

선형 재생 부호 앙상블의 평균 복구 접속 비용

박진수*, 김정현*, 박기현*, 송홍엽^o

Average Repair Read Cost of Linear Repairable Code Ensembles

Jin Soo Park*, Jung-Hyun Kim*, Ki-Hyeon Park*, Hong-Yeop Song^o

요약

본 논문에서는 임의의 선형 재생 부호 앙상블에 대하여 복구 대역폭(Repair bandwidth)과 접속 비용(Repair read cost)의 평균을 유도한다. 한 데이터가 여러 노드에 부호화 되어 분산 저장된 상황에서 하나의 노드가 소실될 경우, 이를 복구하기 위해 필요한 데이터 량을 복구 대역폭, 접속해야 하는 노드 수를 복구 접속 비용이라 한다. 이 때, 선형 재생 부호 앙상블은 데이터 심볼의 수 k 와 패리티 심볼의 수 m , 그리고 그들의 차수 분포로 주어진다. 우리는 이러한 부호들이 시스탬에틱(Systematic)이며 정확한 복구(Exact repair)를 수행하고 $n = k + m$ 개의 모든 저장소(Storages)들이 전부 연결되어 있는 상황을 가정한다. 본 논문의 결과는 파운틴 부호 등과 같이 위와 같은 파라미터들로 랜덤하게 만들어진 부호들에 바로 적용 가능하다. 최종 결과식은 평균 복구 접속 비용이 차수 분포와 n , k 에 따라 결정됨을 보여준다.

Key Words : Distributed storages, regenerating codes, locality, degree distributions, repair read cost

ABSTRACT

In this paper, we derive the average repair bandwidth and/or read cost for arbitrary repairable linear code ensembles. The repair bandwidth and read cost are the required amount of data and access number of nodes to restore a failed node, respectively. Here, the repairable linear code ensemble is given by such parameters as the number k of data symbols, the number m of parity symbols, and their degree distributions. We further assume that the code is systematic, and no other constraint is assumed, except possibly that the exact repair could be done by the parity check-sum relation with fully connected $n = k + m$ storages. This enables one to apply the result of this paper directly to any randomly constructed codes with the above parameters, such as linear fountain codes. The final expression of the average repair read cost shows that it is highly dependent on the degree distribution of parity symbols, and also the values n and k .

I. 서론

최근, 국내외에서 클라우드 서비스에 대한 연구^{1,2)} 뿐만 아니라, 분산 저장 시스템(Distributed storage

system, DSS)을 위한 소실 부호 기술(Erasure coding)이 클라우드 서비스 제공자들과 연구자들의 관심을 끌고 있다^{3,7)}. DSS를 위한 부호 기법의 주요한 두가지 성능 척도는 필요 저장 공간과 하나의 저장소가 소

* This research was supported by Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education(No. 2013R1A1A2062061).

• First Author : School of Electrical and Electronic Engineering, Yonsei University, js.park09@yonsei.ac.kr, 학생회원

◦ Corresponding Author : School of Electrical and Electronic Engineering, Yonsei University, hysong@yonsei.ac.kr, 종신회원

* School of Electrical and Electronic Engineering, Yonsei University, jh.kim06@yonsei.ac.kr, kh.park@yonsei.ac.kr, 학생회원

논문번호 : KICS2014-08-333, Received August 30, 2014; Revised November 6, 2014; Accepted November 6, 2014

실되었을 때 이를 복구하기 위해 필요한 대역폭(즉, 데이터의 양)이다. 이 두가지 척도에 대한 바운드가 잘 알려져 있는데, 어떤 부호가 그 바운드를 만족하면 최적이라 한다⁴⁾. 또한 두 척도에 대한 양극단의 최적 지점을 만족하는 부호를 각각 최소 저장 용량 재생(Minimum storage regenerating, MSR) 부호, 최소 복구 대역폭 재생(Minimum bandwidth regenerating, MBR) 부호라 한다. 기존의 소실 부호에 대한 연구⁸⁾는 이진 소실채널에서 DSS를 고려하지 않고 진행되었으나, 최근에는 DSS를 위해 최적 및 극단 지점을 만족하도록 설계된 repair-by-transfer 부호⁹⁾, product-matrix 알고리즘¹⁰⁾, 그리고 최대 거리 분리가능(Maximum distance separable, MDS) 부호⁵⁾ 등이 제안되었다.

