

이동통신 시스템에서 MAP 기반 터보 부호의 복호에 관한 연구

A Study on Iterative MAP-Based Decoding of Turbo Code in the Mobile Communication System

박 노 진, 강 철 호
No-Jin Park, Chul-Ho Kang

요 약

최근의 이동통신 시스템에서 오류 정정 부호화 방식으로 사용되는 터보 코드의 성능은 부호기에 내재되어 부호의 자유 거리 결정에 큰 영향을 미치는 인터리버와 복호기에서 수행되는 반복 복호에 의한 것이다. 하지만 우수한 성능을 얻기 위해 수신과정에서 많은 지연시간이 요구되는데 이는 주로 인터리버의 크기에 의존하게 된다. 또한 터보 코드는 페이딩 채널 상에서도 신뢰성 있는 강력한 코딩 기법으로 알려져, 최근 ITU 등에서 IMT-2000과 같은 차세대 이동 통신에서 채널 코드의 표준으로 채택되었다. 따라서 본 논문에서는 복잡도는 2배로 증가하나 성능을 개선시킨 병렬 구조의 4-New 터보 복호기를 제안하고, 차세대 무선 멀티미디어 통신에서 실시간 음성 및 비디오 서비스를 제공 시 복호 지연시간을 단축시키기 위해 가변 복호 방법을 이용하여 AWGN과 페이딩 채널 환경에서의 컴퓨터 모의 실험을 통해 성능 분석을 하였다.

ABSTRACT

In the recent mobile communication systems, the performance of Turbo Code using the error correction coding depends on the interleaver influencing the free distance determination and the recursive decoding algorithms that is executed in the turbo decoder. However, performance depends on the interleaver depth that need a large time delay over the reception process. Moreover, Turbo Code has been known as the robust coding method with the confidence over the fading channel. The International Telecommunication Union(ITU) has recently adopted as the standardization of the channel coding over the third generation mobile communications such as IMT-2000. Therefore, in this paper, we proposed of the method to improve the conventional performance with the parallel concatenated 4-New Turbo Decoder using MAP algorithm in spite of complexity increase. In the real-time voice and video service over the third generation mobile communications, the performance of the proposed method was analyzed by the reduced decoding delay using the variable decoding method by computer simulation over AWGN and fading channels.

Keywords : Turbo Codes, MAP Algorithm, Decoding Delay

1. 서 론

터보 코드는 같은 모양의 길쌈 부호 2개 이상을 병렬로 연결하고, 전송 전에 부호 비트의 배열을 바꾸어 주기 위해 각각의 부호화기의 입력으로 들어가는 시퀀스를 인터리버를 통과하게 함으로써 부호화기에서 나오는 출력 시퀀스를 서로 다른 배열로 전송한다. 터보부호의 성능에 영향을 미치는 주요 요소로는 인터리버, 메모리, 펀처링(puncturing)의 유무, 노이즈 레벨과 반복 복호화 횟수 등이 있다^{[1][2]}. 일반적으로 터보 부호는 강력한 오류정정 능력을 갖는 코딩 기법임에도 불구하고 너무 긴 복호 지연

시간으로 인해 성능에 영향을 초래한다. 본 논문에서는 반복 복호 횟수를 감소시키고 성능을 향상시킨 MAP 기반 병렬 구조의 4단 터보 복호기를 제안하고, 이를 컴퓨터 모의 실험을 통해 기존 방법과의 비교 및 성능 분석을 하였다. 본 논문의 구성은 다음과 같다. II장에서는 터보 부호에 대해 알아보고, III장에서는 본 논문에서 사용한 반복 복호 알고리즘에 대해 유도한다. IV장에서는 제안한 터보 부호화기의 대해 기술하고, V장에 모의 실험을 통해 제안한 기법의 성능을 평가해보고, VI장에서 결론을 맺는다.

II 터보 부호기(Turbo Encoder)

그림 1에 나타낸 것처럼 본 논문에서 사용한 터보 부호기는 구속장의 길이(constraint length) $K=3$ (i.e., 메모리 $m=2$), the parity polynomial $g_1(D)=1+D^2$, the feedback polynomial $g_0(D)=1+D+D^2$ 인 부호어를 생성시키는 부호기이다.

