

5G 표준 LDPC 부호의 puncturing 기법 최적화

임성민, 이주형, 윤대영, 곽희열

울산대학교

dlatjdals017@naver.com, wngud387@mail.ulsan.ac.kr

yundy@ulsan.ac.kr, hykwak@ulsan.ac.kr

Optimization of the puncturing scheme for 5G LDPC Codes

SungMin Lim, Juhyeong Lee, DaeYoung Yun, and HeeYoul Kwak

University of Ulsan

요약

본 연구에서는 5G New Radio (NR) 표준에서 사용되는 low-density parity-check (LDPC) 부호에 puncturing 기법을 적용했을 때, 디코딩 알고리즘 종류와 최대 반복 횟수에 따른 성능 변화를 분석한다. 표준적인 sum-product (SP) 알고리즘을 충분히 많은 반복 횟수로 사용할 경우, 초기 두 블록에 puncturing을 적용하더라도 성능 손실이 거의 발생하지 않는다. 반면, 실용적인 환경에서 널리 사용되는 min-sum (MS) 알고리즘을 낮은 반복 횟수로 사용할 경우에는 puncturing으로 인한 성능 저하가 크게 나타난다. 본 연구는 이러한 실용적인 복호 환경에서 puncturing을 적용하지 않고 parity 비트를 일부 덜 사용하는 방식이 오히려 더 우수한 성능을 보일 수 있음을 보인다. 또한, neural min-sum (NMS) 알고리즘을 적용하면 MS 알고리즘에서 발생하는 puncturing 손실을 효과적으로 보완할 수 있음을 확인하였다.

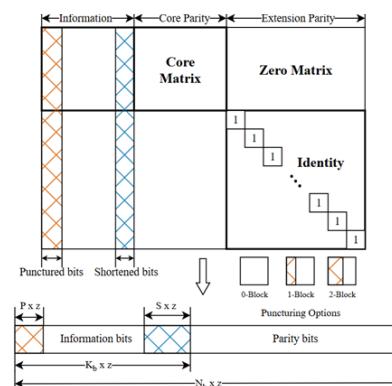
I. 서 론

5G New Radio (NR) 표준은 고속·고신뢰 통신을 위해 low-density parity-check (LDPC) 부호를 채택하였다. 5G NR LDPC 부호는 전송 요구에 맞춰 부호율을 조정하기 위해 rate matching을 사용하며, 이때 부호율을 성능 열화가 거의 없이 높이는 puncturing 기법을 사용한다. 5G NR 표준은 sum-product (SP) 알고리즘과 높은 반복 횟수를 가정한 상황에서 정해졌는데, 이 경우 앞의 2 block을 puncturing하는 방식이 성능 손실이 거의 없다. 그러나 해당 방식은 저복잡도 min-sum (MS) 알고리즘과 저지연 통신을 위한 낮은 반복 횟수 환경에서 성능 열화가 크게 나타난다. 이를 해결하고자 본 연구는 puncturing을 사용하지 않고 일부 parity 비트를 생략하는 전략과 neural MS (NMS) 알고리즘을 사용하는 전략을 제안한다. 실험 결과, 제안한 방법들이 표준에서 사용하는 2 block puncturing 방법보다 성능이 더 좋은 경우가 있음을 확인하였다.

II. 배경

<그림 1>은 5G NR LDPC 표준 부호의 기본 parity 검사 행렬 구조이다. 이 부호는 protograph를 기반으로 한 quasi-cyclic (QC) LDPC 부호 형태로 설계되며, protograph의 각 entry는 lifting factor z 에 따라 $(z \times z)$ 크기의 순환행렬로 치환되어 전체 parity 검사 행렬이 완성된다. 5G NR LDPC 부호는 정보, core parity, extension parity 영역으로 구성되며, 정보 비트와 parity 비트는 systematic 형태로 배치된다. 부호율을 조정하기 위해 rate matching 기법이 적용되며, 대표적으로 puncturing과 shortening 방식이 있다 [1]. puncturing은 부호율을 효율적으로 높이는 기법이며, shortening은 프레임 크기 조정을 위한 방식이다. <그림 1>에서 puncturing은 일부 비트 $(P \times z)$ 를 송신하지

않는 방식으로, 송신되지 않은 비트의 Log Likelihood Ratio (LLR) 값은 디코더에서 0으로 초기화된다. 이 비트들은 반복 복호 과정에서 다른 경로의 신뢰도 메시지를 통해 보강된다면 성능 저하를 거의 일으키지 않는다. 반면 shortening은 일부 비트 $(S \times z)$ 를 송신전에 0으로 고정하고, 디코더에서 해당 비트에서 해당 비트의 LLR값을 최대값으로 설정하여 항상 0으로 복원하는 방식이다. N_b 는 protograph의 총 블록 수, K_b 는 정보 비트 블록 수, P 는 punctured 블록 수, S 는 shortened 블록 수를 의미한다.



