

시간 다이버시티 효과를 증대시키는 새로운 ETD-터보 코드 적용시스템의 성능분석

The performance analysis of a novel enhanced turbo coded system with increased time diversity effect

고 연 화*, 하 덕 호*

Yeon-Hwa Ko, Deock-Ho Ha

요 약

본 논문에서는 시간 다이버시티 효과를 증대시키는 새로운 터보 코드 형태인 ETD-터보 코드(Enhanced Time Diversity-turbo code)를 구성하고, 이를 MC-CDMA 시스템에 적용하여 그 효율성을 입증하였다. ETD-터보 코드는 기존의 터보 코드 구성에 인터리버를 추가하여 구성하고, 모든 패리티 비트가 인터리버 패턴을 갖게 되어 시간 다이버시티 효과가 증대된다. 시스템 성능분석을 위해 인터리버 조합에 따른 ETD-터보 코드의 특성을 비교하고, MC-CDMA 시스템에 채널 부호화 기법으로 기존의 터보 코드와 본 논문에서 제안한 ETD-터보 코드를 적용하여 시스템 성능을 분석한다. 그 결과 ETD-터보 코드는 기존의 터보 코드방식에 비해 비교적 단순한 인터리버 조합으로도 BER을 감소시키고, 반복 복호 횟수를 줄임으로써 복호과정에서의 시간지연을 감소시킬 수 있다. 따라서 ETD-터보 코드를 적용한 MC-CDMA의 전체 시스템 오율특성이 개선됨을 알 수 있다.

Abstract

In this paper, we propose a ETD-turbo code(Enhanced Time Diversity turbo code) which is a novel turbo code configuration to increase the time diversity effect and analyze the performance of ETD-turbo coded MC-CDMA system. The ETD-turbo code, which is added another interleaver to the conventional turbo code structure, is consisted. Time diversity effect of the ETD-turbo code is improved by every parity bits converted into interleaver pattern. In order to the performance of the ETD-turbo code, we conduct a computer simulation about interleaver type. And we make comparison between the performance of ETD-turbo coded MC-CDMA system and the conventional turbo coded MC-CDMA system. By the simulation results, ETD-turbo code has less BER than the conventional turbo code and time delay is decreased by reducing iteration numbers. Therefore, it is defined that the performance of ETD-turbo coded MC-CDMA system is improved than the conventional turbo coded MC-CDMA system.

Key words : ETD-turbo code, Turbo code, Time diversity, MC-CDMA

I. 서론

차세대 이동통신은 대용량의 멀티미디어 정보를 고속의 고효율 전송을 필요로 한다. 특히 IMT-2000의 동기식 표준화 규격을 정의하는 3GPP2(3rd generation partnership project2)에서는 cdma2000을 표준으로 하고 있다. 다중 반송파를 사용하는 MC-CDMA시스템은 협대

역의 다중 반송파를 이용하는 방식으로 IMT-2000 동기식 방식의 순방향 채널(forward channel) 표준으로 사용하며 DS-SS-CDMA에 비해 증가된 용량으로 고속의 데이터 전송이 가능하다. 이동무선채널의 경우 열악한 환경으로 인해 전송의 신뢰성이 떨어지므로 이에 대한 오류 정정 기법이 중요하게 된다. 최근에는 오류 정정 능력이 뛰어난 터보 코드가 차세대 이동통신의 채널 코드의 표준으로 채택되고 있다. 터보 코드는 컨볼루셔널 코드(convolutional code)를 병렬 형태로 구성하고 이들 사이에 인터리버를 위치하도록 구성한 것으로 이론적으로는 Shannon limit의 0.5[dB] 까지 근접하는 우수한 채널 코딩 기법으로 알려져 있다[1]. 앞서 언급한바와 같이 터보

*부경대학교 정보통신공학과

접수 일자 : 2003. 8. 13 수정 완료 : 2003. 10. 15

논문 번호 : 2003-3-13

※이 논문은 2001학년도 부경대학교발전기금재단의 지원에 의하여 연구되었음

코드는 인터리버를 사용하는데, 이것은 입력 데이터열의 패턴을 변화시킴으로써 연접오류를 랜덤 오류로 전환하는 특성을 가진다. 적용하는 인터리버에 따라 성능이 달라지며 복호과정에서 반복복호를 수행함으로써 처리 지연시간이 길어지는 단점을 가진다.

