논문 19-44-02-12 The Journal of Korean Institute of Communications and Information Sciences '19-02 Vol.44 No.02
https://doi.org/10.7840/kics.2019.44.2.274

부분 순환 차족을 이용한 분할 반복 부호의 설계

박 호 성•

Construction of Fractional Repetition Codes Using Partial Cyclic Difference Families

Hosung Park•

요 약

본 논문에서 부분 순환 차족의 한 클래스를 이용하여 분할 반복 부호를 설계하는 방법을 제안한다. 제안하는 부호는 기존에 수학적 구조를 이용해 설계된 부호들과 차별화된 파라미터를 보인다. 또한, 손실 노드 복구시 유연하게 네트워크 대역폭을 사용할 수 있는 장점이 있다.

Key Words : Distributed storage code, fractional repetition code, partial cyclic difference family, replication degree

ABSTRACT

In this paper, we propose a construction method of fractional repetition (FR) codes using a class of partial cyclic difference families. The proposed codes show different parameters from existing FR codes based on mathematical structures. Moreover, they have advantages to enable flexible network bandwidth when repairing a failed node.

Ⅰ. 서 론

현재 발생하는 엄청난 양의 디지털 데이터를 저장하기 위해서 데이터센터의 수가 폭증하고 있으며, 향후 이러한 추세는 더욱 빨라질 것으로 예상되고 있다. 이에 따른 비용 및 공간의 제약 문제는 결국 데이터의 효율적 저장으로 최대한 극복해야 하는 실정이다. 효율적 저장을 위해 핫 데이터 (hot data)를 위한 복제 기반 분산 저장 기술[1]과 콜드 데이터 (cold data)를 위한 소실 부호 (erasure code) 기반 분산 저장 기술[2,3]로 나뉘어 발전하고 있다.

복제 기반 분산 저장 기술은 현재 HDFS[1] 같은 파일시스템에서 이미 사용되고 있는 방법이며, 랜덤한 방법 대신 수학적 구조를 바탕으로 복제된 파일 조각들을 분산시키는 방법은 분할 반복 부호를 통하여 구현될 수 있다. 스테이너 시스템, 그리드, 상호 직교 라틴방진, 하다마드 디자인, 그래프[4], 상대 차족[5], 완전 차족[6] 등을 이용하여 다양한 분할 반복 부호가 설계되었다. [4]와 [5]에서는 사용된 수학적 구조마다 특정 파라미터의 부호들을 얻을 수 있으며, [6]에서는 광범위한 범위의 파라미터에 대해 부호가 설계되었다.

본 논문에서는 반복 차수가 3인 분할 반복 부호를 설계하기 위해 부분 순환 차족[7]을 이용하는 방법을 제안한다. 설계된 부호를 변형하여 여러 파라미터를 가지는 부호들을 만들어낼 수 있으며, 이러한 부호들은 기존에 설계되었던 분할 반복 부호들과 다른 파라미터를 가진다. 제안하는 부호는 손실 노드 복구시 유연하게 네트워크 대역폭을 사용할 수 있는 장점을 보인다.

Ⅱ. 분할 반복 부호

분할 반복 부호는 저장하고자 하는 파일을 여러 조각들로 분할하고, 각 조각을 수차례 복제한 후, 일정한 규칙에 따라 노드들에 분산시킨다. 분할 반복 부호는 복구를 위해 네트워크 대역폭을 최소한으로 이용할 뿐 아니라 디스크의 입출력 및 연산을 최소로 한다. 현재까지 수학적 구조들을 이용하여 다양한 분할 반복 부호들이 제안되었고, 각 설계 방법마다 부호의 파라미터들이 다르므로 보다 다양한 파라미터를 위해서 새로운 설계 방법들이 필요하다.