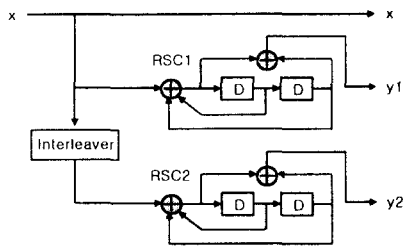


그림 1. 터보 부호기의 구조
Figure 1. structure of the turbo encoder

프레임 길이 L 을 갖는 각 입력 데이터 $\mathbf{x} = \{x_0, x_1, \dots, x_{L-1}\}$ 에 대하여 RSC1은 첫 번째 parity sequence $\mathbf{y}_1 = \{y_{1,0}, y_{1,1}, \dots, y_{1,L-1}\}$, RSC2는 두 번째 parity sequence $\mathbf{y}_2 = \{y_{2,0}, y_{2,1}, \dots, y_{2,L-1}\}$ 를 출력하므로 그림 1로부터 생성되는 부호어의 부호화율은 $1/3$ 이 된다.

III. 반복 복호 알고리즘^{[3][4][5][6]}

본 논문에서 사용한 MAP(Maximum A Posteriori) 알고리즘은 수신 시퀀스를 이용해서 정보의 상태 전이 사후확률(APP: a posteriori probability)과 전송된 메시지 비트 또는 Markov process에 의해 산출되는 코드 심벌 등을 추정하는 것이다. 이는 선형 부호의 심벌 오류 확률을 최소화 할 수 있는 최적의 복호화 방법으로, 1974년 Bahl-CoCke-Jelinek-Raviv에 의해 발표된 심벌 단위의 BCJR Algorithm이 변형된 것이다. 이 절에서는 터보 코드의 반복 복호 알고리즘의 개념을 간략히 유도한다.

1. 터보 부호의 반복 복호화 방법

터보 부호의 복호화는 trellis를 기초로 하는 복호 알고리즘을 이용해서 각각 2개의 Markov process들에 대해서 독립적인 추정을 한다. d_i 가 전송 메시지 비트, y_i 는 수신신호라면 연 판정 복호화기에서의 각 비트에 대한 복호 출력값은 $LLR(\log - \text{likelihood ratio})$ 값으로 구한다.

$$\Phi_i = \log \frac{P(d_i = 1|y)}{P(d_i = 0|y)} \quad (1)$$

통신 시스템 환경이 BPSK로 변조되고 플랫 페이딩 채널이면 수신된 신호는 식(2)와 같이 표현되나, 간략화 하면 식(3)과 같이 표현할 수 있다.

$$y = a\sqrt{E_s}(2d-1) + n \quad (2)$$

여기서 a 는 페이딩 진폭이고 $(2d-1)$ 은 antipodal 형태의 변조된 심벌이고, E_s 는 심벌의 에너지, n 는 평균이 0이며 분산이 $\sigma^2 = N_0/2$ 인 가우시안 랜덤 변수이다. 만약 $a=1$ 이라면 BPSK로 변조되고 채널이 AWGN인 수신 신호를 나타낸다.

$$y = a(2d-1) + n. \quad (3)$$

식(3)에서 잡음의 분산은 $\sigma^2 = N_0/2E_s$ 이다. 한 비트의 수신 신호 y_i 가 주어졌을 때 메시지 비트 d_i 의 LLR은 다음과 같다.

$$L(d_i|y_i) = \log \frac{P(d_i = 1|y_i)}{P(d_i = 0|y_i)} \quad (4)$$

식(4)를 Bayes 이론을 이용하여 정리한다.

$$\begin{aligned} L(d_i|y_i) &= 4a \frac{E_s}{N_0} y_i + \log \frac{P(d_i = 1)}{P(d_i = 0)} \\ &= L_c y_i + z_i \end{aligned} \quad (5)$$

여기서 L_c 는 채널의 신뢰도를 나타내고 z_i 는 priori 값이다. 그러므로 전체 수신 시퀀스에 대한 메시지 비트 d_k 의 LLR을 구하면 식(6)과 같고 이를 강 판정하여 전송한 메시지 비트를 추정한다.

$$\Phi_i = L_c y_i + z_i + \mu_i \quad (6)$$

$$\mu_i = \Phi_i - L_c y_i - z_i \quad (7)$$

식(6)에서 $L_c y_i$ 는 systematic 채널 정보이고 z_i 는 이전 복호화기에서 보내진 priori 정보, μ_i 는 외부 정보(extrinsic information)이다.

