

타부 목록 확률적 순차 가변 임계값 경사 하강 비트 반전 알고리즘

김명인*, 하정석^o

Tabu-List Probabilistic Sequential Adaptive Threshold Gradient Descent

Bit-Flipping Algorithm

Myungin Kim*, Jeongseok Ha^o

요 약

본 논문은 저밀도 패리티 검사 (low-density parity-check, LDPC) 부호의 비트 반전 알고리즘 (bit-flipping algorithm, BFA)의 부호 오류율 (Word Error Rate, WER) 및 복호 속도 개선에 관한 연구이다. 비트 반전 알고리즘은 비트의 신뢰도를 측정하는 측도인 반전 함수 (flipping function) 값의 임계값을 초과하는 비트를 반전시켜 오류를 정정하는 방식으로 동작한다. 본 논문에서는 반전 함수의 국소 최댓값을 이용해 반전 함수의 임계값을 매 반복마다 적응적으로 설정하여 반복 복호 횟수와 임계값을 찾는데 필요한 정렬 복잡도를 감소시키고, 복호 실패의 주 원인 중 하나인 trapping set 문제를 해결하기 위해 확률적으로 비트를 반전시켜 부호 오류율을 개선하였다.

Key Words : LDPC Code, Bit-Flipping Algorithm, Flipping Function, Threshold, Tabu List

ABSTRACT

This work proposes a BFA for LDPC codes aiming to improve the WER and decoding speed. BFAs correct errors by flipping the bits whose

flipping function values are larger than a predetermined threshold. In the proposed BFA, the threshold is adaptively chosen at each iteration by utilizing the local maximum value of the flipping function. It expedites the decoding speed, decreases the sorting complexity for selecting thresholds, and improves the WER performance by probabilistically flipping the bits constituting trapping sets.

I. 서 론

BFA는 LDPC 부호의 복호 알고리즘 중 하나로 각 변수 노드 (variable node)에 할당된 비트의 신뢰도를 측정하는 반전 함수를 정의하여 반전 함수가 특정 임계값 이상의 값을 가지는 비트를 반전시키는 방식으로 동작하는 알고리즘이다. BFA는 기존의 LDPC 부호의 복호 알고리즘인 합곱 알고리즘 (sum-product algorithm, SPA)이나 최소합 알고리즘 (min-sum algorithm, MSA)보다 계산 복잡도가 낮고 연산정 값을 계산할 필요가 없어서 하드웨어 측면에서 구현하기 쉽다는 장점이 있다.

BFA의 오류정정 능력을 개선하기 위해 경사 하강 비트 반전 알고리즘 (gradient-descent BFA, GDBFA)에서는 반전 함수를 채널로부터 수신된 값을 반영하도록 수정하여 BFA의 부호 오류율을 감소시켰다¹⁾. 또한 BFA의 복호 시 반복 횟수를 줄이기 위해 첫 반복에서 반전 함수의 임계값을 적응적으로 설정하는 비트 반전 알고리즘이 제안되었다²⁾. 그러나 BFA의 반복 복호 횟수를 최대로 감소시키기 위해서는 반전 함수 임계값을 매 반복마다 적응적으로 설정하는 알고리즘을 설계할 필요가 있다. 이와 별개로 반전 함수의 최댓값을 구하는데 필요한 정렬 복잡도를 줄이기 위해 반전 함수를 특정 개수의 그룹으로 나누어 그 국소 최댓값을 가지는 비트를 일정 개수만큼 뒤집을 수 있는 비트 반전 알고리즘이 제안되었다³⁾. 위의 방법들을 결합하여 반전 함수의 정의에 채널로부터 수신된 값을 반영하면서 반복 복호 횟수를 감소시키고 정렬 복잡도 문제를 해결하기 위해 반전 함수의 일부 국소 최댓값만을 가지고도 임계값을 매 반복마다 적응

* 이 논문은 2020년도 정부(미래창조과학부)의 재원으로 한국연구재단의 지원을 받아 수행된 기초연구사업임(No. NRF-2018R1A2B6003474)

