

구조적 LDPC 부호의 효율적인 설계

준회원 정 비 응*, 정회원 김 준 성*, 종신회원 송 홍 엽*

Efficient Design of Structured LDPC Codes

Bi-Woong Chung* *Associate Member*, Joon-Sung Kim* *Reguler Member*,
Hong-Yeop Song* *Lifelong Members*

요 약

LDPC 부호의 높은 부호화 복잡도는 구조적인 패리티 검사 행렬의 설계로 해결할 수 있다. 패리티 검사 행렬을 같은 유형의 블록으로 구성한다면 복호화기의 구현이 간단해지고 구조적 복호화가 가능하며 LDPC 부호를 저장하는데 필요한 메모리를 줄일 수 있는 장점이 있다. 본 논문에서는 부행렬 단위의 girth 조건과 PEG 알고리즘, 비트 노드의 connectivity를 이용하여 부행렬이 순환행렬이나 영행렬로 구성되는 짧은 길이를 갖는 구조적 LDPC 부호의 생성 알고리즘을 제안하였다. 이 알고리즘으로 생성된 부호는 구조적 제한이 없이 생성된 부호에 비하여 낮은 SNR에서는 비슷한 성능을, 높은 SNR에서는 더 좋은 성능을 내는 것을 모의 실험을 통해 확인하였다.

Key Words : LDPC codes, circulant permutation matrix, girth, Progressive Edge Growth(PEG), Approximate Cycle EMD(ACE)

ABSTRACT

The high encoding complexity of LDPC codes can be solved by designing structured parity-check matrix. If the parity-check matrix of LDPC codes is composed of same type of blocks, decoder implementation can be simple, this structure allow structured decoding and required memory for storing the parity-check matrix can be reduced largely. In this paper, we propose a construction algorithm for short block length structured LDPC codes based on girth condition, PEG algorithm and variable node connectivity. The code designed by this algorithm shows similar performance to other codes without structured constraint in low SNR and better performance in high SNR than those by simulation

I. 서 론

LDPC 부호(Low Density Parity Check codes)는 1962년 Gallager에 의해 처음 제안되었지만 Shannon limit에 근접하는 뛰어난 성능에도 불구하고 하드웨어 기술력의 제한으로 오랫동안 잊혀져 오다가 1995년 Mackay와 Neal의 재발견 이후 차세대 통신 시스템에 쓰일 부호로 각광받으며 활발한 연구가 이루어지고 있다^{1, 2)}.

LDPC 부호는 생성방법에 따라 크게 랜덤하게

생성된 부호와 블록단위의 구조적 부호로 구분할 수 있다. 구조적 부호는 랜덤부호와 달리 대수학적이거나 기하학적인 방법으로 생성할 수 있다. IEEE 802.16e 시스템에 적합한 채널 부호화 방식으로 연구되고 있는 LDPC 부호들도 대부분 패리티 검사 행렬의 열무게와 행무게가 1인 부행렬 또는 영행렬로 구성된 부호이며 Richardson의 부호화 기법이나 Repeat-Accumulate 구조를 이용한 부호화 기법을 사용하는 구조적 LDPC 부호이다³⁾⁴⁾. 이렇게 구조적인 방법으로 생성된 LDPC 부호는 부호화 과정의

*본 연구는 삼성종합기술원의 "4G wireless system의 연구 개발" 과제의 지원에 의해 이루어졌음.

* 연세대학교 전기전자공학과 부호 및 정보이론 연구실({bw.chung, js.kim, hy.song}@coding.yonsei.ac.kr)

논문번호 : KICS2005-08-344, 접수일자 : 2005년 8월 22일

복잡도를 줄일 수 있을 뿐만 아니라 하나의 행렬로부터 부호율 조정이나 부행렬 길이의 변화가 가능하다는 장점을 가지고 있다. 또한 실제 구현시 랜덤 부호에 비해 단순하며 메모리 사용면에서도 유리하다.