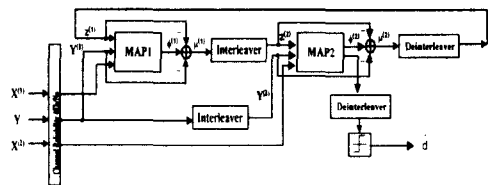


그림 2. 일반적인 터보 복호기
Figure 2. Conventional turbo decoder

그림 2는 일반적으로 사용되는 터보 복호화기의 구조이다. 여기서 비트 오류(BER: bit error rate)을 최소화 하는 알고리즘을 MAP 알고리즘이라 하고, 그림 2에서 보듯 터보 부호의 복호기는 반복 복호를 하는 구조로 되어 있다. 따라서 일정한 횟수의 반복 복호화 과정이 끝나면 최종 정보를 추정하게 되는데 두 번째 복호화기에서

의 LLR 출력을 디인터리빙 한 후에 강 판정을 하여 전송한 메시지 비트 \hat{d} 를 추정한다.

2. MAP 알고리즘

RSC 부호기의 구성 장을 K 라 하면 부호기는 상태 $m=0, 1, \dots, 2^K-1$ 까지 존재하므로, 메시지 비트의 사후확률 $P(d_k=1|y)$ 과 $P(d_k=0|y)$ 이 계산되면 최종적으로 원하는 LLR 형태의 복호화기의 출력을 구할 수 있다.

$$L(d_k) = \log \frac{P(d_k=1|y)}{P(d_k=0|y)} + \log \frac{\sum_{\substack{(m, m) \\ d_k=1}} P(S_{k-1}=m, S_k=m, y)}{\sum_{\substack{(m, m) \\ d_k=0}} P(S_{k-1}=m, S_k=m, y)} \quad (8)$$

메시지 비트의 사후 확률을 구하기 위해서는 상태 천이 사후확률 $P(S_{k-1}=m, S_k=ml|y)$ 을 구해야 하는데 그 과정은 다음과 같다. 조건부 확률의 정의로부터

$$P(S_{k-1}=m, S_k=ml|y) = \frac{P(S_{k-1}=m, S_k=ml, y)}{P(y)} \quad (9)$$

위 식의 분자는 Markov process의 성질에 의해 식(10)처럼 분리할 수 있다.

$$P(S_{k-1}=m, S_k=ml|y) = P(S_{k-1}=m, y_{j<k}) \cdot P(y_{j>k}|S_k=m) \cdot P(S_k=ml|S_{k-1}=m) \quad (10)$$

여기서 $y_{j<k}$ 는 처음 시간부터 $k-1$ 까지의 수신 시퀀스 y_j 이고, $y_{j>k}$ 는 시간 $k+1$ 부터 마지막 수신 시퀀스까지를 나타낸다. 이제 식(11), (12), (13)처럼 확률 함수들을 정의한다.

$$\alpha(S_{k-1}) = P(S_{k-1}=m, y_{j<k}) \quad (11)$$

$$\lambda(S_{k-1}, S_k) = P(S_{k-1}=m, y_k|S_{k-1}=m) \quad (12)$$

$$\beta(S_k) = P(y_{j>k}|S_k=m) \quad (13)$$

위의 식(11~13)들을 식(10)에 대입하면 식(14)와 같다.

$$P(S_{k-1}=m, S_k=ml, y) = \alpha(S_{k-1}) \cdot \lambda(S_{k-1}, S_k) \cdot \beta(S_k) \quad (14)$$

식(14)에서 $\lambda(S_{k-1}, S_k)$ 는 상태 S_{k-1} 가 S_k 로 천이 할 때 관계하는 branch metric을 나타내며 식(15)와 같이 변경된다.

