

論文2000-37TC-02-1

3GPP2에 적용된 터보부호의 성능 분석 (Analysis of the performance of Turbo codes on 3GPP2)

玄 星 煥 * , 李 旻 秀 * , 朴 晨 鐘 *

(Sunghwan Hyun, Gyongsu Lee, and Sinchong Park)

요 약

본 논문에서는 3GPP와 3GPP2에서 제시된 표준안을 사용하여 모의실험을 수행하였다. 모의실험은 AWGN 채널상에서 표준안으로 제시된 터보부호의 사양에 따른 성능분석을 중심으로 3GPP에 제안되어 있는 터보부호의 성능을 비교하였다. AWGN 채널과의 비교대상으로 실제 채널과 보다 근접한 레일리 페이딩 채널 하에서 3GPP2에 제시된 여러 블록 중 한 블록 크기에 대하여 수행된 결과를 제시하였으며, Max_Log MAP방식과 Log MAP 방식의 터보복호기의 성능도 비교하였다.

Abstract

In this paper, simulations are performed using standards proposed by 3GPP and 3GPP2. From the view point of performance analysis based on Turbo code specification, performance of Turbo code in AWGN, proposed by 3GPP, is compared. To make comparison, experimental result of Turbo code that is performed on one of the many block sizes proposed by 3GPP2 in Rayleigh fading channel, is provided. Performance of Turbo decoders using Max_Log MAP and Log MAP are compared.

I. 서 론

1993년에 개발된 터보부호^[1]는 그 자체의 놀라운 성능과 더 좋은 성능을 구현 가능케 하는 구성부호와 인터리버를 찾고자 하는 노력, 그리고 터보부호의 약점으로 지적되어온 복호시의 큰 시간지연과 하드웨어 구현시 필수적인 복호기의 복잡성을 줄이려는 전세계 많은 연구자들의 노력의 결과로 터보부호는 차세대 이동 통신인 IMT2000에서 길쌈부호의 역할을 대신하게 되어왔다.

1997년부터 진행 되어온 IMT2000 표준화 작업은 크게 3GPP(3rd Generation Partnership Project) 그룹과 3GPP2 그룹으로 양분되어 있고, 1999년 초부터 OHG(Operator Harmonization Group)에 의한 harmonization작업이 진행 중에 있다. 여기서 3GPP는

ETSI주도의 GSM에 기반을 둔 WCDMA방식이고, 3GPP2는 TTA주도의 IS-95 기반의 cdma2000방식이다. 이 두 그룹의 각 표준이 서로 양립하게 되어 두 그룹사이의 harmonization작업이 필요하게 되었다. 이 작업은 OHG에 의해 이루어진다. 이 두 그룹의 제안서 모두 길쌈 부호와 터보 부호에 대한 사양을 포함한다. 본 논문에서는 3GPP2에 제안된 터보부호의 사양에 따른 성능의 분석을 중심으로 하여 3GPP에 제안되어 있는 터보부호의 성능을 비교하였다.

이 논문의 구성은 다음과 같다. 2장에서는 3GPP와 3GPP2에서 제시된 표준안과 모의실험에 사용된 터보 부호기와 복호기의 사양을 살펴보고, 3장에서는 모의 실험을 바탕으로 두 그룹에서 제시된 표준사양의 성능을 분석해보았다. 그리고 4장에서는 결론을 맺는다.

* 正會員, 韓國情報通信大學院大學校

(Information and Communications University)

接受日字:1999年11月22日, 수정완료일:2000年1月20日

II. 시스템 사양

I. MT2000에서의 터보부호 표준

IMT-2000의 표준안은 현재 크게 3GPP에서 제안하고 있는 비동기 방식과 3GPP2에서 제안하고 있는 동기 방식이 양대 산맥을 이루고 있다. 이 두 표준안은 모두 채널부호의 방법으로 길쌈부호뿐만 아니라 터보부호를 받아들이고 있다^{2, 3)}. 터보부호는 BER (bit error rate) 이 10^{-3} 에서 10^{-6} 사이의 서비스 품질을 요구하는 데이터 서비스에서 사용하도록 되어 있으며, 인터리버를 중심으로 두 개의 RSC (recursive systematic convolutional) 부호기가 병렬로 연결되어 있는 형태이다.