본 논문에서는 기존의 터보 코드의 성능을 개선하기 위해 새로운 터보 코드를 구성하고 이를 MC-CDMA 시스템에 적용하여 그 성능을 분석하고자 한다. 제안된 터보 코드는 시간 다이버시티 효과를 증대시키는 터보 코드라는 의미로 "ETD(Enhanced Time Diversity)-터보 코드"로 정의한다. 기존의 터보 코드의 구성에서 첫 번째 RSC(recursive systematic convolutional)와 두 번째 RSC 사이에 인터리버를 위치하도록 한 반면에 ETD-터보 코드의 경우 두 RSC가 각각의 인터리버를 통과한 데이터를 부호화하도록 구성하였다. 따라서 부호화된 데이터 열은 기존의 터보 코드에 비해 시간다이버시티 효과가 증대되며 block 인터리버와 같이 단순한 인터리버들을 사용하여 우수한 시스템 성능을 얻을 수 있다. 뿐만 아니라 반복 복호 횟수를 줄임으로써 시스템의 처리지연시간을 줄일 수 있는 방식이다.

논문의 구성은 II장에서 MC-CDMA 시스템에 대해 간략히 설명하고, III장에서는 기존의 터보 부호와 제안하는 ETD-터보 코드를 구성을 비교하고 인터리버의 종류에 대해 간단히 설명한다. IV장에서는 인터리버의 조합에 따른 기존의 터보 코드와 ETD-터보 코드의 오율특성과 ETD-터보 코드를 적용한 MC-CDMA 시스템의 성능을 컴퓨터 시뮬레이션을 통해 나타낸다. 그리고 V장에서는 결론을 맺고 향후 연구 방향에 대해 논한다.

II. MC-CDMA 시스템

1993년 N. Yee와 J. P. Linnartz에 의해 처음으로 제안된 MC-CDMA 시스템은 주어진 확산부호계열을 이용하여 데이터의 주파수 스펙트럼을 확산한 뒤 주파수 영역에서 서로 다른 반송파 즉, 다수개의 협대역 부반송파를 이용하여 전송하는 방식이다[2]. 기존의 DS-CDMA 방식에서는 채널의 지연 확산이 데이터의 심볼 구간보다 클 경우에 ISI(Inter-symbol interference)가 발생하고, ISI의 영향을 없애기 위해 낮은 전송률 전송할 경우에는 ICI(Inter-chip interference)가 발생하는 문제점이 있었다. 이를 해결하기 위해서는 채널의 추정, 등화기, 전력제어기, RAKE 수신기 등의 부가적인 장치를 사용하면 되지만 시스템의 구성이 복잡해지게 된다. 따라서 이와 같은 문제점들을 해결하기 위해 다수의 부반송파를 이용한 변조 기술로 MC-CDMA 방식이 제안되었다. 이 방식은 반송파 단위의 대역 확산 없이 협대역의 직교 부반송파에 동일한 데이터 심볼을 전송함으로써 ICI 문제를 개선할 수 있으며, 주파수 다이버시티 효과를 얻을 수 있다.

그림 1과 그림 2는 각각 MC-CDMA 시스템의 송신

기와 수신기 구조이다. 데이터를 주어진 확산 부호를 사용하여 주파수 영역에서 서로 다른 반송파 즉, 다수개의 협대역 부반송파를 통하여 전송한다. 각 부반송파별로 협대역으로 확산이 수행되므로 확산 부호가 커질 때 발생하는 신호 처리의 복잡성이 줄어들며 동일한 부반송파내에서 각 사용자는 직교 부호에 의해 구별된다.

