완전 부호의 부분접속수 분석

남미영, 김정현, 송홍엽
연세대학교, 채널 부호 및 암호 연구실

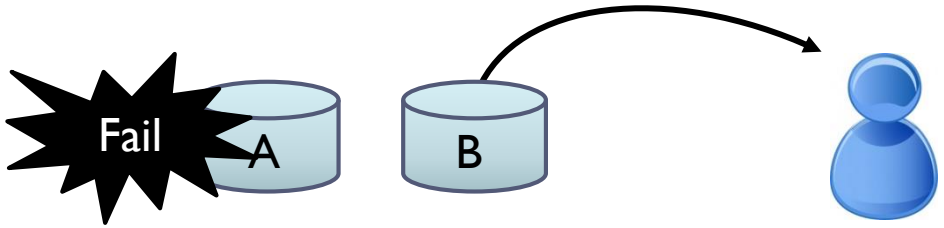
2015 / 01 / 22
2015 한국통신학회 동계종합학술발표회

Abstract

- 분산저장시스템에 부호 적용의 필요성
- 분산저장시스템에 적용될 부호의 요구조건
  - 낮은 부분접속수
- 완전부호의 부분접속수 분석
  - 해밍부호
  - 골레이부호

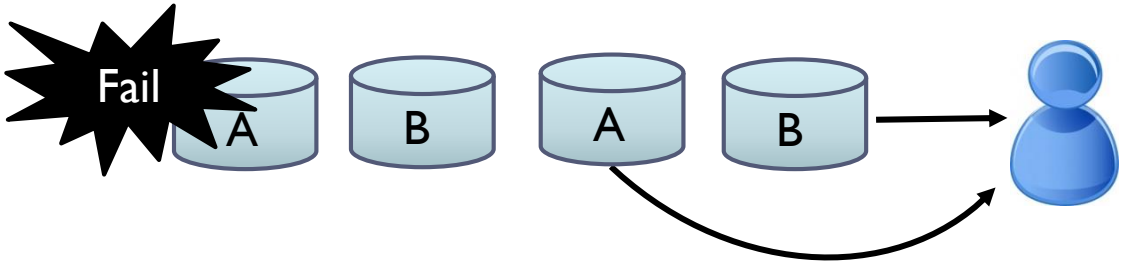
Distributed Storage Systems

- 저장장치의 불안정성으로 인한 데이터 소실



사용자가 A 블록을 복구할 방법이 없다

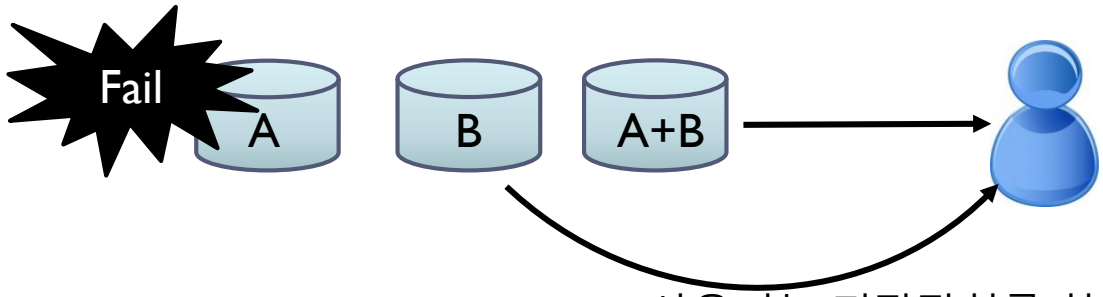
- 데이터 소실을 막기 위한 방법