$$\lambda(S_{k-1}, S_k) = P(S_k=ml|S_{k-1}=m) \cdot P(y_k|S_{k-1}=m, S_k=ml) = P(d_k) \cdot P(y_k|x_k) \quad (15)$$

위의 식(15)에서 d_k 와 x_k 는 상태 S_{k-1} 가 S_k 로 천이 할 때 관계되는 메시지 비트와 부호화기의 출력 값이고, 첫 번째 항 $P(d_k)$ 는 priori 정보로서 z_k 로부터 얻어진

값이다. 그리고 $P(y_k|x_k)$ 는 채널 환경과 변조의 함수이다. 본 논문에서 가정한 BPSK로 변조되고 채널이 플랫 페이딩인 경우 $P(y_k|x_k)$ 는 다음과 같다.

$$P(y_k|x_k) = \frac{1}{\sqrt{\pi N_0/E_s}} e^{-\frac{E_s}{N_0} \sum_{i=1}^2 [y_{k,i} - a_{k,i}(2x_{k,i}-1)]^2} \quad (16)$$

식(14)의 확률 함수 $\alpha(S_k)$ 는 순방향 재귀(forward recursion)로 구하고, 비슷한 방법으로 $\beta(S_k)$ 는 역방향 재귀(backward recursion)로 구한다면 식(17)과 식(18)과 같다.

$$\alpha(S_k) = \sum_{m=0}^{M-1} P(S_{k-1}=m, S_k=m, y_{j<k+1}) = \sum_{S_{k-1}=0}^{M-1} \alpha(S_{k-1}) \cdot \lambda(S_{k-1}, S_k) \quad (17)$$

$$\beta(S_k) = \sum_{m=0}^{M-1} P(S_{k+1}=m, y_{j>k}|S_k=m) = \sum_{S_{k+1}=0}^{M-1} \beta(S_{k+1}) \cdot \lambda(S_{k+1}, S_k) \quad (18)$$

식(14)에서와 같이 정보의 상태 천이 사후확률 $P(S_{k-1}=m, S_k=ml|y)$ 이 구해지면 식(19)처럼 각각의 메시지 비트에 대한 사후 확률을 구할 수 있기 때문에 최종적으로 구해지는 LLR은 식(20)과 같이 나타내게 된다.

$$P(d_k=\pm 1|y) = \sum_{\substack{(m, m) \\ d_k=\pm 1}} P(S_{k-1}=m, S_k=ml, y) \quad (19)$$

$$L(d_k) = \log \frac{\sum_{\substack{(S_{k-1}, S_k) \\ d_k=1}} \alpha(S_{k-1}) \cdot \lambda(S_{k-1}, S_k) \cdot \beta(S_k)}{\sum_{\substack{(S_{k-1}, S_k) \\ d_k=0}} \alpha(S_{k-1}) \cdot \lambda(S_{k-1}, S_k) \cdot \beta(S_k)} \quad (20)$$

개념적으로 MAP 알고리즘이 메시지 비트의 사후확률을 정확하게 추정한다 하더라도 한 비트의 추정에 약 6×2^M 의 곱셈과 덧셈을 필요로 하는 많은 계산량과 round-off 에러에 아주 민감하다. 이러한 문제들은 곱셈을 덧셈으로 바꾸어 계산하는 로그 계산을 하여 어느 정도 해결하고 있는데 이를 위해 Jacobian logarithm을 이용한 로그 계산을 수행하고, 이를 일반화하면 식(21~23)와 같이 나타낼 수 있다.

$$\bar{\lambda}(S_{k-1}, S_k) = \log \lambda(S_{k-1}, S_k) = \log P(d_k) + \log P(y_k|x_k) \quad (21)$$

$$\begin{aligned} \bar{\alpha}(S_k) &= \log \alpha(S_k) = \log \left(\sum_{S_{k-1}=0}^{M-1} e^{[\bar{\alpha}(S_{k-1}) + \bar{\lambda}(S_{k-1}, S_k)]} \right) \\ &= \max_{S_{k-1}} [\bar{\beta}(S_{k+1}) + \bar{\lambda}(S_{k+1}, S_k)] \end{aligned} \quad (22)$$

$$\begin{aligned} \bar{\beta}(S_k) &= \log \beta(S_k) = \log \left(\sum_{S_{k-1}=0}^{M-1} e^{[\bar{\beta}(S_{k-1}) + \bar{\lambda}(S_{k-1}, S_k)]} \right) \\ &= \max_{S_{k-1}} [\bar{\beta}(S_{k+1}) + \bar{\lambda}(S_{k+1}, S_k)] \end{aligned} \quad (23)$$