본 논문에서는 부 행렬이 단위 행렬의 순환 행렬로 구성되는 구조적 LDPC 부호의 패리티 검사 행렬을 효과적으로 생성하는 기법을 제안한다. 일반적으로 사용되는 짧은 길이의 균일, 비균일 LDPC 부호를 생성하고 같은 파라미터를 갖고 구조적 제한이 없는 랜덤 부호화의 성능 비교를 통해 제안하고 있는 구조적 LDPC 부호 설계 기법의 효율성을 보인다.

II. 패리티 검사 행렬 구조

2.1 QC-LDPC 부호

본 논문에서 생성하고자 하는 구조적 LDPC 부호는 패리티 검사 행렬이 단위 행렬의 순환 행렬이나 영행렬인 작은 길이의 부행렬로 구성되므로 Quasi-cyclic LDPC 부호의 일종이다. X 를 $N_s \times N_s$ 단위 행렬을 오른쪽으로 한번 cyclic shift한 순환 행렬이라 정하겠다.

$$X = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \\ 1 & 0 & 0 & \cdots & 0 \end{bmatrix} \quad (2-1)$$

X^i 는 단위 행렬을 i 번 cyclic shift한 순환 행렬이고 X^∞ 는 영행렬을 나타낸다. 이 표시 방법을 사용하면 LDPC 부호의 패리티 검사 행렬 H 를 다음과 같이 정의할 수 있다.

$$H = \begin{bmatrix} X^{s_{0,0}} & X^{s_{0,1}} & \cdots & X^{s_{0,(n-1)}} \\ X^{s_{1,0}} & X^{s_{1,1}} & \cdots & X^{s_{1,(n-1)}} \\ \vdots & \vdots & \ddots & \vdots \\ X^{s_{(m-1),0}} & X^{s_{(m-1),1}} & \cdots & X^{s_{(m-1),(n-1)}} \end{bmatrix} \quad (2-2)$$

여기서 $s_{i,j}$ 는 i 행 j 열에 해당하는 부행렬의 shift-value를 나타내고 $s_{i,j} \in \{0, 1, \dots, N_s - 1, \infty\}$ 이다. H 는 $m \times n$ 개의 부행렬로 이루어지기 때문에 $mN_s \times nN_s$ 의 크기를 가지며 부호어의 길이는 nN_s , 부호율은 full-rank를 가정하면 다음과 같다.

$$R = \frac{nN_s - mN_s}{nN_s} = 1 - \frac{m}{n} \quad (2-3)$$

2.2 효과적인 부호화 알고리즘

패리티 검사 행렬을 다음과 같이 6부분으로 나누어 설계하면 Richardson의 효율적인 부호화 기법을 사용할 수 있다.

$$H = \begin{bmatrix} A & B & T \\ C & D & E \end{bmatrix} \quad (2-4)$$

여기서 $\phi = -ET^{-1}B + D$ 로 정의하면 다음의 수식을 이용하여 패리티 비트를 얻을 수 있다.

$$\begin{aligned} p_1^T &= -\phi^{-1}(-ET^{-1}A + C)u^T \\ p_2^T &= -T^{-1}(Au^T + Bp_1^T) \end{aligned} \quad (2-5)$$

더 간단한 부호화를 위해 B, T, D, E 를 $\phi = I$ 가 되도록 조정하면 그림 1과 같은 부호화기를 사용할 수 있다.

III. 구조적 LDPC 부호 생성 알고리즘

본 논문에서 제안하고 있는 구조적 LDPC 부호 생성 알고리즘은 임의의 파라미터 m 과 n , 그리고 부행렬의 크기 N_s 에 대하여 주어진 무게 분포에 따라 성능이 좋은 $(mN_s) \times (nN_s)$ 패리티 검사 행렬 H 를 만들 수 있다. 부행렬 단위의 girth 조건과 비트 노드의 connectivity를 기준으로 하고 무게가 작은 비트 노드끼리의 짧은 cycle을 피하기 위해 비트 노드의 무게가 작은 부행렬부터 생성을 한다.