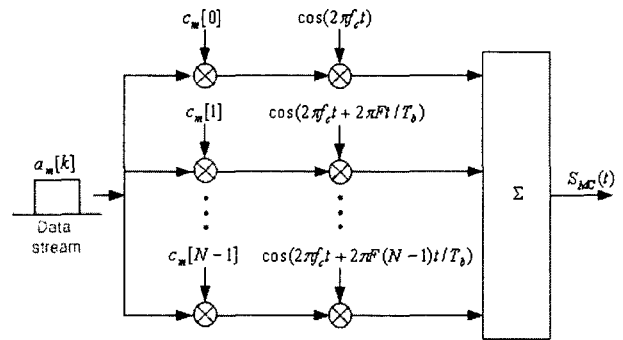


그림 1. MC-CDMA 송신기
Fig. 1 MC-CDMA transmitter

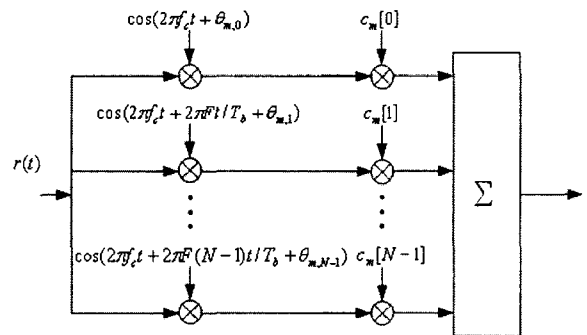


그림 2. MC-CDMA 수신기
Fig. 2 MC-CDMA receiver

m 번째 사용자의 k 번째 데이터 비트에 대응되는 송신 신호는 다음 식과 같이 표현된다.

$$s_{MC}(t) = \sum_{i=1}^{N-1} c_m[i] a_m[k] c \left(2\pi f_c t + 2\pi i \frac{F}{T_b} t \right) \cdot p_{T_b}(t - kT_b) \quad (1)$$

여기서, $c_m[i] \in \{-1, 1\}$ 는 $i = 0, 1, \dots, N-1$ 일 때 m 번째 사용자의 확산 부호이고, $p_{T_b}(t)$ 는 구간 $[0, T_b]$ 에서 단위 크기를 갖는 펄스 파형이다. 입력 데이터 심볼 $a_m[k]$ 는 $\{-1, 1\}$ 로 동일 확률로 전송된다. F/T_b 는 부반송파의 간격을 나타내는 것으로서 T_b 는 비트구간이고, F 는 정수이다. 그리고 채널을 통해 수신된 신호는 다음과 같이 표현된다.

$$r(t) = \sum_{m=0}^{M-1} \sum_{i=0}^{N-1} \rho_{m,i} c_m [i] a_m [k] \cos(2\pi f_c t + 2\pi i \frac{F}{T_b} t + \theta_{m,i}) + n(t) \quad (2)$$

이에서 $\rho_{m,i}$ 와 $\theta_{m,i}$ 는 각각 채널에 의한 랜덤 진폭과 위상 오프셋(offset)이며, $n(t)$ 는 AWGN이다.

III. 터보 코드와 ETD-터보 코드

1993년 Berrou, Glavieux, Thitimajshima에 의해 소개된 터보 코드는 BCJR 알고리즘[3]을 개량한 방식으로, 인터리버와 반복 복호 알고리즘을 통해 Shannon의 한계까지 근접하는 오류 정정 성능을 나타낸다. 특히, AWGN 채널에서 코드율을 1/2, 인터리버의 크기를 256×256, 반복 복호 횟수를 18회 정도로 하였을 때 10^{-5} 의 BER을 얻기 위해 필요한 E_b/N_0 가 0.7[dB]까지 근접함을 밝혔다[3]. 터보코드는 반복 복호 횟수와 인터리버의 크기를 조절할 수 있어 원하는 성능을 가변적으로 얻을 수 있으며, 우수한 오류 정정 특성으로 IMT-2000과 위성 통신 등에 채널부호로 채택되고 있다[4].

