

3개의 직렬 MAP 복호기를 이용한 반복 터보 복호화기

정희원 김 동 원*, 이 호 응**, 강 철 호***

Iterative Turbo Decoding Using Three Cascade MAP Decoder

Dong-Won Kim*, Ho-Woong Lee**, Chul-Ho Kang*** *Regular Members*

요 약

반복 복호 알고리즘에 의해 복호화된 터보 코드는 가산성 백색 가우시안 잡음(AWGN) 채널 환경에서 이론적으로 Shannon의 한계에 근접한 뛰어난 코딩 이득을 나타내는 것으로 보여지고 있다. 그러나, 터보 코드의 성능은 터보 부호화기에서 프레임의 크기 즉, 인터리버의 크기에 의존한다. IMT-2000과 같은 이동 통신 채널 환경에서 음성 전송하는 경우에는 터보 코드의 프레임 크기는 매우 작다. 그리고, 그것은 터보 코드의 성능을 떨어뜨리는 직접적인 원인이 된다. 본 논문에서는 차세대 이동 통신 시스템에서 프레임 크기가 작은 음성 프레임을 이용하여 터보 코드의 성능을 검증하며, 작은 프레임 크기에 알맞은 3개의 직렬 MAP(Maximum A Posteriori probability) 복호기를 이용한 반복 복호의 터보 코드를 제안하고 부호율 1/3, 구속장의 길이 3 또는 4, 프레임 크기 24, 192 비트에 대하여 컴퓨터 모의실험을 통해 터보 코드의 성능을 분석한다.

ABSTRACT

Turbo codes, decoded through an iterative decoding algorithm, have recently been shown to yield remarkable coding gain close to theoretical Shannon's limit in the Additive White Gaussian Noise(AWGN) channel environment. But the performance of turbo codes depends on the frame size of turbo encoder, i. e, interleaver size. In the case of voice transmission in a mobile communication environment, its frame sizes are very small and result in degrading the performance of turbo codes. In this paper has analyzed the performance of turbo code which is based on voice frames with short frame sizes in the future mobile communication system. So we propose of turbo codes based on iterative decoding using three cascade MAP(Maximum A Posteriori probability) decoder suitable for small frame sizes and analyzed its performance with a rate 1/3, constraint length $v=3$ and 4, frame size of 24 and 192 bits by computer simulation.

I. 서론

이동 통신 채널 환경에서는 채널의 특성상 데이터를 전송하는 경우 여러 가지 원인에 의해 오류가 발생할 확률이 높다. 그러므로, 이동 통신 채널 환경에서 이용되는 채널 부호화 기술은 매우 중요하게 인식되고 있다. 이제까지 주로 이용된 채널 부호화 기술로는 부호기단에서는 convolutional 코드, 복호기 단에서는 Viterbi가 사용되었는데, 최근에는 18

번의 반복 복호시 0.7dB라는 낮은 신호 대 잡음비에서 $P_b=10^{-5}$ 의 획기적인 반복 복호 성능 때문에 터보 코드가 여러 분야에서 관심의 대상이 되고 있다. 터보 코드는 1993년 Berrou, Glavieux, Thitimajshima에 의하여 제안된 길쌈 부호를 병렬로 연결시킨 부호로서 프레임의 크기 즉, 인터리버(interleaver)의 크기가 큰 경우에 Shannon의 한계에 근접하는 우수한 성능을 갖는다고 알려져 있다^[1].

* 삼성전자 디지털영상사업부(dongdog7@samsung.co.kr),

** 동원대학 정보통신과(hwlee@tongwon.ac.kr),

*** 팽문대학교 전자통신공학과

논문번호 : 00307-0731, 접수일자 : 2000년 7월 31일)

이와 같은 터보 코드의 구조를 살펴보면, 부호화기는 두 개의 재귀 조직 길쌈 부호(RSC:Recursive Systematic Convolutional)와 인터리버(interleaver)로 구성되어 있다. RSC 부호는 조직 길쌈부호(systematic convolutional code)에 궤환(feedback)이 첨가된 형태이고, 인터리버는 긴 버스트(burst)에러를 매우 작은 버스트 에러로 변환하는 역할을 수행한다. 터보 복호화기는 부호화기의 두 RSC 부호에 대한 복호화기가 직렬로 연결되어 있으며, 각각의 복호화기에서는 MAP(Maximum A Posteriori probability) 복호 알고리즘 및 SOVA (Soft Output Viterbi Algorithm) 복호 알고리즘을 이용하여 복호를 수행한다. MAP 알고리즘은 이산 무기억 채널을 통하여 전송된 Markov source의 상태와 천이에 대한 사후 확률(A Posteriori Probability, APP)을 얻기 위하여 1974년 Bahl 등에 의해서 처음으로 제안되었으며^[2], 잡음이 첨가된 수신 신호로부터 사후 확률을 계산하는 알고리즘으로 각 수신된 비트의 최대 근사화 비(Logarithm of Likelihood Ration, LLR)를 계산하기 위해 매우 복잡한 계산이 필요하였다. 그러나, Pietrobon이 1994년에 MAP 알고리즘의 계산량을 줄인 간단한 복호 알고리즘을 제안하였다^{[3][4]}.