사용자가 A, B 두 개의 블록을 모두 복구할 수 있다

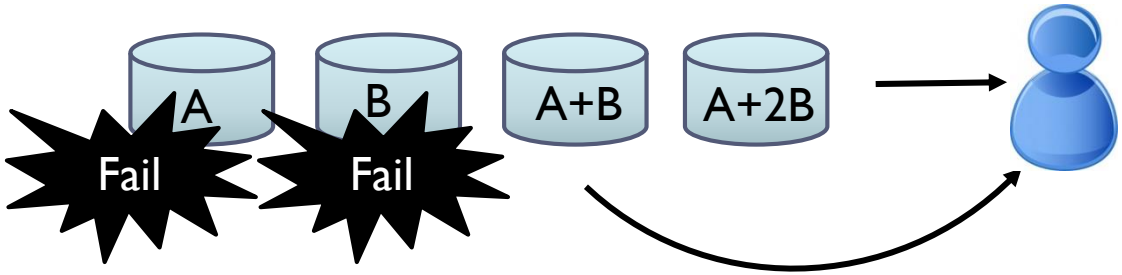
Codes for Distributed Storage

- MDS 부호의 적용
  - (3,2) MDS 부호



사용되는 저장장치를 한 개 줄이면서
시스템의 안정성은 그대로 유지할 수 있다

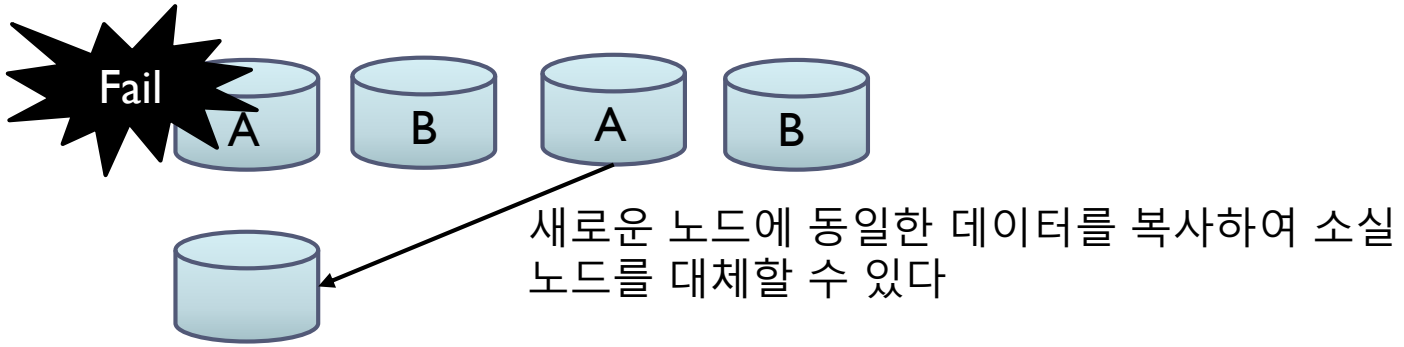
  - (4,2) MDS 부호



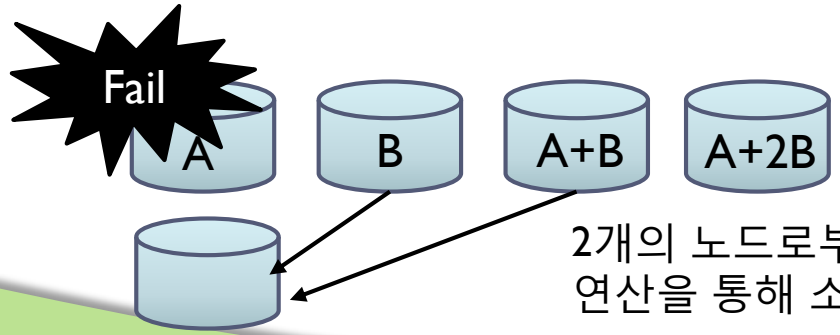
2개의 노드에 데이터 소실이 발생해도
사용자가 A와 B를 모두 복구할 수 있다

The Repair Problem

- 소실 노드의 복구
  - 시스템의 안정성을 일정하게 유지하기 위해 소실된 노드를 효과적으로 복구할 수 있어야 한다.
  - 반복부호에서의 소실 노드 복구



