可能なパリティグループを選択し，パリティグループに含まれる利用可能なノードからデータを読み出し，排他的論理和を計算することで消失データを復元する．

3 分散ファイル記憶システム用符号の条件

3.1 準備

ベクトル$\boldsymbol{u}_i, \boldsymbol{u}_j$の論理和，及び論理積のハミング重み（単に重みという）を$w_H(\boldsymbol{u}_i \vee \boldsymbol{u}_j), w_H(\boldsymbol{u}_i \wedge \boldsymbol{u}_j)$と表し，それぞれ$\boldsymbol{u}_i, \boldsymbol{u}_j$の広がり，及び会合数と呼ぶ．

$M \times (K+M)$行列$H = [P|I]$を用いて，M個のパリティグループを表現する．ここで，Pは$M \times K$行列であり，Iは単位行列である．Hの各行をパリティグループに，各列をノードに対応付ける．第iノード$(0 \leq i \leq K+M-1)$が第jパリティグループ$(0 \leq j \leq M-1)$に含まれるとき，Hの第(j,i)要素を1とし，それ以外のとき0とする．

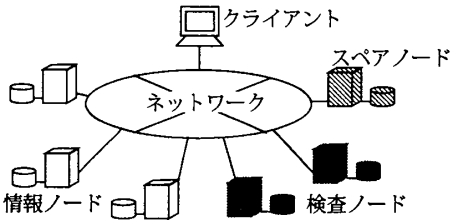

図1: 符号適用分散ファイル記憶システムのモデル

3.2 符号の条件

大規模で高信頼高性能なシステムを実現するために，(1) 同時に複数のノード故障からも消失データを復元でき，(2) 更新ペナルティが最小となり，(3) 消失データ復元時の読み出しデータ量のできるだけ小さい符号を構成する．

任意のt個の同時ノード故障から消失データを復元できるための必要十分条件は，$H = [P|I]$を検査行列として持つ符号の最小距離が$t+1$以上となることである[3]．最小距離が$t+1$以上で更新ペナルティを最小にするために，行列Pのどの列も重みがtであり，どの2列も互いに異なるようにPを構成する．しかし，これらの条件を満たす符号をそのまま適用して，消失データ復元に必要な読み出しデータ量を小さくすることは難しい．ところで，HはMが大きくなるに従い1の要素数の少ない行列，すなわち低密度な行列となる．Hが示すM個のパリティグループだけを用いてt消失すべてを復元できれば，低密度な分だけ復元時の読み出しデータ量を小さく抑えることができる．そのようなHをt消失復元可能な検査行列と呼ぶことにする．本稿では，Pがどの列も重みtを持ち，$H = [P|I]$がt消失復元可能な検査行列となるようにHを構成することにする．

4 符号構成法

$t = 3$の場合の構成法を与える．

定理 1 与えられた$H = [P|I]$が3消失復元可能となるための必要十分条件は，Pの任意に選択した相異なる3列$\boldsymbol{u}_i, \boldsymbol{u}_j, \boldsymbol{u}_k$の広がりが5以上になることである：$w_H(\boldsymbol{u}_i \vee \boldsymbol{u}_j \vee \boldsymbol{u}_k) \geq 5$

3個の故障ノードに対応した3列の広がりが5以上のとき，またそのときに限り，復元可能な故障ノードが1つ以上存在する．更にHのどの2列も互いに異なることから，残りの故障ノードも復元できる．

†東京工業大学大学院情報理工学研究科

定理 2 与えられた部分行列 $P^{(0)}$, 及び $P^{(1)}$ が次の 3 条件を満足するとき, $H = [P^{(0)}|P^{(1)}|I]$ は 3 消失復元可能な検査行列となる：
条件 1　行列 $[P^{(0)}|P^{(1)}]$ のどの列も重み 3 で互いに異なる
条件 2　$P^{(0)}$ のどの 2 列も会合数が 1 以下になる
条件 3　$P^{(1)}$ のどの 2 列も会合数が 1 以下になる

$P = [P^{(0)}|P^{(1)}]$ からどの 3 列を選択しても, その中に会合数 1 以下の 2 列の組が必ず含まれる. よって, 3 列の広がりが 5 以上となることから定理 2 を証明できる.