터보 코드는 인터리버의 크기가 큰 경우에 대해서는 성능이 우수한 것으로 알려져 있는데, 이동 통신채널 환경 등에서 2kbit/sec 정도의 저속 데이터 통신과 음성 통신의 신호를 전송할 경우에는 프레임의 크기가 너무 작아서 터보 코드를 사용하는 경우 월등한 성능 향상을 볼 수가 없었다.

본 논문에서는 이러한 성질을 개선하기 위하여 기존의 터보 부호화기 단은 그대로 사용하면서 복호화기 단에 MAP 복호화기를 하나 더 연결하여 IS-95^[5]에서 제안되어 사용되고 있는 프레임과 같은 크기의 인터리버인 24 및 192인 경우 부호율 1/3인 터보 코드에 적용하였고, 기존의 논문^[6]에 비해 구조가 간단하고 컴퓨터 모의실험을 통해 그 성능을 검증하였다. 본 논문은 다음과 같이 구성된다. II장에서는 기존 터보 코드의 부호기 개념과 Pietrobon에 의해 간략화된 MAP 알고리즘에 대해 살펴본다. III장에서는 본 논문에서 제안한 3개의 직렬 MAP 복호기를 이용한 반복 터보 코드를 살펴본다. IV장에서는 제안한 방식에 대하여 AWGN채널 환경에서 컴퓨터 모의실험을 통하여 성능을 분석한다. 마지막으로 V장에서는 결론을 맺도록 한다.

II. Turbo 코드의 기본이론

1. Turbo 코드의 부호화기

1993년 Berrou 등에 의해서 제안된 터보 코드의 구조는 부호율이 1/2로 만들어 전송하는 구조로 프레임 크기가 65,536인 인터리버를 사용하였고, 여기에 MAP 알고리즘을 이용하여 반복적인 복호를 수행함으로써 Shannon의 한계에 근접하는 성능을 보여 주었다^[1].

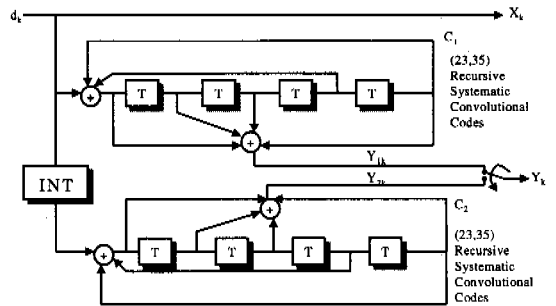


그림 1. 부호율 1/2인 터보 코드

그림 1의 부호화기 구조에서 C_1 과 C_2 는 각각 구성 부호의 생성 다항식을 나타내며, 평행링 작용에 의해 Y_{1k} 와 Y_{2k} 를 교대로 전송한다. 터보 코드의 부호화기는 N 정보 비트의 프레임으로 이루어진 입력을 이용하여 패리티 심볼을 만드는 두 개의 비교적 간단한 재귀 조직 길쌈 부호(RSC)를 병렬로 연결한 구조로 되어 있으며, 부호율 및 오류 정정 능력을 고려하여 원하는 만큼의 구성 부호를 둘 수 있다. 인터리버는 두 번째 재귀 조직 길쌈부호의 입력에 들어가는 정보 비트의 순서를 바꾸어 주는 역할을 하며, N 정보 비트의 프레임과 동일한 크기를 가진다. 따라서, 터보 코드의 출력은 재귀 조직 길쌈 부호의 출력뿐만 아니라 인터리버를 통해 변형된 출력으로 인해 상관관계(correlation)가 작아지는 이중의 패리티 정보를 지니게 된다.

2. Turbo 코드의 복호화기

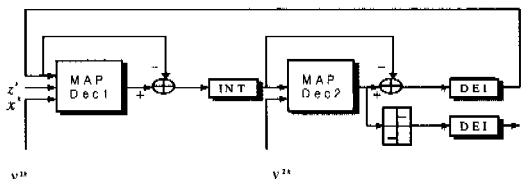


그림 2. 터보 코드의 복호화기 구조

여기에서,

- x_k : systematic data(information sequence),
- y_{1k}, y_{2k} : parity(redundancy) sequence,
- z_k : extrinsic information,
- d_k : decoded bit

