

## 유전 알고리즘을 기반한 저밀도 패리티 검사 부호 설계 성능 분석

김인기, 안현태, 주효상, 김상호\*

성균관대학교

\*iamshkim@skku.edu

## Performance Analysis of LDPC Code Design based on Genetic Algorithm

Ingi Kim, Hyun Tae Ahn, Hyosang Ju, Sang-Hyo Kim\*

Department of Electrical and Computer Engineering, Sungkyunkwan University

## 요약

유전 알고리즘은 전역 최적화에 유리한 메타 휴리스틱 기법으로, 복잡한 탐색 공간을 갖는 저밀도 패리티 검사 부호 설계에 효과적으로 적용될 수 있다. 본 연구는 유전 알고리즘을 활용한 저밀도 패리티 검사 부호 설계 방법을 소개하고, 다양한 실험을 통해 그 성능을 정량적으로 평가함으로써 기존 방법의 한계를 진단하고 개선 방향을 모색한다.

## I. 서론

유전 알고리즘(Generic Algorithm, GA)은 생물학적 진화 이론 중에서도 자연 선택과 유전 원리를 활용하여 설계된 진화 기반 메타 휴리스틱 최적화 기법으로 1975년 John Holland에 의해 최초로 체계화되어 제안되었다 [1]. GA는 복잡하고 비선형적인 탐색 공간을 갖는 다양한 최적화 문제에 효과적으로 적용되어 왔다. 본 알고리즘은 후보 해들로 구성된 개체군(population)을 기반으로 하며 세대를 거쳐 선택(selection), 교차(crossover), 돌연변이(mutation) 등의 유전 연산을 반복 수행함으로써 해 공간을 탐색하고 점진적으로 우수한 해를 도출한다. 이와 같은 메커니즘은 GA의 전역 탐색 능력을 향상시켜, 해 공간 내 전역 최적해(global optimum)에 도달할 가능성을 높인다.

저밀도 패리티 검사(Low-Density Parity-Check, LDPC) 부호는 1의 수가 매우 적은 패리티 검사 행렬에 의해서 정의되는 선형 부호(linear code)로, 특히 긴 길이에서 샤논(Shannon)의 한계에 근접하는 성능을 달성하는 오류정정부호이다 [2]. LDPC 부호는 우수한 복호 성능을 기반으로 5G NR(New-Radio) 데이터용 채널 부호로 사용되고 있다 [3].

LDPC 부호는 패리티 검사 행렬에 따라 설계됨에 따라 그 성능이 패리티 검사 행렬의 구조 즉, 엣지(edge)라 불리는 그래프의 연결 형태와 노드의 차수 분포에 크게 의존한다. 이에 따라 LDPC 부호 설계는 패리티 검사 행렬 설계 문제를 핵심으로 간주될 수 있다. 이때 패리티 검사 행렬의 최적 구조를 찾는 것은 조합 최적화(combinatorial optimization) 문제로서 가능한 설계 공간이 매우 크고 비선형적인 특성을 지녀 전통적인 설계 방식에만 의존하기엔 어려움이 있다.

이러한 배경에서 탐색 공간 내에서 효율적으로 우수한 성능을 지니는 구조를 찾는 것에 이점이 있는 GA를 포함한 휴리스틱 기반 기법들이 LDPC 부호 설계 문제에 적극적으로 적용되고 있다. 본 논문에서는 GA를 기반한 LDPC 부호 설계 기법을 소개하고 [4], 이에 대한 성능 분석을 통해 해당 방법의 장점과 한계점을 규명한다. 이를 바탕으로 향후 LDPC 부호 설계의 성능 향상을 위한 개선 방향을 제시한다.

## II. 본론

## 2.1. 유전 알고리즘 기반 LDPC 부호 설계 방법

본 방법은 LDPC 부호 및 패리티 검사 행렬 설계를 최적화 문제로 접근하며, 특정 SNR(Signal-to-Noise Ratio)에서의 블록 오류율(Block Error Rate, BLER)을 최소화하는 것을 목표로 한다. LDPC 부호의 특성에 따라 생성된 행렬  $\mathbf{H}$ 는 (수식 1)과 같은 제약 조건을 만족하여야 한다.

$$R_c = 1 - \frac{\text{rank}(\mathbf{H})}{n} \quad (\text{수식 1})$$

패리티 검사 행렬의 크기를 결정하는 변수노드(variable node) 수  $n$ 과 체크노드(check node) 수는 고정되며 행렬  $\mathbf{H}$ 가 full-rank를 만족하면 부호율  $R_c$ 는 (수식 1)을 만족한다.

GA 기반 LDPC 부호 설계 알고리즘은 초기 집단 생성(initialization), 적합도 평가(fitness evaluation), 선택(selection), 돌연변이(mutation), 교차(crossover), 다음 세대 구성(next generation) 단계 순으로 수행되며 사전에 정의된 세대 수까지 반복적으로 진행된다. 특히 돌연변이 및 교차 연산은 탐색 공간의 다양성을 확보하여 지역 최적해(local optimum)에 수렴하는 것을 효과적으로 방지하는 데 핵심적인 역할을 수행한다. 다음은 각 단계의 세부 설명이다.