바운드의 중간 부분에 대한 재생 부호(Regenerating codes) 또한 실제적으로 중요한데, 이러한 관점에서 부분성(Locality) 개념이 도입되었다. 만약 어떤 저장소가 소실되어 새로운 저장소에 복구해 놓아야 하는 경우, 전체 데이터 심볼 수 대비 훨씬 적은 양의 저장소만 접속하여 복구할 수 있으면 부분 접속으로 복구 가능(Locally decodable)하다고 부른다⁷⁾. 그러한 부호를 부분접속 복구 부호(Locally repairable code, LRC)라 하는데, 이에 대한 간단하며 실용적인 연구가 있었으며^{11,12)} 부호 파라미터와 부분성에 연구가 진행되었다^{13,14)}. 또한 저장 공간과 부분성에 대한 trade-off를 주는 피라미드 부호가 제안되었으며¹⁵⁾, 자연적으로 부분접속 복구 성질을 갖는 파운틴 부호에 대한 near-MDS 성질의 연구가 발표되었다¹⁶⁾. 또한, 복구를 위한 접속 노드 수를 줄일 수 있는 간섭 정렬(Interference alignment, IA) 기법^{17,18)}이 발표되었다.

본 논문에서는 선형 재생 부호 이상상의 평균 복구 접속 비용과 대역폭을 유도한다. 우리는 평균 복구 접속 비용, 즉 하나의 소실 저장소를 복구하기 위해 필요한 다른 저장소 접속 수를 주로 고려하며 이로부터 복구 대역폭을 쉽게 유도할 수 있다. 필요한 입력 파라미터로는 데이터 심볼의 수 k , 패리티 심볼의 수 m , 그리고 그들의 차수 분포이다. 각 패리티 심볼은 데이터 심볼들의 선형 결합으로 만들어지며, $n = k + m$ 개의 부호 심볼들은 n 개의 저장소에 각각 저장되며 모든 저장소는 연결되어있다고 가정하자. 이 때, 패리티 심볼과 데이터 심볼간 연관성은 차수 분포를 제외하고는 특정짓지 않도록 한다. 이러한 가정은 본 논문의 가정을 재생 파운틴 부호¹³⁾ 등에 직접 적용 가능하게 한다.

본 논문은 II장에서 선형 재생 부호를 간략히 설명하고, III장에서는 주 결과인 평균 복구 접속 비용과 대역폭을 유도한다. 마지막 IV장에서는 시뮬레이션 결과들과 결론으로 논문을 마친다.

II. 시스템 모델

본 장에서는 일반적인 선형 재생 부호에 대하여 설명한다^{3-6,12)}. 본 논문에서는 오직 시스터메틱(Systematic) 부호와 정확한 복구(Exact repair)만을 가정한다⁴⁾. 부호화의 첫 과정은 한 데이터 파일을 k 개로 조각내어 k 개의 데이터 심볼로 삼는 것이다. 이 데이터 심볼들을 시스터메틱 심볼이라 하고, 부호화기는 시스터메틱 심볼을 선형 결합하여 m 개의 패리티 심볼을 생성한다. 이렇게 만들어진 $n = k + m$ 개의 부호화된 심볼들은 n 개의 서로 다른 저장소에 저장된다. 심볼의 크기가 α 라 하면, 각 저장소에는 α 만큼의 데이터가 저장된다. 한 저장소의 복구를 위해서는 살아있는 다른 저장소들에서 데이터를 가져와야 하는데, 이 때 각 저장소들에서 α 만큼의 데이터를 모두 가져오는 것으로 가정한다. 본 논문에서 S 와 P 는 각각 모든 시스터메틱 심볼과 패리티 심볼의 집합을 나타낸다. 어떻게 부호화를 하는지에 따라 그림 1과 같은 이분 그래프를 그릴 수 있다. 그림에서 원과 '+' 마크된 사각형은 각각 시스터메틱 심볼과 패리티 심볼을 의미하며 한 패리티 심볼의 체크섬 방정식에 참여하는 시스터메틱 심볼들은 해당 패리티 심볼과 엣지로 연결된다. 이 때, 각 엣지는 해당 체크섬 방정식에서의 해당 시스터메틱 심볼들의 계수값을 가지고 있다. 한 심볼에 연결되어있는 엣지의 수를 본 논문에서는 해당 심볼의 차수라고 부르도록 한다. 이 때, 각 시스터