RSC는 모두 구축장 4를 가지는 것을 사용한다. 그러나 3GPP는 부호율이 1/3인 경우만 사용하고, 3GPP2는 1/2, 1/3, 1/4에 대해서 각각 정의되어 있다. 부호율이 1/2과 1/3인 경우에는 두 표준안 모두 생성다항식이 $G(D) = \left[1 \frac{n_0(D)}{d(D)} \right]$ 인 것을 동일하게 사용한다 ($d(D) = 1 + D^2 + D^3$, $n_0(D) = 1 + D + D^3$). 부호율이 1/4인 터보부호의 경우는 두 개의 생성다항식, 각각 $G(D) = \left[1 \frac{n_0(D)}{d(D)} \right]$ 인 것과 $G(D) = \left[1 \frac{n_0(D)}{d_1(D)} \right]$ 인 것을 평치링하여 사용한다 ($d_1(D) = 1 + D + D^2 + D^3$). 또한 3GPP에서는 정보 비트가 320이상일 때, 3GPP2에서는 360이상일 경우에 대해서 정의되어 있다. 이 조건에 3GPP는 10msec 프레임을 사용하고, 3GPP2는 20msec 프레임을 기본프레임으로 사용하는 것을 고려한다면, 앞에서 이야기한 정보 비트의 수를 만족하는 경우는 3GPP에서는 60kbps이상일 때, 3GPP2에서는 19.2kbps이상일 때 사용한다.

인터리버의 구조는 서로 다른 것을 사용한다. 3GPP의 내부 인터리버는 세 단계로 구성된다. 먼저 연속된 인터리버의 입력을 RxC 사각행렬에 한 행씩 차례차례 쌓아 나간다. 다음 단계는 각 행 안에서 한번 더 위치를 바꾸어 주고, 마지막 단계에서 한 열씩 차례차례 읽어 나간다. 3GPP2에서도 마찬가지로 행 단위로 입력을 표현하고, 각 행 내에서 위치 변화를 선형합동수열 (linear congruential sequence)에 의해 수행하고, 마지막으로 열 단위로 읽어 낸다. 큰 골격은 유사하나 3GPP에서는 행(R)의 수를 20으로 고정하게 되어 있고¹⁾, 3GPP2는 열(C)의 수를 32로 고정하게 되

1) 정보비트의 크기가 481에서 530인 경우 행의 개수는 10, 그 이외의 크기에서 행의 개수는 20으로 정해져 있다. 하지만 bit rate와 함께 고려할 때 모두 후자에 속한다.

어 있다. 또 3GPP는 열의 수의 성질에 따라 두 번째 단계에서 사용하는 난수열의 식이 결정되고, 3GPP2는 행의 수에 따라 난수열의 식이 결정된다.

Trellis의 종료에 사용되는 방법은 두 표준안이 모두 두 개의 RSC중에서 위의 RSC를 먼저 종료시키고 나서 두 번째 RSC를 종료시킨다. 각각의 RSC는 3개의 레지스터로 구성되어 있기 때문에 3비트의 종료에 3비트의 값이 필요하다. 하지만 3GPP에서 종료 비트의 출력 순서는 첫 번째 RSC의 출력 x, y를 3번 반복한 이후 두 번째 RSC의 출력을 3번 반복한다. 즉 종료시 총 12심벌을 출력하게 되어 있다. 마치 code rate가 1/2인 것처럼 사용하지만 3비트를 반복하기 때문에 3의 배수인 12심벌을 맞추어 낼 수 있게 설계되어 있다. 반면 3GPP2에서는 code rate가 1/2일 경우 앞에서 말한 3GPP의 종료와 같은 출력을 생성하고 1/3일 경우는 출력 x를 한번 더 반복하여 x, x, y의 순서로 출력한다. 즉 각각의 RSC에서 3비트에 해당하는 9심벌을 생성하기 때문에 총 18 심벌이 종료시에 출력된다.

2. 터보부호

(1) 터보부호기 :

일반적으로 터보부호기는 [그림 1]과 같이 구성되어 있다. 이러한 터보부호기는 보통 동일한 RSC부호기를 구성부호기로 사용한다.