본 논문에서 데이터의 각 심볼들은 F_q의 원소이고, 파일은 외각 부호화를 거친 후 (생략가능) θ개의 심볼로 이루어지며, 데이터를 저장하기 위한 노드의 수는 n이라고 가정하자. (n,α,ρ) 분할 반복 부호 C는 다음의 조건을 만족시키는 $\{0,1,\ldots,\theta-1\}$의 n개의 부분집합 $N_0,N_1,\ldots,N_{n-1}$으로 정의된다. 1) $n\alpha=\theta\rho$; 2)

※ 본 연구는 2018년도 정부(교육부)의 재원으로 한국연구재단의 지원을 받아 수행된 기초연구사업임 (No. NRF-2018R1D1A1B07051108)
• First Author : School of Electronics and Computer Engineering, Chonnam National University, hpark1@jnu.ac.kr, 종신회원
논문번호 : 201812-396-A-LU, Received December 26, 2018; Revised January 23, 2019; Accepted January 28, 2019

$|N_i| = \alpha$, $i = 0,...,n-1$; 3) 심볼 j는 N_i들에 정확히 ρ번 포함된다. 이와 같이 정의된 분할 반복 부호 C에 의해 노드 i는 N_i에 있는 심볼들을 저장한다.

분할 반복 부호 C는 접속 행렬 $I(C)$에 의해 표현되기도 하는데, 이는 $n \times \theta$의 크기를 가지고 그 (i,j) 원소는 N_i가 심볼 j를 포함하면 1, 그렇지 않으면 0이 된다. 모든 α_i가 α로 일정하고, ρ_j 역시 ρ로 일정할 때 C는 (n,α,ρ) 정규 분할 반복 부호라고 불리고, 그 접속행렬은 행가중치와 열가중치가 각각 일정한 정규 행렬이 된다. 정규 분할 반복 부호가 아니면 비정규 분할 반복 부호라 부른다.

분할 반복 부호를 사용한 분산 저장 장치에서 하나의 노드 i가 손실되었을 때 다른 노드들로부터 α개의 심볼을 다운로드 받으면 그 노드는 복구될 수 있다. 그리고 사용자가 임의의 k개의 노드에 접속해서 다운로드 받을 수 있는 최소의 심볼 수 $M(k)$는 다음과 같이 표현된다.

$$M(k) = \min_{\substack{I \subset \{0,...,n-1\} \\ |I| = k}} |\cup_{i \in I} N_i|$$

주어진 k에 대하여 $M(k)$는 분할 반복 부호가 저장을 보장하는 심볼의 개수를 의미하며, 저장하고자 하는 파일의 크기와 일치한다.

Ⅲ. 부분 순환 차족을 이용한 분할 반복 부호 설계

본 논문에서 부분 순환 차족을 이용하여 분할 반복 부호를 설계하는데, 이를 위해 디자인 이론 [7]에서의 정의들을 설명한다.

정의 1: $B = \{b_0, b_1, ..., b_{k-1}\}$를 가산 군 (additive group) G의 부분집합이라고 하자. B의 G-안정화기 (stabilizer)는 $B + g = B$를 만족시키는 $g \in G$들로 구성된 G의 부분군 (subgroup) G_B로 정의된다. $G_B = \{0\}$인 경우 B는 전블록 (full block)으로, 그렇지 않은 경우는 단블록 (short block)으로 불린다.

정의 2: B의 차 리스트는 멀티셋 $\Delta B = \{b_i - b_j \mid i,j = 0,...,k-1;\ i \neq j\}$으로 정의된다. ΔB의 한 원소 $g \in G$의 중복도 (multiplicity)는 정수 μ_g에 대하여 $\mu_g |G_B|$의 형태가 된다. B의 부분 차 리스트는 각 $g \in G$가 정확히 μ_g번 포함되는 멀티셋 ∂B로 정의된다.

참고로 $\Delta B = \partial B$는 B가 전블록일 필요충분조건이다.

정의 3: G를 차수가 v인 군이라고 하자. 크기가 k인 G의 부분집합(블록)들 $B_0, ..., B_{t-1}$은 0을 제외한 G의 모든 원소들이 $\partial B_0 \cup \cdots \cup \partial B_{t-1}$에 정확히 λ번 나타날 때 (v,k,λ) 차족으로 정의된다. 차족이 적어도 하나의 단블록을 포함할 때 부분 차족으로 불린다.

G가 $Z_v = \{0,1,...,v-1\}$과 동형 (isomorphic)이면 해당하는 차족은 (v,k,λ) 순환 차족으로 불린다. 본 논문에서는 $k=3$인 부분 순환 차족을 이용하여 분할 반복 부호를 설계하고자 한다. 이에 앞서 해당 차족이 존재하는 충분조건은 다음 정리와 같다 [7, p. 614].

정리 1: $v \neq 9$, $v \equiv 3 \pmod 6$ 에 대하여 $(v,3,1)$ 부분 순환 차족이 존재한다.