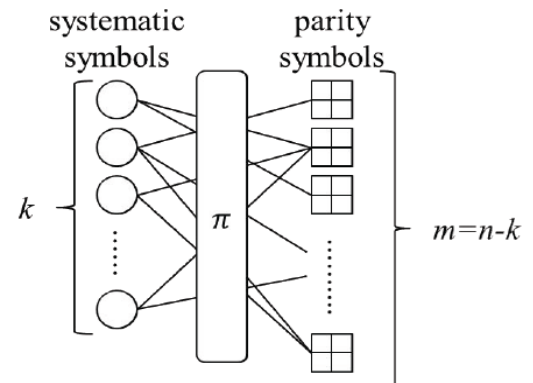


그림 1. 시스터메틱 선형 부호의 일반적인 그래프 표현
Fig. 1. Graph representation of a systematic linear code

메틱, 패리티 심볼들의 차수 분포를 각각 $\mu^s = (\mu_1^s, \mu_2^s, \dots, \mu_m^s)$, $\mu^p = (\mu_1^p, \mu_2^p, \dots, \mu_k^p)$ 로 적는다. μ_i^s 와 μ_i^p 는 각각 k 개의 시스터메틱 심볼들 또는 m 개의 패리티 심볼들 중 차수가 i 인 심볼들의 비율을 나타낸다.

일반적으로, 시스터메틱 심볼과 패리티 심볼들의 정확한 복구를 하는 과정은 서로 다르다. 차수가 q 인 패리티 심볼을 복구하기 위해서는 그래프 상에서의 q 개의 이웃한 시스터메틱 심볼들(즉, 패리티 심볼을 만들었던 q 개 시스터메틱 심볼들)을 읽어오면 된다. 따라서 복구 접속 비용은 q 가 된다. 하지만 차수가 d 인 시스터메틱 심볼을 최소 복구 비용으로 복구하려면 우선 해당 시스터메틱 심볼의 이웃한 d 개의 패리티 심볼들 중 차수가 가장 작은 심볼을 골라야 한다. 그 패리티 심볼의 차수가 q 라고 하면, 해당 패리티 심볼과 그 패리티 심볼에 이웃한 나머지 $q-1$ 개의 시스터메틱 심볼들에 접속해야 한다. B_{p_l} 과 B_{s_j} 를 각각 l 번째 패리티 심볼 p_l 과 j 번째 시스터메틱 심볼 s_j 의 복구 접속 비용이라 하자. 그러면 $l \in \{1, \dots, m\}$, $j \in \{1, \dots, k\}$ 에 대하여 다음과 같이 적을 수 있다.

$$\begin{aligned} B_{p_l} &= \deg(p_l) \\ B_{s_j} &= \min_{l \in N(s_j)} \deg(p_l) \end{aligned} \quad (1)$$

이 때, $\deg(x)$ 는 어떤 심볼 x 의 차수를 뜻하며, $N(x)$ 는 어떤 심볼 x 의 이웃한 노드들의 인덱스 집합을 뜻한다. 본 논문에서는 하나의 저장소, 즉 하나의 심볼이 소실되면 곧바로 정확한 복구를 하는 것으로 가정한다. 또한 본 논문의 결과는 $GF(2)$ 를 포함하여 어떤 크기의 필드에서도 적용 가능하다.