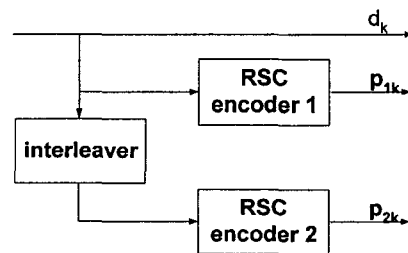


그림 1. 터보 부호기

Fig. 2. Turbo Encoder.

IMT2000에 제안된 터보부호기의 구조를 살펴보면 3GPP와 3GPP2 모두 구축장 4인 구성부호기를 사용하고, 그 생성다항식은 부호율이 1/3인 경우 동일하므로 두 그룹에서 제안된 터보부호기의 성능 고찰을 위하여 [그림 2]의 구성부호기를 사용한 터보부호기의 경우에 대하여 모의실험을 수행하였다.

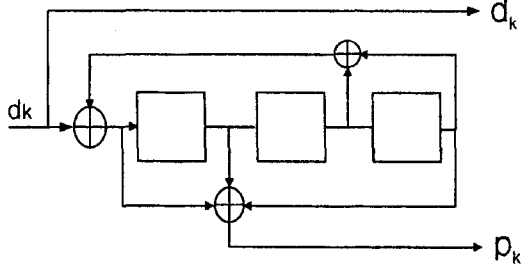


그림 2. RSC부호기
Fig. 2. RSC Encoder.

(2) 터보복호기 :

터보복호기의 구조는 [그림 3]와 같다. 이 복호기에서 사용된 인터리버는 터보 부호기에서 사용한 것과 같은 것을 사용하며 DEC1과 DEC2에는 보통 같은 방식의 복호 알고리즘을 사용한다.

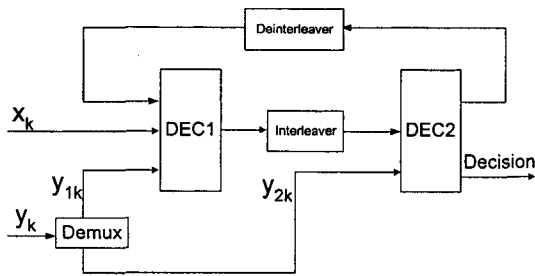


그림 3. 터보 복호기
Fig. 3. Turbo Decoder.

터보복호기에 사용되는 복호 알고리즘은 크게 SOVA(Soft Output Viterbi Algorithm) 방식과 MAP(Maximum A Posteriori) 방식으로 나눌 수 있다. SOVA방식은 MAP에 비해서 하드웨어의 복잡성이 낮은 반면에 대략 0.7dB정도의 감쇠를 가지는 것으로 알려져 있다^[4]. 이 이유로 MAP을 사용하기 위해서 하드웨어의 복잡성을 낮출 필요가 있었고, Pietrobon과 Pieter Jung 등과 같은 연구자들이 수정된 BCJR 알고리즘을 더욱더 간단히 고쳐 터보복호를 구현하였다^[5]. 이번 모의실험에서는 Pietrobon에 의해 제시된 간단한 BCJR 알고리즘을 사용하여 수정되어 졌는데, 이 알고리즘은 기존의 수정된 BCJR 알고리즘에 밑이 $\exp(\frac{2}{\sigma^2})$ 인 로그를 취하여 곱셈은 덧셈으로 바꾸었기 때문에 하드웨어로 구현이 쉬워진다. 또, 덧셈은 아래와 같이 최대 E함수로 바꿀 수 있다.