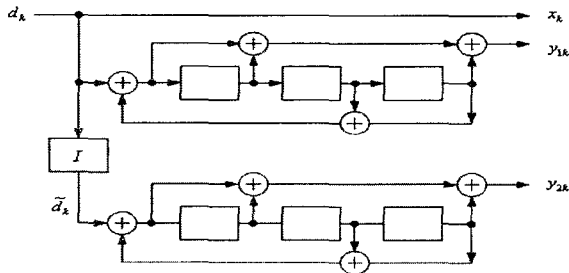


그림 3. 기존의 터보 코드 부호기
Fig. 3 Conventional turbo encoder

그림 3은 일반적인 터보 코드 부호기로서 하나의 인터리버와 두 개의 RSC(recursive systematic convolutional) 부호기가 병렬로 구성되어 있다[5]. 첫 번째 RSC 부호화는 입력 프레임의 비트(d_k)의 원래 순서에 따라 부호화 (y_{1k})를 수행하고, 두 번째 RSC 부호화는 입력 프레임의 비트가 인터리버에 의해 재배열(\tilde{d}_k)된 후 부호화되어 y_{2k} 가 생성된다. 따라서 터보 부호의 출력은 RSC 부호기의 출력 뿐 아니라, 인터리버에 의해 재배열된 출력으로 인한 이중의 패리티 정보를 지니게 된다. 또한 두 개의 구성 부호화기로부터 나온 패리티 스트림은 펀칭(puncturing) 과정을 통해 원하는 코드율을 얻을 수 있다. 터보 부호화 과정에서의 인터리버는 병렬 연결된 한 RSC 부호화기로부터 나오는 낮은 가중치의 코드워드와 다른 RSC 부호화기의 낮은 가중치의 코드워드와 서로 엮이 되는 것을 피하게 해주고, 출력되는 코드워드의 순서를 변화시켜서 복호 과정에서 시간 다이버시티 효과를

얻게 해준다.

터보 코드는 상대적으로 단순한 경관정(soft decision) 디코딩 알고리즘을 사용하여 강력한 코드를 형성하는 능력을 제공한다. 이것은 터보 복호기가 비트 신뢰도 지수를 공유하는 단순한 구성 복호화기로 구성되어 있기 때문이다. 터보 복호기에 사용되는 알고리즘으로는 MAP(Maximum A Posteriori)와 SOVA(Soft Output Viterbi Algorithm)이 있다. MAP 알고리즘은 성능이 우수하지만 복호 매트릭에 많은 지수항수를 포함하므로 복잡도가 높다. 이러한 점을 개선시킨 Log-MAP과 Max-Log-MAP 알고리즘이 많이 사용된다. SOVA는 Viterbi 알고리즘을 변형한 형태로 복잡도는 낮지만 성능이 떨어진다.

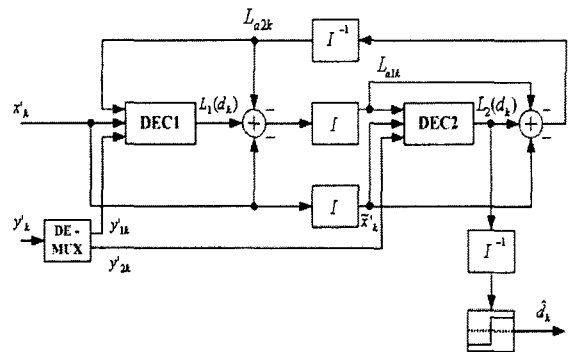


그림 4. 기존의 터보 코드 복호기
Fig. 4 Conventional turbo decoder

그림 4는 터보 코드 복호기로서 구성 복호기(DEC1, DEC2), 인터리버와 디인터리버로 구성되어 있으며 반복 복호(iterative decoding)를 통해 BER성능을 향상시킨다. 구성 복호기에는 수신된 정보 비트 x'_k 와 패리티 비트 y'_k , 그리고 복호 과정에서 생성된 사전 정보(priori information)인 L_{ak} 가 입력되고, 각 비트에 대한 LLR(Log-Likelihood Ratio) 값인 $L(d_k)$ 가 출력된다. 각 복호기의 출력 $L(d_k)$ 는 다른 복호기의 사전정보 L_{ak} 를 생성하여 반복 복호 과정에서 사용된다. 반복 복호 단계에서 DEC1은 x'_k, y'_{1k} , DEC2에 의해 생성된 사전정보 L_{a2k} 를 사용하여 LLR값인 $L_1(d_k)$ 를 생성한다. 이러한 $L_1(d_k)$ 는 입력된 정보와 상관성을 없애기 위해 DEC1로 입력된 x'_k 와 사전정보인 L_{a2k} 가 제거되어 DEC1의 외부정보가 되고, 인터리버를 거치면서 DEC2를 위한 사전정보 L_{a1k} 가 된다. DEC2에는 \tilde{x}'_k, y'_{2k} 그리고 DEC1에서 얻어진 사전정보 L_{a1k} 가 입력되고 $L_2(d_k)$ 가 출력된다. 여기서 수신된 정보비트 x'_k 와 L_{a1k} 가 인터리빙되어 DEC2에 입력되는 이유는 y'_{2k} 가 부호기에서 인터리버를 거친 입력열의 출력 패리티 비트이므로 인터리버 순서를 맞춰야 복호가 가능하기 때문이다. 그리고 $L_2(d_k)$ 에서 \tilde{x}'_k 와 L_{a1k} 가 제거되어 DEC2에 대한 외부정보가 되고, 디인터리버를 통해 DEC1

