

복잡도를 줄인 LDPC 복호를 위한 새로운 Simplified Sum-Product 알고리즘

준회원 한 제 희*, 중신회원 선우명훈**

New Simplified Sum-Product Algorithm for Low Complexity LDPC Decoding

Jae Hee Han* Associate Member, Myung Hoon Sunwoo** Lifelong Member

요 약

본 논문은 BER 성능을 높인 LDPC 복호를 위한 새로운 SSP 알고리즘을 제안한다. 제안하는 SSP 알고리즘은 추가적인 연산 없이 곱셈 연산과 나눗셈 연산을 덧셈 연산과 뺄셈 연산으로 대체가 가능하다. 제안하는 SSP 알고리즘은 $\ln[\tanh(x)]$ 함수와 $\tanh^{-1}[\exp(x)]$ 함수를 각각의 양자화 테이블을 사용하여 단순화하기 때문에 연산 복잡도를 줄일 수 있다. 시뮬레이션 결과 제안하는 SSP 알고리즘은 기존의 근사화 SP 알고리즘과 비교하여 BER 성능을 0.3 ~ 0.8 dB 향상 시킨 것을 보여준다.

Key Words : Iterative decoding, sum-product algorithm, forward error correction, LDPC, piecewise linear function approximation

ABSTRACT

This paper proposes new simplified sum-product (SSP) decoding algorithm to improve BER performance for low-density parity-check codes. The proposed SSP algorithm can replace multiplications and divisions with additions and subtractions without extra computations. In addition, the proposed SSP algorithm can simplify both the $\ln[\tanh(x)]$ and $\tanh^{-1}[\exp(x)]$ by using two quantization tables which can reduce tremendous computational complexity. Moreover, the simulation results show that the proposed SSP algorithm can improve about 0.3 ~ 0.8 dB of BER performance compared with the existing modified sum-product algorithms.

I. 서 론

LDPC(Low-Density Parity-Check) 부호는 1962년 Gallager에 의해 처음 제안되었다^[1]. 그러나 LDPC 부호는 하드웨어 기술의 한계로 인하여 1996년 MacKay와 Neal에 의해 재발견될 때까지 주목받지 못하였다^[2]. 순방향 오류 정정 (Forward

Error Correction) 기술 중 Shannon 한계에 가장 근접하는 오류 정정 능력을 보이는 LDPC 부호는 DVB-S2 (Digital Video Broadcasting - Satellite - Second Generation)^[3], WLAN(IEEE 802.11n)^[4], WiMAX (IEEE 802.16e)^[5] 등의 최신 통신 시스템 표준에 채택되어 사용되고 있으며, DVB-T2(Digital Video Broadcasting - Terrestrial - Second Generation),

※ 본 연구는 교육인적자원부 2단계 BK(Brain Korea) 21 과제와 지식경제부가 지원하는 국가 반도체 연구개발사업인 “시스템집적반도체 기반기술개발사업(시스템 IC 2010)” 그리고 반도체설계교육센터(IDECE)의 지원을 받아 수행되었습니다.

* 한국전자통신연구원 방송통신융합연구부문 위성무선융합연구부 위성방통융합연구팀 (jhan1@etri.re.kr)

** 아주대학교 정보통신대학 전자공학부 SoC 연구실(sunwoo@ajou.ac.kr)

논문번호 : KICS2008-11-525, 접수일자 : 2008년 11월 26일, 최종논문접수일자 : 2009년 3월 4일

DVB-C2(Digital Video Broadcasting - Cable - Second Generation), IMT-Advanced 이동 통신 시스템 등의 차세대 표준에서 순방향 오류 정정 부호로 채택이 활발히 논의 되고 있다⁶⁾.

LDPC 복호 알고리즘에 대한 연구는 크게 연산 복잡도를 줄이는 방법과 반복 복호 횟수를 줄이는 방법, BER (Bit Error Ratio) 성능을 높이는 방법에 대한 연구로 나누어 볼 수 있다. BP(Belief propagation) 알고리즘이라고도 불리는 SP(sum-product) 알고리즘은 가장 좋은 BER 성능을 보이고 있지만 연산 복잡도가 높은 단점이 있다^{7), 8)}. 대조적으로 MS (min-sum) 알고리즘은 연산 복잡도를 줄이는데 초점을 맞추고 있어 연산 복잡도는 낮은 반면 BER 성능은 SP 알고리즘에 비해 감소하는 단점이 있다⁹⁾.