여기서, $L_c = (2/\sigma^2)$ 로 두면,

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

을 얻을 수 있다. 최대E함수를 이용하여 변조 방법으로 BPSK 변조방식을 사용하고, AWGN 채널 하에서 전송메트릭과 상태메트릭을 정의하면 다음과 같다.

$$D_i(R_k, m) = x_k i + y_k p_k \quad (2)$$

$$A_k^i(m) = D_i(R_k, m) + E_{j=0}^1 A_{k-1}^j(m') \quad (3)$$

$$B_k^i(m) = E_{j=0}^1 [B_{k+1}^j(m') + D_j(R_{k+1}, m')] \quad (4)$$

최종적으로 판정을 위한 LLR을 나타내면 다음과 같다.

$$\begin{aligned} L(d_k) &= E_{m=0}^{2^v-1} \{A_k^1(m) + B_k^1(m)\} \\ &\quad - E_{m=0}^{2^v-1} \{A_k^0(m) + B_k^0(m)\} \end{aligned} \quad (5)$$

여기서 $E_{m=0}^{2^v-1} X(m) = X(0) E X(1) E \dots X(2^v-1)$ 이다.

수신단이 코히어런트 검출기능을 가졌다고 가정하였을 때, 정보 심벌 $X \in \{\pm\sqrt{E_s}\}$ 의 BPSK 전송에 대하여 간략히 레일리 페이딩 채널을 모델링하면 다음과 같다^[6].

$$Y = AX + N \quad (6)$$

여기서, E_s 는 심벌당 전송된 에너지이고 A 는 레일리 분포를 갖는 페이딩 변수이며, N 은 zero 평균과 분산 $N_0/2$ 를 갖는 AWGN이다. 레일리 페이딩 채널 모델에 관하여 생각해 보면, 앞선 BCJR 알고리즘의 유도는 변화가 없다. 단지 전송메트릭만이 페이딩 분포에 영향을 받을 뿐이다. 따라서, 가산성 잡음을 가진 레일리 페이딩 채널 상에서 전송메트릭을 구하여 보면 다음과 같다.

$$D_i(R_k, m) = a_k^i (x_k i + y_k p_k) \quad (7)$$

여기서, a_k^i 는 $d_k = i$ 이고 $t = k$ 일 경우의 레일리 분포 함수를 갖는 랜덤한 페이딩 크기이다^[7].

$$f_A(a) = 2a e^{-a^2}, \quad a \geq 0 \quad (8)$$

III. 모의실험

IMT2000의 데이터 전송시 요구되는 BER이 10^{-6}

이므로 우리의 모의실험은 BER : 10^{-6} order를 만족하는 범위에서 수행되었다.

1. AWGN 환경에서 모의실험 결과 :

터보부호는 복호를 할 때 iteration 수를 많이 하면 할수록, 인터리버 크기를 크게 하면 할수록 성능이 좋아진다고 알려져 있고 이것은 [그림 4]의 모의실험 결과로도 확인할 수 있다. [그림 4]의 (a)에서 나타나있는 것처럼 블록 크기 378에서는 iteration 3회 이상이면, Eb/No가 3dB일 때 BER : 10^{-6} order를 만족하는 것을 볼 수 있다. (b)의 결과는 블록 크기 762에서는 iteration 2회 이상이면, Eb/No가 3dB일 때 BER : 10^{-6} order를 만족하는 것을 볼 수 있다. 블록 크기가 1530이상에서는 1번의 iteration 수행으로도 Eb/No가 3dB에서 BER : 10^{-6} order를 만족하였다.