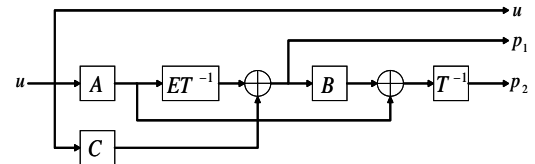


그림 1. Richardson의 부호화기 구조

실제 부행렬이 들어갈 위치는 현재의 그래프상에서 연결할 최선의 edge를 결정하는 Progressive Edge Growth(PEG) 알고리즘을 사용하고 주어진 비트 노드의 무게 분포로부터 검사 노드 분포의 균형을 맞추는 역할을 한다^{[5][6]}.

비균일 LDPC 부호의 경우 cycle 외에 connectivity라는 조건이 사용된다. 어떤 비트 노드의 집합에 연결된 검사 노드 중 이 집합에 하나의 edge로만 연결된 검사 노드의 수는 extrinsic message degree(EMD)로 정의 된다. EMD를 쉽게 구하기 위해

한 cycle안에 포함된 비트 노드의 무게를 다음과 같이 더한 값이 approximate cycle EMD(ACE)이다⁷⁾.

$$\sum_i (d_i - 2) \quad (3-1)$$

여기서는 어떤 비트 노드에 걸린 cycle안에 포함된 비트 노드 무게의 합을 이 노드의 connectivity라고 정하겠다.

부호 생성 과정은 다음과 같다.

- 1) 원하는 부호율의 최적화된 무게 분포를 density evolution을 이용하여 얻고 m 과 n 의 크기에 따라 수정한다⁸⁾.
- 2) 비균일 LDPC 부호의 경우 효과적인 부호화가 가능한 구조로 H 의 패리티 부분을 초기화한다.
- 3) PEG 알고리즘을 이용하여 영행렬이 아닌 부행렬이 들어갈 위치를 결정한다.
- 4) 3)에서 결정된 위치 (i, j) 에 대해 shift-value를 결정한다.
- 5) 해당하는 순환행렬을 그래프 상에 추가시키고 위치 (m, n) 까지 4), 5) 과정을 반복한다.

과정 4)의 shift-value $s_{i,j}$ 를 결정하는 알고리즘은 다음과 같다. Insert $(i, j, s_{i,j})$ 는 위치 (i, j) 에 $X^{s_{i,j}}$ 를 넣어 Tanner 그래프 상에 N_s 개의 edge를 연결시키는 함수이고 Delete (i, j) 는 위치 (i, j) 에 있는 부행렬을 빼어 N_s 개의 edge를 제거시키는 함수이다. check-local-girth (x) 와 check-connectivity (x) 는 각각 현재의 그래프 상에서 x 번째 비트 노드의 cycle과 connectivity를 구하는 함수를 의미한다^{9),10)}.

```

Decide  $s_{i,j}$ 
begin
if  $i = 0$ 
 $s_{i,j} \leftarrow 0$ 
else
 $s_{i,j} - temp \leftarrow 0$ 
Insert  $(i, j, s_{i,j} - temp)$ 
 $cycle\_p \leftarrow check\_local\_girth(j \times N_s)$ 
 $ACE(0) \leftarrow check\_connectivity(j \times N_s)$ 
Delete  $(i, j)$ 
for  $k = 1$  to  $N_s - 1$ 
begin
Insert  $(i, j, k)$ 
 $cycle\_c \leftarrow check\_local\_girth(j \times N_s)$ 
 $ACE(k) \leftarrow check\_connectivity(j \times N_s)$ 
Update  $s_{i,j} - temp$ 
end
 $s_{i,j} \leftarrow s_{i,j} - temp$ 
end
    
```