본 논문에서는 $\ln[\tanh(x)]$ 와 $\tanh^{-1}[\exp(x)]$ 함수를 각각 하나의 양자화 테이블로 한 번에 처리할 수 있는 SSP (simplified sum-product) 알고리즘을 제안한다. 제안하는 SSP 알고리즘은 기존의 근사화 알고리즘인 modified SP 알고리즘⁸⁾과 비교해 성능 감소를 최소화하였다. 제안하는 SSP 알고리즘은 곱셈 연산과 나눗셈 연산을 덧셈 연산과 뺄셈 연산으로 대체할 수 있어 연산 복잡도를 기존의 SP 알고리즘과 비교해 현저히 줄일 수 있다.

이 논문의 구성은 다음과 같다. 2장에서는 LDPC 부호에 대해 살펴본다. 3장에서는 기존의 LDPC 복호 알고리즘에 대해 설명하고, 4장에서는 제안하는 SSP 알고리즘에 대해 설명한다. 5장에서는 기존의 SP 알고리즘과 제안하는 SSP 알고리즘과의 성능을 비교한다. 마지막으로 6장에서 결론을 맺는다.

II. LDPC 부호

(N, K) LDPC 부호는 블록 코드의 한 종류로 N 은 부호 (codeword) 비트 수를 의미하고, K 는 정보 (information) 비트 수를 의미한다. LDPC 부호는 $(N-K)$ 행과 N 열로 이루어진 패리티 검사 행렬 H 로 정의된다. 부호율 (code rate) 은 K/N 으로 정의된다. 그림 1은 $(12, 6)$ LDPC 부호의 부호율 1/2인 패리티 검사 행렬 $H_{6 \times 12}$ 을 보여준다.

LDPC 부호는 Tanner 그래프로 복호 과정에 대한 표현이 가능하다. 그림 2는 비트 노드와 체크 노드로 구성되는 Tanner 그래프를 보여주고 있다. 한 개의 비트 노드는 패리티 검사 행렬 H 의 같은 행에서 열의 값이 '1'인 체크 노드들과 연결된다. 같은 방법으로 한 개의 체크 노드는 패리티 검사

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

그림 1. $(12, 6)$ LDPC 부호의 패리티 검사 행렬

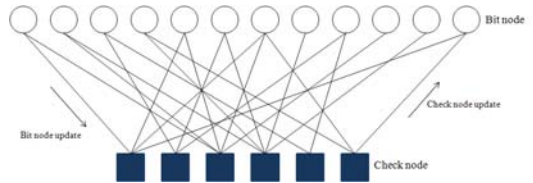


그림 2. 그림 1의 $(12, 6)$ LDPC 부호에 대한 Tanner 그래프

행렬 H 의 같은 열에서 행의 값이 '1'인 비트 노드들과 연결된다.

III. 기존의 LDPC 복호 알고리즘

LDPC 복호 알고리즘은 서론에서 언급한 것과 같이 크게 SP 복호 알고리즘과 MS 복호 알고리즘으로 분류할 수 있다. SP 복호 알고리즘은 LDPC 복호 알고리즘 중 가장 높은 BER 성능을 보여준다⁷⁾. 그러나 SP 복호 알고리즘은 하이퍼볼릭 탄젠트 연산과 역 하이퍼볼릭 탄젠트 연산이 사용되기 때문에 연산 복잡도가 매우 높다. 이러한 연산 복잡도를 줄이기 위하여 modified SP 복호 알고리즘이 제안되었다⁸⁾.