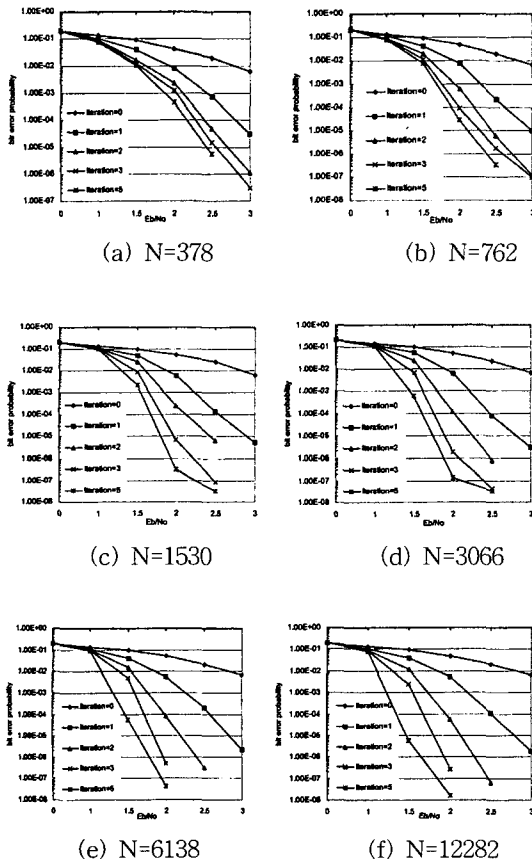
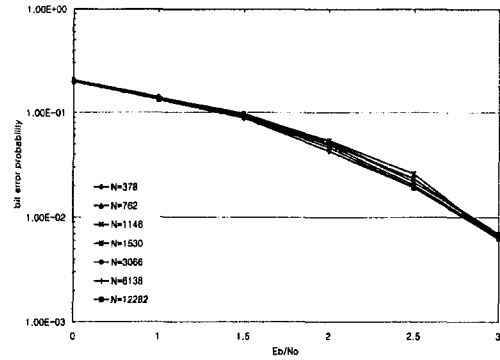
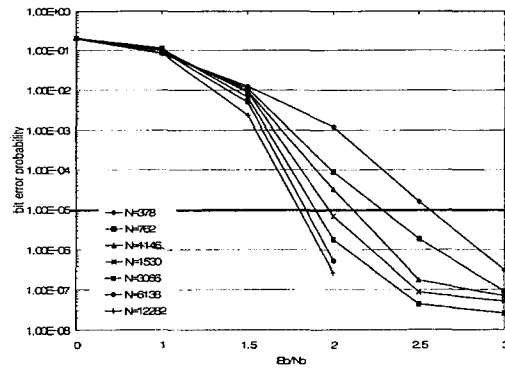


그림 4 부호율에 따른 3GPP2 터보부호화기
Fig. 4. Performances of Turbo encoder according to bit rates.

Iteration 횟수는 크게 할수록 더 좋은 성능을 보이거나 5번이 넘었을 때는 거의 일정한 값을 유지하였다. [그림 5]에서 보여지는 것처럼, iteration을 수행하지 않았을 때는 0~3dB의 Eb/No에서 블록크기에 상관없이 거의 일정한 BER을 유지하였다. 복호 할 때 1회 이상의 iteration은 터보부호의 성능에 커다란 개선을 가져오나, 3회 iteration이상의 iteration은 성능의 개선에 영향을 끼치지 않음을 볼 수 있다.



(a) iteration 횟수 : 1



(b) iteration 횟수 : 3

그림 5. Iteration 횟수에 따른 3GPP2 터보부호기 성능
Fig. 5. Performances of Turbo encoder according to iteration number.

2. 3GPP와 3GPP2에서의 터보부호기 성능 고찰 :

두 그룹에서 각각 제시된 터보부호의 성능을 서로 비슷한 비트율에서 살펴봄으로써 AWGN 채널 하에서 각각의 터보부호의 성능을 검증함과 동시에 간접적인 비교를 해보기 위하여 다음과 같은 사양에서 모의실험을 수행하였다. [그림 6]에서 보여지는 결과는 각각 프레임 크기가 10msec이므로 비트율이 60kbps에서 인

터리버 크기가 600인 3GPP의 터보부호로 수행한 결과와 프레임 크기가 20msec이므로 비트율이 57.6kbps에서 인터리버 크기가 1146인 3GPP2의 터보부호로 수행한 결과이다. 5번의 복호 iteration으로 BER = 10⁻⁶에서 약 0.25dB의 성능의 차이를 보이는데, 이 결과는 서로 비슷한 비트율에서 3GPP2의 인터리버 크기가 3GPP의 인터리버 크기보다 약 2배정도 크다는 점에서 기인하는 것으로 여겨진다. 따라서, 시간당 같은 양의 데이터를 전송할 때 3GPP2의 터보부호의 성능이 3GPP의 터보부호보다 우수한 성능을 나타낸다.