III. 평균 복구 접속 비용과 대역폭

본 장에서는 어떤 주어진 차수 분포 μ^s , μ^p 와 파라미터 n , k (또는 m)에 대한 평균 복구 접속 비용 \bar{B} 를 유도한다. 평균 복구 대역폭은 본 시스템 모델에서는 간단히 $\alpha \bar{B}$ 가 된다. 우선 전체 엣지의 수는 $E_T = k \sum_j j \mu_j^s = m \sum_i i \mu_i^p$ 과 같이 구할 수 있다. 하나의 소실된 심볼을 erasure(ϵ)로 부르면, 평균 복구 접속 비용은

$$\begin{aligned} \bar{B} &= \Pr[\epsilon \text{ in } S] \bar{B}_s + \Pr[\epsilon \text{ in } P] \bar{B}_p \\ &= \frac{k}{n} \bar{B}_s + \frac{m}{n} \bar{B}_p \end{aligned} \quad (2)$$

로 쓸 수 있다. 이 때, \bar{B}_s 와 \bar{B}_p 는 각각 S 와 P 의 평균 복구 접속 비용이다. 만약 엣지 연결까지도 주어진 상황이라면, 각각은 다음과 같이 간단히 표현 가능하다.

$$\begin{aligned} \bar{B}_s &= \frac{1}{k} \sum_{j=1}^k B_{s_j} = \frac{1}{k} \sum_{j=1}^k \min_{l \in N(s_j)} \deg(p_l) \\ \bar{B}_p &= \frac{1}{m} \sum_{l=1}^m B_{p_l} = \frac{E_T}{m} \end{aligned} \quad (3)$$

하지만 엣지 연결은 정의되지 않은 부호 앙상블에 대한 \bar{B}_s 를 다시 쓰면 다음과 같다.

$$\bar{B}_s = \sum_{d=1}^m \mu_d^s E[\min_{l \in N(s')} \deg(p_l) | \deg(s') = d] \quad (4)$$

이 때, s' 는 차수가 d 인 어떤 임의의 시스터메틱 심볼이며, 평균은 주어진 d 값에 대하여 이웃한 패리티 심볼들 중 최소 차수가 어떤 확률로 나타나는가에 따라 계산된다. 모든 패리티 심볼의 차수는 1부터 k 까지 될 수 있으므로,

$$E[\min_{l \in N(s')} \deg(p_l) | \deg(s') = d] = \sum_{q=1}^k q F(q, d) \quad (5)$$

로 계산할 수 있으며, 이 때 $F(q, d)$ 는 다음과 같이 정의된다.

$$F(q, d) = \Pr[\min_{l \in N(s')} \deg(p_l) = q | \deg(s') = d] \quad (6)$$

먼저, $\min_{l \in N(s')} \deg(p_l)$ 이 가질 수 있는 값의 최대 값을 구하자. 이를 위해 패리티 심볼들을 차수에 따라 내림차순으로 재정렬 하자. s' 의 이웃으로 처음 d 개의 패리티 심볼을 선택하면, 마지막 연결된 패리티 심볼의 차수가 곧 $\min_{l \in N(s')} \deg(p_l)$ 의 최대값이 된다. 즉, $Q(d)$ 를 d 번째 패리티 심볼의 차수라고 정의하면, 차수가 d 인 한 시스터메틱 심볼 s' 의 $\min_{l \in N(s')} \deg(p_l)$ 값은 최대 $Q(d)$ 가 된다. 따라서, $F(q, d)$ 는 $q > Q(d)$ 일 때 0이 된다. $q \leq Q(d)$ 인 경우, q 가 $N(s')$ 에 해당하는 패리티 심볼들 중의 최소 차수를 뜻하므로 다음을 만족해야 한다.

- s' 와 차수가 q 보다 작은 패리티 심볼간은 엣지 연결이 없다.
- s' 은 반드시 차수가 q 인 패리티 심볼들 중 하나 이상을 이웃해야 한다.
- s' 의 나머지 $d-1$ 엣지들은 반드시 차수가 q 이상인 패리티 심볼들에 연결되어야 한다.