3.1 SP 복호 알고리즘

다음에 사용하게 되는 SP 복호 알고리즘의 과정 및 표기는^{6), 8)}과 유사하다. (N, K) LDPC 코드는 패리티 검사 행렬 $H_{(N-K) \times N}$ 으로 정의된다. y_n 은 채널을 통과한 수신된 n 번째의 비트 노드 부호이며, u_n 은 송신하는 n 번째의 비트 노드 부호라고 하자. $q_{n \rightarrow m}(x)$, $x \in \{0, 1\}$ 은 n 번째 비트 노드에서 m 번째 체크 노드로 보낼 때, n 번째 비트 노드의 값이 0 또는 1이 될 확률을 나타낸다. 같은 방법으로 $r_{m \rightarrow n}(x)$, $x \in \{0, 1\}$ 은 m 번째 체크 노드에서 n 번째 비트 노드로 보낼 때, m 번째 체크 노드의 값이 0 또는 1이 될 확률을 의미한다.

LDPC 부호의 SP 알고리즘 복호 과정은 크게 4

단계로 나뉘지며 첫째 초기화 단계, 둘째 체크 노드 업데이트 단계, 셋째 비트 노드 업데이트 단계. 마지막으로 판정 단계로 이루어진다.

3.1.1 초기화 단계

송신하는 부호 비트인 u_n 은 $L(u_n) = L_c \cdot y_n$ 으로 n 번째 비트 노드에 할당 된다. 이 때, $L(u_n)$ 은 사후 확률(posteriori probability)을 의미한다. LDPC로 부호화된 정보 비트가 AWGN 채널을 통과한다고 가정한다면 사후 확률의 신뢰값(reliability value) L_c 는 $2/\sigma^2$ 이 된다. 이 때, σ^2 은 잡음 분산(noise variance)을 의미한다. 패리티 검사 행렬 H 에서 $H_{m,n} = 1$ 인 모든 (m, n) 위치의 사후 확률 값은 식 (1)과 식 (2)로 초기화 된다.

$$\lambda_{n \rightarrow m}(u_n) = L(u_n) \tag{1}$$

$$A_{m \rightarrow n}(u_n) = 0 \tag{2}$$

$\lambda_{n \rightarrow m}(u_n)$ 과 $A_{m \rightarrow n}(u_n)$ 은 각각 식 (3)과 식 (4)와 같이 LLR(log-likelihood ratio)로 표현되는데 $\lambda_{n \rightarrow m}(u_n)$ 은 n 번째 비트 노드에서 m 번째 체크 노드로 전달되는 메시지를, $A_{m \rightarrow n}(u_n)$ 은 m 번째 체크 노드에서 n 번째 비트 노드로 전달되는 메시지를 의미한다..

$$\lambda_{n \rightarrow m}(u_n) = \log \left\{ \frac{q_{n \rightarrow m}(0)}{q_{n \rightarrow m}(1)} \right\} \tag{3}$$

$$A_{m \rightarrow n}(u_n) = \log \left\{ \frac{r_{m \rightarrow n}(0)}{r_{m \rightarrow n}(1)} \right\} \tag{4}$$

3.1.2 체크 노드 업데이트 단계

$N(m)$ 이 패리티 검사 행렬 H 의 m 번째 행에서 $H_{m,n} = 1$ 인, 즉, m 번째 체크 노드와 연결되어 있는 비트 노드들의 집합이라고 하자. m 번째 체크 노드에서 n 번째 비트 노드로 보내지는 $A_{m \rightarrow n}(u_n)$ 값은 식 (5)의 과정을 거쳐 업데이트하게 된다.

$$A_{m \rightarrow n}(u_n) = 2 \left\{ \prod_{n' \in N(m) \setminus n} \text{sign}[\lambda_{n' \rightarrow m}(u_{n'})] \right\} \times \tanh^{-1} \left\{ \prod_{n' \in N(m) \setminus n} \tanh \left[\frac{|\lambda_{n' \rightarrow m}(u_{n'})|}{2} \right] \right\} \tag{5}$$

이 때, $N(m) \setminus n$ 은 n 번째 비트 노드를 제외한

나머지 $N(m)$ 의 집합을 의미한다.