定理 3 任意の自然数 q に対し, $P^{(0)}, P^{(1)}$ を次のように定義すると, $H = [P^{(0)}|P^{(1)}|I]$ はパラメータ $M = 6q+3, K = M(M-2)/3$ を持つ 3 消失復元可能な検査行列となる：

$$P^{(0)} = \begin{bmatrix} P_S^{(0)} & O & I & \cdots & P_S^{(q-1)} & O & I & I \\ I & P_S^{(0)} & O & \cdots & I & P_S^{(q-1)} & & I \\ O & I & P_S^{(0)} & \cdots & O & I & P_S^{(q-1)} & I \end{bmatrix} \quad (1)$$

$$P^{(1)} = \begin{bmatrix} P_S^{(0)} & O & I & \cdots & P_S^{(q-1)} & O & I \\ O & I & P_S^{(0)} & \cdots & O & I & P_S^{(q-1)} \\ I & P_S^{(0)} & O & \cdots & I & P_S^{(q-1)} & O \end{bmatrix} \quad (2)$$

ここで, $P_S^{(i)} (0 \leqq i \leqq q-1)$ は第 $i+1$ 要素, 及び第 $(2q-i)$ 要素が 1 でありそれ以外が 0 となる $2q+1$ 次元列ベクトルと, それを下方に巡回シフトして得られる $2q$ 個の相異なる列ベクトルを左から並べて得られる $2q+1$ 次正方行列である. O は零行列を表す.

例 1 定理 3 による符号構成例 $(q = 1; M = 9)$ を次に示す：

$$H = \left[\begin{array}{cccccccc|c} 011 & 000 & 100 & 100 & 011 & 000 & 100 & 100000000 \\ 101 & 000 & 010 & 010 & 101 & 000 & 010 & 010000000 \\ 110 & 000 & 001 & 001 & 110 & 000 & 001 & 001000000 \\ \hline 100 & 011 & 000 & 100 & 000 & 100 & 011 & 000100000 \\ 010 & 101 & 000 & 010 & 000 & 010 & 101 & 000010000 \\ 001 & 110 & 000 & 001 & 000 & 001 & 110 & 000001000 \\ \hline 000 & 100 & 011 & 100 & 100 & 011 & 000 & 000000100 \\ 000 & 010 & 101 & 010 & 010 & 101 & 000 & 000000010 \\ 000 & 001 & 110 & 001 & 001 & 110 & 000 & 000000001 \end{array}\right]$$

定理 4 与えられた $M(\geq 4)$ に対し, $\sum_{i=0}^{M-1} i(\boldsymbol{u})_i \equiv 0 \pmod{M}$ を満たす重み 3 の相異なる列ベクトル $\boldsymbol{u} \in \mathrm{GF}(2)^M$ をすべて並べて得られる行列を $P^{(0)}$ とし, $P^{(0)}$ の行を下方に 1 だけ巡回シフトして得られる行列を $P^{(1)}$ とする. このとき, $H = [P^{(0)}|P^{(1)}|I]$ は $K = 2(\lfloor M(M-3)/6 \rfloor + 1)$ の 3 消失復元可能な検査行列となる. ここで, $(\boldsymbol{u})_i (0 \leq i \leq M-1)$ は $\boldsymbol{u}$ の第 i 成分を表す.