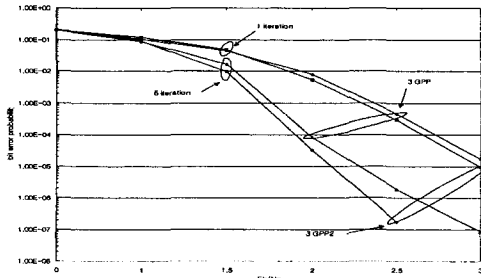


그림 6. 3GPP와 3GPP2의 성능
Fig. 6. Performances of 3GPP and 3GPP2

3. 페이딩 채널 하에서의 모의실험

실제 이동 전송 채널 하에서의 성능 검증을 위해서 슬로우 레일리 페이딩 채널 하에서 모의실험을 수행하였다. 이동체의 속도는 108km/h이고 캐리어주파수는 2Ghz이다. 수신단에서는 완벽한 채널판정이 수행되어진다고 가정한다.

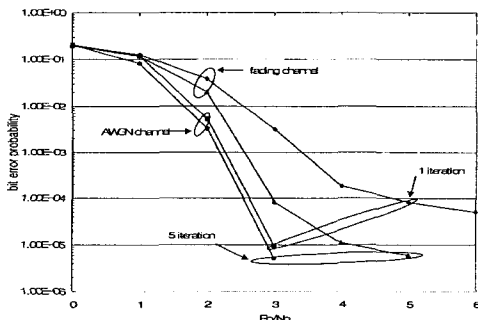


그림 7. AWGN과 페이딩 채널하에서의 성능
Fig. 7. Performances in AWGN and fading channel.

페이딩 채널 하에서 성능 검증을 하기 위해서는 추가적인 채널 인터리버가 필요하게 되는데, 3GPP2 터

보부호의 경우는 20msec의 프레임 주기 동안에 57.6kbps의 일정한 데이터 비율로 프레임 크기가 1146인 채널 블록 인터리버를 사용하였다. [그림 7]에서 보여지는 결과를 AWGN 채널에서 수행된 결과와 비교해 보면, 5번의 복호 iteration으로 BER = 10⁻⁵에서 약 2dB의 성능의 감소가 있음을 볼 수 있다.

4. Log-MAP과 MAX Log-MAP의 비교

터보 복호기의 계산에 필수적인 계산 회로로 최대함수 계산블록이 있다. 기존의 MAP 알고리즘에 있는 곱셈연산을 덧셈연산으로 대체하기 위한 방법으로 로그를 취하게 되고 이 결과를 최대함수로 표현한 것이다. 최대E함수는 식(1)에서 보여지는 것과 같이 두 입력의 크기를 비교하여 큰 값을 골라내는 부분과 두 값의 차이에 해당하는 인수에 대한 로그값 부분이 존재한다. 그리고 최대E함수를 계산하기 위해서는 채널의 정보에 의한 변수인 L_c값도 알아야 하는데 이는 양변에 로그를 취할 때 밑이 exp(2/σ²)인 로그를 취함으로써 해결할 수 있다. f(x) = ln(1 + e^{-L_cx})는 최대값으로 1/L_c ln(1+1) ≈ 0.693/L_c 을 가지고, 최소값으로 0 을 가지므로 이를 테일러로 만들면 최대E함수의 연산 수행시 걸리는 로드를 많이 줄일 수 있다.

다음의 [그림 8]은 수식(1)에 의한 최대 E함수를 사용하여 복호한 결과와 로그 항을 무시하고 max(x, y)만 사용하여 구한 결과의 비교이다. 그림을 보면 BER : 10⁻⁴ order에서는 Max_Log MAP복호 방식이 약 0.5 dB의 Eb/No감소가 발생되나, BER : 10⁻⁶ order 이상에서는 적은 Eb/No의 감소를 보이거나 Max_Log MAP복호 방식과 Log MAP복호 방식사이의 성능의 차이가 거의 없는 것으로 볼 수 있다.

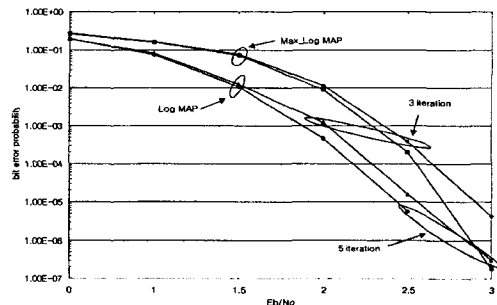


그림 8. Log-MAP과 MAX Log-MAP의 성능
Fig. 8. Performances in the case of Log-MAP and MAX Log-MAP.

