

PBCA의 상태전이행렬을 이용한 이중 오류정정부호의 설계[†]

조성진*, 허성훈**, 김석태***

*부경대학교 수리과학부

**부경대학교 정보보호학과

***부경대학교 전자컴퓨터정보통신공학부

e-mail:sjcho@pknu.ac.kr

Design of Double-Error Correcting Code using the Transition Matrix of PBCA

Sung-Jin Cho*, Seong-Hun Heo**, Seok-Tae Kim***

*Division of Mathematical Sciences, Pukyong National Univ.

**Dept. of Information Security, Pukyong National Univ.

***Dept. of Telematics Engineering, Pukyong National Univ.

요 약

초고속 유무선 통신망이 확대됨에 따라 디지털 데이터 통신 및 저장 시스템 설계 시 통신채널에서 발생하는 오류를 효율적으로 제어하기 위한 오류정정부호 장치가 중요한 요소가 되었다. 본 논문에서는 기존의 셀룰라 오토마타 기반의 오류정정부호를 개선시킨 이중 오류정정부호를 설계하는 방법을 제안한다.

1. 서론

많은 양의 데이터가 다양한 컴퓨터 시스템과 서버 시스템 사이에서 디지털 논리 회로와 상호 연결선을 통하여 전송된다. 시스템의 신뢰성은 회로 모듈 사이에서 데이터 전송의 무오류성(error-free)에 의해서 좌우된다[1]. 하지만 전자적 잡음, 장치 결함, 시간 오류 등으로 인하여 시스템에는 항상 언제 발생할지 모르는 잠재적 오류가 존재한다[1][2]. 따라서 시스템의 신뢰성을 향상시키기 위해서 오류정정부호 장치가 필수불가결한 요소가 되었다. 일반적으로 오류정정부호는 (n, k, d) 부호로 나타내며, n 은 부호어의 길이, k 는 정보어의 길이, d 는 최소거리를 나타낸다[2][3]. 기존의 오류정정부호는 k 값이 커짐에 따라 부호화 및 복호화 회로가 복잡해지는 단점이 있다. 본 논문에서는 셀룰라 오토마타(Cellular Automata, 이하 CA)의 간단하고, 규칙적이며, 모듈

화한 특성을 이용하여 특히, PBCA의 상태전이행렬을 이용하여 이중 오류정정부호를 구성하는 방법을 제안한다.

2. 셀룰라 오토마타

CA는 동역학계(dynamical system)를 해석하는 하나의 방법으로 공간과 시간을 이산적으로 다루는 시스템이며, 셀룰라 공간(cellular space)의 기본 단위인 각 셀(cell)이 취할 수 있는 상태를 유한하게 처리하며, 각 셀들의 상태가 국소적인 상호작용에 의해서 동시에 갱신되는 시스템이다. 1차원 CA에서는 모든 셀들이 선형으로 배열되어 있고 국소적 상호작용이 세 개의 셀, 즉 자신과 인접한 두 셀에 의해 이루어지는 CA를 3-이웃(3-neighborhood) CA라 한다. 본 논문에서 다루는 CA는 3-이웃 1차원 CA에 국한시킨다.

CA를 설명하기 위해서 다음 기호들이 사용된다.

[†] 본 연구는 정보통신부 2003 기초기술연구(IITA:03-기초-0047)지원으로 연구되었음.

- i : 일차원으로 배열되어 있는 각 셀들의 위치
- t : 시간 단계
- $q_i(t)$: 시간 t 에서 i 번째 셀의 상태
- $q_i(t+1)$: 시간 $t+1$ 에서 i 번째 셀의 상태

세 개의 이웃을 가지는 CA에 대한 다음상태 전이함수(transition function)는 다음과 같이 나타낸다.

$$q_i(t+1) = f[q_{i-1}(t), q_i(t), q_{i+1}(t)]$$

여기서 f 는 결합논리를 가지는 국소 전이함수이다. f 는 3개의 변수를 가지는 Boolean 함수이므로 2^3 , 즉 256개의 다음 상태 전이함수들이 있으며 이것을 CA의 rule이라고 한다.