• First Author : (ORCID:0000-0002-0720-2671)School of Electrical Engineering, Korea Advanced Institute of Science and Technology, kmi9212@kaist.ac.kr, 학생(박사과정), 학생회원

o Corresponding Author : (ORCID:0000-0003-1262-151X)School of Electrical Engineering, Korea Advanced Institute of Science and Technology, jsha@kaist.edu, 정교수, 종신회원

논문번호 : 202012-325-A-LU, Received December 23, 2020; Revised January 22, 2021; Accepted January 25, 2021

적으로 설정할 수 있는 순차 가변 임계값 경사 하강 비트 반전 알고리즘 (sequential adaptive threshold GDBFA, SAT GDBFA)이 제안되었다⁴⁾.

그러나 임계값 이상의 반전 함수 값을 가지는 비트를 일괄적으로 뒤집을 경우 오류인 비트를 반복적으로 반전시키는 현상, 즉 trapping set에 빠지는 문제가 발생한다. 이를 해결하기 위해 임계값 이상의 반전 함수 값을 가지는 비트를 확률적으로 반전시키는 확률적 병렬 비트 반전 알고리즘 (Probabilistic Parallel BFA, PPBFA)과 이전 반복에서 반전시킨 비트는 다시 반전시키지 않는 타부 목록 랜덤 페널티 경사 하강 비트 반전 알고리즘 (Tabu List Random Penalty GDBFA, TR GDBFA)이 제안되었다^{5,6)}. 본 논문에서는 기존의 SAT GDBFA에서 발생하는 trapping set 문제와 변수 노드 차수 (variable node degree)가 낮을수록 오류율을 최소화시키기 위한 최적 임계값을 찾기 어려운 문제를 해결하기 위해 확률적으로 비트를 뒤집으면서 타부 목록을 생성하여 오류정정능력을 개선한 타부 목록 확률적 순차 가변 임계값 경사 하강 비트 반전 알고리즘 (Tabu-list Probabilistic Sequential Adaptive Threshold GDBFA, TB PSAT GDBFA)을 제안하였다.

II. 제안한 알고리즘의 동작 방식

제안한 TB PSAT GDBFA는 PPBFA에서 확률적으로 비트를 반전시키는 방법과 TR GDBFA에서 타부 목록을 생성하는 방법을 SAT GDBFA에 적용하여 설계된다.

실험 환경으로는 변수 노드 차수와 검사 노드 차수 (check node degree)가 각각 d_v, d_c 로 일정한 정규 (regular) LDPC 부호와 교차 확률 (crossover probability)이 ϵ 인 이진 대칭 채널 (binary symmetric channel, BSC)을 가정한다. 채널을 통해 수신한 벡터를 $\mathbf{r} = (r_1, r_2, \dots, r_N)$, 복호기에서 매 반복마다 결과로 얻어지는 벡터를 $\mathbf{z} = (z_1, z_2, \dots, z_N)$ 라고 하자. 알고리즘의 반복 횟수는 ℓ , 최대 반복 횟수는 I_{\max} 로 표기한다. 제안한 PSAT GDBFA의 동작 방식은 다음과 같다.

1단계 : 각 검사 노드 m 에서 신드롬 값 s_m 을 검사 노드 m 과 연결된 변수 노드들의 집합인 $N(m)$ 에 있는 변수 노드 n 의 모든 비트 값 z_n 을 이진 덧셈

하여 계산한다.

$$s_m = \sum_{n \in N(m)} z_n \pmod{2} \quad (1)$$

2단계 : 각 변수 노드 n 에서 변수 노드와 연결된 검사 노드들의 집합인 $M(n)$ 에 있는 모든 검사 노드 m 으로부터 계산된 신드롬값 s_m 의 합과 채널로부터 받은 비트값 r_n 과 현재 비트값 z_n 사이의 연관성을 반영하는 식을 더하여 반전 함수 E_n 을 계산한다. 여기서 \oplus 연산은 이진 덧셈 연산을 의미한다.