<그림 1> 5G NR LDPC 부호의 rate-matching 기법

LDPC 부호의 복호 과정은 반복적인 메시지 전달로 이루어진다 [2]. SP 알고리즘은 각 variable node (VN)에서 채널로부터 입력된 LLR을 연결된 check node (CN)로 전달하고, CN은 인접한 VN들로부터 받은 메시지를 기반으로 extrinsic 값을 계산하여 다시 VN으로 전달한다. CN에서 VN으로의 메시지 전달은 다음과 같이 정의된다 (1).

$$m_{c \rightarrow v}^{(\ell)} = 2 \tanh^{-1} \left(\prod_{v' \in N(c) \setminus v} \tanh \left(\frac{1}{2} m_{v' \rightarrow c}^{(\ell)} \right) \right) \quad \dots (1)$$

VN에서의 업데이트는 채널 LLR과 인접한 CN들로부터 전달된 메시지의 합으로 계산된다 (2).

$$m_{v \rightarrow c}^{(\ell)} = L_v^{ch} + \sum_{c' \in N(v) \setminus c} m_{c' \rightarrow v}^{(\ell-1)} \quad \dots (2)$$

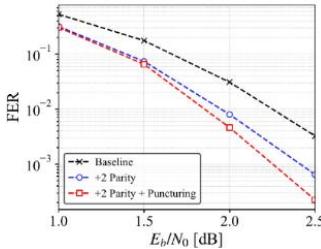
한편, MS 알고리즘은 SP의 계산 복잡도를 줄이기 위해 최소 연산을 사용한 다음 식을 사용한다 (3).

$$m_{c \rightarrow v}^{(\ell)} = \left(\prod_{v' \in N(c) \setminus v} \text{sgn}(m_{v' \rightarrow c}^{(\ell)}) \right) \min_{v' \in N(c) \setminus v} |m_{v' \rightarrow c}^{(\ell)}| \quad \dots (3)$$

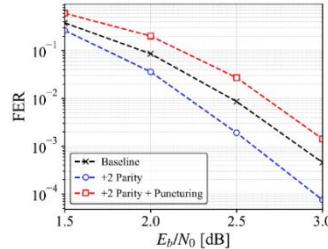
이 근사화로 복호 복잡도는 감소하지만, SP 알고리즘 대비 성능 열화가 발생한다.

III. 실험 설정 및 결과

먼저 우리는 puncturing 사용 여부에 따른 성능 변화를 확인하였다. baseline을 puncturing이 없고 부호율 0.5, 길이 640인 5G LDPC 부호라고 하자. <그림 2> 와 <그림 3>에 SP와 MS 알고리즘에 대해서 i) baseline ii) baseline에 2개의 추가적인 parity block을 추가한 경우 (부호율 0.45) iii) baseline에 2개의 parity block을 추가하고 앞 2 block을 puncturing 한 경우 (부호율 0.5)을 비교하였다. additive white Gaussian noise (AWGN) 채널과 iteration 50을 가정하였다.



<그림 2> SP 알고리즘 복호 성능



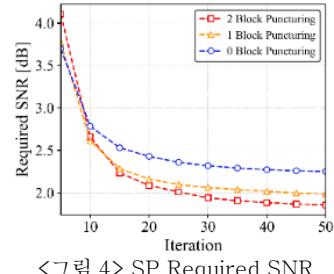
<그림 3> MS 알고리즘 복호 성능

<그림 2>에 따르면 SP 알고리즘에서의 복호 성능으로, 2 block을 puncturing 시켰을 때 성능이 가장 우수하다. 반면 <그림 3>의 MS 알고리즘 성능은 2 block puncturing의 성능이 baseline보다 나쁘다. 따라서 MS의 경우 puncturing을 사용하지 않고 추가적인 parity block도 사용하지 않는 baseline 전략이 더 나은 성능을 보인다.