의 사전정보 L_{a2k} 가 된다. 이러한 과정으로 적절한 횟수의 반복 복호가 이루어지면 최종 $L_2(d_k)$ 를 디인터리빙하여 경관정함으로써 \hat{d}_k 를 결정하게 된다.

본 논문에서 제안하는 시간 다이버시티 효과를 증가시키는 ETD-터보 코드는 기존의 터보 코드의 구성에 인터리버를 추가하여 입력 데이터는 인터리버를 통과한 후 RSC 부호기에 의해 부호화하도록 구성한 것이다. 그림 5는 ETD-터보 코드에 대한 부호기 구성도이다. 이를 기존의 터보 코드 부호기와 비교하여보자. 우선 기존의 터보 코드 부호기는 첫 번째 RSC 부호기의 결과 데이터 y_{1k} 는 입력 데이터와 동일한 패턴을 유지하지만, ETD-터보 코드는 첫 번째 RSC 부호기 앞단에 인터리버 I_1 이 위치함으로써 결과 데이터인 y_{1k} 는 입력 데이터의 열과 동일한 패턴을 가지지 않는다. 뿐만 아니라 두 번째 RSC 부호기는 두 개의 인터리버를 통과한 데이터를 입력으로 하여 결과데이터 y_{2k} 는 입력 데이터뿐만 아니라 y_{1k} 와도 다른 패턴을 가지게 된다. 그림에서 d_k 는 입력 비트 열이고, I_1 과 I_2 는 인터리버이다. x_k 는 입력 비트 열과 동일한 패턴을 가지는 출력 비트 열이고, y_{1k} 은 인터리버 I_1 에 의한 첫 번째 RSC의 출력 비트 열이며 y_{2k} 는 인터리버 I_2 에 의한 두 번째 RSC의 출력 비트 열에 해당한다. 여기서 k 는 입력 비트열의 입력 순서를 의미한다. 만일 $I_1=[5 6 9 1 4 2 8 3 7]$, I_2 과=[8 2 7 4 3 6 9 5 1]과 같은 패턴을 가지는 경우라면, 첫 번째 RSC의 입력 열 \tilde{d}_k 는 I_1 과 동일한 인터리빙 패턴으로 나타내고, 두 번째 RSC의 입력 열 $\tilde{\tilde{d}}_k$ 는 I_1 과 I_2 를 거쳐 새로운 인터리빙 패턴인 $I_3=[3 6 8 1 9 2 7 4 5]$ 와 같은 순서를 가지게 된다. 이러한 ETD-터보 코드 부호기의 출력 비트 열은 표 1과 같이 나타낼 수 있다.

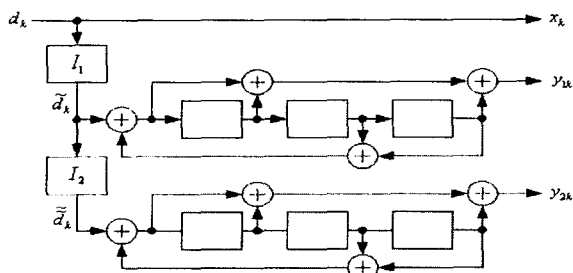


그림 5. ETD-터보 코드 부호기
Fig. 5 ETD-turbo encoder

표 4. ETD-터보 코드 부호기의 출력
Table 1. Output of the ETD-turbo encoder

x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9
y_{15}	y_{16}	y_{19}	y_{11}	y_{14}	y_{12}	y_{18}	y_{13}	y_{17}
y_{23}	y_{26}	y_{28}	y_{21}	y_{29}	y_{22}	y_{27}	y_{24}	y_{25}