모든 부행렬이 영행렬이거나 단위행렬의 순환 행렬이기 때문에 어떤 하나의 비트 노드의 cycle과 그 노드가 속한 부행렬에 포함된 다른 비트 노드들의 cycle은 같게 된다. 이 성질을 이용하여 각 블록에서 첫 번째 비트 노드만을 검사를 하여 조건에 따라 shift-value를 갱신하게 되고 조건은 다음과 같다.

```

Update  $s_{i,j} - temp$ 
begin
if  $cycle\_c > cycle\_p$ 
if  $ACE(k) \geq ACE(s_{i,j} - temp)$ 
 $cycle\_p \leftarrow cycle\_c$ 
 $s_{i,j} - temp \leftarrow k$ 
else if  $cycle\_c = cycle\_p$ 
 $cycle\_sum\_c \leftarrow \sum_{l=0}^{j-1} check\_local\_girth(l \times N_s)$ 
delete  $(i, j)$ 
Insert  $(i, j, s_{i,j} - temp)$ 
 $cycle\_sum\_p \leftarrow \sum_{l=0}^{j-1} check\_local\_girth(l \times N_s)$ 
if  $cycle\_sum\_c > cycle\_sum\_p$ 
 $s_{i,j} - temp \leftarrow k$ 
delete  $(i, j)$ 
end
    
```

현재 shift-value에 해당하는 부행렬이 그래프 상에서 추가될 경우의 cycle과 connectivity를 구하여 cycle이 크고 connectivity가 크거나 같으면 현재 값으로 갱신하고 cycle이 같으면 현재까지 생성된 그래프 상에서 전체 cycle의 합을 비교하여 갱신한다. 이 방법은 추가할 부행렬의 cycle뿐만 아니라 이미 결정된 부행렬의 cycle까지 고려하기 때문에 평균 local girth를 높일 수 있고 하나의 shift-value를 결정하기 위해 가능한 모든 값을 넣어 검사를 하지만 N_s 의 비트 노드중 하나만을 검사하기 때문에 m 과 n 의 값이 크지 않으면 알고리즘의 복잡도는 크지 않게 된다.

균일 LDPC 부호의 생성일 경우는 cycle이 크다면 connectivity도 크기 때문에 connectivity 조건은 빠지고 비균일 LDPC 부호의 경우는 cycle이 크면서 connectivity가 작은 경우를 배제하여 높은 SNR에서의 성능을 향상시킬 수 있다.

IV. 모의 실험 결과 및 고찰

실험은 가우시안(AWGN) 채널에서 SPA(sum product algorithm)으로 복호를 하였으며 최대 반복 횟수를 50번으로 제한시켰다. 본 논문에서 제안한 알고리즘으로 생성한 LDPC 부호는 Proposed LDPC

로 표현을 하였다.

4.1 균일 LDPC 부호

$m=3$, $n=6$ 이고 N_s 는 167과 334로 길이 1002와 2004의 부호를 0.5 균일 Proposed LDPC 부호와 같은 파라미터를 갖는 Mackay의 랜덤 LDPC 부호, PEG 부호를 비교하였다⁹⁾. 표 1은 cycle 분포를 나타내고 있는데 Proposed LDPC 부호의 girth가 두 길이에서 10으로 Mackay 부호보다 높고 PEG 부호와 비슷함을 알 수 있다.

표 1. 부호율 0.5 균일 LDPC 부호의 cycle 분포

length	code	4-cycle	6-cycle	8-cycle	10-cycle
1002	Mackay	0	544	457	1
	PEG	0	0	78	904
	Proposed	0	0	0	1002
2004	Mackay	0	666	1313	25
	PEG	0	0	0	2004
	Proposed 1	0	0	0	2004
	Proposed 2	0	0	0	2004