$$E_n = \sum_{m \in M(n)} s_m + z_n \oplus r_n \quad (2)$$

3단계 : 먼저 이전 $\ell - 1$ 번째 반복에서 반전시킨 비트의 집합 $B^{(\ell-1)}$ 을 현재 반복 ℓ 에서 반전시키지 않기 위해 이를 타부 목록으로 정의한다. 반전 함수 값이 임계값 $\theta^{(\ell)}$ 이상이면 타부 목록에 들어가있지 않는 변수 노드의 집합인 $B^{(\ell)}$ 에 있는 비트 값을 확률 p 로 (5)와 같이 반전시킨다. 반전 함수의 임계값 설정 방식은 그림 1과 같이 반전 함수의 국소 최댓값에 따른 순람표 (look-up table, LUT)를 관찰하여 결정된다.

$$B^{(\ell)} = \{n \mid E_n \geq \theta^{(\ell)}\} \cap B^{(\ell-1)c} \quad (3)$$

$$a_n \sim \text{Bern}(p) \quad (4)$$

$$z_n = z_n \oplus a_n \quad \forall n \in B^{(\ell)} \quad (5)$$

4단계 : 반복 횟수 ℓ 을 1 증가시키고 1-3단계를 모

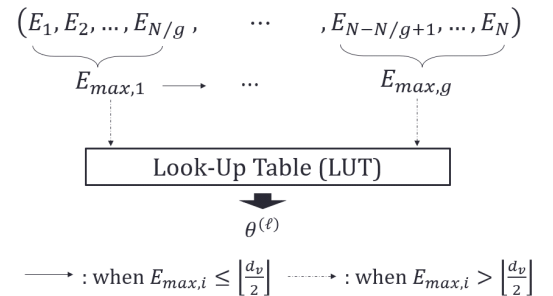


그림 1. PSAT GDBFA의 임계값 설정 방식
Fig. 1. Threshold determination method in PSAT GDBFA

든 신드롬 값이 0이 되어 부호어가 복호되거나 $\ell = I_{\max}$ 가 될 때까지 알고리즘을 반복한다.

그림 1의 제안한 PSAT GDBFA의 임계값 설정 방식은 다음과 같다. 먼저 반전 함수의 벡터 $\mathbf{E} = (E_1, E_2, \dots, E_N)$ 를 g 개의 그룹으로 분할한다. 그리고 첫 번째 그룹부터 순차적으로 국소 최댓값을 구하기 시작하여 만약 그룹에서 구한 국소 최댓값이 $\lfloor d_v/2 \rfloor$ 보다 큰 경우 순람표를 관찰하여 반전 함수 임계값을 구하고, $\lfloor d_v/2 \rfloor$ 보다 작은 경우 다음 그룹에서 국소 최댓값을 구하여 같은 과정을 반복한다. 여기서 반전 함수 임계값을 구하기 위해 필요한 그룹의 개수를 g^* 이라 하자. 만약 모든 그룹에서 구한 국소 최댓값이 $\lfloor d_v/2 \rfloor$ 보다 작거나 같을 경우 임계값은 반전 함수 값 전체의 최댓값인 $\max_n E_n$ 으로 설정한다. 제안한 알고리즘의 경우 평균적으로 g^* 가 전체 그룹의 개수 g 보다 작을 때 반전 함수 임계값을 구할 수 있어 모든 반전 함수의 최댓값을 구하는 기존 방식과 비교하여 임계값을 구하기 위해 필요한 비교 유닛의 개수를 감소시킬 수 있다.

여기서 순람표는 genie-aided 경사 하강 비트 반전 알고리즘 (genie-aided GDBFA, GA GDBFA)이 반전 함수 벡터의 국소 최댓값에 어떤 임계값을 선택하는지 경향성을 분석하여 설계하였다. GA GDBFA는 매 반복마다 오류율을 최대로 낮출 수 있는 임계값을 설정하여 오류를 정정하는데 필요한 반복 횟수를 최소화시키고 오류정정능력을 극대화시킨 가상 알고리즘이다. GA GDBFA의 임계값 설정방식은 (6)과 같다.

$$\theta^{(\ell)} = \min_{\theta \in \{1, 2, \dots, d_v + 1\}} P_e^{(\ell+1)}(\theta) \quad (6)$$

여기서, $P_e^{(\ell+1)}(\theta)$ 은 ℓ 번째 반복에서 반전 함수의 임계값을 θ 로 잡았을 때 $\ell + 1$ 번째 반복에서 비트 오류율을 나타낸 것이다.