3.1.3 비트 노드 업데이트 단계

$M(n)$ 이 패리티 검사 행렬 H 의 n 번째 열에서 $H_{m,n} = 1$ 인, 즉, n 번째 비트 노드와 연결되어 있는 체크 노드들의 집합이라고 하자. m 번째 체크 노드에서 n 번째 비트 노드로 보내지는 $\lambda_{n \rightarrow m}(u_n)$ 값은 식 (6)의 과정을 거쳐 업데이트하게 된다.

$$\lambda_{n \rightarrow m}(u_n) = L(u_n) + \sum_{m' \in M(n) \setminus m} A_{m' \rightarrow n}(u_n) \tag{6}$$

이 때, $M(n) \setminus m$ 은 m 번째 체크 노드를 제외한 나머지 $M(n)$ 의 집합을 의미한다.

n 번째의 비트 노드의 $\lambda_n(u_n)$ 의 값은 다음 식 (7)과 같이 계산되고 이 값은 판정 단계에서 사용되게 된다.

$$\lambda_n(u_n) = L(u_n) + \sum_{m \in M(n)} A_{m \rightarrow n}(u_n) \tag{7}$$

3.1.4 판정 단계

복호된 부호어를 $d = \{d_1, d_2, d_3, \dots, d_n\}$ 이라고 하자. 이 때, $\lambda_n(u_n) \geq 0$ 이면, d_n 은 0으로 $\lambda_n(u_n) < 0$ 이면, d_n 은 1로 양자화 된다. 만약 행렬 연산 $dH^T=0$ 을 만족하면 복호 과정을 중단하고 d 를 복호된 최종 결과로 남겨주게 되고, 행렬 연산 $dH^T \neq 0$ 이면 2 단계인 체크 노드 업데이트 단계로 돌아가 $dH^T=0$ 을 만족하거나 최대 반복 복호 횟수 만큼 복호 과정을 다시 수행하게 된다.

높은 BER 성능을 보이는 SP 복호 알고리즘은 앞에서 언급한 것과 같이 하이퍼볼릭 탄젠트와 역 하이퍼볼릭 탄젠트를 이용하여 체크 노드 업데이트를 수행한다. 따라서, 4 단계의 LDPC 복호 과정 중에서 체크 노드 업데이트 과정에서 가장 높은 연산 복잡도를 필요로 한다.

3.2 Modified SP 복호 알고리즘

SP 복호 알고리즘이 가장 높은 BER 성능을 보여주고는 있지만 하이퍼볼릭 탄젠트와 역 하이퍼볼릭 탄젠트를 연산하기 때문에 높은 연산 복잡도를 필요로 한다. 연산 복잡도를 줄이기 위하여 modified SP 알고리즘이 제안되었다^[8]. modified SP 알고리즘은 하이퍼볼릭 탄젠트 함수와 역 하이퍼볼릭 탄젠트 함수를 각각 7개의 구간으로 나누고, 이렇게 나눈 7개의 구간을 대표하는 8개의 양자화 값을 선정하였다. 표 1은 $\tanh(x)$ 의 양자화 테이블을 보여주며, 표 2

표 1. $\tanh(x)$ 의 양자화 테이블

x	$\tanh(x)$
$-7 < x \leq -3$	-0.99991
$-3 < x \leq -1.6$	-0.9801
$-1.6 < x \leq -0.8$	-0.8337
$-0.8 < x \leq 0$	-0.3799
$0 < x \leq 0.8$	0.3799
$0.8 < x \leq 1.6$	0.8337
$1.6 < x \leq 3$	0.9801
$3 < x \leq 7$	0.99991

표 2. $\tanh^{-1}(x)$ 의 양자화 테이블

x	$\tanh^{-1}(x)$
$-0.99998 < x \leq -0.9951$	-3.3516
$-0.9951 < x \leq -0.9217$	-1.9259
$-0.9217 < x \leq -0.6640$	-1.0791
$-0.6640 < x \leq 0.0$	-0.3451
$0.0 < x \leq 0.6640$	0.3451
$0.6640 < x \leq 0.9217$	1.0791
$0.9217 < x \leq 0.9951$	1.9259
$0.9951 < x \leq 0.99998$	3.3516

는 $\tanh^{-1}(x)$ 의 양자화 테이블을 나타낸다 [8].