- (1) 초기 집단 생성: 고정된  $n, m$ 에 대하여 초기 세대(generation 1)를 이루는 인구 40명을 무작위로 생성한다. 이때 각 인구는 2차원 행렬을 의미하며, 모든 행렬들은 패리티 검사 행렬에 해당하고 (수식 1)을 만족한다.
- (2) 적합도 평가: 현 세대(generation  $i$ )에 속하는 모든 행렬들에 대하여 적합도를 계산한다. 적합도 함수(Fitness function)로는 특정 SNR에서의 BLER을 사용하며 다음 세대 생성을 위한 선택 기준이 된다.
- (3) 선택: 적합도 평가에 따라 성능이 가장 우수한 상위  $T$ 개의 인구/행렬을 선택한다. 이 상위 개체들은 다음 세대에도 그대로 유지되며 교차 및 돌연변이를 위한 기반이 된다. 즉, 부모 역할을 수행한다.
- (4) 돌연변이: 선택된 부모 개체로부터 다음과 같은 변화를 주어 자식 행렬을 생성한다. 임의 위치에서 엣지를 하나

제거, 하나 추가, 제거와 추가를 모두 수행 각 부모에 대하여 총 3가지의 변화를 주어 돌연변이 인구를 생성한다.  
 (5) 교차: 두개의 부모 행렬( $H_1, H_2$ ) 간 2차원 대칭 교차를 적용한다.  $H_1$ 의 좌측(또는 상단) 절반과  $H_2$ 의 우측(또는 하단) 절반을 결합함으로써 자식 행렬을 생성한다.  
 (6) 다음 세대 구성: 선택된 부모 행렬들과 이를 돌연변이 및 교차를 통해 변형한 자식 행렬들로 다음 세대(generation  $i + 1$ )를 구성한다.

## 2.2. 유전 알고리즘 기반 LDPC 부호 설계 성능 분석

GA기반 LDPC 부호 설계의 성능을 평가하기 위해  $(n, m) = (128, 64)$ 로 설정된 파라미터 하에서 분석하였다. 적합도 함수는 BPSK(Binary Phase Shift Keying) 변조 및 AWGN(Additive White Gaussian Noise) 채널 환경에서  $E_b/N_0 = 4.5$  dB 조건하에 측정된 BLER 성능을 기반으로 정의하였다.

그림 1은 각 세대마다 최고 성능을 보이는 행렬의 4.5dB에서의 BLER을 나타낸 그래프로 GA의 탐색 성능 분석을 위한 것이다. 그래프에서 특정 지점(Point)으로 표시된 것은, 이전 모든 세대를 포함해 새로운 최고 성능의 부호도 출된 경우를 의미한다. 이를 통해 구현한 GA가 세대를 거치며 점진적으로 더 우수한 해를 찾아가는 과정을 확인할 수 있다.

그림 2는 설계된 LDPC 부호의 실제 성능을 평가하기 위해, AWGN 채널 환경에서 유사한 부호어 길이와 부호율을 갖는 5G-NR LDPC 부호 [3]와의 비교를 수행하였다. 제안된 LDPC 부호는 다양한 SNR 영역에서 5G-NR LDPC 부호에 비해 낮은 비트 오류율(Bit Error Rate, BER) 및 BLER을 기록하였으며, 이를 통해 GA 기반 설계 기법의 우수성과 실효성을 입증할 수 있었다. 다만, 고 SNR 영역에서 두 부호 간 성능이 교차하는 현상에 대해서는 추가적인 분석이 필요하다.

## 2.3. 오류 마루 현상 및 알고리즘 개선 방향

오류 마루(Error floor) 현상은 LDPC 부호와 같은 오류 정정 부호에서 높은 SNR 영역에서도 BER 및 BLER이 더 이상 감소하지 않고 정체되는 현상을 의미한다. 일반적으로 복호화 반복 과정에서 일부 비트 오류가 계속 발생하도록 만드는 트래핑 셋(trapping set), 짧은 길이의 사이클(cycle)이 오류 마루 현상의 주된 원인이다 [5].

앞서 제시한 실험 결과를 통해, 5G NR 표준 LDPC 부호에 비해 GA 기반으로 설계된 LDPC 부호는 오류 마루 현상이 상대적으로 이른 SNR 영역에서 발생함을 확인할 수 있었다. 이러한 성능 저하를 개선하기 위해, 초기 집단 생성, 돌연변이, 교차 연산 과정에 제약 조건(constraint) 또는 구조적 설계 요소를 추가적으로 도입함으로써 성능 향상이 기대된다.