CA의 셀들의 상태를 0과 1의 두 가지 값으로 다루고 주어진 CA의 다음상태 전이함수를 아래와 같이 표현한다. 여기서 첫 행은 시간 t 에서 인접한 세 개의 셀들의 가능한 8가지 상태의 배열이고 다음 행들은 시간 $t+1$ 에서 i 번째 셀의 갱신된 상태이다.

CA는 적용되는 rule의 논리에 따라 선형 CA(Linear CA), 가산 CA(Additive CA), 비가산 CA(Nonadditive CA)로 분류되는데 각 셀에 적용된 rule이 XOR 논리로만 이루어진 CA는 선형 CA이다. 선형 CA의 상태 전이 함수는 행렬로 표현 될 수 있고 이 행렬을 특성행렬이라 한다. 또한 셀에 적용되는 rule이 XNOR과 XOR논리로 이루어진 CA를 여원 CA(Complemented CA)라 하고 선형 CA와 여원 CA를 가산 CA라 한다. 셀들의 rule이 AND-OR논리로 이루어진 CA를 비가산 CA라 한다. 표 1은 가산 CA에서 선형 rule(90, 102, 150)을 보여준다.

rule 90	$q_i(t+1) = q_{i-1}(t) \oplus q_{i+1}(t)$
rule 102	$q_i(t+1) = q_i(t) \oplus q_{i+1}(t)$
rule 150	$q_i(t+1) = q_{i-1}(t) \oplus q_i(t) \oplus q_{i+1}(t)$

표 1. CA의 선형 rule

CA의 각 셀에 모두 같은 rule이 적용된 CA를 uniform CA, 그렇지 않고 2가지 이상의 서

로 다른 rule이 적용된 CA를 hybrid CA라 한다. CA의 rule에 의해 변화되는 상태를 나타낸 상태전이 그래프의 형태에 따라 Group CA와 Nongroup CA로 분류할 수 있다. Group CA는 모든 셀들의 상태가 몇 개의 사이클을 이루며 반복되는 CA로 임의의 한 상태에 대한 이전상태가 유일하다. 이와 달리 Nongroup CA는 상태전이 그래프가 트리 구조를 이루고 있으며 상태전이 함수에 의해 얻어질 수 있는 상태인 도달 가능한 상태와 상태전이 함수에 의해 나타날 수 없는 도달 불가능한 상태로 나누어진다. CA에서 가장 왼쪽과 오른쪽의 셀은 2개의 이웃만을 가지므로 세 번째 이웃의 상태를 결정해 주어야 한다. 이것을 CA의 경계조건이라고 하고 일반적으로 다음 세 가지의 경계조건을 이용한다. 제일 왼쪽과 오른쪽의 셀들이 0상태에 연결되어 있는 NBCA (Null Boundary CA), 양끝의 셀들이 서로 연결되어 있는 PBCA (Periodic Boundary CA), 가장 왼쪽(오른쪽) 셀의 다음 상태가 그 자신과 그것의 오른쪽(왼쪽) 이웃, 두 번째 오른쪽(왼쪽) 이웃 셀의 상태에 의존하는 IBCA(Intermediate Boundary CA)이다[5][6][7].

3. 제안된 이중 오류정정부호

(1) 부호화 방법

CA의 상태전이행렬을 이용하여 이중 오류정정부호를 생성하는 방법을 알아본다.

[정리 1] [2] 선형 (n, k) 부호 C 의 최소거리가 d 이기 위한 필요충분조건은 C 의 패리티 검사행렬 H 에서 서로 다른 $d-1$ 개 이하의 열벡터가 독립이다.

[따름정리 1] 행렬 T 가 다음 두 조건을 만족한다면, CA는 $(n, k, 5)$ 부호를 생성한다:

- (a) T 의 모든 열벡터는 적어도 네 개의 1을 포함한다.
- (b) T 의 서로 다른 네 개 이하의 열벡터의 합은 영벡터가 아니다.