이렇게 순람표를 관찰하여 반전 함수의 임계값을 적응적으로 선택하는 방식은 변수 노드의 차수가 큰 경우에는 오류율을 최소화시키기 위한 최적의 임계값을 쉽게 찾을 수 있어 반복 복호 횟수를 크게 줄일 수 있다. 그러나 변수 노드 차수가 낮을수록 GA GDBFA의 반전 함수의 국소 최댓값에 따른 반전 함수 임계값의 통계적인 수치가 임계값에 따라 확률적으로 비슷한 분포를 보이는 현상이 발생하여 최적 임

계값을 설정하기 어려운 문제가 발생한다. 제안한 확률적 비트 반전 기법을 적용시킬 경우 최적의 반전 함수 임계값을 찾지 못하더라도 오류인 비트의 개수가 급격하게 증가하는 현상을 방지할 수 있어 오류율 감소에 기여할 수 있다.

III. 실험 결과

표 1, 그림 2, 그림 3은 길이가 1296, $d_v = 3$, 그리고 부호율 (code rate)이 0.5인 LDPC code에서 진행한 비트 반전 알고리즘의 실험 결과이다. 그림 2, 그림 3의 Tabu-list Probabilistic Parallel GDBFA (TB PP GDBFA)는 타부 목록 및 확률적 비트 반전 기법을 GDBFA에 적용시킨 알고리즘을 의미하고 PSAT GDBFA는 SAT GDBFA에서 타부 목록을 생성하지 않고 확률적으로만 비트를 반전시키는 알고리즘이다. 표 1은 제안한 TB PSAT GDBFA을 $\epsilon = 0.04$ 인 BSC 환경에서 g 를 변화시켰을 때 구한 평균 WER, 복호 시 얻어진 g^* 의 평균값, 매 반복에서 복호기가 필요로 하는 평균 비교 유닛 (compare unit, CU) 개수, 단위 부호어에서 복호기가 필요한 평균 비교 유닛 개수, 매 반복에서 기존의 복호 방식 즉 $g = 1$ 인 경우와 비교하여 그림 1에서 제안된 방식을 사용할 때 감소하는 평균 비교 유닛 개수의 비율, 단위 부호어 측면에서 봤을 때 감소하는 평균 비교 유닛 개수의 비율을 나타낸 것이다. 표 1에서의 결과를 통해 필요한 비교 유닛 개수와 부호 오류율을 최소화시키기 위하여 그림 2와 그림 3의 성능 비교 그래프에서는 g 를 108로 설정하였다. 표 1을 통해 제안한 알고리즘이 기존의 반전 함수의 최댓값을 임계값으로 설정하는 알고리즘과 비교하여 평균적으로 절반 정도의 비교 유

표 1. PSAT GDBFA의 그룹의 개수 변화에 따른 성능 및 복잡도
Table 1. Performance and complexity of PSAT GDBFA for various g 's

| g | 81 | 108 | 144 |
|--------------------------|-------|-------|-------|
| WER | 0.01 | 0.008 | 0.009 |
| $E[g^*]$ | 41 | 54 | 72 |
| $E[CU]$ (/iter.) | 652 | 645 | 647 |
| $E[CU]$ (/word) | 22660 | 22071 | 22262 |
| reduced $E[CU]$ (/iter.) | 0.500 | 0.505 | 0.504 |
| reduced $E[CU]$ (/word) | 0.508 | 0.520 | 0.516 |