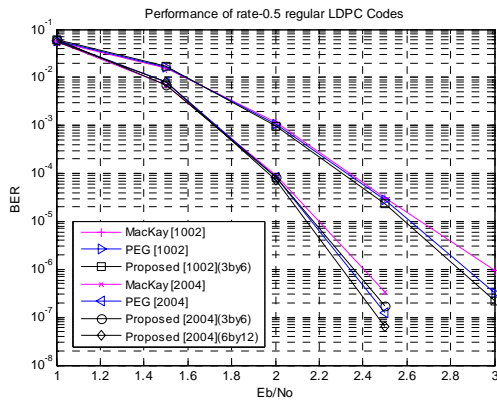


그림 2. 부호율 0.5 균일 LDPC 부호 성능

그림 2는 부호율 0.5인 균일 LDPC 부호의 성능을 비교한 결과이다. Proposed LDPC 부호의 성능은 높은 SNR에서 Mackay 부호보다 좋고 PEG 부호와 거의 비슷한 결과를 보여준다. 길이 2004에서 살펴보면 $m=3$, $n=6$ 인 경우 높은 SNR에서 PEG 부호보다 약간 성능이 떨어지는데 $m=6$, $n=12$ 인 경우는 더 좋은 성능을 보인다. 이것은 전체 길이에 비해 상대적으로 부행렬의 크기가 크면 무작위성이 떨어지기 때문이다. 짧은 길이에서 성능이 좋은 PEG 부호보다 구조적 제한을 두고 생성한 Proposed LDPC 부호의 성능이 더 좋으므로

본 논문에서 제안한 생성 알고리즘이 효과적이라 할 수 있다.

4.2 비균일 LDPC 부호

비균일 Proposed LDPC 부호와 같은 파라미터를 갖는 improved PEG 부호의 성능을 비교하였다. 비트 노드 분포는 다음과 같다

$$\lambda(x) = 0.2895x + 0.3158x^2 + 0.3947x^5 \quad (4-1)$$

$m=12$, $n=24$, $N_s=24,48$ 로 길이 576, 1152 부호에 대해 살펴 보았다. 표 2는 cycle 분포를 나타내는데 improved PEG 부호와 비슷한 평균 local girth를 보이고 girth는 같다.

표 2. 부호율 0.5 비균일 LDPC 부호의 cycle 분포

length	code	4-cycle	6-cycle	8-cycle	10-cycle
576	imPEG	0	68	508	0
	Proposed	0	192	384	0
1152	imPEG	0	0	1148	4
	Proposed	0	0	1008	144

그림 3과 그림 4는 각각 길이 576과 1152의 비균일 LDPC 부호의 성능 그래프이다. Improved PEG 알고리즘은 짧은 길이의 비균일 LDPC 부호를 edge 단위로 효과적으로 생성하는 기법으로 낮은 오류 마루를 갖는 부호를 만든다. Proposed LDPC 부호가 높은 SNR에서 성능을 향상시킨 improved PEG 부호와 비교하여 낮은 SNR에서는 비슷한 성능을, 높은 SNR에서는 더 좋은 성능을 보이기 때문에 제안된 알고리즘이 구조적 비균일 LDPC 부호의 생성에도 매우 효과적임을 알 수 있다.

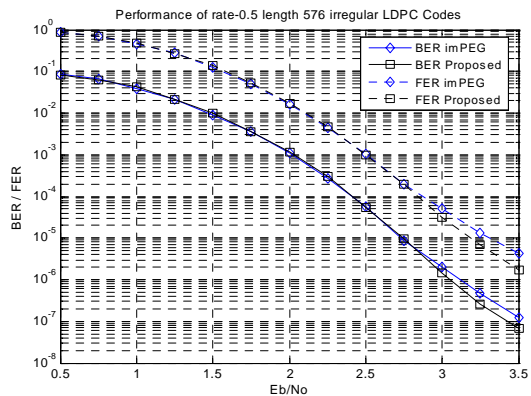


그림 3. 부호율 0.5 길이 576 비균일 LDPC 부호 성능

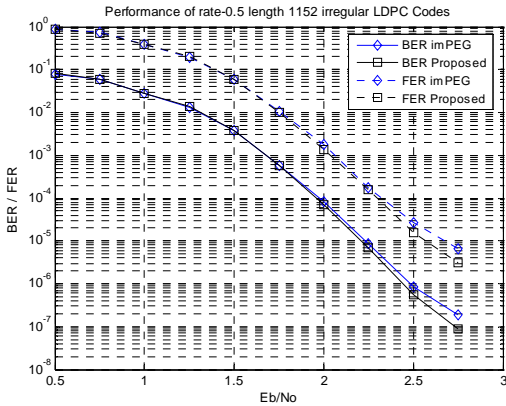


그림 4. 부호율 0.5 길이 1152 비균일 LDPC 부호 성능

V. 결론

본 논문에서는 부행렬 단위의 구조적인 균일, 비균일 LDPC 부호를 효과적으로 생성하는 알고리즘을 제안하였다. 이 알고리즘은 다양한 길이와 부호율, 무게 분포에 대해 girth가 크고 높은 SNR까지 성능이 뛰어난 부호의 설계를 가능하게 하여 짧은 길이의 구조적 LDPC 부호를 생성하는데 효과적이다.