  - MDS 부호에서의 소실 노드 복구

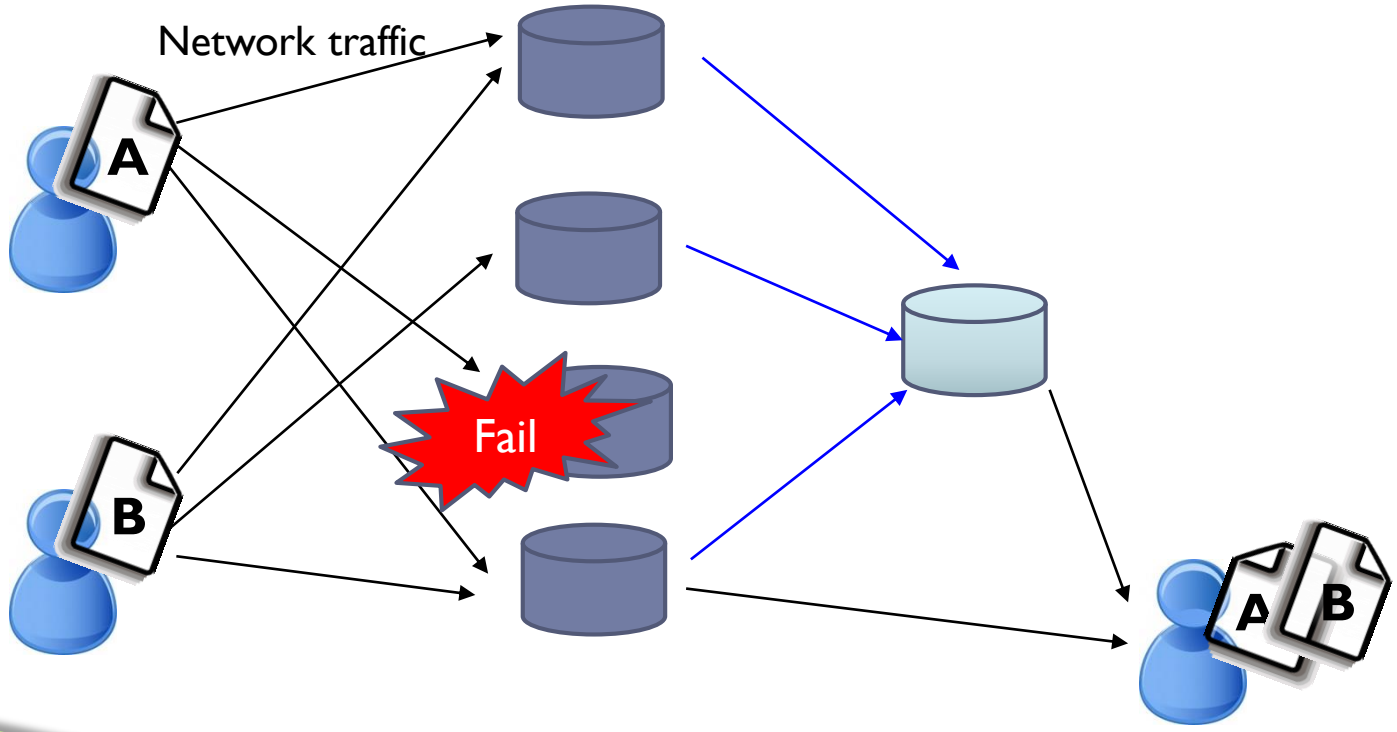


New Open Problems

- 시스템에 최적인 부호??
  - 통신량을 최소화하는 부호

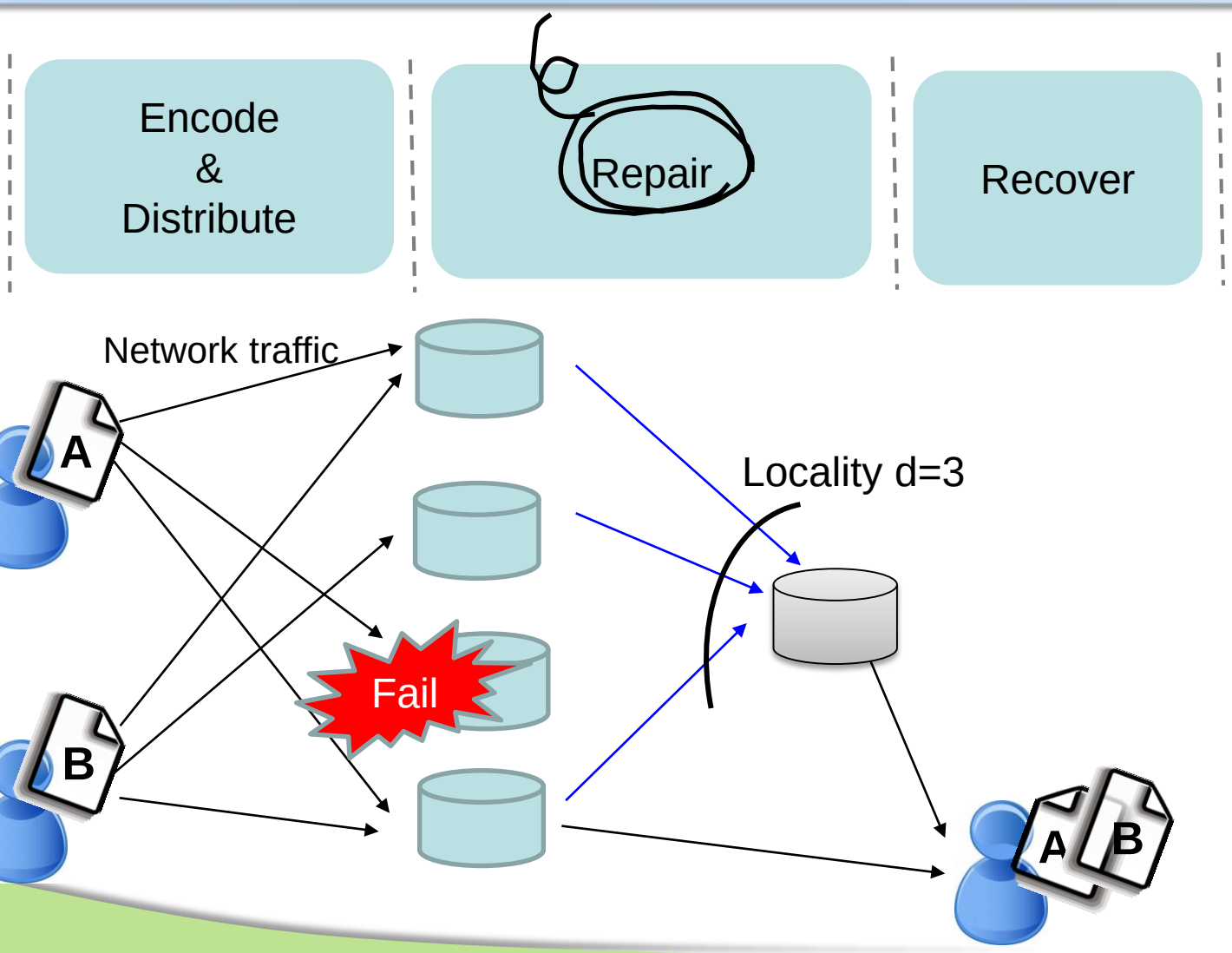
    복구시 필요한 통신량을 최소화하는 부호 (Repair cost)

    저장장치의 사용을 최소화 하는 부호



Performance Metric

We focus on the number of nodes accessed during a repair



Locality(부분접속수) of Linear Codes

- **Definition.** [3]

$C \in F_q^n$는 선형 $[n, k, d]_q$ 부호라 하자. 임의의 정수 $i \in \{1, \dots, n\}$에 대해서, C의 한 부호어의 i번째 심볼 Y_i가 다음을 만족한다고 하자.

$$Y_i = \sum_{j \in R(i)} \lambda_j Y_j$$

여기서 $\lambda_j \in F_q$이다.

위 식을 만족하는 $R(i)$는 심볼 Y_i의 복구집합(repair set)이라고 한다. 한 심볼은 서로 다른 복구집합을 여러 개 가질 수 있다. 한 심볼의 복구집합 중 가장 작은 크기를 갖는 복구집합의 크기 r을 해당 심볼의 부분접속수(locality)라고 한다. C에 속하는 모든 n개의 심볼의 부분접속수 중 가장 큰 값을 부호 C의 부분접속수라고 한다.

[3] P. Gopalan, C. Huang, H. Simitci, and S. Yekhanin, “On the locality of codeword symbols,” IEEE Trans. Inform. Theory, vol. 58, no. 11, pp. 6925-6934, Nov. 2012.

Locality(부분접속수) of Linear Codes

- **Remark.** [5]

  C가 $[n, k, d]$ 선형 부호라 하자. C의 쌍대부호 $C^{\perp}$의 부호어 중 i번째 위치가 0이 아닌 모든 부호어 중 최소의 무게를 갖는 부호어의 무게가 $r+1$이면 부호 C의 i번째 부호의 부분접속수는 r이다.