비트 오류 확률(Bit Error Probability)을 최소화하는 알고리즘을 MAP 알고리즘이라 부르며, 연성 출력을 이용해서 연판정(soft decision)된 출력값을 생성한다. 이때 생성된 값은 정확한 판정에 대한 척도로서 사후 확률이라 부른다. 재귀 조직 길쌈 부호의 출력값을 d_k (부호화 되지 않은 정보 비트)와 Y_k (부호화된 패리티 비트)라 하고, 이 출력 값이 BPSK 변조되어 가산 백색 가우시안 잡음(Additive White Gaussian Noise, AWGN) 채널을 통해 수신된 결과를 $R_k=(x_k, y_k)$ 라 하면, 최대 근사화 비 $L(d_k)$ 는 다음의 수식과 같이 정의할 수 있다.

$$L(d_k) = \log \frac{\Pr(d_k=1 | R_k^N)}{\Pr(d_k=0 | R_k^N)} \quad (1)$$

위 식에서 분모항과 분자항은 각각 1, ..., N 까지 수신된 모든 심볼에 대해 $d_k=1$ 인 확률과 $d_k=0$ 인 확률을 나타내며, 경판정(hard decision)의 경우 $L(d_k)$ 값이 0보다 작으면 k번째 데이터를 0으로 판정하고, 0보다 크면 k번째 데이터를 1로 판정한다.

재귀 조직 길쌈 부호의 구성장의 길이를 v 라 하면, 부호기의 상태는 $m=0, 1, \dots, 2^v-1$ 까지 존재하며, $\lambda_k^i(m)$ 을 아래와 같이 정의하면 (1)식은 (2)식으로 표현할 수 있다.

$$\lambda_k^i(m) = \Pr(d_k=i, S_k=m | R_k^N)$$

$$L(d_k) = \log \frac{\sum_m \lambda_k^1(m)}{\sum_m \lambda_k^0(m)} \quad (2)$$

계속해서 $\alpha_k^i(m)$ 과 $\beta_k^i(m)$ 을 다음과 같이 정의하면, (2)식은 (3)식으로 표현된다.

$$\alpha_k^i(m) = \Pr(d_k=i, S_k=m, R_k^k)$$

$$\beta_k^i(m) = \Pr(R_{k+1}^N | d_k=i, S_k=m)$$

$$L(d_k) = \log \frac{\sum_m \alpha_k^1(m) \beta_k^1(m)}{\sum_m \alpha_k^0(m) \beta_k^0(m)} \quad (3)$$

위에서 정의한 $\alpha_k^i(m)$ 를 전개하여 풀어쓰면 식(4)로 표현되며, 이 값은 시간 k를 기준으로 과거의 상태값 $\alpha_k^i(m)$ 를 사용한 순방향 함수(forward recursive function)를 나타낸다.

$$\alpha_k^i(m) = \frac{1}{2} \Pr(R_k | d_k=i, S_k=m) \sum_{j=0}^1 \alpha_{k-1}^j(S_b^j(m)) \quad (4)$$

마찬가지로 $\beta_k^i(m)$ 를 전개하여 풀어쓰면 (5)식으로 표현되며, 이 값은 시간 k를 기준으로 미래 상태값 $\beta_k^i(m)$ 를 사용한 역방향 함수(backward recursive function)를 나타낸다.

$$\beta_k^i(m) = \frac{1}{2} \sum_{j=0}^1 \beta_{k+1}^j(S_b^j(m)) \Pr(R_{k+1} | d_{k+1}=j, S_{k+1}=S_b^j(m)) \quad (5)$$

MAP 알고리즘이란 전체 수신된 신호 프레임을 관찰한 후 임의의 시점에서 정보 비트가 0일 확률과 1일 확률을 비교하여 확률이 더 큰 값을 복호값으로 선택하는 알고리즘이다. 즉, MAP 복호기의 출력 값으로 다음과 같이 최대 근사화 비(LLR)를 생성하며, 최대 근사화 비를 구하는 과정은 식(6)과 같다. 여기에서, 순방향 상태 메트릭, 역방향 상태 메트릭, 그리고 가지 메트릭 연산식은 다음과 같이 표현할 수 있다.

$$\Lambda_k = \log \frac{\sum_m \alpha_k^1(m) \beta_k^1(m)}{\sum_m \alpha_k^0(m) \beta_k^0(m)} \quad (6)$$

$$\alpha_k^i(m) = \delta_i(R_k, m) \sum_{j=0}^1 \alpha_{k-1}^j(S_b^j(m)) \quad (7)$$

$$\beta_k^i(m) = \sum_{j=0}^1 \beta_{k+1}^j(S_b^j(m)) \delta_j(R_{k+1}, S_b^j(m)) \quad (8)$$

$$\delta_i(R_k, m) = \exp[-\frac{2}{\sigma^2} (x_k i + y_k Y_k(i, m))] \quad (9)$$