$(v,3,1)$ 부분 순환 차족의 블록의 개수는 $t = (v+3)/6$이 된다. 아래 표는 $15 \leq v \leq 39$에 대하여 $(v,3,1)$ 부분 순환 차족의 예시이다. 더 많은 파라미터에 대한 설계방법 및 예시는 [7]을 참고할 수 있다.

이제 $(v,3,1)$ 부분 순환 차족을 이용하여 분할 반복 부호를 설계하는 방법을 제안한다. 먼저 $v \neq 9$, $v \equiv 3 \pmod 6$ 에 대하여 $(v,3,1)$ 부분 순환 차족 $B_i = \{b_{i,0}, b_{i,1}, b_{i,2}\}$ $(i = 0, ..., (v-3)/6)$을 설계한다. 여기서 항상 마지막 블록이 단블록이 되도록 한다. 그리고 분할 반복 부호에서 노드의 수 n을 부분 순환 차족의 v와 동일하게 설정하고,

$$N_i = \bigcup_{j=0}^{(v-3)/6} \bigcup_{k=0}^{2} \{(b_{j,k} + i) \bmod v + jv\} \qquad (i = 0, ..., v-1)$$

가 되도록 한다. 그러면 분할 부호는 ($n = v$, $\alpha = (v+3)/2, \rho = 3$)의 파라미터를 가지게 된다.

아래 그림은 표 1의 (15,3,1) 부분 순환 차족을 이

표 1. $(v,3,1)$ 부분 순환 차족의 예 (기울임꼴은 단블록을 나타낸다).
Table 1. Examples for $(v,3,1)$ partial cyclic difference family (short blocks are slanted).

v	Blocks
15	{0,1,4}, {0,2,9}, *{0,5,10}*
21	{0,1,3}, {0,4,12}, {0,5,11}, *{0,7,14}*
27	{0,1,3}, {0,4,11}, {0,5,15}, {0,6,14}, *{0,9,18}*
33	{0,1,3}, {0,4,10}, {0,5,18}, {0,7,19}, {0,8,17}, *{0,11,22}*
39	{0,1,3}, {0,4,18}, {0,5,27}, {0,6,16}, {0,7,15}, {0,9,20}, *{0,13,26}*
45	{0,1,3}, {0,4,10}, {0,5,28}, {0,7,34}, {0,8,32}, {0,9,29}, {0,12,26}, *{0,15,30}*

용하여 상기 설계 방법으로 얻은 (15,9,3) 분할 반복 부호의 접속 행렬이다. 행렬에서 0을 모두 생략하고 1만 표시하였다.

그림 1에서 볼 수 있듯이 제안하는 부호는 마지막 v개의 파일 심볼들에 대하여 3개의 노드에 동일한 3개의 파일 심볼을 분배하여 저장하는 것을 볼 수 있다. 분할 반복 부호에서는 보통 하나의 노드로부터 하나의 파일 심볼을 전달받아 손실된 노드를 복구하는 것을 가정하는데, 제안하는 부호의 상기 성질은 노드 복구를 위해서 하나의 노드로부터 최대 4개의 파일 심볼을 전달받는 것이 가능해진다. 따라서 손실 노드 복구시 동일한 랙에 위치한 노드들처럼 네트워크 연결에 좋은 노드들 간에 대역폭을 유연하게 사용할 수 있는 장점을 제공한다.

한편, 제안하는 부호의 접속행렬에서 맨 뒤의 정사각형에 해당하는 부분행렬에서 마지막 $v/3$개 또는 $2v/3$개의 열을 제거해도 행가중치 및 열가중치가 일정한 정규행렬이 됨을 알 수 있다. 위의 방법으로 $(n=v,\alpha=(v+1)/2,\rho=3)$ 분할 반복 부호와 $(n=v,\alpha=(v-1)/2,\rho=3)$ 분할 반복 부호를 설계할 수 있다.

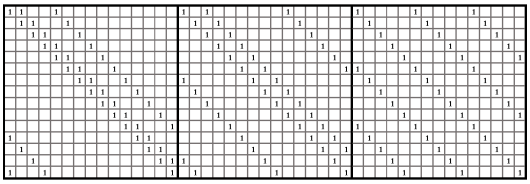

그림 1. 제안하는 (15,9,3) 분할 반복 부호의 접속 행렬.
Fig. 1. Incidence matrix of the proposed (15,9,3) fractional repetition code.