- Example: (4,2) MDS 부호

$$G = \begin{bmatrix} 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 2 \end{bmatrix} \rightarrow H = \begin{bmatrix} 2 & 2 & 1 & 0 \\ 2 & 1 & 0 & 1 \end{bmatrix}$$

$$C^{\perp} = \{(0000), (2210), (2101), (1120), (1202), (1011), (0112), (0221), (2022)\}$$

$$r = \min_{j \in \{2,3,4,5,6,9\}} \left| supp\left(c_j^{\perp}\right) \right| - 1 = 2$$

[5] P. Gopalan, C. Huang, B. Jenkins and S. Yekhanin, "Explicit maximally recoverable codes with locality," IEEE Trans. Inform. Theory, vol. 60, no. 9, pp. 5245-5256, Jun. 2014.

Locality of Some Optimal Codes

- 그리스머 바운드를 만족하는 최적부호 [2]
  - $[2^k - 1, k, 2^{k-1}]$ 심플렉스 부호:
    k에 관계 없이 항상 부분접속수 2를 갖는다.
  - $[2^k - 1 + k, k, 2^{k-1} + 1]$ IR-심플렉스 부호:
    k에 관계 없이 항상 부분접속수 2를 갖는다.
- 다른 종류의 최적부호의 부분접속수??
  - 완전부호
    : 해밍바운드 측면에서 최적인 부호

[2] 김정현,남미영,박기현,송홍엽,"IR-심플렉스 부호 설계 및 부분접속수 분석", 2015 한국통신학회 동계학술대회

Locality of Hamming Codes

- **Theorem.**

  임의의 정수 $m \geq 2$에 대해 $[2^m - 1, 2^m - m - 1, 3]$ 해밍부호 C의 부분접속수는 $2^{m-1} - 1$이다.

- Example. [7,4,3] 해밍부호

$$H = \begin{bmatrix} 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{bmatrix}$$

Locality is 4.

$$p_0 = \{1, 3, 5, 7\}$$
$$p_1 = \{2, 3, 6, 7\}$$
$$p_2 = \{4, 5, 6, 7\}$$

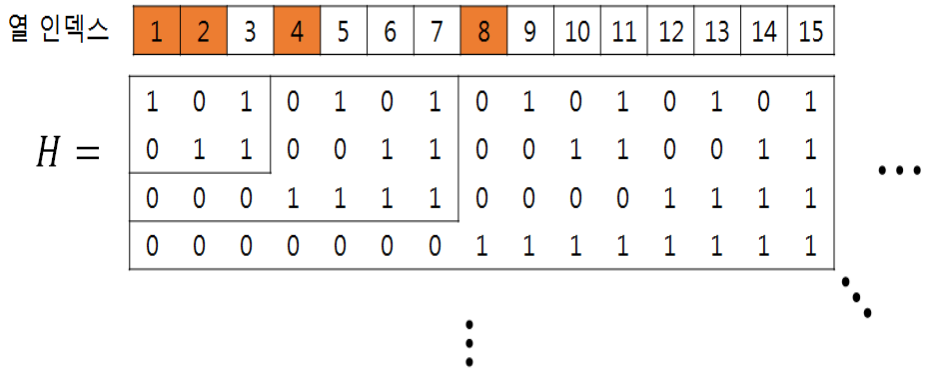
Parity-check set

$R(1) = p_0 \backslash \{1\} = \{3,5,7\}$	$R(5) = p_0 \backslash \{5\} = \{1,3,7\}$
$R(2) = p_1 \backslash \{2\} = \{3,6,7\}$	$R(6) = p_1 \backslash \{6\} = \{2,3,7\}$
$R(3) = p_0 \backslash \{3\} = \{1,5,7\}$	$R(7) = p_0 \backslash \{7\} = \{1,3,5\}$
$R(4) = p_2 \backslash \{4\} = \{5,6,7\}$	

Repair set

Repair Set of Hamming Codes

- 해밍부호의 패리티 검사 행렬의 생성과 확장



- m개의 패리티 검사식에 포함되는 인덱스의 집합 $p_0, \dots, p_{m-1}$.
  위의 패리티 검사 행렬 표현 방법을 이용해 다음과 같이 정리할 수 있다.

$$p_0 = \{1,3,5, \dots, 2^m - 1\}$$
$$p_1 = \{2,3,6,7, \dots, 10,11, \dots, 2^m - 2\}$$
$$\vdots$$
$$p_j = \left\{\left(2^j + 2^{j+1}l + 0\right), \dots, \left(2^j + 2^{j+1}l + \left(2^j - 1\right)\right)\right\}, l$$
$$= 0, \dots, 2^{m-1-j} - 1$$
$$\vdots$$