최대 근사화 비를 구하기 위한 위의 식 (6), (7), (8), (9)에는 곱하기, 지수 함수, 로그 함수 등의 복잡한 연산이 포함되어 있다. 따라서, E 연산자를 아래 식 (10)와 같이 정의하고, 이를 이용해서 식 (6), (7), (8), (9)를 변환하여 새로운 식 (11), (12), (13), (14)를 살펴보면 곱하기, 지수 함수, 로그 함수 등의 연산이 단순한 더하기 빼기와 E 함수의 호출로 변환되었음을 알 수 있다. E 함수는 그 값의 특성 상 작은 크기의 look-up 테이블로 구성할 수 있다.

$$x E y = -\frac{1}{L_c} \ln(e^{-L_c x} + e^{-L_c y}) \quad (10)$$

$$D_i(R_k, m) = x_k i + y_k Y_k(i, m) \quad (11)$$

$$A_k^i(m) = D_i(R_k, m) + \sum_{j=0}^i A_{k-1}^j(S_j^i(m)) \quad (12)$$

$$B_k^i(m) = \sum_{j=0}^i [B_{k+1}^j(S_j^i(m)) + D_j(R_{k+1}, S_j^i(m))] \quad (13)$$

$$\Lambda_k = \sum_{m=0}^{2^i-1} [A_k^i(m) + B_k^i(m)] + \sum_{m=0}^{2^i-1} [A_k^0(m) + B_k^0(m)] \quad (14)$$

위에서 살펴봤던 E 연산자를 좀 더 자세히 전개하면 다음과 같고 E 함수 부분을 작은 look-up 테이블을 이용하여 연산에 사용하면 Log-MAP이 되며, 마지막 식의 뒷부분이 "0"이 되면 Max-Log-MAP(Sub-MAP)이 된다.

$$\begin{aligned} x E y &= -\frac{1}{L_c} \ln(e^{-L_c x} + e^{-L_c y}) \\ &= -\frac{1}{L_c} \ln(e^{-L_c x} (1 + e^{-L_c y + L_c x})) \\ &= x - \frac{1}{L_c} \ln(1 + e^{L_c(x-y)}) \\ &= y - \frac{1}{L_c} \ln(1 + e^{L_c(y-x)}) \\ &= \min(x, y) - \frac{1}{L_c} \ln(1 + e^{-L_c|y-x|}) \end{aligned} \quad (15)$$

III. 제안한 3개의 직렬 MAP 복호화기

일반적인 연쇄 부호는 첫 번째 복호기의 신뢰도 값을 이용하여, 두 번째 복호기에서 복호를 하게 되면, 복호의 과정은 종료된다. 첫 번째 복호기가 두 번째 복호기에서 출력된 신뢰도 값을 이용하게 된다면, 그만큼의 성능 향상을 기대할 수 있다. 일반적인 터보 코드의 구조가 이러한 반복 복호 알고리즘을 사용하는 형태이다. 따라서, 이러한 터보 코드의 반복적인 복호 알고리즘을 이용하여 성능 향상을 기대할 수 있다. 이때, 첫 번째 복호기에서 두 번째 복호기로 전환되는 두 번째 복호기의 신뢰도 값을 부가 정보라고 한다. 부호화된 정보를 복호화할 때, 수신단에서 부호화된 정보를 이상적으로는 한 번에 복호를 수행해야 하지만, 실질적으로는 거의 불가능한 일이다. 일반적인 연쇄 부호는 두 번째 단에서 부호화된 정보를 복호하고 복호된 그 정보를 사용하여 첫 번째 단을 복호하게 되면 복호의

과정은 종료된다. 이때, 첫 번째 단의 복호된 정보를 다시 두 번째 단의 복호에 사용하는 반복 복호 알고리즘을 사용한다면 그만큼의 성능 향상이 이루어지게 된다. 즉, 수평적인 부호기에 입력된 정보를 먼저 복호하여 그 연관정 값을 다음 단에서 사용하여 순서가 재배열된 부호를 다시 연관정 한다. 제안된 터보 코드의 반복적인 복호는 3차원으로 생각하여 복호를 할 수 있다. 이 3차원의 정보를 이상적으로는 한번에 복호를 해야 하지만, 실질적으로는 거의 불가능한 일이므로 먼저 첫 번째 부호를 복호하고 두 번째 부호를 복호한 후 마지막으로 세 번째 부호를 복호한다. 이러한 사항을 고려하여, 본 논문에서 사용한 부호화기와 복호화기 블록도는 그림 3과 그림 4와 같다. 여기서는 소단원에 관한 내용을 간단히 살펴본다.