다음은 최소거리가 5인 부호를 생성하기 위하여 따름정리 1을 만족하는 T 를 구성하는 알고리즘이다.

알고리즘 I. T구성 알고리즘

Step 1. rule이 <102, 102, ..., 102, 150>인 PBCA의 특성행렬 T_k 을 구성한다.

Step 2. T_k^3 을 구한다.

Step 3. T_k^3 의 (1, k)성분을 1로 바꾸고, 행렬 $\begin{bmatrix} 101010\dots \\ 010101\dots \end{bmatrix}$ 을 덧붙인다.

Step 4. Step 3에서 구한 행렬을 T_k 라 하면 같은 위치에서 1을 갖지 않는 행끼리 더하여 [따름정리 1]을 만족하는지 검사하여 행의 수를 최대한으로 줄인다.

Step 5. Step 4에서 구한 행렬을 T 로 둔다.

	Chowdhury [4]	본 연구
메시지 (k)	검사비트(n-k)	검사비트(n-k)
7	9	8
8	9	8
9	9	9
10	10	9
11	10	9
12	11	9
13	11	10
14	11	10
15	11	10
16	12	11
17	13	11
18	13	12
19	13	12
20	13	12
32	17	16

표 2. 최소거리가 5인 부호의 검사비트

알고리즘 I에 의해서 구성된 T 를 이용하여 부호어를 생성한다. k 비트의 메시지 (M)가 로드(load) 되면 T 를 이용하여 검사비트를 생성하고 메시지에 검사비트 (CB)를 연결하여 부호어 (C)를 생성한다.

표 2는 최소거리가 5인 부호에서 메시지에 대한 검사비트의 크기를 Chowdhury, et, al.[4]의 결과와 본 연구를 비교한 것이다. 표에 의해서 알고리즘 I에 의해서 구성된 T 를 이용한 방법이 1비트 또는 2비트가 더 줄어들음을 알 수 있다.

(2) 복호화 방법

수신된 부호어에서 오류가 있는 비트를 정정하기 위한 복호화 방법을 알아본다. 복호화하는 첫 번째 단계는 신드롬을 계산하는 것이다.

수신된 부호어가 $C' (= (M'|CB'))$ 이라면 신드롬은 $S(C') = H C'^t = H [M'|CB']^t$ 이다. 여기서, H 는 알고리즘 I에 의해서 생성된 T 와 단위행렬 I_k 를 연결하여 형성된 패리티 검사행렬이다. 즉, $H = [T|I_k]$ 이다.

따라서,

$$S(C') = H [M'|CB']^t = [T|I_k] [M'|CB']^t = T M'^t \oplus I_k CB'^t = T M'^t \oplus CB'^t$$

로 표현될 수 있다.

신드롬의 값이 0이면 오류가 없음을 나타내고 0이 아니면 수신된 부호어에 오류가 있음을 나타낸다.

수신된 부호어에서 M_e 과 CB_e 를 각각 메시지와 검사비트에 대응되는 오류 벡터라고 하자. 그러면, 수신된 메시지는 $M' = M \oplus M_e$ 이고, 수신된 검사비트는 $CB' = CB \oplus CB_e$ 이다.

$$H [M|CB]^t \oplus H [M_e|CB_e]^t = S$$

여기서, $H [M|CB]^t = 0$ 이므로, $H [M_e|CB_e]^t = S$ 이 된다.