즉, s' 는 반드시 차수가 q 이상인 패리티 심볼들에 만 엣지로 연결되어야 하며, 차수가 q 인 패리티 심볼에 최소한 한번 이상은 엣지로 연결되어야 한다. 이때, A_i 를 차수가 i 이상인 패리티 심볼의 개수로 정의 하자.

$$A_i = m \sum_{j=i}^k \mu_j^p \quad (7)$$

전체 m 개의 패리티 심볼들 중 d 개를 고르는 경우들 중에, 위 세가지 조건에 만족하도록 $F(q,d)$ 를 정리하면 다음과 같다.

$$F(q,d) = \begin{cases} \binom{A_q}{d} - \binom{A_{q+1}}{d}, & q \leq Q(d) \\ 0, & q > Q(d) \end{cases} \quad (8)$$

최종적으로, 평균 복구 접속 비용 \bar{B} 를 정리하면 다음과 같다.

$$\bar{B} = \frac{k}{n} \sum_{d=1}^m \mu_d^s \sum_{q=1}^{Q(d)} q F(q,d) + \frac{E_T}{n} \quad (9)$$

식 (9)에서 $F(q,d)$ 를 식 (8)로 치환하면 다음과 같이 정리할 수 있다.

$$\bar{B} = \frac{k}{n} \sum_{d=1}^m \mu_d^s \left(\sum_{q=1}^{Q(d)} \binom{A_q}{d} - Q(d) \binom{A_{Q(d)+1}}{d} \right) / \binom{m}{d} + \frac{E_T}{n} \quad (10)$$

IV. 시뮬레이션 및 분석

식 (10)은 주어진 파라미터 n, k, μ^s, μ^p 에 대한 평균 복구 접속 비용의 성능 척도가 될 수 있다. 우선, $\mu^p = (\mu_3^p, \mu_4^p, \mu_5^p, \mu_6^p) = (1/5, 3/5, 1/5)$, $\mu_5^s = 1, k/n = 0.5$ 로 고정해놓고 k 를 10부터 수천까지 변화시킨 경우

의 \bar{B} 가 그림 2의 왼쪽 그래프에 나타나있다. 그림 2의 우측은 다른 파라미터는 동일하고 $\mu^p = (\mu_3^p, \mu_4^p, \mu_5^p, \mu_6^p, \mu_7^p) = (1/5, 1/5, 1/5, 1/5, 1/5)$ 인 경우의 그래프이다.

정해진 k/n 과 어떤 d 에 대하여 $\mu^s = (\mu_d^s) = (1)$ 일 때, 식 (10)의 \bar{B} 는 k 에 따라 단조증가하며 바운드 되어있다. 그림 2에 따르면, 두 경우에 이러한 바운드가 각각 약 4.67과 4.21 근처에서 나타나는 것으로 보인다. 이러한 바운드는 아직 정확히 구해지지 않았다. 단, \bar{B}_p 와 \bar{B}_s 에 대하여 각각 $\max_{s \in S} \deg(s), \max_{p \in P} \deg(p)$ 의 자명한 바운드가 있다. \bar{B}_s 의 경우, 여전히 자명하지만 조금 더 좋은 바운드로는 $Q(d)$ 가 있다. 다만, k 또는 m 이 매우 작을 때를 제외하면 거의 $\max_{p \in P} \deg(p)$ 와 동일하다. 또한, 그림 2의 두 그래프는 μ^p 가 달라질 때 바운드가 서로 다를 수 있음을 보여준다. 우측 그래프가 같은 k 에 대하여 더 낮은 \bar{B} 값을 보이는데, 이는 $\min_{p \in P} \deg(p)$ 가 \bar{B} 에 주도적인 역할을 하기 때문으로 해석된다.

위에서 구한 식이 실제로 잘 맞는지를 몇가지 시뮬레이션 결과들을 표 1을 통해 확인해 보았다. 이 때, $n = 20, k = 10$ 으로 고정하였고 테스트 해본 차수 분포들은 표 1에 나타나있다. 각 열은 한 쌍의 μ^s 와 μ^p 을 나타내며 \bar{B}_{sim} 은 10,000개의 선형 재생 부호를 랜덤하게 생성하고 각 부호들에 대해 하나의 심볼을 삭제한 후에 필요한 복구 접속 수의 평균을 계산한 것