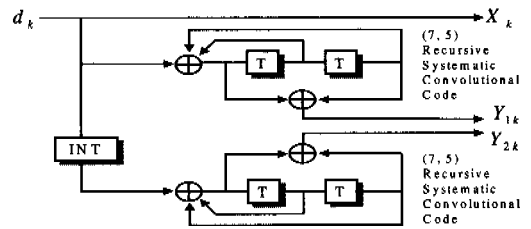


그림 3. 부호율 1/3인 터보 코드

그림 4에서 각각의 MAP 복호기는 각 정보 비트 d_k 에 대한 최대 근사화 비를 생성하며, 정보 비트 d_k 에 대한 a priori 값 $L_a(d_k)$, 채널의 신뢰도 값(reliability value)을 L_c 라 하면, 부호화된 비트가 채널을 통과한 후의 값 $L_c x_k$ 와 이전의 반복 복호에서 출력된 외부 정보 $L_e(d_k)$ 을 이용하여 복호를 수행하게 된다

$$L(d_k) = L_c x_k + L_a(d_k) + L_e(d_k) \quad (16)$$

첫 번째 복호화기에서는 a priori 값이 없으므로, 즉 $L_a(d_k) = 0$ 이 된다.

$$L(d_k) = L_c x_k + L_e D_1(d_k) \quad (17)$$

그리고, 두 번째 복호화기 (D_2)에서는 첫 번째 복호화기 (D_1)로부터의 a priori 정보를 가지므로 아래와 같은 출력을 가지게 된다.

$$L(d_k) = L_c x_k + L_e D_1(d_k) + L_e D_2(d_k) \quad (18)$$

세 번째 복호화기 (D_3)의 수직적 반복 복호가 수행된 후의 최대 근사화 비는 아래 식과 같다.

$$L(d_k) = L_c x_k + L_e D_1(d_k) + L_e D_2(d_k) + L_e D_3(d_k) \quad (19)$$

여기에서, 세 번째 복호화기에서 출력된 부가 정보 $L_e D_3(d_k)$ 는 다음 단계의 반복 복호에서 첫 번째 복호화기의 a priori 값 $L_a(d_k)$ 가 된다.

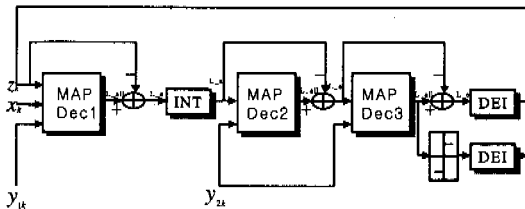


그림 4. 제안한 터보 복호화기의 블록도

IV. 컴퓨터 시뮬레이션을 이용한 성능분석

본 논문에서 제안한 방식은 구속장의 길이 3 또는 4 즉, 구성 부호의 생성 다항식은 (7, 5) 또는 (17,15)일 때 음성과 같은 저속의 데이터에 적합한 부호율이 1/3인 터보 코드를 모의 실험하였다. 블록 크기는 IS-95^[5]에서 제안되어 사용되고 있는 프레임과 같은 크기의 인터리버인 24 및 192인 경우를 고려하였고, 채널 조건은 AWGN을 적용하였고, 변조 방식은 BPSK, 인터리버는 랜덤 인터리버를 사용하였고, 터보 코드 복호화기의 반복 복호 횟수는 3회로 하였다^[7]. 이것은 기존의 터보 코드의 반복 복호 횟수가 5~6회를 수행하여 최대한의 이점을 보인 것에 비하면 거의 반으로 반복 복호 횟수를 줄인 것으로, 기존의 MAP 복호기에 1단을 더 첨가하여 반복 복호 3회에서도 충분히 효과를 낼 수 있음에 기인한다. 즉, 부호율이 1/3, 반복 복호 3회의 터보 코드에 대해 랜덤 인터리버, 채널 환경은 평균이 0 이고 분산이 $N_0/2$ 인 AWGN 채널을 모델링하여 송신 신호에 잡음이 첨가되도록 하였고, 각각의 E_b/N_0 [dB]에 대한 복호화기의 비트 오류율(BER)과 프레임 오류율(FER)의 성능을 비교·분석 하였다. 컴퓨터 모의실험에서 사용한 파라미터는 표 1과 같다.

IS-95에서 제안되어 사용되고 있는 방식인 프레임 크기 24, 192에 대하여 구속장의 길이를 3과 4

표 1. 모의 실험에 사용된 파라미터

부 호 율	1/3
프레임의 크기	24, 192
생성다항식	(7,5), (17,15)
채널 환경	AWGN
인터리버 종류	랜덤 인터리버
반복 복호 횟수	3

로 변화시켜가면서 실험을 행하여 기존의 방식인 구속장 길이 9인 경우의 convolutional 코드와 비교 하였다. 또, 터보 코드에서 가장 널리 사용되고 있는 방식인 구속장 길이 5와 비교하여 제안한 방식과 성능 차이를 계산하였다.