결과적으로, 기존의 터보 코드 부호기의 출력 x_k 와 y_{1k} 는 인터리버의 영향을 받지 않으므로 입력 비트 열과 동일한 순서를 갖게 되고 인터리버를 통과한 y_{2k} 만 인터리빙 패턴의 순서를 가지고 채널을 통과하게 되지만, ETD-터보 코드 부호기의 출력은 x_k 를 제외한 y_{1k} 와 y_{2k} 가 서로 다른 인터리빙 패턴으로 채널을 통과하게 되므로 더 큰 시간 다이버시티 효과를 기대할 수 있다.

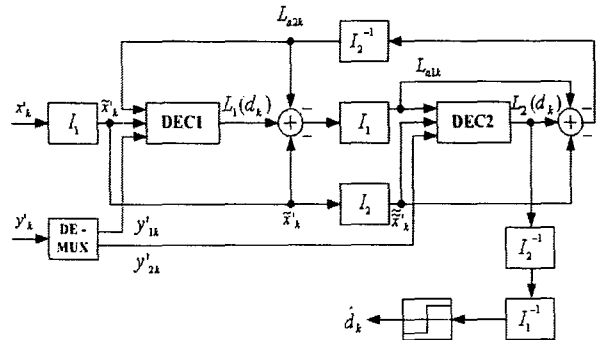


그림 6. ETD-터보 코드 복호기
Fig. 6 ETD-turbo decoder

그림 6은 ETD-터보 코드 복호기의 구조이다. 복호기에는 수신된 x'_k 와 y'_k 가 입력된다. 여기서 x'_k 는 부호기의 정보 출력 비트 x_k 가 채널을 통과한 형태를 의미하고, y'_k 는 패리티 출력비트가 채널을 통과한 형태로서 역다중화기를 이용하여 y'_{1k} 와 y'_{2k} 를 분리한다. y'_{1k} 는 첫 번째 인터리버 I_1 을 거친 출력 패리티 비트 그리고 y'_{2k} 는 I_1 과 I_2 를 모두 거친 두 번째 패리티 비트를 의미한다. ETD-터보 코드 복호기는 DEC1과 DEC2 사이에 인터리버와 디인터리버를 이용하여 디코더 사이에 전달되는 LLR값의 순서를 맞추어 주게 된다.

표 5. DEC1의 입력
Table 2. Input of the DEC1

\tilde{x}'_5	\tilde{x}'_6	\tilde{x}'_9	\tilde{x}'_1	\tilde{x}'_4	\tilde{x}'_2	\tilde{x}'_8	\tilde{x}'_3	\tilde{x}'_7
y'_{15}	y'_{16}	y'_{19}	y'_{11}	y'_{14}	y'_{12}	y'_{18}	y'_{13}	y'_{17}

표 6. DEC2의 입력

Table 3. Input of the DEC2

$\tilde{\tilde{x}}'_3$	$\tilde{\tilde{x}}'_6$	$\tilde{\tilde{x}}'_8$	$\tilde{\tilde{x}}'_1$	$\tilde{\tilde{x}}'_9$	$\tilde{\tilde{x}}'_2$	$\tilde{\tilde{x}}'_7$	$\tilde{\tilde{x}}'_4$	$\tilde{\tilde{x}}'_5$
y'_{23}	y'_{26}	y'_{28}	y'_{21}	y'_{29}	y'_{22}	y'_{27}	y'_{24}	y'_{25}

표 2는 DEC1의 입력 순서를 나타내고 있다. x'_k 는 ETD-터보 부호기에서 입력된 입력 비트열의 순서 [1 2 3 4 5 6 7 8 9]를 그대로 가지고 있지만, y'_{1k} 와 순서를 맞추어서 복호하기 위해서 DEC1에 입력되기 전 $I_1=[5 6 7 8 9 1 2 3 4]$ 에 의해 순서가 변환된 \tilde{x}'_k 가 DEC1에 입력된다. 표 3은 DEC2의 입력 순서를 나타내고 있다. 여기서도 DEC1과 마찬가지로 x'_k 를 y'_{2k} 와 순서를 동일하게 하기 위해 I_1 과 I_2 를 거친 $I_3=[3 6 8 1 9 2 7 4 5]$ 의 순서로 DEC2에 입력 되게 된다.

