

# 그래프 신경망 노드 속성에 의한 채널 복호의 개선

우우석, 이예진, 박호성\*

전남대학교 컴퓨터정보통신공학과, \*ICT 융합시스템공학과

182870@jnu.ac.kr, 201622@jnu.ac.kr, \*hpark1@jnu.ac.kr

## Channel Decoding improved by Node Attributes in Graph Neural Networks

Wooseok Woo ,Yejin Lee, \*Hosung Park

Chonnam National University

### 요 약

채널 복호와 통신을 위한 여러 가지 딥러닝 방법의 하나로 그래프 신경망 (graph neural network: GNN)을 이용한 방식이 있다. GNN 기반 복호가 다른 딥러닝 기반 방식과 다른 점은 임의의 블록 길이에 대한 확장성을 누릴 수 있다는 점과 차원성에 제한받지 않는다는 점이다. 본 논문은 기존 GNN 기반의 딥러닝 학습을 통한 복호에서 더 성능적으로 우수한 방법을 제안하며 GNN의 특징 중 하나인 속성 (attribute)을 이용하는 방안을 제시하고 결과를 비교 분석하였다.

### I. 서 론

최근 몇 년간 채널 복호와 통신을 위한 딥러닝 분야의 방식들이 빠르게 발전하고 있다. 이러한 방식 중 하나로 그래프 구조를 신경망을 통해 학습하는 GNN을 이용한 방식이 있다. GNN의 데이터 구조인 그래프는 노드, 간선, 전역 맥락 (global context)로 이루어져 있고, 이 안에 value라고 불리는 임베딩과 속성 정보들을 저장하여 업데이트한다. 이 방식이 다른 딥러닝 기반 방식과 다른 점은 임의의 블록 길이에 대한 확장성을 누릴 수 있다는 점과 차원성에 제한받지 않는다는 점이다.

저밀도 패리티 체크 (low-density parity-check: LDPC) 부호를 복호하는 기존의 방식으로 신뢰 전파 (belief propagation: BP) 복호 알고리즘이 있다. 이 알고리즘은 낮은 복잡성을 가지고 반복적인 메시지 패싱 복호 방법인 BP 복호를 사용한다. 현재 가중치를 이용한 BP 복호 이외의 다른 방식들은 확장성에 의해 제한된다는 점과 엄청난 양의 학습 가능한 매개변수를 갖는다는 문제점을 가지고 있다. Sebastian Cammerer와 Jakob Hoydis는 기존의 BP 복호 방식보다 성능에서 더 뛰어난 GNN 구조의 복호 방식[1]을 제안하였다. 이 방식은 BCH 코드 같은 경우에, BP보다 적은 반복을 통해 비슷한 성능을 내는 것을 관찰할 수 있었다.

본 논문은 기존 GNN 기반의 딥러닝 학습을 통한 복호기에서 더 성능적으로 우수한 방법을 제안하며 GNN의 특징 중 하나인 속성을 이용하는 방안을 제시하고 결과를 비교 분석하였다.

### II. 본론

#### 2.1 기존의 GNN 기반 채널 복호

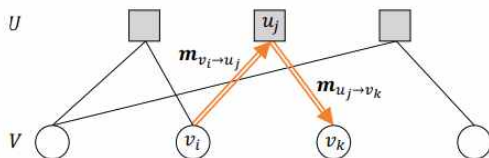


그림1. GNN 업데이트 방식 [1]

$$\begin{aligned} m_{v_i \rightarrow u_j} &= f^{F_u}([h_{v_i} \| h_{u_j} \| g_{m_{v_i \rightarrow u_j}}], \theta_{m_{v_i \rightarrow u_j}}) \\ m_{u_j \rightarrow v_i} &= f^{F_u}([h_{u_j} \| h_{v_i} \| g_{m_{u_j \rightarrow v_i}}], \theta_{m_{u_j \rightarrow v_i}}) \\ h'_{u_j} &= f^{F_u}([h_{u_j} \| \frac{1}{|V(u_j)|} \sum_{v_i \in V(u_j)} m_{v_i \rightarrow u_j} \| g_{u_j}], \theta_{u_j}) \\ h'_{v_i} &= f^{F_u}([h_{v_i} \| \frac{1}{|U(v_i)|} \sum_{u_j \in U(v_i)} m_{u_j \rightarrow v_i} \| g_{v_i}], \theta_{v_i}) \end{aligned}$$

그림 1은 기존 논문[1]에서 사용하는 GNN 업데이트 방식이다. 해당 식의  $g$ 가 속성을 나타낸다. 기존 논문은 속성을 이용하여 갱신하는 방식은 나와 있지만 실제로는 속성값을 사용하지 않는다.