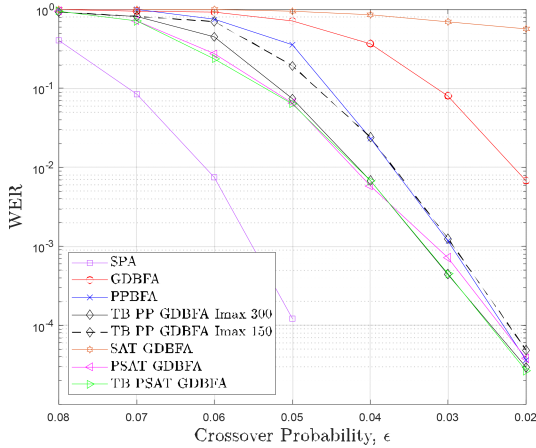


그림 2. 기존의 BFA와 제안한 BFA의 WER 비교
Fig. 2. Comparison of WER between the conventional BFAs and the proposed one

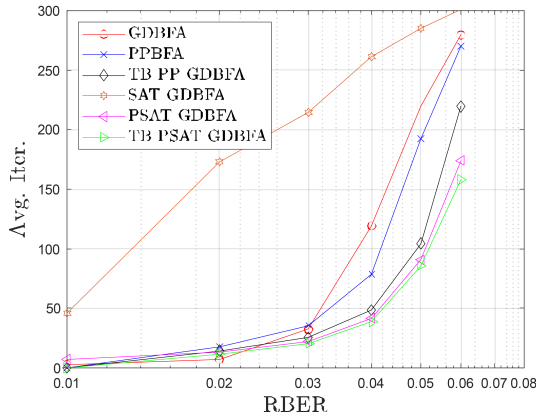


그림 3. 기존의 BFA와 제안한 BFA의 평균 복호 횟수 비교
Fig. 3. Comparison of average decoding iterations between the conventional BFAs and the proposed one

닛을 사용하더라도 반전 함수 임계값을 설정할 수 있음을 보여준다. 그림 2와 그림 3에서 I_{max} 는 300으로 설정하였고, TB PP GDBFA의 경우 제안한 알고리즘과 동일한 복잡도에서 성능 비교를 보이기 위해 I_{max} 가 150인 성능 그래프를 추가하였다. 그림 2와 그림 3을 통해 제안한 알고리즘이 동일한 복잡도에서 TB PP GDBFA보다 우수한 성능을 보임을 확인할 수 있고 평균적으로 더 적은 반복 횟수로도 동일한 성능을 얻을 수 있음을 알 수 있다.

IV. 결론

본 논문에서는 순차 가변 비트 반전 알고리즘에서 확률적으로 비트를 반전시키면서 타부 목록을 생성하여 trapping set문제를 해결함으로써 성능을 향상시켰다. 제안한 TB PSAT GDBFA는 경관정 정보만이 제공되어있는 BSC와 같은 환경에서 부호 오류율, 반전 함수의 임계값을 찾기 위한 정렬 복잡도, 평균 복호 횟수 측면에서 모두 우수한 성능을 보이므로 저복잡도 오류정정 능력이 필요한 영역에서 응용이 가능할 것으로 기대된다.

References

- [1] T. Wadayama, et al., "Gradient descent bit flipping algorithms for decoding LDPC codes," in *IEEE Trans. Commun.*, vol. 58, no. 6, pp.1610-1614, Jun. 2010.
- [2] J. Jung and I. Park, "Multi-bit flipping decoding of LDPC codes for NAND storage systems," in *IEEE Commun. Lett.*, vol. 21, no. 5, pp. 979-982, May 2017.
- [3] J. Cho and W. Sung, "Adaptive threshold technique for bit-flipping decoding of low-density parity-check codes," in *IEEE Commun. Lett.*, vol. 14, no. 9, pp. 857-859, Sep. 2010.
- [4] M. I. Kim, et al., "A fast convergence bit-flipping algorithm for LDPC codes over BSC," in *Proc. KICS*, pp. 36-37, Yongpyong Korea, Jan. 2019.
- [5] K. Le, et al., "A probabilistic parallel bit-flipping decoder for low-density parity-check codes," in *IEEE Trans. Cir. Syst. I Regul. Pap.*, vol. 66, no. 1, pp. 403-416, Jan. 2019.
- [6] H. Cui, et al., "An improved gradient descent bit-flipping decoder for LDPC codes," in *IEEE Trans. Cir. Syst. I Regul. Pap.*, vol. 66, no. 8, pp. 3188-3200, Aug. 2019.