DEC1에서 출력된 LLR값인 $\tilde{L}_1(d_k)$ 는 \tilde{x}'_k 와 사전정보 \tilde{L}_{e2k} 가 제거되면서 DEC1에서 얻어진 순수한 외부정보인 \tilde{L}_{e1k} 가 된다. 이러한 외부정보는 다시 두 번째 인터리버인 I_2 를 거치면서 DEC2의 사전정보 \tilde{L}_{e1k} 가 된다. DEC2에는 사전정보 \tilde{L}_{e1k} , 두 개의 인터리버를 거친 \tilde{x}'_k 와 y'_{2k} 가 입력된다. 여기서 tilde 표시가 두 번 사용된 것은 인터리버 I_1 과 I_2 를 모두 거친 열이라는 것을 나타내고 있다. DEC2에서 출력된 $\tilde{L}_2(d_k)$ 는 \tilde{x}'_k 와 \tilde{L}_{e1k} 가 제거되어 DEC2에서 얻어진 외부정보 \tilde{L}_{e2k} 가 된다. \tilde{L}_{e2k} 는 다시 I_2^{-1} 에 의해 디인터리빙 되면서 DEC1의 사전정보 \tilde{L}_{e1k} 가 된다. 이러한 과정을 통해 충분한 반복 복호가 이루어진 후 $\tilde{L}_2(d_k)$ 는 I_3^{-1} 에 의해 디인터리빙 되어 I_1 과 I_2 의 영향을 벗어나게 되고 경판정에 의해 정보 비트 \hat{d}_k 를 결정하게 된다. 여기서 MAP 디코더의 디코딩 과정은 다음과 같다. 만약 입력 정보 비트를 d_k , 시간 k 에서의 패리티 비트를 y_k 라 하고 채널을 통해 수신된 시퀀스가 $R_1^N = (R_1, R_2, \dots, R_N)$ 라고 가정하면, 시간 k 에서 수신된 시퀀스는 $R_k = (x'_k, y'_k)$ 이다. 그리고 수신된 정보비트 x'_k 와 패리티 비트 y'_k 는 다음과 같이 나타낼 수 있다[6].

$$x'_k = (2d_k - 1) + p_k \quad (3)$$

$$y'_k = (2q_k - 1) + q_k \quad (4)$$

여기서, p_k 와 q_k 는 각각 평균이 0이고 분산이 $\sigma^2 = N_0/2$ 인 랜덤 변수를 의미한다. 시간 k 에서의 입력 심볼에 대한 LLR은 다음과 같다.

$$L(d_k) = \log \left(\frac{P(d_k = 1 | R_1^N)}{P(d_k = 0 | R_1^N)} \right) \quad (5)$$

여기서 $P(d_k = i | R_1^N)$ 는 수신된 시퀀스 R_1^N 중에서 $d_k = i$ ($i = 0, 1$)일 때 사전 확률값 (posteriori probability)이다. 수신된 심볼에 대한 최종 $L(d_k)$ 는 구성부호기 정보와 사전정보, 그리고 외부정보(extrinsic information)

로 결정되며 아래와 같이 나타낼 수 있다.

$$L(d_k) = \log \left(\frac{P(y'_k | d_k = 1)}{P(y'_k | d_k = 0)} \right) + \log \left(\frac{P(d_k = 1)}{P(d_k = 0)} \right) + \log \left(\frac{\sum_{m=0}^{q-1} \sum_{m'=0}^{q-1} \gamma'_k(R_k, m', m) \alpha_{k-1}(m') \beta_k(m)}{\sum_{m=0}^{q-1} \sum_{m'=0}^{q-1} \gamma''_k(R_k, m', m) \alpha_{k-1}(m') \beta_k(m)} \right) \quad (6)$$