#### 2.2 변수 노드 속성에 의한 복호 성능 개선

본 연구에서는 GNN 구조에서 유의미한 속성 초기값을 사용해 보는 것에 중점을 두었다. 속성 사용 시 기존의 방식보다 매개변수가 많아져서 복잡성은 높아지지만, 그만큼 성능의 유의미한 차이를 내는 것을 목표로 하였다. 사용해 본 속성 초기값들은 다음과 같다.

- A1. 속성값 사용 하지 않음 (기존 방식)
- A2. all zero
- A3. node degree
- A4. node degree normalized
- A5. cycles metric
- A6. cycles metric normalized

parameter	input
use attributes	True
node attributes initialize (variable node)	A1 ~ A6
node attributes initialize (check node)	all zero
edge attributes initialize	all zero
node attribute dims	1
edge attribute dims	1

표 1. 변경하여 실험한 매개변수 표

모든 속성의 차원은 1로 설정함으로써 기존 GNN과 복잡도가 큰 차이가 나지 않게 하였다.

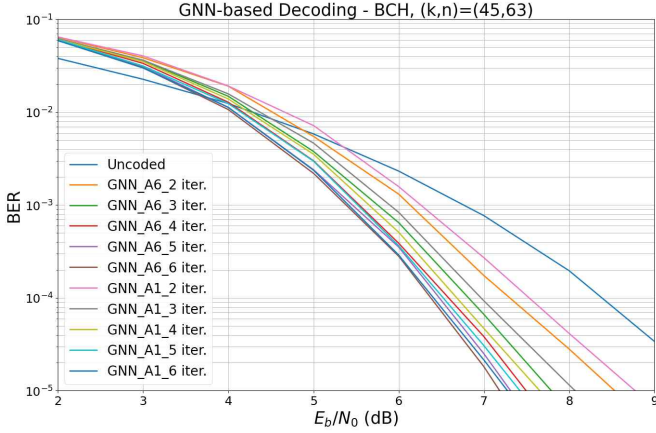


그림 2. 정규화된 사이클 메트릭 속성 사용 여부에 따른 BER 성능 비교

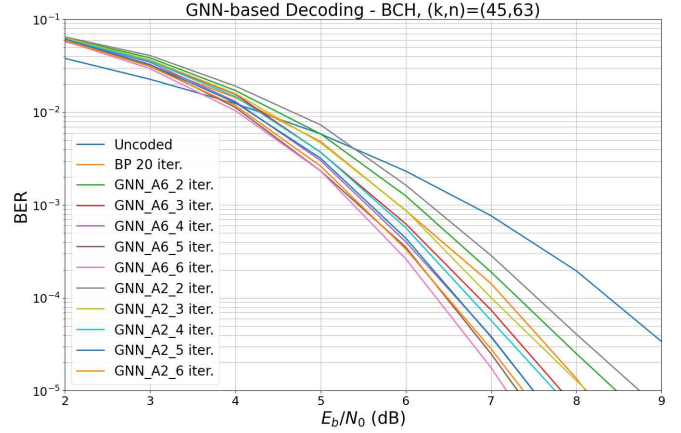


그림 3. 정규화된 사이클 메트릭과 All Zero와의 BER 성능 비교

속성 초깃값 중 하나인 사이클 메트릭 (cycles metric) [2]은

$$M_{v_p c_i} = \sum_{l=2}^{q-1} \beta^{l-2} \cdot N_{2l}^{j,l}, \quad M_{v_j} = \sum_{c_i \in (v_j)} M_{v_p c_i}$$

을 이용하여 각각 가변 노드의 사이클 메트릭을 구하였고 그 값을 0~1 사이의 값을 정규화해주었다. 사이클은 태너 그래프 상에서 노드가 원래 지나간 길을 제외하고 다시 자기 자신에게 돌아오는 경로를 말하고, 가장 작은 사이클을 겐스(girth)라고 한다. 위 식에서  $M$ 은 사이클 메트릭,  $v$ 는 가변 노드 (variable node),  $c$ 는 체크 노드 (check node),  $g$ 는 겐스,  $N_{2l}^{j,l}$ 는 에지  $E_{j,i}$ (edge)를 포함하는 사이클의 수를 나타낸다. 이때  $\beta$ 는 0.01이다.