즉, $H E^t = S$ 에서 오류벡터 E 는

$E^t = H^{-1} S$ 이 된다. 하지만 H^{-1} 이 존재하려면 H 는 $n \times n$ 정방행렬이어야 한다. 하지만, H 가 정방행렬이 아니므로 패리티 검사 행렬 $H_{(n-k) \times n}$ 를 정방행렬 $T_{aug} n \times n$ 로 변환하기 위해서 k 개의 추가 행을 $H_{(n-k) \times n}$ 에 덧붙인다. 이 때 $T_{aug} n \times n$ 가 역행렬이 존재(즉, group CA)하도록 ($\det [T_{aug}] = 1$) 구성한다.

$$T_{aug} = \begin{bmatrix} H_{(n-k) \times n} \\ \dots \\ [추가행]_{k \times n} \end{bmatrix}_{n \times n}$$

$$T_{aug} E^t = \begin{bmatrix} H \\ \dots \\ [추가행] \end{bmatrix}_{n \times n} [M_e|CB_e]^t = \begin{bmatrix} S \\ \dots \\ S_{aug} \end{bmatrix}_{n \times 1}$$

$$E^t = T_{aug}^{-1} \begin{bmatrix} S \\ S_{aug} \end{bmatrix}$$

<정의1> 복호화에 사용되는 T_{aug} 는 <알고리즘 I>에 의해서 구성된 T 를 이용하여 다음과 같이 구성한다.

$$T_{aug} = \begin{bmatrix} T_{(n-k) \times k} & I_{n-k} \\ I_k & O_{k \times (n-k)} \end{bmatrix}_{n \times n}$$

[정리 2] 정의 1에서 구성된 T_{aug} 의 역행렬은 다음과 같다.

$$T_{aug}^{-1} = \begin{bmatrix} O_{k \times (n-k)} & I_k \\ I_{n-k} & T_{(n-k) \times k} \end{bmatrix}_{n \times n}$$

다음은 이중 오류정정부호의 복호화 알고리즘이다.

알고리즘 II. 복호화 알고리즘

- Step 1. 신드롬 S 를 구한다.
- Step 2. T_{aug} 와 T_{aug}^{-1} 를 구한다.
- Step 3. PLA를 이용하여 S_{aug} 를 대응시킨다.
- Step 4. $E^t = T_{aug}^{-1} \begin{bmatrix} S \\ S_{aug} \end{bmatrix}$ 에 의해서 오류벡터 E 를 구한다.
- Step 5. $C = C' \oplus E$ 에 의해서 부호어를 복호한다.

4. 결론

본 논문에서는 PBCA의 상태전이행렬을 이용하여 이중 오류를 정정하기 위하여 최소거리가 5인 부호를 생성하는 방법 및 복호화 방법을 제안하였다. 본 연구에서 제안된 방법은 부호화에 필요한 행렬을 더욱 효율적으로 구성하였고, 검사비트의 수를 기존연구보다 줄임으로써 전송되는 부호어의 길이를 감소시켰다. 또한 복호화에 쓰이는 행렬 T_{aug} 를 쉽게 구성하는 방법을 제안하였다.

참고문헌

- [1] P. P. Chaudhuri, et. al., "Additive cellular automata theory and applications", Vol. 1,

IEEE Computer Society Press, California, USA, 1997.

[2] T.R.N. Rao and E. Fujiwara, "Error-Control Coding for Computer System", Prentice Hall, Englewood Cliffs, N.J., 1989.

[3] S. Lin and D.J. Costello, "Error Control Coding: Fundamentals and Applications", Prentice Hall, Englewood Cliffs, N.J., 1983.

[4] D.R. Chowdhury, S. Basu, I.S. Gupta and P.P. Chaudhuri, "Design of CAECC - Cellular Automata Based Error Correcting Code", IEEE Trans. Computers, Vol. 43, June 1994, pp. 759-764.

[5] S.J. Cho, U.S. Choi and H.D. Kim, "Linear nongroup one-dimensional cellular automata characterization on GF(2)", J. Korea Multimedia Soc., Vol. 4, No. 1, 2001, pp. 91-95.

[6] S.J. Cho, H.D. Kim and U.S. Choi, "Behavior of Complemented CA Whose Complement Vector is Acyclic in a Linear TPMACA", Mathematical and Computer Modelling, July, 2002, pp. 980-986.

[7] S.J. Cho, H.D. Kim and U.S. Choi, "Analysis of complemented CA derived from a linear TPMACA", Computers & Mathematics with Applications, Vol. 45, 2003, pp. 689-698.