그러나 modified SP 알고리즘 [8]은 SP 알고리즘 [7]과 비교해 비교적 큰 BER 성능 저하가 발생한다. modified SP 알고리즘 [8]은 SP 알고리즘 [7]과 마찬가지로 체크 노드 업데이트시 곱셈 연산과 나눗셈 연산으로 인하여 높은 연산 복잡도를 필요로 한다.

IV. 제안하는 SSP 복호 알고리즘

BER 성능을 기존의 modified SP 알고리즘과 비교해 향상된 성능을 보이면서 연산 복잡도를 줄이기 위하여 SSP 알고리즘을 제안한다. 식 (1)에서 비트 노드의 초기값 $\lambda_{n \rightarrow m}(u_n)$ 을 식 (8)과 같이 변경하면 식 (5)의 체크 노드 업데이트에서 1비트 오른쪽 쉬프트 연산과 1비트 왼쪽 쉬프트 연산을 제거할 수 있다.

$$\lambda_{n \rightarrow m}(u_n) = \frac{L(u_n)}{2} \quad (8)$$

곱셈 연산과 나눗셈 연산을 덧셈 연산과 뺄셈 연산으로 대체하기 위해서 제안하는 SSP 알고리즘에서는 로그 함수와 지수 함수를 추가하였다. 식 (9)는 새로운 체크 노드 업데이트를 나타낸다.

표 3. 제안하는 $\ln[\tanh(x)]$ 의 양자화 테이블

x	$\ln[\tanh(x)]$
$0 \leq x < 0.5$	1.1000
$0.5 \leq x < 1.0$	0.5000
$1.0 \leq x < 1.8$	0.1250
$1.8 \leq x < 2.5$	0.0250
$2.5 \leq x < 3.0$	0.0100
$x \geq 3.0$	0.0025

표 4. 제안하는 $\tanh^{-1}[\exp(x)]$ 의 양자화 테이블

x	$\tanh^{-1}[\exp(x)]$
$0 \leq x < 0.006$	3.25
$0.006 \leq x < 0.018$	2.50
$0.018 \leq x < 0.075$	2.00
$0.075 \leq x < 0.400$	1.00
$0.400 \leq x < 1.000$	0.50
$x \geq 1.0000$	0.25

$$\Lambda_{m \rightarrow n}(u_n) = \prod_{n' \in N(m) \setminus n} \text{sign}[\lambda_{n' \rightarrow m}(u_{n'})] \times \tanh^{-1} \left(\exp \left(\sum_{n' \in N(m) \setminus n} \ln(\tanh[|\lambda_{n' \rightarrow m}(u_{n'})|]) \right) \right) \quad (9)$$

식 (9)에서와 같이 하이퍼볼릭 탄젠트 연산과 역 하이퍼볼릭 탄젠트 연산에서는 양의 값만을 필요로 하기 때문에 제안하는 SSP 알고리즘에서는 양수만을 고려하였다. 그러므로 제안하는 SSP 알고리즘은 양자화 테이블의 음수 부분을 제거하여 modified SP 알고리즘과 비교하여 양자화 테이블의 크기를 줄일 수 있다.

제안하는 SSP 알고리즘은 로그 함수와 하이퍼볼릭 탄젠트 함수를 동시에 하나의 양자화 테이블을 이용하여 양자화 한다. 그러므로 로그 함수가 추가되지만 체크 노드 업데이트에서 추가적인 연산이 필요하지 않는다. 제안하는 SSP 알고리즘은 $\ln[\tanh(x)]$ 함수를 6개의 구간으로 나누고, 각 구간에 대해 6개의 양자화 값을 이용하여 함수 연산을 수행한다. 각 구간에 대한 양자화 값은 실험적인 시뮬레이션을 통하여 선택되었다. 표 3은 제안하는 $\ln[\tanh(x)]$ 함수의 양자화 테이블을 나타낸다. 같은 방법으로 지수 함수와 역 하이퍼볼릭 탄젠트 함수를 하나의 양자화 테이블을 이용하여 양자화 하였다. 표 4는 제안하는 $\tanh^{-1}[\exp(x)]$ 함수의 양자화 테이블을 나타낸다.