그러므로 trellis의 모든 상태들에 대해 $\bar{\alpha}(S_k)$ 와 $\bar{\beta}(S_k)$ 가 구해지면 각 복호화기의 LLR은 식(24)과 같이 계산된다.

$$\begin{aligned} L(d_k) &= \max_{d_k=1} (S_{k-1}, S_k) \{ \bar{\alpha}(S_k) + \bar{\beta}(S_{k+1}) + \bar{\lambda}(S_{k-1}, S_k) \} \\ &- \max_{d_k=0} (S_{k-1}, S_k) \{ \bar{\alpha}(S_k) + \bar{\beta}(S_{k+1}) + \bar{\lambda}(S_{k-1}, S_k) \} \end{aligned} \quad (24)$$

이러한 MAP 알고리즘은 직접적으로 사후 확률을 계산하기가 효과적이지만 많은 계산 과정과 메모리 및 복호 지연을 필요로 하기 때문에 현재 이를 간소화하기 위한 여러 연구들이 수행되어 왔다. 또한 log MAP 알고리즘은 correlation factor에 대한 look-up table이 필요하지만, max-log 알고리즘은 look-up table이 불필요하고, 복잡성도 1/2 정도이다.

IV. 제안한 방법의 터보 복호기

터보 부호의 구조 및 복호 방법은 복호 지연과 성능을 결정하는 중요한 요소이다. 특히 복호화기의 구조는 하드웨어의 복잡도와 성능에 따라 결정되어야 하고, 반복 복호화의 수는 복호 지연과 성능을 잘 고려하여 선택되어야 한다. 그리고 복호화기의 출력은 연성 입·출력(SISO: Soft In Soft Output) 데이터이므로 반복적인 복호가 가능하며, 반복 복호의 횟수가 증가할수록 성능이 향상되나 복호 지연과 계산의 복잡성, 연산량 증가 등의 문제를 가져오므로 대략 6~10회 정도의 반복 복호를 수행하면 만족할 만큼의 성능을 얻을 수 있다. 하지만 이러한 반복 복호를 수행함으로써 실시간 통신에서 커다란 장애라 할 수 있는 큰 지연을 수반하게 된다. 특히 직렬 복호화기를 3단 이상 연결하여 사용 시에는 하드웨어의 복잡성과 더불어 많은 복호 지연을 수반하게 된다. 따라서 본 논문에서는 기존의 2단 복호화기와 비교하여 하드웨어의 복잡도는 2배로 증가되나 성능은 향상시키고 복호 지연은 감소시킨 4단 병렬 복호화기를 제안한다. 그림

3는 본 논문에서 제안한 병렬 구조의 4-New 터보 복호화기이다.

그림 3에서 $L_1^{(m)}$ 과 $L_2^{(m)}$ 는 m 번째 복호화가 진행된 후 하단 복호기 1과 상단 복호기 2의 LLR(log-likelihood ratio) 출력 값이다. 그리고 $L_3^{(m)}$ 는 $L_1^{(m)}$ 과 $L_2^{(m)}$ 값을 합한 LLR 값이다. 매 복호시마다 하단 복호기 1과 상단 복호기 2에서 출력되는 LLR 값은 서로 다른 값들의 분포를 가지므로, 각각의 LLR 출력 값을 강 판정(hard decision)한 시퀀스 \hat{u}_1, \hat{u}_2 도 서로 다른 분포 값을 가지며, 각각의 오류를 계산해 보면 두 오류 값이 서로 다르다.