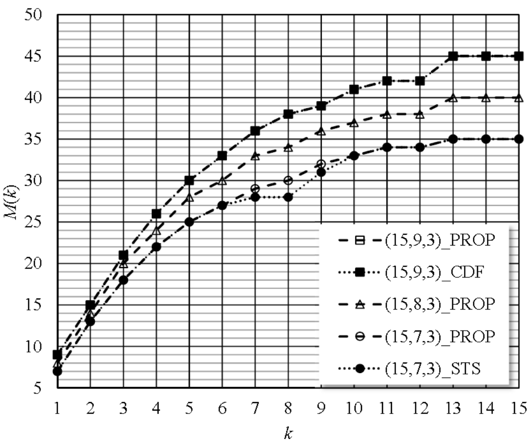

그림 2. (15,3,1) 부분 순환 차족으로부터 생성된 제안하는 3종류의 부호들의 파일 크기 비교
Fig. 2. File size comparison for the three proposed codes constructed from (15,3,1) partial cyclic difference family.

Ⅳ. 지원하는 파일 크기 모의실험

제안하는 분할 반복 부호 중 $(v,(v+3)/2,3)$ 및 $(v,(v+1)/2,3)$부호는 기존 설계된 부호 중 동일한 파라미터를 가지는 것이 없고, $(v,(v-1)/2,3)$ 부호는 스테이너 3중 시스템 (Steiner triple system, STS)으로부터 설계된 분할 반복 부호 [4]와 동일한 파라미터를 가진다. 그럼에도 불구하고 $(v,(v+3)/2,3)$부호에 대해서 지원하는 파일크기를 검증하기 위해 순환 차족 (cyclic difference family, CDF)으로부터 설계된 분할 반복 부호 [6]에서 부호의 길이를 짧게 변형하여 제안한 부호와 동일한 파라미터의 부호를 얻은 후 비교한다. 파일 크기 시뮬레이션을 위해 그림 1의 제안하는 (15,9,3) 부호와 (19,3,1) CDF로부터 얻은 (15,9,3) 부호를 비교하고, 제안하는 부호에서 마지막 열들을 제거해서 얻은 (15,8,3), (15,7,3) 부호들과 STS(15)부터 얻은 (15,7,3) 부호를 비교하여 그림 2에 나타내었다. 제안하는 (15,9,3) 부호는 CDF로부터 얻은 부호와 정확히 동일한 파일 크기를 지원하며, 제안하는 (15,7,3) 부호는 STS로부터 얻은 부호에 비해 $7 \le k \le 9$에 대해 더 큰 파일 크기를 가진다.

References

[1] K. Shvachko, H. Kuang, S. Radia, and R. Chansler, "The Hadoop distributed file system," in *Proc. IEEE Symp. Mass Storage Syst. Technol.*, pp. 1-5, Incline Village, NV, USA, May 2010.

[2] D. S. Papailiopoulos and A. G. Dimakis, "Locally repairable codes," *IEEE Trans. Inf. Theory*, vol. 60, no. 10, pp. 5843-5855, Oct. 2014.

[3] M.-Y. Nam, J.-H. Kim, and H.-Y. Song, "Locally repairable codes for distributed storage systems," *KICS Inf. and Commun. Mag.*, vol. 32, no. 6, pp. 3-8, Jun. 2015.

[4] O. Olmez and A. Ramamoorthy, "Fractional repetition codes with flexible repair from combinatorial designs," *IEEE Trans. Inf. Theory*, vol. 62, no. 4, pp. 1565-1591, Apr. 2016.

[5] H. Park and C.-S. Kim, "Distributed storage

codes with multiple replication degrees using relative difference families,” *J. KICS*, vol. 41, no. 12, pp. 1768-1770, Dec. 2016.

[6] H. Park and Y.-S. Kim, “Construction of fractional repetition codes with variable parameters for distributed storage systems,” *Entropy*, vol. 18, no. 12, pp. 441:1-8, Dec. 2016.

[7] C. J. Colbourn and J. H. Dinitz, *The CRC Handbook of Combinatorial Designs,* 2nd Ed., Boca Raton, FL, USA: Chapman and Hall/CRC, 2006.