

QC-LDPC 부호의 모든 작은 사이클을 검출하는 알고리즘

윤대영, 노종선, 김재원*
서울대학교, *경상대학교

dyyun@ccl.snu.ac.kr, jsno@snu.ac.kr, *jaewon07.kim@gnu.ac.kr

Algorithm for Detecting the All Short Cycles of QC-LDPC Codes

Dae-Young Yun, Jong-Seon No, Jae-won Kim*
Seoul National Univ., *Gyeongsang National University

요 약

본 논문에서는 Quasi-Cyclic Low Density Parity Check(QC-LDPC) 부호에서 짧은 cycle 을 검출하는 알고리즘을 제시한다. $2g$ -cycle 을 검출하는 알고리즘은 parity check matrix(PCM)의 m 개의 row 중에서 g 개를 골라 sub matrix 를 구성하고, 그 중 column weight 가 2 이상인 column 에서 cycle traveling 을 수행한다. QC-LDPC 의 특성을 이용하여 짧은 길이 n 을 갖는 base matrix 를 이용하여 z -lifting 한 길이 nz 의 전체 QC-LDPC 부호 속 cycle 을 구하며, $O(n^g)$ 의 복잡도를 보인다. 해당 알고리즘을 통해 cycle 의 개수 뿐만 아니라, 위치 및 경로까지 검출할 수 있다. 본 연구에서는 길이 400 이하의 QC base matrix 에서 길이 12 이하의 cycle 을 모두 검출해 보았다.

I. 서 론

LDPC 의 PCM 에 존재하는 cycle 들은 BP decoder 의 동작 기전을 방해하여 복호 성능을 떨어트린다. 특히 길이가 짧은 cycle 일 수록 치명적인 성능 열화를 일으켜 이른 오류 마루 현상을 보이게 한다. 때문에 그동안 cycle 을 최대한 피하며 부호를 만드는 여러가지 알고리즘들과 cycle 의 부정적인 효과를 억제하는 복호 방식들이 만들어 졌다.

그러나 주어진 부호에서 cycle 을 찾아서 취약한 변수 노드를 보호하는 식의 연구는 비교적 드문데, 모든 cycle 을 찾는 일은 계산 복잡도가 높아 일정 부호 길이 이상에서는 수행하기 어렵기 때문이다. 본 논문에서는 취약 변수 노드를 선별적으로 보호하는 향후 연구를 위해, QC-LDPC 의 성질을 이용하여 base matrix 기준 길이 400, 전체 코드 기준 길이 30000 까지의 부호에서 길이 12 이하의 cycle 을 찾는 작업을 수행하였다.

II. 본 론

가. PCM 및 QC base matrix 상에서 cycle 의 성질

먼저 cycle 을 찾는 알고리즘을 구상하자. 그림 1 예 cycle 6 예시가 나타나 있다. 우리가 PCM 에서 문제라고 하는 cycle 들은 각 column 마다 2 개의 entry 1 으로 구성되어 있다. 예시의 경우 3 개의 column 들이 서로 연결되어 있어 길이 6 의 cycle 을 구성하게 된다. 일반적으로는 길이 $2g$ 의 cycle 의 경우 g 개 row, g 개 column 에서 각각 2 개의 entry 1 들이 서로 연결되어 cycle 을 구성하게 된다. 각 entry 들은 어느 방향으로 이웃을 탐색해도 결국 자기 자신으로 돌아오게 된다.

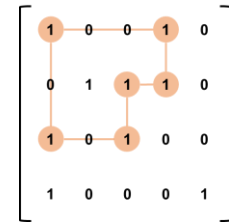


그림 1. Binary parity check matrix 에서 6-cycle 예시

QC-LDPC 는 base matrix 에 shift value 값을 entry 로 갖는다. 이는 z lifting 시 (z, z) circulant permutation matrix 로 바뀌어 실제 PCM 을 구성하게 된다. 이러한 성질을 이용하면 lifting 후의 크기가 매우 큰 PCM 상의 cycle 을 비교적 크기가 작은 base matrix 상에서 구할 수 있다. QC base matrix 상에서의 cycle 은 일반적인 PCM 의 cycle 요건에 shift value 의 plus-minus 값이 0 이 되어야 한다는 조건이 붙는다. 그림 2 의 예시에서 빨간색 entry 의 shift value 를 plus, 파란색 entry 의 shift value 를 minus 하여 합치면 0 이 나온다.

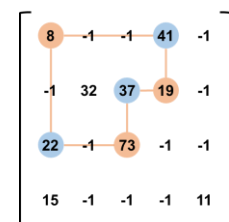


그림 2. QC-LDPC 의 base matrix 에서 6-cycle 예시

나. 알고리즘 구성

아래에 본 논문의 알고리즘이 제시되어 있다. $m \times n$ base matrix 에서 m 개의 row 중 g 개를 골라 submatrix 를 만들고, 그 안에서 degree 가 2 이상인 column 중 g 개를 골라 $g \times g$ submatrix 를 구성한다. 그 안에서 각 column 마다 1 entry 를 두개씩 골라 cycle 후보가 될 최종 submatrix H_{sub4} 를 만든다.