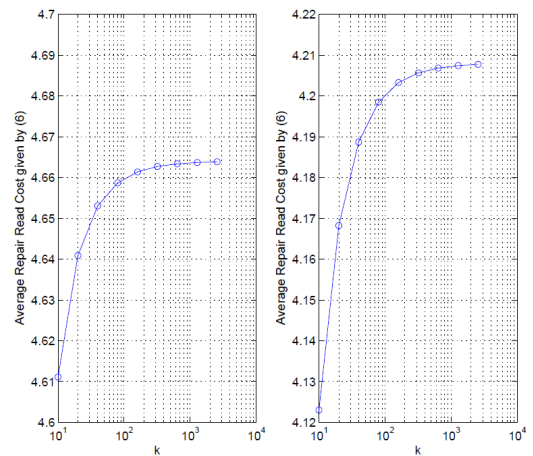


그림 2. k/n 이 고정되어 있을 때, k 의 변화에 따른 식 (10)의 평균 복구 비용 \bar{B} 의 변화.
Fig. 2. Average repair read cost \bar{B} given by (10) for various k with fixed $k/n = 0.5$.

표 1. 식 (10)과 그 경향을 확인하기 위해 임의 설정한 타겟 차수 분포들과 그때의 결과
Table 1. Some target degree distributions and results for confirming the equation (10) and their tendency.

μ_3^s					0.2			0.2
μ_4^s		0.2		0.2	0.2		0.2	0.2
μ_5^s	1	0.6	1	0.6	0.2	1	0.6	0.2
μ_6^s		0.2		0.2	0.2		0.2	0.2
μ_7^s					0.2			0.2
μ_3^p						0.2	0.2	0.2
μ_4^p			0.2	0.2	0.2	0.2	0.2	0.2
μ_5^p	1	1	0.6	0.6	0.6	0.2	0.2	0.2
μ_6^p			0.2	0.2	0.2	0.2	0.2	0.2
μ_7^p						0.2	0.2	0.2
\overline{B}_{sim}	5.0000	5.0000	4.6672	4.6687	4.6779	4.2902	4.3021	4.3464
\overline{B}_{cal}	5.0000	5.0000	4.6111	4.6133	4.6222	4.1230	4.1286	4.1527
error(%)	0	0	1.20	1.19	1.19	3.90	4.03	4.46

이다. \overline{B}_{cal} 은 식 (10)으로 계산된 값이다. 이 때 이들의 오차는 $(\overline{B}_{sim} - \overline{B}_{cal}) / \overline{B}_{sim}$ 으로 구하였다. 이 오차는 표 1의 마지막 행에 기록하였다. 모든 패리티 심볼들이 같은 차수를 갖는다면 당연히 그 차수가 곧 평균 복구 접속 비용이 된다. 이는 표 1의 처음 두 개의 차수 분포 쌍에서 나타난다. 표 1의 3-5번째 차수 분포 쌍과 6-8번째 차수 분포 쌍은 각각 동일한 μ^p 를 갖는다. 같은 μ^p 에 대해서는 평균 복구 접속 비용은 거의 비슷한 값을 갖는다. 즉, 평균 복구 접속 비용 또는 대역폭에 대해서는 μ^s 보다 μ^p 가 더 적절히 설계되어야 함을 의미한다.

본 논문에서는 부호 앙상블에 대해 고려하였으나, 부호 구조가 정해져있는 피라미드 부호^[15]와 부분 접속 복구 부호^[11]에 대해서도 적용해 보았다. 몇가지 파라미터에 대해서 실험하여 보았을 때, 본 논문의 결과와 비슷한 값을 보이는데, 예를 들어 (10,6) 기본 피라미드 부호(Basic pyramid code)의 평균 복구 접속 비용은 3.6이다. 이 부호의 차수 분포인 $\mu_3^s = 1$, $\mu_5^p = 0.5$, $\mu_6^p = 0.5$ 를 (10)에 적용하면 동일한 3.6이 계산된다.

V. 결론

본 논문에서는 부호의 길이와 시스터메틱 심볼의 수가 주어졌을 때 같은 차수 분포를 갖는 부호 앙상블에 대하여, 하나의 노드가 소실되었을 경우 이를 복구

하기 위해 몇 개의 다른 노드에 접속해야 하는지를 계산하였다. 이를 평균 복구 접속 비용이라 하며, 본문에서 유도한 식에 대한 결과를 실험적으로 확인하였다.