한편, 본 알고리즘은 적합도 평가 단계에서 몬테카를로(Monte-Carlo) 시뮬레이션 기반의 성능 측정을 활용하고 있으며, 이에 따라 높은 계산 복잡도(computational complexity)를 수반하게 된다. 부호 블록 길이 또는 패리티 검사 행렬의 크기가 커질수록 시뮬레이션 시간이 급격히 늘어나며, 이는 긴 길이에서 뛰어난 성능을 보이는 LDPC 부호의 설계에 비효율적인 측면이 있다. 또한, 현재 적합도 함수는 단일 SNR 지점에서의 BLER 성능을 기준으로 평가되므로, 다른 SNR 영역에서의 성능에 대한 일반화 가능성이 제한되는 단점이 존재한다. 따라서 보다 효율적이고 일반화된 성능 예측이 가능한 대체 적합도 함수의 설계는 계산 복잡도 절감 뿐 아니라, 다양한 채널 조건에 대한 강건한 성능 확보를 위한 중요한 개선 방향으로 고려될 수 있다.

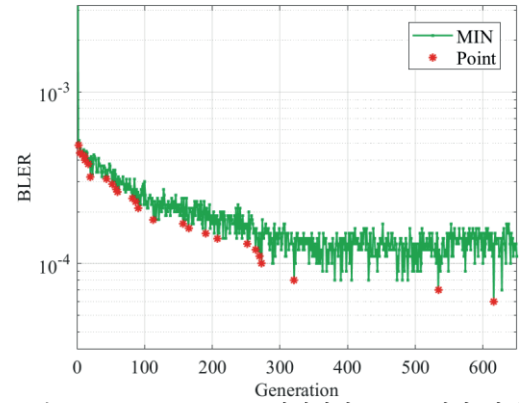


그림 1.  $E_b/N_0 = 4.5$  dB 조건에서의 BLER 진화 과정

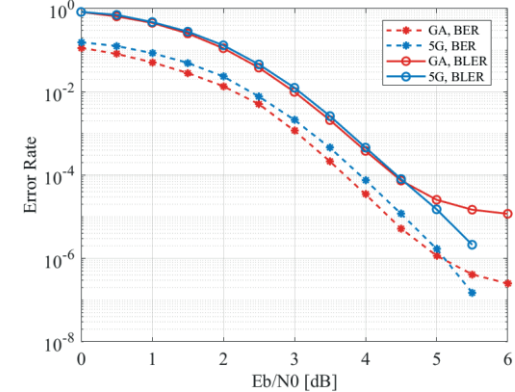


그림 2. GA 기반으로 설계된 부호와 5G NR 부호의 성능 비교

## III. 결 론

본 논문에서는 메타 휴리스틱 최적화 기법인 GA를 기반으로 한 LDPC 부호 설계 방법을 제안하고, 이에 대한 성능을 분석하였다. GA는 전역 최적해 탐색에 강점을 가지며, 실험 결과를 통해 해당 기법이 5G 표준 LDPC 부호보다 우수한 성능을 발휘할 수 있음을 확인하였다. 그러나 제안된 알고리즘은 특정 한계점도 동반하는데, 특히 높은 SNR 영역에서의 오류 마루 현상, 높은 계산 복잡도, 그리고 적합도 함수의 SNR 의존성 등이 성능 저하의 요인으로 작용할 수 있음을 확인하였다. 이에 따라 본 연구에서는 이러한 한계를 극복하기 위한 설계 제약조건의 도입, 적합도 함수 개선 등 여러 가능성 있는 개선 방향을 제시하였다. 향후에는 차세대 6G 통신 시스템을 위한 더 높은 부호화 이득(coding gain)과 고 SNR 영역에서의 강건한 오류 성능 확보를 목표로 하기에, LDPC 부호 설계에 대한 지속적인 연구가 요구된다.

## ACKNOWLEDGMENT

이 논문은 2025년도 정부(과학기술정보통신부)의 재원으로 정보통신기획평가원의 지원(RS-2024-00398449), 한국연구재단의 지원(RS-2024-00343913) 및 정보통신기획평가원의 지원을 받아 수행된 연구임(RS-2024-00397216)

## 참 고 문 헌

- [1] J. H. Holland, "Adaptation in Natural and Artificial Systems,," Ann Arbor, MI: Univ. of Michigan Press, 1975.
- [2] R. Gallager, "Low-density parity check codes," IRE Trans. Inf. Theory, vol. 8, no. 1, pp. 21-28, Jan. 1962.
- [3] 3GPP, NR; Multiplexing and channel coding, TS 38.212, v 15.2.0, Dec. 2018.
- [4] A. Elkelesh, M. Ebada, S. Cammerer, L. Schmalen, and S. ten Brink, "Decoder-in-the-Loop: Genetic Optimization-Based LDPC Code Design," in IEEE Access, vol. 7, pp. 141161-141170, 2019.
- [5] T. J. Richardson, "Error floors of LDPC codes," in Proc. 41st Annu. Allerton Conf. Commun., Control, and Comput., 2003.