이제 column degree 2 를 갖는 $g \times g$ submatrix 에서 cycle traveling 을 시작한다. 그림 3에 cycle traveling 의 과정이 나타나 있다. 각 entry 는 자신과 row 혹은 column 이 같은 entry 들을 이웃으로 생각한다. 첫 entry 에서 시작하여 자신의 이웃 entry 를 탐색하는데, 방문한 entry 는 재방문하지 않는다. 이때 주어진 submatrix 가 cycle 을 형성한다면 entry 는 travel 의 결과로 자기 자신 에게 돌아오고, 형성하지 않는다면 돌아오지 못한다.

Cycle traveling 이 성공한다면, protograph level 에서 cycle 이 성립한다. 이제 shift value 를 고려해야 한다. traveling 하며 계산해둔 해당 entry 의 shift value plus-minus 값이 0 이라면, 해당 entry 들은 $2g$ -cycle 로 최종 확정된다. Cycle path list 에 해당 entry 들을 추가하고, 이 과정을 모든 H_{sub4} 조합에 대하여 수행한다.

Algorithm 1. detect all $2g$ cycle in a parity check matrix

Input: target cycle number g , $m \times n$ integer matrix H_{base}

1: while (choose all mC_g row groups) **do**

2: Selection: pick g rows and make $g \times n$ submatrix H_{sub1}
pick all columns with a degree 2+ (let the number of them h)
and make $g \times h$ submatrix H_{sub2}

if ($h < g$) **do go 2:**

end if

3: while (choose all hC_g col groups in H_{sub2}) **do**

Selection: pick g cols and make $g \times g$ submatrix H_{sub3}

while (choose all degree 2 combination from each column in H_{sub3}) **do**

4: Selection: pick 2 entries from each column in H_{sub3} ,
make $g \times g$ VN degree 2 submatrix H_{sub4} (cycle candidate)

5: Cycle traveling: travel the 1 entries in H_{sub4}

if (travel success) **do**

Shift value calculation: do plus-minus shift values

if (calculation result = 0) **do**

add travel path to cycle path list

end if

end if

6: end while

7: end while

8: end while

9: return cycle path list

알고리즘 1. $2g$ -cycle detection 기법의 pseudo code

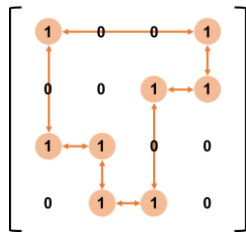


그림 3. 8-cycle 후보군에서 cycle traveling 예시

다. 실험 결과

본 연구에서는 Generalized LDPC 연구에 필요한 길이 400 의 variable node(VN) degree 2 를 갖는 regular QC base matrix 에 대하여 cycle detection 을 수행하였다. 각 base matrix 는 progressive edge growth(PEG) 알고리즘 을 통해 만들었다. 제안한 알고리즘을 통해 길이 12 이하의 cycle 의 개수 뿐만 아니라 위치, 경로까지 성공적으로 도출하였다. 검출해낸 cycle 의 개수는 표 1 과 같다.

row number of base matrix	8-cycle	10-cycle	12-cycle
96	2	21	257
100	2	21	168
104	1	15	126
108	0	8	91

표 1. 길이 400 의 VN degree2 regular base matrix 에서 cycle detection 결과

구현한 알고리즘의 복잡도를 분석해보면 행과 열을 뒤집어도 동치의 알고리즘이 구현 가능하므로, 총 n 개 column 에서 g 개를 선택해서 알고리즘을 수행한다고 생각하면 $O(n^g)$ 의 복잡도를 보인다. 따라서 주어진 base matrix 의 길이가 길어지고 구하는 cycle 의 크기가 커질수록 검출에 소요되는 시간은 가파른 속도로 늘어나게 된다. 실험에 사용한 길이 400 의 경우, 16-cycle 이상은 real time 안에 검출하기 어려웠다.

III. 결론

작은 크기의 사이클에 대하여 QC base matrix 에서 모든 사이클을 구하는 알고리즘을 구현하였다. 구현한 알고리즘은 PCM 상에서 cycle 이 각 column 마다 두 개의 entry 만 갖는다는 것을 이용하여 $g \times g$ submatrix 를 골라내고 그 안에서 cycle traveling 을 수행하며 cycle 여부를 판정한다. 실험 결과 길이 400 의 base matrix 에서 길이 12 까지의 cycle 을 성공적으로 검출하였으며, 알고리즘의 시간 복잡도는 $O(n^g)$ 이다.

참 고 문 헌

- [1] Marc P. C. Fossorier, "Quasi-Cyclic Low-Density Parity-Check Codes From Circulant Permutation Matrices," *IEEE Transactions on Information Theory*, 2004.
- [2] Pan Hu and Hongyu Zhao, "Improved Method for Detecting the Short Cycles of LDPC Codes," *IEEE International Conference on Information Theory and Information Security*, 2010.
- [3] Xiao-Yu Hu, E. Eleftheriou, and D.M. Arnold, "Regular and irregular progressive edge-growth tanner graphs," *IEEE Transactions on Information Theory*, 2005.