프레임의 크기와 동일한 인터리버 크기인 24 비트의 경우에 그림 5와 그림 9에서 BER의 성능을 제안 한 방식과 비교하였다. 그림 5에서 구속장의 길이가 3인 경우에 convolutional 코드와 비교하면, 제안한 방식이 BER 10^{-4} 에서 0.11[dB]정도 값이 떨어졌지만, 구속장의 길이가 5인 터보 코드와 비교하여 보면 BER 10^{-3} 에서 1.7[dB]정도 값이 우수하였다. 그림 9의 구속장의 길이가 4인 경우에 convolutional 코드와 비교하면, BER 10^{-4} 에서 0.65 [dB]정도 값이 우수하였다. 구속장의 길이가 5인 터보 코드와 비교하여 보면 BER 10^{-3} 에서 약 2.0[dB]정도 값이 우수하였다.

동일한 환경에서 그림 6과 그림 10은 FER의 성능을 비교하였다. 그림 6에서 구속장의 길이가 3인 경우에 convolutional 코드와 비교하면, FER 10^{-3} 에서 0.5[dB]정도 값이 떨어졌지만, 구속장의 길이가 5인 터보 코드와 비교하여 보면, FER 10^{-2} 에서 1.8[dB]정도 값이 우수하였다. 그림 10의 구속장의 길이가 4인 경우에 convolutional 코드와 비교하면, FER 10^{-4} 에서 0.4[dB]정도 값이 떨어졌지만, 구속장의 길이가 5인 터보 코드와 비교하여 보면 FER 10^{-2} 에서 약 2.25[dB]정도 값이 우수하였다.

프레임의 크기와 동일한 인터리버 크기인 192 비트의 경우는 그림 7와 그림 11에서 BER의 성능을 제안한 방식과 비교하였다. 그림 7에서 구속장의 길이가 3인 경우에 convolutional 코드, 구속장의 길이가 5인 터보 코드와 비교하면, 제안한 방식이 BER 10^{-4} 에서 1.29[dB], 1.07[dB]정도 값이 우수하였다. 그림 11의 구속장의 길이가 4인 경우에 convolutional 코드, 구속장의 길이가 5인 터보 코드

와 비교하면, BER 10^{-4} 에서 1.3[dB], 1.01[dB]정도 값이 우수하였다.

동일한 환경에서 그림 8와 그림 12는 FER의 성능을 비교하였다. 그림 8에서 구속장의 길이가 3인 경우에 convolutional 코드와 비교하면, FER 10^{-2} 에서 1.25[dB]정도 값이 우수하지만, 구속장의 길이가 5인 터보 코드와 비교하여 보면, FER 10^{-3} 에서 0.5[dB]정도 값이 우수하였다. 그림 12의 구속장의 길이가 4인 경우에 convolutional 코드와 비교하면, FER 10^{-2} 에서 1.1[dB]정도 값이 우수하였지만, 구속장의 길이가 5인 터보 코드와 비교하여 보면 FER 10^{-3} 에서 약 0.2[dB]정도 값이 우수하였다.