GNN에서 속성은 노드 속성과 간선 속성이 있다. 우리 연구에서는 노드의 초깃값을 설정하였고 그중에서도 가변 노드의 속성 초깃값을 설정해 주었다. 나머지 간선 속성과 검사 노드의 속성 초깃값은 원래대로인 0으로 설정하였다.

### 2.3 실험 결과

실험 환경은 BCH(63, 45) 코드를 사용하였다. 이진 위상변이 변조(Binary Phase Shift Keying: BPSK) 된 신호를 가산성 백색 가우시안 잡음(Additive White Gaussian Noise: AWGN) 채널 환경에서 복호화 하는 과정을 실험해서 비트 오류율(Bit Error Rate: BER) 결과를 출력하였고 비교 분석하였다.

사용해 본 6가지의 속성 초깃값들 중에서 가장 좋은 성능을 나타낸 것은 2.2의 A6 cycles metric normalized, 정규화된 사이클 메트릭이었다. 속성값을 사용하지 않은 기존 GNN은 속성값을 0으로 사용했을 때와 값이 비슷하게 나타났다. 그리고 노드 차수와 사이클 메트릭 둘 다 0~1 사이로 정규화한 값이 정규화하지 않은 값보다 성능이 좋게 나오는 것도 확인할 수 있었다. 사이클 메트릭을 사용하는 경우에는 기존 GNN보다 많은 매개변수를 가지기에 최적의 학습을 찾기 위해 기존보다 학습 수를 늘렸다. 이러한 과정은 처음 한 번 진행된다.

그림 2는 기존 GNN과 정규화된 사이클 메트릭을 속성 초깃값으로 사용한 GNN 비교를 한 것이고, 그림 3은 All Zero 속성값을 사용한 GNN과 정규화된 사이클 메트릭을 속성 초깃값으로 사용한 GNN과의 비교를 진

행한 것을 나타낸다. 기존 GNN과 정규화된 사이클 메트릭을 비교한 그림 2에서는 더 높은 신호 대 잡음 비와 비교적 낮은 반복(2, 3, 4)에서 BER 성능 차이가 명확하게 나타났다. 그리고 All Zero 속성값과 사이클 메트릭을 비교한 그림 3에서는 All Zero를 속성 초깃값으로 사용하였을 때보다 사이클 메트릭 값을 넣었을 때 성능적으로 좋은 것을 확인할 수 있었다. 이는 속성값에 의미 있는 값을 넣었을 때 더 좋은 결과를 얻을 수 있다는 사실을 알 수 있다.

### III. 결론

본 논문에서는 선형 블록 코드의 복호를 위한 GNN 기반 알고리즘에서 추가적인 성능 향상을 위한 방법으로 속성값을 적극적으로 사용하였고, 관련된 실험을 진행하였다. 결과적으로 BCH 코드에서 기존 GNN보다 효과적인 성능 향상을 확인할 수 있었고, 이러한 과정은 기존의 GNN보다 나은 성능을 보이며 복잡도 측면에서 많이 늘어나지 않기에 더욱 효과적이다. 그렇지만 여전히 딥러닝의 방법은 계산 복잡도가 높다는 문제를 가지고 있다. 그렇지만 기존에 복호 과정에서 사용하지 않았던 GNN에 있던 속성값을 적극적으로 사용하여 효과를 알게 되었고, 이에 따라 다양한 곳에서 실험해 볼 수 있을 것이다.

### ACKNOWLEDGMENT

이 논문은 2022년도 정부(과학기술정보통신부)의 재원으로 정보통신기획평가원의 지원을 받아 수행된 연구임 (No.2020-0-00846, 5G와 방송망(ATSC 3.0) 연동 전송 기술 개발) 또한 이 논문은 2023년도 정부(과학기술정보통신부)의 재원으로 정보통신기획평가원의 지원을 받아 수행된 연구임 (No. 2021001016001, 6G/B5G xURLLC를 위한 유연한 신뢰도의 채널코딩)

### 참 고 문 헌

- [1] Sebastian Cammerer, Jakob Hoydis, Fayçal Aït Aoudia, and Alexander Keller, "Graph Neural Networks for Channel Decoding". arXiv:2207.14742
- [2] Xingcheng Liu, Feng Xiong, Zhongfeng Wang, and Shuo Liang, "Design of Binary LDPC Codes With Parallel Vector Message Passing", *IEEE Transactions on Communications*, vol. 66, pp. 1363-1375, 2018.