IV. 결론

이제까지 IMT2000에 제시된 3GPP와 3GPP2의 터보부호의 성능에 대하여 AWGN 채널과 레일리 페이딩 채널 하에서 수행된 모의실험결과와 Max_Log MAP방식과 Log MAP방식의 터보부호의 성능에 관하여 비교 분석해 보았다. IMT2000의 터보부호로써 3GPP와 3GPP2에서 각각 제시된 터보부호의 성능은 비슷한 블록 크기에서 거의 차이가 없으며, 약간의 성능 감쇄를 감수할 수 있다고 한다면, 하드웨어 복호기 설계시 시간지연이 적고 훨씬 덜 복잡한 방식인 Max_Log MAP을 선택하여 복호기를 설계하는 것이 타당하다고 여겨진다. 터보 부호에서 선택해야 하는 인자 중 iteration 횟수가 있는데, 이 횟수는 AWGN 채널 상에서 최소 3번 이상만 수행하면 Eb/No가 3dB에서 BER : 10^{-6} order의 충분한 성능을 얻을 수 있다.

터보 부호의 MAP 복호를 이용하여 BER을 계산하는데 사용한 컴퓨터는 SUN의 Enterprise450 4cpu 1대와 SUN Ultra10 8대였다. 그럼에도 불구하고 터보 MAP 복호의 엄청난 계산량 때문에 보다 3GPP를 비롯한 페이딩 채널하의 모의실험의 충분한 데이터를 제공하지 못하였다. 향후 계속적인 모의실험을 통하여 미흡한 데이터를 보완하고, 실제 채널환경에 보다 가까운 채널환경을 구축하고, ISI(Inter Symbol Interference)와 Power Control scheme 등을 비롯한 시스템적인 요소들을 고려함으로써 시스템 측면에서의 모의실험이 수행되어 채널부호화 방법 선택과 방식에 보다 많은 정보를 제공하여야 한다.

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)*, 1993, pp. 1064-1070.

[2] 3GPP, " <http://www.3gpp.org>".

[3] 3GPP2, " <http://www.3gpp2.org>".

[4] L. Papke, P. Robertson, and E. Villebrun, "Improved decoding with the SOVA in a parallel concatenated (turbo-code) scheme," *Proc. of ICC '96*, Dallas, TX, USA, June, 1996, pp. 102-106.

[5] S. S. Pietrobon, "Implementation and performance of a serial MAP decoder for use in an iterative turbo decoder," *IEEE Int. Symp. Inform. theory*, Whistler, British Columbia, Canada, p.471, Sep.1995.

[6] J. Hagenauer, "Viterbi Decoding of Convolutional codes for Fading and Burst Channels." *Proc. of the 1980 zurich Seminar on Digital Communication*. pp. G2.1-G2.7, 1980.

[7] S. G. Wilson, *Digital Modulation and Coding*. Prentice Hall, NJ, 1996.

저 자 소 개



玄星煥(正會員)
1971년 12월 24일생, 1998년 2월 KAIST 전기및전자공학과 학사 졸업, 2000년 2월 ICU 석사 졸업, 2000년 1월 LG 종합기술원 정보기술연구소 입사 (CT그룹). 주관심분야는 터보코드, WCDMA 단말기 구현



朴 晨 鐘(正會員)
1945년 10월 8일생, 1968년 서울공대 응용물리학과 학사, 1978년 University of Minnesota, Min- nesota, U.S.A. Ph. D., 1978~1982 국방과학연구소 책임연구원, 1983~1985 KIET 책임연구원, 1985~1997 ETRI 반도체담당장, 1996~1997 KAIST 초빙교수, 1998~현재 ICU 교수. 주관심분야: Coding, Modulation, 이동통신 단말기술, 반도체 설계



李 秉 秀(正會員)
1974년 6월 28일생, 1998년 2월 KAIST 전기및전자공학과 학사 졸업, 2000년 2월 ICU 석사 졸업. 주관심분야: 코드동기, 단말기 시스템, 터보코드, 무선단말기 구현