참 고 문 헌

[1] R. G. Gallager, "Low-density parity-check codes," *IRE Trans. Inform. Theory*, vol. IT-8, pp. 21-28, Jan. 1962.

[2] D. J. C. MacKay, R. M. Neal, "Near Shannon limit performance of low-density parity-check codes," *Electron. Lett.*, vol. 33, pp.457-458, Mar. 1997.

[3] Thomas J. Richardson and R. L. Urbanke, "Efficient Encoding of Low-Density Parity-Check Codes," *IEEE Trans. IT*, vol. 47, pp. 638-656, Feb. 2001.

[4] Michael Yang, "Design of Efficiently Encodable Moderate-Length High-Rate Irregular LDPC Codes," *IEEE Trans. comm*, Vol. 52. pp. 564-571, April. 2004.

[5] X. Hu, E. Eleftheriou, and D.-M. Arnold, "Progressive Edge-Growth Tanner Graphs," in *IEEE GlobeCom*, vol. 2, pp. 995-1001, Nov. 2001.

[6] Hua Xiao and Amir H. Banihasemi, "Impro-

ved Progressive-Edge-Growth(PEG) Construction of Irregular LDPC Codes," *IEEE Trans. Commun.*, vol. 8, pp. 715-717, Dec. 2004.

[7] T. Tian, C. Jones, J. D. Villasenor, and R. D. Wesel, "Selective Avoidance of Cycles in Irregular LDPC Code Construction," *IEEE Trans. Commun*, vol. 52, pp. 1242-1247, Aug. 2004.

[8] T. Richardson, A. Shokrollahi, and R. Urbanke, "Design of capacity-approaching irregular low-density parity-check codes," *IEEE Trans. Inform. Theory*, vol. 47, pp. 619-637, Fer. 2001.

[9] David J. C. Mackay, "Good error-correcting codes based on very sparse matrices," *IEEE Trans. IT*, vol. 45, pp. 399-431, Mar. 1999.

[10] S. Lin, L. Chen, J. Xu, and I. Djurdjevic, "Near Shannon limit quasi-cyclic low-density parity-check codes," *IEEE Trans. Inform. Theory*, vol. 50, pp. 2030-2035, June. 2003.

[11] A. Ramamoorthy, R. Wesel, "Construction of short block length irregular low-density parity-check codes," *IEEE Trans. Commun*, vol. 1, pp. 410-414, June. 2004.

정 비 응 (Bi-Woong Chung)

준회원



2004년 2월 연세대학교 전기전자공학과 졸업
2004년 3월~현재 연세대학교 전기전자공학과 석사과정
<관심분야> Error Correcting Codes, PN Sequences, LDPC Codes

김 준 성 (Joon-Sung Kim)

정회원



2001년 2월 연세대학교 전파공학과 졸업
2003년 2월 연세대학교 전기전자공학과 석사
2003년 3월~현재 연세대학교 전기전자공학과 박사과정
<관심분야> Error Correcting Codes, PN Sequences, LDPC Codes

송 홍 엽 (Hong-Yeop Song)

중신회원



1984년 2월 연세대학교 전자공
학과 졸업

1986년 5월 USC 대학원 전자공
학과 석사

1991년 12월 USC 대학원 전자
공학과 박사

1992년~1993년 Post Doc., USC

전자공학과

1994년~1995년 Qualcomm Inc., 선임연구원

1995년 9월~현재 연세대학교 전기전자공학과 교수

<관심분야> PN sequences, Error Correcting Codes,
Spread Spectrum Communication Systems, Stream
Cipher Systems