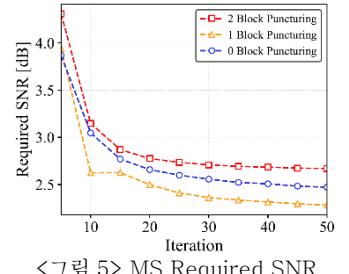
추가로 각 알고리즘에 대해 puncturing block 개수의 최적 값을 탐색하였다. 이를 위해 $FER = 10^{-2}$ 를 만족하는 required SNR 변화를 관찰하였다. SP, MS 알고리즘에 대해 각각 block puncturing 수를 2, 1, 0으로, iteration 횟수는 5, 10, ..., 50으로 변경하면서 복호를 수행하였다. 모든 결과의 부호율은 같게 추가적인 parity block을 추가하였다. <그림 4>의 SP 알고리즘은 반복 횟수가 증가함에 따라 2 block puncturing 이 가장 낮은 required SNR으로 가장 우수한 성능을 보인다. 반면, <그림 5>의 MS 알고리즘은 1 block만 puncturing을 했을 때 가장 우수한 성능을 보이며 2 block puncturing에서 가장 낮은 성능을 보인다.

이 다음으로, MS 알고리즘의 puncturing으로 인한 성능 손실을 완화하기 위해 VN 연산에 학습 가능한 가중치를 도입한 NMS 알고리즘을 사용한다 [3]. NMS 알고리즘에서의 VN 연산은 다음과 같이 정의된다 (4).

$$m_{v \rightarrow c}^{(\ell)} = w_{ch} L_v^{ch} + \sum_{c' \in N(v) \setminus c} w_{c \rightarrow v} m_{c' \rightarrow v}^{(\ell-1)} \dots (4)$$

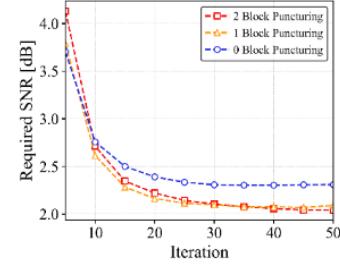


<그림 4> SP Required SNR



<그림 5> MS Required SNR

여기서 w_{ch} 는 채널 LLR의 신뢰도를 조정하는 가중치이며, $w_{c \rightarrow v}$ 는 CN에서 전달되는 메시지의 중요도를 조절하는 가중치를 의미한다. 가중치 학습 시 초기 w_{ch} 값은 0.75, 초기 $w_{c \rightarrow v}$ 값은 0.75, batch size=200, 학습률 $1 \cdot 10^{-4}$, epoch수는 10으로 설정하였고, 옵티마이저는 Adam 옵티마이저 [4]를 사용하였다.



<그림 6> NMS Required SNR

<그림 6>은 NMS 알고리즘의 required SNR 변화를 나타낸다. MS에서 2 block puncturing시 발생한 성능 손실이 N-MS 알고리즘 적용 후 완화되어 1 block puncturing과 동등 성능을 달성한다.

V. 결론

본 논문에서는 5G NR LDPC 부호에서 저복잡도 실용 환경인 MS 알고리즘에서 2 block puncturing의 대안과 성능 손실을 줄이는 방법을 확인해 보았다. MS 알고리즘에서 2 block을 puncturing 하는 것보다 일부 parity를 사용하지 않는 것이 더 우수한 성능을 보이며, 2 block puncturing의 손실을 학습 가능한 NMS 알고리즘을 사용하여 보완할 수 있는 것을 확인하였다.

ACKNOWLEDGMENT

이 논문은 2024년도 정부(과학기술정보통신부)의 재원으로 정보통신기획평가원의 지원(RS-2024-00398449, 네트워크 전문연구실 (NRC): 통신 세대 진화를 위한 채널 부호 부복호 및 채널 추정 기술, 기여율 50%)과 2024년도 정부(교육부)의 재원으로 한국연구재단원의 지원을 받아 수행된 연구임(No. RS-2023-00247197, LDPC 부호의 한계 극복으로 응용 분야 확대, 기여율 50%)

참고 문헌

- [1] Richardson, T., & Kudekar, S. (2018). Design of low-density parity check codes for 5G new radio. *IEEE Communications Magazine*, 56(3), 28–34.
- [2] Johnson, S. J. (2006). Introducing low-density parity check codes. *University of Newcastle, Australia*, 1, 2006, 31–39.
- [3] Nachimani, E., Marciano, E., Lugosch, L., Gross, W. J., Burshtein, D., & Be'ery, Y. (2018). Deep learning methods for improved decoding of linear codes. *IEEE Journal of Selected Topics in Signal Processing*, 12(1), 119–131.
- [4] Kinga, D., & Adam, J. B. (2015, May). A method for stochastic optimization. In *International conference on learning representations (ICLR)* (Vol. 5, No. 6).