Frame size: 192[bits], Code rate: 1/3, AWGN, Iterations: 3

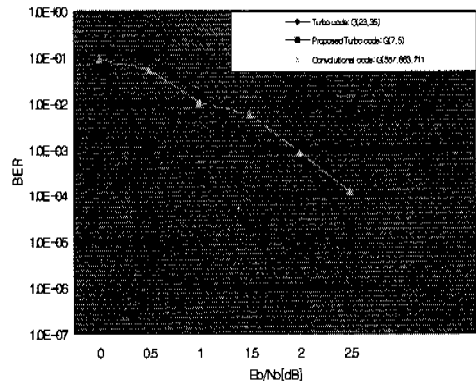


그림 7. AWGN 채널에서 프레임 크기가 192 비트인 BER 성능

Frame size: 24[bits], Code rate: 1/3, AWGN, Iterations: 3

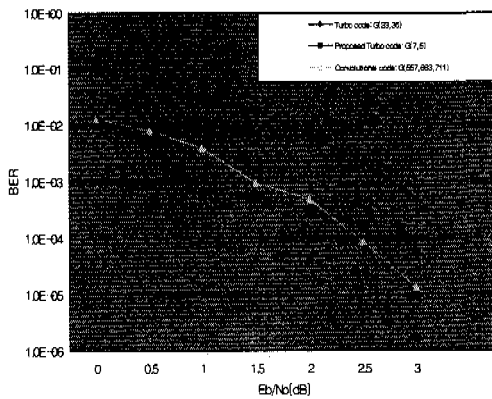


그림 5. AWGN 채널에서 프레임의 크기가 24 비트인 BER 성능

Frame size: 192[bits], Code rate: 1/3, AWGN, Iterations: 3

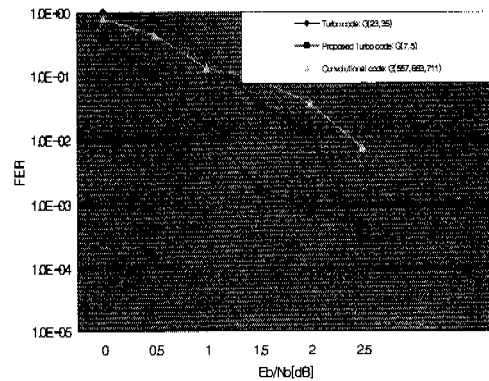


그림 8. AWGN 채널에서 프레임 크기가 192 비트인 FER 성능

Frame size: 24[bits], Code rate: 1/3, AWGN, Iterations: 3

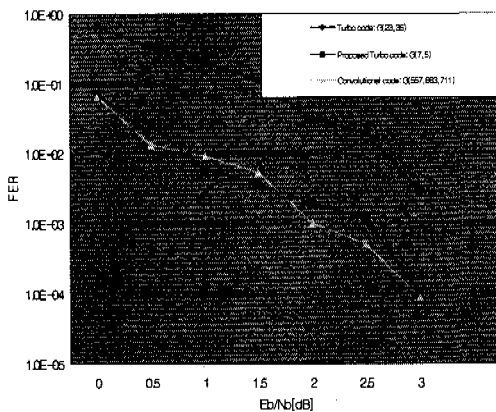


그림 6. AWGN 채널에서 프레임 크기가 24 비트인 FER 성능

Frame size: 24[bits], Code rate: 1/3, AWGN, Iterations: 3

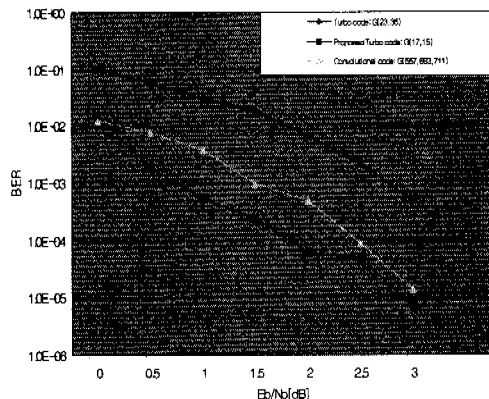


그림 9. AWGN 채널에서 프레임 크기가 24 비트인 BER 성능

Frame size: 24[bits], Code rate: 1/3, AWGN, iterations: 3

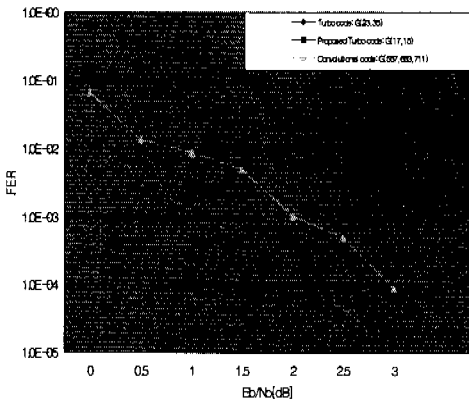


그림 10. AWGN 채널에서 프레임 크기가 24 비트인 FER 성능

Frame size: 192[bits], Code rate: 1/3, AWGN, iterations: 3

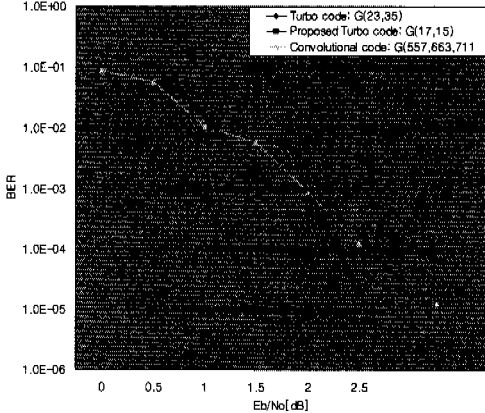


그림 11. AWGN 채널에서 프레임 크기가 192 비트인 BER 성능

Frame size: 192[bits], Code rate: 1/3, AWGN, iterations: 3

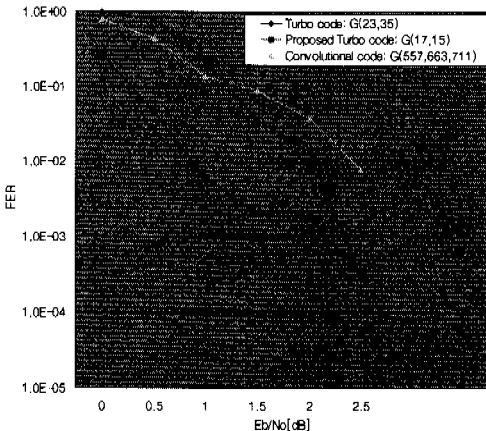


그림 12. AWGN 채널에서 프레임 크기가 192 비트인 FER 성능

V. 결론

본 논문에서는 MAP 복호기를 하나 더 첨가시키지만, 구속장의 길이는 줄이면서 널리 사용되고 있는 방식인 convolutional 코드와 구속장 5인 터보 코드와 비교하여 실험을 해본 결과, 프레임 크기가 24 비트인 경우에 구속장의 길이가 3일 때는 convolutional 코드보다 성능이 떨어졌지만, 구속장의 길이를 4로 실험을 했을 때는 성능이 우수하였다. 그리고, 구속장의 길이 5인 터보 코드에 비해서는 성능이 훨씬 더 우수하였다. 프레임 크기를 증가시켜 192 비트로 실험을 했을 때는 convolutional 코드와 구속장의 길이 5인 터보 코드보다 성능이 우수하였지만, FER 성능에서는 구속장의 길이 5인 터보 코드가 제안한 방식보다 성능이 약간 떨어졌다. 즉, 프레임 크기가 192 비트인 경우에는 제안한 방식은 모든 면에선 비교적 우수한 성능을 나타내었지만, FER 성능에서는 BER 성능에서 보여줬던 것만큼의 뛰어난 성능을 나타내지 못하는 단점이 발생하였다.

모의 실험에서 제안한 방식은 MAP 복호기의 첨가로 반복 복호 횟수가 3회에서 충분한 성능을 발휘하지만, 기존의 널리 사용되고 있는 방식인 구속장 길이 5인 터보 코드는 반복 복호를 5~6회에서 더 최적의 성능을 나타내는데, 반복 복호 횟수를 제안한 방식과의 비교를 위하여 3회로 제한시킴으로써 성능 저하 원인이 다르리라 본다.

앞으로의 연구 과제는 제안한 방식에 대한 더욱 다양한 부호율과 터보 코드의 구속장 길이 변화에 따른 최적의 생성 다항식을 찾아 실험을 통해 보다 일반화시키고, 다양한 채널 환경에서의 실험을 통해 특성을 분석하며, 실제의 하드웨어 제작에 적합한 터보 코드의 복호화 알고리즘을 개발하는 것이다.