본 논문에서는 오직 각 부호 심볼들 간의 연결에 대해서만 고려하였고, 간접 정렬 기법 (IA)^[17,18]을 사용한 경우는 고려하지 않았다. IA 기법을 사용하면 복구 접속 비용은 더 줄어 들 수 있지만, IA 기법을 본 논문에서와 같이 분석하는 것은 그 동작을 예측하기 어려워, 본 논문에서는 고려하지 않았다. IA 기법까지 고려한 환경에 대해서는 앞으로 더욱 연구 되어야 할 것으로 생각된다.

또한 본 논문에서는 부호의 전체적인 신뢰도를 고려하지 않았다. 실제로, 평균 복구 접속 비용을 최소화 하려면 $\mu_1^p = 1$ 로 만들면 되지만, 이 경우 신뢰도의 문제가 생기게 된다. 이러한 신뢰도 분석을 위하여, 주어진 차수 분포에 대한 파운틴 부호의 분석 방법^[19]을 본 논문에서 고려하는 시스터메틱 상황에 응용할 수 있을 것이다. 또한, 실험 결과에 따라, k 가 증가할수록 평균 복구 접속 비용이 증가하게 되지만, k 가 증가할수록 신뢰도에서 이득을 볼 수 있다. 따라서, k 와 n 은 신뢰도와 평균 복구 접속 비용의 tradeoff 관계를 고려하여 결정되어야 한다.

References

- [1] T.-H. Kim, J. Kim, and Y. I. Eom, "A scheme on high-performance caching and high-

- capacity file transmission for cloud storage optimization,” *J. KICS*, vol. 37C, no. 8, Aug. 2012.
- [2] I. Y. Jung, I. Jo, and Y. Yu, “Trust Assurance of Data in Cloud Computing Environment,” *J. KICS*, vol. 36B, no. 9, Sept., 2011.
- [3] J.-H. Kim, J. S. Park, K.-H. Park, M. Y. Nam, and H.-Y. Song, “Trends of regenerating codes for next-generation cloud storage systems,” *Inf. Commun. Mag.*, vol. 31, no. 2, pp. 125-131, Feb. 2014.
- [4] A. G. Dimakis, P. B. Godfrey, Y. Wu, M. J. Wainwright, and K. Ramchandran, “Network coding for distributed storage systems,” *IEEE Trans. Inf. Theory*, vol. 56, no. 9, Sept. 2010.
- [5] A. G. Dimakis, K. Ramchandran, Y. Wu, and C. Suh, “A survey on network codes for distributed storage,” in *Proc. IEEE*, vol. 99, no. 3, Mar. 2011.
- [6] J. Li and B. Li, “Erasure coding for cloud storage systems: a survey,” *Tsinghua Sci. Technol.*, vol. 18, no. 3, Jun. 2013.
- [7] F. Oggier and A. Datta, *Coding Techniques for Repairability in Networked Distributed Storage Systems*, Foundations and Trends in Communications and Information Theory, MA: Now Publishers Inc., 2013.
- [8] H. T. Lim, M. J. Park, S. G. Kang, and E. K. Joo, “Performance analysis of RS, turbo and LDPC code in the binary symmetric erasure channel,” *J. KICS*, vol. 35, no. 2, Feb. 2010.
- [9] N. B. Shah, K. V. Rashmi, P. V. Kumar, and K. Ramchandran, “Distributed storage codes with repair-by-transfer and nonachievability of interior points on the storage-bandwidth tradeoff,” *IEEE Trans. Inf. Theory*, vol. 58, no. 3, Mar. 2012.
- [10] K. V. Rashmi, N. B. Shah, and P. V. Kumar, “Optimal exact-regenerating codes for distributed storage at the MSR and MBR points via a product-matrix construction,” *IEEE Trans. Inf. Theory*, vol. 57, no. 8, Aug. 2011.
- [11] M. Sathiamoorthy, M. Asteris, D. Papailiopoulos, A. G. Dimakis, R. Vadali, S. Chen, and D. Borthakur, “XORing elephants: Novel erasure codes for big data,” in *Proc. 39th Int. Conf. Very Large Data Bases*, vol. 6, no. 5, pp. 325-336, Mar. 2013.
- [12] C. Huang, H. Simitci, A. Ogun, B. Calder, P. Gopalan, J. Li, and S. Yekhanin, “Erasure coding in windows azure storage,” in *Proc. 2012 USENIX Annu. Tech. Conf. (ATC '12)*, Boston, MA, Jun. 2012.
- [13] P. Gopalan, C. Huang, H. Simitci, and S. Yekhanin, “On the locality of codeword symbols,” *IEEE Trans. Inf. Theory*, vol. 58, no. 11, Nov. 2012.
- [14] A. Mazumdar, V. Chandar, and G. W. Wornell, “Local recovery properties of capacity achieving codes,” in *Proc. Inf. Theory Appl. Workshop (ITA)*, pp. 1-3, San Diego, CA, Feb. 2013.
- [15] C. Huang, M. Chen, and J. Li, “Pyramid codes: Flexible schemes to trade space for access efficiency in reliable data storage systems,” *ACM Trans. Storage*, vol. 9, no. 1, Article 3, Mar. 2013.
- [16] M. Asteris and A. G. Dimakis, “Repairable fountain codes,” in *Proc. IEEE Int. Symp. Inf. Theory (ISIT '12)*, vol. 32, no. 5, pp. 1037-1047, May 2014.
- [17] Y. Wu and A. G. Dimakis, “Reducing repair traffic for erasure coding-based storage via interference alignment,” in *Proc. 2009 IEEE Int. Symp. Info. Theory (ISIT '09)*, pp. 2276-2280, Seoul, Jun.-Jul. 2009.
- [18] V. R. Cadambe, S. A. Jafar, H. Maleki, K. Ramchandran, and C. Suh, “Asymptotic interference alignment for optimal repair of MDS codes in distributed storage,” *IEEE Trans. Inf. Theory*, vol. 59, no. 5, May 2013.
- [19] M. Luby, “LT-codes,” in *Proc. 43rd Annu. IEEE Symp. Foundations of Comput. Sci. (FOCS)*, Vancouver, BC, Canada, Nov. 2002.