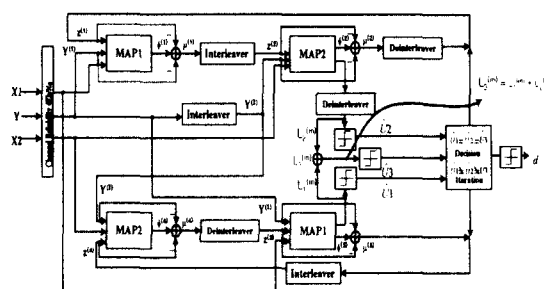


그림 3. 제안한 병렬 구조의 터보 복호기
Figure 3. Turbo decoder with proposed parallel structure

강 판정된 \hat{u}_1 와 \hat{u}_2 의 분포는 서로 다르지만, 복호 횟수 m 이 증가할수록 두 값의 신뢰도가 높아지는 분포로 새롭게 갱신되어 더 이상 성능 향상이 없는 오류 값으로 수렴하게 된다. 기존의 터보 복호기에서는 요구되는 반복 복호횟수 m 번의 복호를 진행한 후 \hat{u}_2 를 추정한다. 이런 경우 오류가 줄어드는 방향으로 수렴이 빨리 이루어지지 않으면 정해진 복호 횟수만큼 복호를 해야 하겠지만, 그렇지 않고 정해진 복호 횟수 이전에 수렴이 빨리 이루어질 때는 그만큼의 불필요한 복호가 이루어져 바람직한 현상이라 할 수 없다. 특히 비트 에너지 대 잡음의 전력 비가 클수록 오류분포 수렴 속도가 빨라져 불필요한 복호로 인한 지연은 더 크다고 할 수 있다. 이 때문에 본 논문에서는 위에서 언급한 각각의 LLR 값이 서로 다른 분포 특성을 이용하여 복호 횟수를 상당량 줄일 수 있는 방법을 사용하였다^[7].

제안한 병렬 구조의 복호 절차는 매 복호시마다 \hat{u}_1 과 \hat{u}_2, \hat{u}_3 를 서로 비교함으로써 오류 분포가 수렴하게 되며, 강 판정한 3개의 출력 값이 같으면 더 이상 복호를 진행하지 않고 \hat{u}_3 를 판정하고, 그렇지 않으면 정해진 복호 횟수만큼 반복 복호를 진행하게 되므로 기존의

복호 지연 시간을 상당히 단축시킬 수 있다. 또한 일반적인 병렬 구조의 복호기를 이용하는 것 보다 제안한 방식을 이용하여 복호시 상단과 하단의 복호기들이 각각 분리 동작하면서 LLR 값을 출력한다. 따라서 더욱 신뢰성 있는 복호 과정이 진행되고 성능을 향상시킬 수 있음이 컴퓨터 모의 실험을 통해 확인할 수 있었다. 제안한 4-New 복호화기 구조는 기존의 2단 직렬 복호기에 비해 하드웨어의 복잡도는 2배이나, 그에 대한 보상으로 성능은 향상시키면서 복호 횟수를 가변적으로 적용할 수 있는 방법을 사용하여 복호 지연을 감소시킬 수 있다. 모의 실험을 통해 프레임 사이즈가 증가할수록 성능이 증가됨을 알 수 있었고, 전체적으로 4096 비트 이하의 프레임에서는 약 0.5~0.8dB 정도의 성능을 향상시킬 수 있음을 확인할 수 있었다.