참고 문헌

- [1] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near Shannon limit error-correcting coding and decoding turbo codes," in Proc. *International Conference on Communications (ICC'93)*, pp. 1064-1070, 1993.
- [2] L. Bahl, J. Coche, F. jelinek, and J. Raviv, "Optimal decoding of linear codes for minimizing symbol error rate," *IEEE Trans.*

Inform. Theory, vol. IT-20, pp. 284-287, Mar. 1974.

- [3] S. Pietrobon and A. Barbulescu, "A simplification of the modified Bahl decoding algorithm for systematic convolutional Codes," in Proc. *International Symposium on Information Theory and Its Applications(ISITA'94)*, pp. 1073-1077, 1994.
- [4] S. Pietrobon, "Implementation and performance of a serial MAP decoder for use in an iterative turbo decoder," in Proc. *International Symposium on Information Theory(ISIT'95)*, pp. 471, 1995.
- [5] TIA/EIA/IS-95, Mobile Station-Base Station Compatibility Standard for Dual-Mode Wideband Spread Spectrum Cellular System, published by Telecommunication Industry Association as a North American 1.5 GHz Cellular CDMA Air-Interface Standard, July 1993.
- [6] 김 준범, 문 태현, 임 승주, 주 판유, 홍 대식, 강 창언, "감소된 복호 지연을 갖는 Turbo Code", *한국통신학회 논문지*, '97-7 Vol.22 No.7
- [7] 최 회동, 노 종선, 강 회원, "인터리버 크기가 작은 터보 부호의 새로운 종료 알고리즘", *한국통신학회 논문지*, '99-3 Vol.24 No.3A

이 호 웅(Ho-Woong Lee)

정회원



1984년 2월 : 광운대학교
전자통신공학과(공학사)
1986년 2월 : 광운대학교 대학원
전자통신공학과(공학석사)
2000년 2월 : 광운대학교 대학원
전자통신공학과(공학박사)

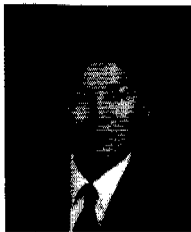
1988년 4월~1996년 2월 : LG전자 영상미디어연구소(선임연구원)
1990년 8월~1995년 8월 : LG전자 시카고연구소(Zenith Electronics, U.S.A)
1996년 3월~1997년 2월 : 대원과학대학 전자통신과
1997년 3월~현재 : 동원대학 정보통신과 조교수
<주관심 분야> 통신 및 영상신호처리, HDTV

강 철 호(Chul-Ho Kang)

한국통신학회논문지 제26권 제1B호 참조

김 동 원(Dong-won Kim)

정회원



1999 2월 : 광운대학교 전자
통신공학과(공학사)
2000년 2월 : 광운대학교 대학원
전자통신공학과(공학석사)
2000년 2월~현재 : 삼성전자
DTV사업팀 연구원

<주관심 분야> 채널코딩, DTV