V. 성능 비교

제안하는 SSP 알고리즘은 MATLAB을 이용하여 시뮬레이션을 수행하고 기존의 SP 알고리즘 [7, 8]과

성능을 비교하였다. 제안하는 SSP 알고리즘과 기존의 SP 알고리즘 [7, 8]은 WLAN 표준 [4]에 적용되어 있는 12개의 LDPC 부호를 이용하여 성능 평가를 진행하였다. 본 논문에서는 WLAN 표준에 적용되어 있는 LDPC 부호 중 $(N, K) = (648, 324)$ 와 $(1296, 864)$ 에 대한 성능 평가를 보여 준다.

부호화된 데이터 비트가 BPSK로 변조되고 AWGN 채널을 통과한다고 가정한다. LDPC 복호를 위한 반복 복호 횟수는 20번과 40번으로 제한한다. 그림 3과 그림 4는 반복 복호 횟수를 20번으로 제한하였을 때, 제안하는 SSP 알고리즘과 기존의 SP 알고리즘 [7, 8]의 BER 성능을 보여준다.

그림 3과 그림 4에서와 같이 BER 값이 10^{-3} 일 때, 제안하는 SSP 알고리즘은 기존의 높은 연산 복잡도를 보이는 SP 알고리즘 [7]과 비교해 $(648, 324)$

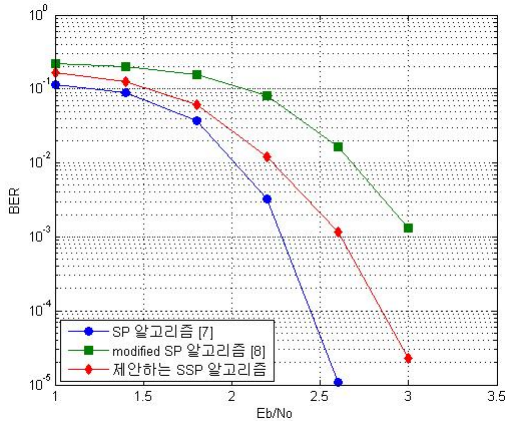


그림 3. 반복 복호 횟수를 20으로 했을 때, $(648, 324)$ LDPC 부호의 BER 성능 비교

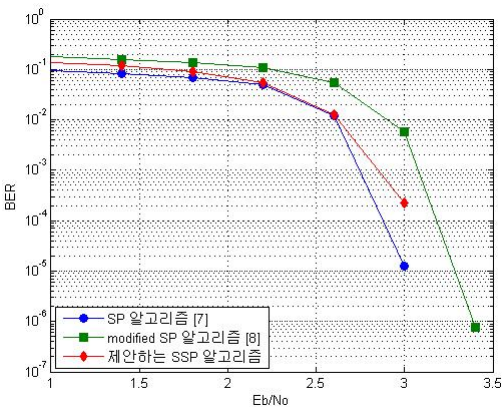


그림 4. 반복 복호 횟수를 20으로 했을 때, $(1296, 864)$ LDPC 부호의 BER 성능 비교

LDPC 부호에서 약 0.2 dB의 성능 저하가, $(1296, 864)$ LDPC 부호와 비교해 약 0.1 dB의 성능 저하가 발생하였다. 대조적으로 modified SP 알고리즘 [8]은 $(648, 324)$ 와 $(1296, 864)$ LDPC 부호에서 SP 알고리즘 [7]과 비교해 모두 약 0.5 dB의 성능 저하가 발생하였다. 그러므로 제안하는 SSP 알고리즘은 기존의 근사화 SP 알고리즘인 modified SP 알고리즘 [8]과 비교해 $(648, 324)$ 와 $(1296, 864)$ LDPC 부호에서 각각 약 0.3 dB와 0.4 dB의 성능 향상을 보인다.

같은 방법으로 그림 5와 그림 6은 최대 반복 복호 횟수가 40으로 제한되었을 때 제안하는 SSP 알고리즘과 기존의 SP 알고리즘간의 BER 성능 비교를 나타낸다.