V. 모의 실험

본 논문에서는 이동통신 시스템에서 잡음의 영향으로부터 전송된 신호를 복원하기 위해 필수적인 오류 정정 부호 중 IS-95 및 cdma 2000에서도 채택한 터보 부호에 대해 4-New 병렬 구조의 터보 복호화기를 제안하고 이의 이론적 고찰과 모의실험을 통한 분석을 하였다. 복호화 알고리즘은 MAP를 사용하였고, 터보 부호화기는 동일한 2개의 RSC 부호화기를 사용하였다. BPSK 변조를 가정하였으며, 구속장의 길이 $K=3$, 구성 부호화기의 생성 다항식은 $(7, 5)_8$ 부호율은 평처링(puncturing)을 사용하지 않은 $1/3$ 을 사용하였다. CDMA 채널 환경에서의 성능 분석을 위해 채널 모델은 AWGN 과 플랫 페이딩 모델을 사용하였다. 반복 복호 횟수는 최대 5회를 적용하여 가변 복호 하도록 하였으며, 정보 시퀀스의 프레임 크기 L 은 256, 512, 1024, 4096 비트인 경우에 대하여 모의 실험하였다. 그림 4와 그림 5는 제안한 방식의 4-New 복호기와 기존의 유사한 터보 복호기(2-Kye^[4], 4Benedetto^[6], 2Benedetto^[8])와의 오차성능(BER)을 비교한 것인데, 성능에 커다란 변화가 있음을 알 수 있다. AWGN 채널 환경에서 각 프레임 길이 256, 512비트에 대한 평균 이득은 약 0.4dB 정도의 성능 향상을 보였고, 1024, 4096비트는 약 0.8 dB 정도의 이득을 보여 프레임 사이즈가 증가할수록 제안한 4-New 복호화기의 성능 향상이 두드러지게 개선됨을 알 수 있다. 그리고 페이딩 채널 환경에서도 약 0.5~0.85 dB 정도의 꾸준한 성능 개선이 이루어지며, 프레임 사이즈가 증가될수록 점점 뚜렷한 성능 개선이 이루어짐을 알 수 있다.

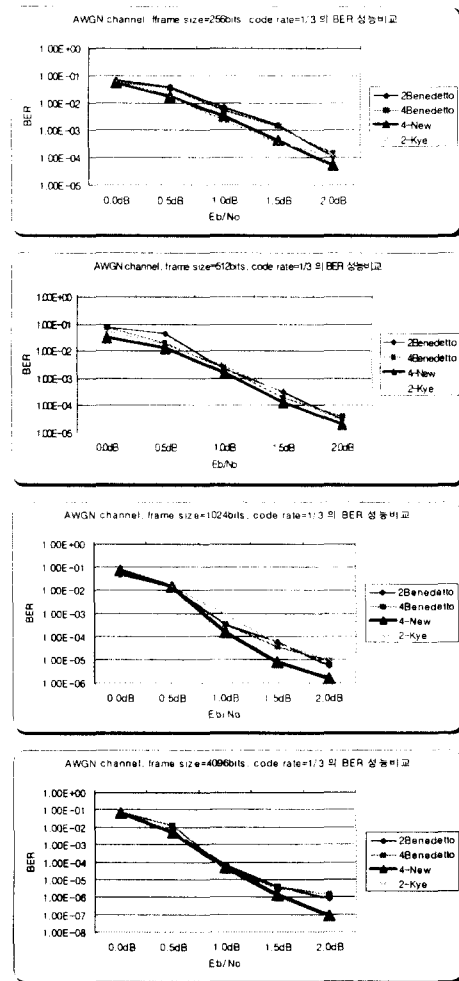
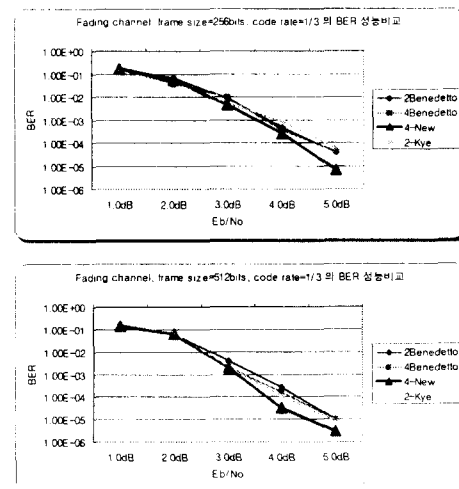


그림 4. AWGN 채널 환경에서 프레임 길이 256, 512, 1024, 4096 비트에 따른 성능
Figure 4. BER performance with 256, 512, 1024, 4096 bits in AWGN Channel