- 심볼 i의 복구집합

$$R(i) = p_j \backslash \{i\} \; for \; \forall p_j \; containing \; i$$

Locality of Golay Codes

- $[23, 12, 7]$ 이진 골레이 부호

$$G = \left[I_{12} \middle| \begin{matrix} 0 & 1 & \cdots & 1 \\ 1 & & & \\ \vdots & & B & \\ 1 & & & \end{matrix} \right], \quad B = \begin{bmatrix} 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \end{bmatrix}$$

- **Theorem.**

  [23,12,7] 이진 골레이 부호의 부분접속수는 7이다.

- **Corollary.**

  [24,12,8] 이진 골레이 부호의 부분접속수는 7이다.

Locality of Golay Codes

- $[11, 6, 5]$ 삼진 골레이 부호

$$G = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 2 & 2 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 2 & 2 \\ 0 & 0 & 0 & 1 & 0 & 0 & 2 & 1 & 0 & 1 & 2 \\ 0 & 0 & 0 & 0 & 1 & 0 & 2 & 2 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 2 & 2 & 1 & 0 \end{bmatrix}$$

- **Theorem.**

  [11,6,5] 삼진 골레이 부호의 부분접속수는 5이다.

- **Corollary.**

  [12,6,2] 삼진 골레이 부호의 부분접속수는 5이다.

Repair Set of Golay Code

$R(1) = \{2,3,5,6,7,11,14\}$	$R(13) = \{1,4,8,9,10,12,14\}$
$R(2) = \{1,3,5,6,7,11,14\}$	$R(14) = \{1,4,8,9,10,12,13\}$
$R(3) = \{1,2,5,6,7,11,14\}$	$R(15) = \{1,2,4,5,6,10,12\}$
$R(4) = \{1,8,9,10,12,13,14\}$	$R(16) = \{1,3,4,5,9,11,12\}$
$R(5) = \{1,2,4,6,7,11,14\}$	$R(17) = \{1,2,3,4,8,10,12\}$
$R(6) = \{1,2,3,5,7,11,14\}$	$R(18) = \{1,2,3,7,9,10,12\}$
$R(7) = \{1,2,3,5,6,11,14\}$	$R(19) = \{1,2,6,8,9,11,12\}$
$R(8) = \{1,4,9,10,12,13,14\}$	$R(20) = \{1,5,7,8,10,11,12\}$
$R(9) = \{1,4,8,10,12,13,14\}$	$R(21) = \{1,4,6,7,9,10,11\}$
$R(10) = \{1,4,8,9,12,13,14\}$	$R(22) = \{1,3,5,6,8,9,10\}$
$R(11) = \{1,2,3,5,6,7,14\}$	$R(23) = \{1,2,4,5,7,8,9\}$
$R(12) = \{1,4,8,9,10,13,14\}$	

[23,12,7] 골레이부호의 복구집합

$R(1) = \{3,4,5,6,7\}$	$R(7) = \{1,3,4,5,6\}$
$R(2) = \{1,4,5,6,8\}$	$R(8) = \{1,2,4,5,6\}$
$R(3) = \{1,4,5,6,7\}$	$R(9) = \{1,2,3,5,6\}$
$R(4) = \{1,3,5,6,7\}$	$R(10) = \{1,2,3,4,6\}$
$R(5) = \{1,3,4,6,7\}$	$R(11) = \{1,2,3,4,5\}$
$R(6) = \{1,3,4,5,7\}$	

[11,6,5] 골레이부호의 복구집합

Conclusion

- 해밍부호는 동일한 최소거리를 갖는 반복부호에 비해 저장공간 측면에서 효율적인 대신 더 큰 부분접속수를 갖는다.
- 해밍부호의 최소거리가 증가할수록 부분접속수 역시 증가한다.
- 골레이부호는 반복부호에 비해 최소거리가 증가하고 저장공간 효율성이 높아지고 부분접속수는 증가한다.

	$[9,3,3]_2$ *Repetition code*	$[7,4,3]_2$ *Hamming code*	$[15,11,3]_2$ *Hamming code*	$[23,12,7]_2$ *Golay code*
Storage overhead	3	1.75	1.36	1.92
Fault tolerance	2	2	2	6
Locality	1	3	7	7

<부호간 성능척도 비교>