例 2 定理 4 による符号構成例 $(M = 9)$ を次に示す：

$$H = \left[\begin{array}{ccc} 1111000000 & 1000000110 & 100000000 \\ 1000110000 & 1111000000 & 010000000 \\ 0100101000 & 1000110000 & 001000000 \\ 0010011100 & 0100101000 & 000100000 \\ 0001001010 & 0010011100 & 000010000 \\ 0001010001 & 0001001010 & 000001000 \\ 0010100011 & 0001010001 & 000000100 \\ 0100000101 & 0010100011 & 000000010 \\ 1000000110 & 0100000101 & 000000001 \end{array}\right]$$

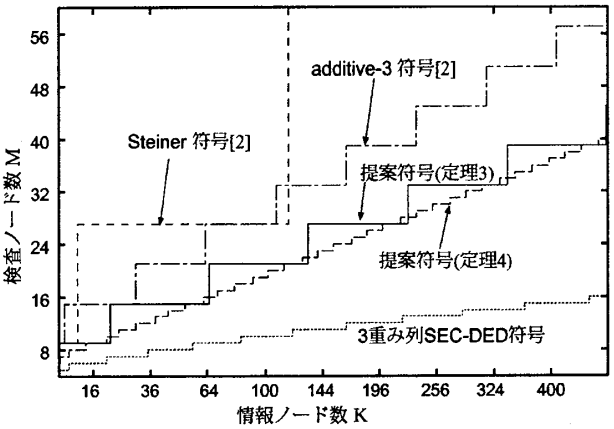

図 2: 情報ノード数 K と検査ノード数 M の関係

表 1: 復元時のデータ読出ノード数の上限 (故障 1 個あたりの個数；() 内は符号化率 $K/(K+M)$ を表す)

K	additive-3	定理 3	定理 4	SEC-DED
30	6(0.67)	6(0.67)	10(0.73)	13 以上 (0.81)
120	11(0.78)	18(0.85)	18(0.85)	36 以上 (0.92)
400	24(0.89)	31(0.91)	33(0.92)	80 以上 (0.96)

5 評価

(1) 記憶容量の効率（符号化率）, 及び (2) 消失データ復元時のデータ読み出しノード数を評価する. 提案符号の K と M の関係を図 2 に示す. 例えば $K = 120$ のとき, Steiner 符号 [2] では $M = 81$, additive-3 符号 [2] では $M = 33$ となるのに対し, 提案符号では $M = 21$ 個の検査ノードで構成できる. なお, 3 重み列 SEC-DED 符号は効率的であるが, 復元時のデータ読み出しノード数の上限を小さくできる保証はない.

3 個までの同時ノード故障に対するデータ読み出しノード数の上限を表 1 に示す. 提案符号では, 3 個までの同時ノード故障に対し故障ノード 1 個あたり $\lceil 3K/M \rceil$ 個以下の読み出しノード数で復元できる. これは, H に示された M 個のパリティグループだけで復元を行う場合の最小の読み出しノード数上限である. 例えば $K = 120$ のとき, 提案符号では 3 個同時故障に対し 1 個あたり $\lceil 3K/M \rceil = 18$ 個以下の読み出しノード数で復元できる. なお 3 重み列 SEC-DED 符号では, M 個のパリティグループだけで復元できない 3 個故障が生じ, 読み出しノード数が $\lceil 3K/M \rceil$ 個を上回ることがある.

6 結論

分散ファイル記憶システムのための低密度符号を提案し, 構成法を与えた. 本符号は, Steiner 符号, 及び additive-3 符号よりも記憶容量の効率（符号化率）が高く, 3 個までのノード故障に対し消失データの復元時に計算するデータ量が, 情報ノード数 K, 及び検査ノード数 M に対して消失データの $\lceil 3K/M \rceil$ 倍以下となる.

参考文献

[1] M. Blaum, et al., "EVENODD: An efficient scheme for tolerating double disk failures in RAID architectures," IEEE Trans. Computers, 44(2):192-202, 1995.
[2] L. Hellerstein, et al., "Coding Techniques for Handling Failures in Large Disk Arrays," Algorithmica, 12(2–3), 1994.
[3] T.R.N. Rao and E. Fujiwara, "Error-Control Coding for Computer Systems," Prentice Hall, 1989.