그림 5에서와 같이 BER 값이 10^{-3} 일 때, 제안하는 SSP 알고리즘은 $(648, 324)$ LDPC 부호에서 SP

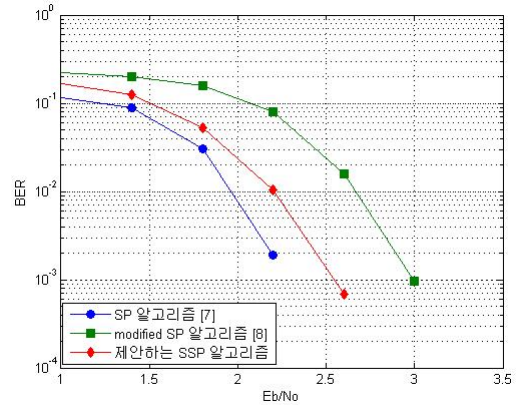


그림 5. 반복 복호 횟수를 40으로 했을 때, $(648, 324)$ LDPC 부호의 BER 성능 비교

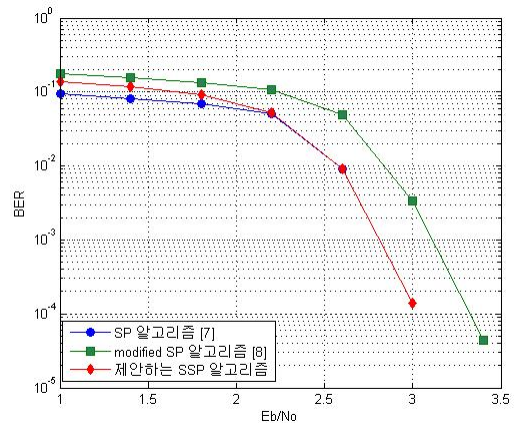


그림 6. 반복 복호 횟수를 40으로 했을 때, $(1296, 864)$ LDPC 부호의 BER 성능 비교

표 5. (648, 324) LDPC 부호의 체크 노드 업데이트에서 연산 복잡도 비교

	SP 알고리즘 [7]	modified SP 알고리즘 [8]	제안하는 SSP 알고리즘
곱셈 연산	76	76	0
나눗셈 연산	88	88	0
덧셈 연산	0	0	76
뺄셈 연산	0	0	88

알고리즘 [7]과 비교해 약 0.3 dB의 성능 저하를 보인다. 특히 그림 6에서 보는 것과 같이 (1296, 864) LDPC 부호에서는 기존의 SP 알고리즘 [7]과 유사한 성능을 보인다. 대조적으로 modified SP 알고리즘 [8]은 SP 알고리즘 [7]과 비교해 (648, 324) LDPC 부호에서는 약 0.8 dB, (1296, 864) LDPC 부호에서는 약 0.3 dB의 성능 저하를 보인다. 그러므로 제안하는 SSP 알고리즘은 modified SP 알고리즘 [7]과 비교해 (648, 324) LDPC 부호와 (1296, 864) LDPC 부호에서 각각 0.5 dB, 0.3 dB의 BER 성능 향상을 보여준다.

반복 복호 횟수를 20회와 40회로 제한하였을 때, 모든 경우에서 modified SP 알고리즘 [8]과 비교하여 BER 성능 향상을 보인다. 또한, 성능 평가 결과 반복 복호 횟수를 늘릴수록 modified SP 알고리즘과 비교하여 BER 성능이 향상된다.

표 5는 WLAN 표준 [4]에 적용되어 있는 (648, 324) LDPC 부호에서 기존의 SP 알고리즘과 제안하는 SSP 알고리즘의 연산 복잡도를 비교한 표이다. 표 5에서 보는 것과 같이 제안하는 SSP 알고리즘은 기존의 SP 알고리즘 [7, 8]과 비교해 곱셈 연산과 나눗셈 연산을 덧셈 연산과 뺄셈 연산으로 대체가 가능하여 연산 복잡도를 줄일 수 있다.