박진수 (Jin Soo Park)



2009년 2월 : 연세대학교 전기
전자공학과 졸업
2009년 3월~현재 : 연세대학교
전기전자공학과 석박사 통합
과정
<관심분야> 통신공학, 부호이
론, 분산저장시스템, 전자전
통신

박기현 (Ki-Hyeon Park)



2007년 2월 : 연세대학교 전
기전자공학과 졸업
2009년 2월 : 연세대학교 전
기전자공학과 석사
2009년 3월~현재 : 연세대학교
전기전자공학과 박사과정
<관심분야> 통신공학, 정보이
론, 암호이론, 이산수학

김정현 (Jung-Hyun Kim)



2006년 8월 : 연세대학교 전기
전자공학과 졸업
2008년 8월 : 연세대학교 전기
전자공학과 석사
2010년 7월~2013년 2월 : 한국
전자통신연구원 연구원
2013년 3월~현재 : 연세대학교
전기전자공학과 박사과정

<관심분야> 통신공학, 부호이론, 전자전 통신

송홍엽 (Hong-Yeop Song)



1984년 2월 : 연세대학교 전자
공학과 졸업
1986년 5월 : University of
Southern California Dept. of
EE. Systems 석사
1991년 12월 : University of
Southern California Dept. of
EE. Systems 박사

1992년 1월~1993년 12월 : Post-Doc Research Associate,
University of Southern California Dept. of EE.
Systems

1994년 1월~1995년 8월 : Senior Engineer, Qualcomm
Inc., San Diego, California.

2002년 3월~2003년 2월 : Visiting Professor, University
of Waterloo, Canada

1995년 9월~현재 : 연세대학교 전기전자공학과 교수
<관심분야> 통신공학, 정보이론, 부호이론, 암호이론,
이산수학