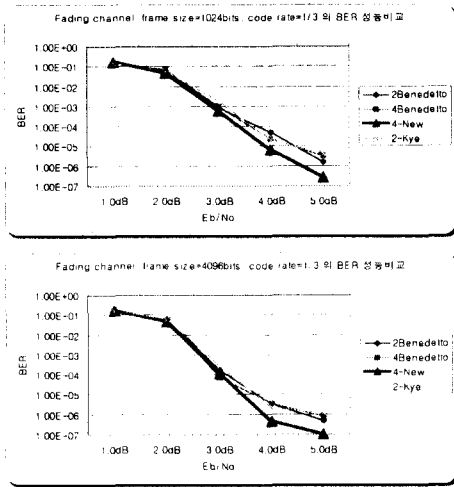


그림 5. 페이딩 채널 환경에서 프레임 길이 256, 512, 1024, 4096 비트에 따른 성능

Figure 4. BER performance with 256, 512, 1024, 4096 bits in Fading Channel

VI. 결론

본 논문에서는 차세대 이동통신 시스템에서 사용되는 Turbo Code에 대해 복잡도는 2배로 증가하나 성능을 개선하고 복호 지연을 감소시킨 병렬 터보 복호화기를 제안하였다. 제안한 4-New 복호화기는 기존의 유사한 터보 복호화기와 비교하여 하드웨어의 복잡도는 2배로 증가하나 성능은 향상시키고, 복호 지연을 개선시켜 실시간 통신 및 신뢰성이 요구되는 통신 시스템에도 적용할 수 있기를 기대한다. 앞으로의 연구 과제는 현재의 반도체 기술의 발전에 부응하여 신뢰성이 요구되는 시스템에 적용하기 위해 하드웨어의 복잡도는 증가하나 터보 코드의 지연을 감소시키고, 성능은 획기적으로 향상시키기 위한 연구가 이루어져야 한다.

접수일자 : 2000. 11. 28 수정완료 : 2001. 3. 30

참고 문헌

[1] C. Berrou, A. Glavieux, P. Thitimajshima. "Near Shannon Limit Error-Correcting Coding and Decoding : Turbo Codes," in ICC, pp. 1064-1070, 1993.
 [2] P. Robertson. Improving decoder and code structure of parallel concatenated recursive systematic (turbo)codes, in Proc. IEEE ICUPC, pp. 183-187, 1994.
 [3] Matthew C. Valenti, Iterative detection and decoding for wireless communications, Ph.D.

Dissertation, Virginia Polytechnic Institute and State Univ, Sept. 1998.

[4] 계환원, 구정우, 송홍엽, "병렬구조 복호기에서 LLR의 합을 사용한 터보 부호의 성능분석", 한국통신학회 논문지, 제 24권, 제 11A호, pp. 1678-1684, 1999.
 [5] L. R. Bahl et al., Optimal Decoding of Linear Codes for Minimizing Symbol Error Rate, IEEE Trans. Info. Theory, vol. IT-20, Mar. pp. 248-87, 1974.
 [6] S. Benedetto, D.Divsalar and F.Pollara, Soft-Output Decoding Algorithm in Iterative Decoding of Turbo Codes, TDA progress rep. 42-124, Jet Propulsion Lab., Pasadena, CA, Feb. 15, pp. 63-86, 1995.
 [7] 강경우, 박노진, 강철호, "감소된 복호 지연을 갖는 SOVA 기반 터보 부호에 관한 연구," 2000년 제 13회 신호처리합동학술대회 논문집, Vol. 13, No. 1, 2000년 9월.
 [8] S. Benedetto and G. Montorsi, "Design of Parallel Concatenated Convolutional Codes," IEEE Trans. Commun., vol. 44, iss. 5, pp. 591-600, June 1996.



박노진(No-jin Park)

正會員

1992년 서울산업대학교 매체공학과 공학사

1998년 광운대학교 전자통신공학과 공학석사

1998년~현재 광운대학교

전자통신공학과 박사과정

관심분야: 비동기 CDMA, 통신신호처리, 채널 코딩



강철호(Chul-ho Kang)

正會員

1975년 한양대학교 전자공학과 공학사
 1979년 서울대학교 전자공학과 공학석사

1977년~1981년 국방과학연구소 (연구원)

1988년 서울대학교 전자공학과 공학박사

1994년 4월~현재 광운대학교 전자공학부 정교수
 관심분야: 통신 및 음성 신호처리, 이동통신