VI. 결 론

본 논문에서는 LDPC 부호를 위한 새로운 SSP 복호 알고리즘을 제안하였다. 제안하는 SSP 알고리즘은 로그 함수와 지수 함수를 추가하여 기존 SP 알고리즘 [7, 8]의 체크 노드 업데이트 연산에서 사용하는 곱셈 연산과 나눗셈 연산을 덧셈 연산과 뺄셈 연산으로 대체가 가능하다. $\ln[\tanh(x)]$ 연산과 $\tanh^{-1}[\exp(x)]$ 연산을 위한 6개의 구간과 6개의 양자화 값을 가지는 2개의 양자화 테이블을 제안하였

다. 제안하는 SSP 알고리즘은 $\ln[\tanh(x)]$ 연산과 $\tanh^{-1}[\exp(x)]$ 연산 없이 체크 노드 업데이트가 가능하다. 또한, 제안하는 SSP 알고리즘은 기존의 SP 알고리즘 [7, 8]과 비교하여 연산 복잡도를 현저히 줄일 수 있으며 modified SP 알고리즘 [8]과 비교해 BER 성능 저하를 줄일 수 있다. 시뮬레이션 결과 제안하는 SSP 알고리즘은 modified SP 알고리즘 [8]에 비해 최대 0.8 dB의 BER 성능을 향상시켰다.

참 고 문 헌

- [1] R. G. Gallager, "Low-Density Parity-Check Codes," *IRE Trans. Inform. Theory*, vol.IT-8, pp. 21-28, Jan. 1962
- [2] D. J. MacKay, "Good error correcting codes based on very sparse matrices," *IEEE Trans. Inform. Theory*, vol. 45, pp. 399-431, Mar. 1999
- [3] ETSI, "Digital Video Broadcasting (DVB); Second generation framing structure, channel coding and modulation systems for broadcasting, interactive services, news gathering and other broadband satellite applications," *ETIS EN 302 307 v1.1.2*, Jun. 2006
- [4] IEEE, "Draft STANDARD Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications: Enhancement for Higher Throughput," *IEEE 802.11nTM/D2.00*, Feb. 2007
- [5] IEEE, "IEEE Standard for Local and metropolitan area networks Part 16: Air Interface for Fixed Broadband Wireless Access Systems," *IEEE Std 802.16TM-2004*, Oct. 2004
- [6] 한국과학기술정보연구원, "LDPC," 2005 Tech-Issue Emerging S&T Report. 2005년 12월
- [7] J. Chen, A. Dholakia, E. Eleftheriou, M. P. C. Fossorier and X. Hu, "Reduced-Complexity Decoding of LDPC Codes," *IEEE Trans. Commun.*, vol. 53, pp. 1288-1299, Aug. 2005
- [8] S. Papaharalabos, P. Sweeney, B. G. Evans, P. T. Mathiopoulos, G. Albertazzi, A. Vanelli-Coralli and G. E. Corazza, "Modified sum-product algorithms for decoding low-density parity-check codes," *IET Commun.*, vol. 1. pp. 294-300, Jun. 2007
- [9] J. Hagenauer, E. Offer and L. Papke, "Iterative Decoding of Binary Block and Convolutional Codes," *IEEE Trans. Info. Theory*, vol. 42, pp. 429-445, Mar. 1996

한 제 희 (Jae Hee Han)

준회원



2007년 2월 아주대학교 전자공학부 졸업

2009년 2월 아주대학교 전자공학과 석사

2009년 3월~현재 한국전자통신연구원 (ETRI)

<관심분야> 위성통신 및 케이

블 통신 모듈 동기부, LDPC 복호기, SoC 설계

선우명훈 (Myung Hoon Sunwoo)

중신회원



1980년 2월 서강대학교 전자공학과 졸업

1982년 2월 한국과학기술원 전자공학과 석사

1982년 3월~1985년 8월 한국전자통신연구원(ETRI)

1985년 9월~1990년 8월 Univ.

of Texas at Austin 전자공학과 박사

1992년 8월~1996년 10월 아주대학교 전기전자공학부 조교수

1996년 10월~2001년 9월 아주대학교 전자공학부 부교수

2001년 10월~현재 아주대학교 전자공학부 교수

<관심분야> VLSI 및 Parallel Architecture, 통신 멀티미디어용 DSP 칩 및 SoC 설계