$$= L_{\text{systematic}} + L_{\text{priori}} + L_{\text{extrinsic}}$$

여기서, m 은 시간 k 에서의 부호기 상태를, m' 은 시간 $k-1$ 에서의 부호기 상태를 의미한다. $\gamma'_k(R_k, m', m)$ 은 심볼 x_k 와 y_k 가 상태 m' 에서 m 으로의 천이에 의해 얻어지는 천이 확률이다. $\alpha_k(m)$ 과 $\beta_k(m)$ 은 각각 순방향과 역방향에 대한 시간 k 에서의 상태 메트릭에 대한 가능도비이다. 그리고, $L_{\text{systematic}}$ 은 시간 k 에서 수신된 정보 심볼에 대한 확률 값으로 systematic 데이터이고, L_{priori} 는 입력 비트 d_k 에 대한 사전정보이며, $L_{\text{extrinsic}}$ 은 입력 비트 d_k 에 대한 외부정보이다. 복호기는 경판정 출력 $L(d_k)$ 를 임계값과 비교하여 정보비트 \hat{d}_k 를 결정한다.

$$\hat{d}_k = \begin{cases} 1 & : L(d_k) \geq 0 \\ 0 & : L(d_k) < 0 \end{cases} \quad (7)$$

일반적으로 인터리버는 채널에서 발생하는 연접 오류를 랜덤 오류로 전환하는 역할을 한다. 터보 부호에서 부호화한 비트들 간의 상관성이 크면 복호과정에 필요한 많은 정보를 얻을 수 있지만 오류에 대한 상관성도 증가하게 된다. 이때 인터리버에 의해 오류에 대한 상관성을 제거하게 된다. 그러나 인터리버의 크기나 종류에 따라 그 성능이 달라진다.

블록 인터리버는(block interleaver) 일반적으로 $m \times n$ 형태의 행렬로 구성되며 정보는 각 행에 순차적으로 쓰여 지고 열의 방향으로 정보를 읽게 된다. 구성 형태는 간단하지만 연접오류가 행렬의 길이보다 큰 경우 완전한 랜덤 오류로 전환할 수 없는 단점이 있다. 헬리컬 인터리버(helical interleaver)는 일반적으로 $m \times (m+1)$ 행렬로 구성되며 정보는 블록 인터리버와 마찬가지로 쓰이지만, 행렬의 $(m, 1)$ 위치에서 대각선으로 읽음으로써 블록 인터리버보다는 좀더 랜덤한 특성을 갖는다. 랜덤 인터리버(random interleaver)는 인터리빙 패턴을 랜덤하게 선택하는 것이다. 따라서 실시간 형태의 랜덤 인터리버는 구현하기가 어려우며 단지 랜덤 수열 발생기에 의해 발생하는 수열을 저장하여 사용함으로써 랜덤 인터리버를 구성하게 된다. 성능 면에서는 랜덤 인터리버가 가장 우수한 것으로 알려져 있으며 터보 부호의 성능 특성과 터보 부호를 위한 인터리버를 연구하는데 있어서 기준이 되는 인터리버이다.

본 논문에서 제안하는 ETD-터보 코드의 경우 두 개의 RSC 모두 각각의 인터리버를 경유하게 되므로 기존

의 터보 코드가 하나의 인터리버로 RSC를 구분하였던 것과는 다른 성능을 나타낼 것이다. 따라서 인터리버 선택에 따른 성능 분석은 IV장의 시뮬레이션 결과에서 다시 논하도록 한다.

IV. 시뮬레이션 결과

본 논문에서 사용한 시뮬레이션은 AWGN 환경에서의 기존의 터보 코드방식과 새로 제안한 ETD-터보 코드 방식을 여러 형태로 비교하였고, 이들 채널코딩 기법을 MC-CDMA 시스템에 적용하여 Rayleigh 페이딩 채널에서 시스템의 성능을 평가하였다. 터보 코드와 ETD-터보 코드는 프레임 길이를 1024로 하였고, 복호과정은 Log-MAP 알고리즘을 사용하였으며, 부호율은 1/2로 하였다[7]. 사용된 인터리버로는 블록 인터리버, 헤리컬 인터리버 그리고 랜덤 인터리버이며, ETD-터보 코드의 경우 이들 세 인터리버들을 조합하여 적용하였다. 그리고 MC-CDMA 시스템의 경우 부반송파의 수를 128개로 하였고, BPSK 변조를 하였다.

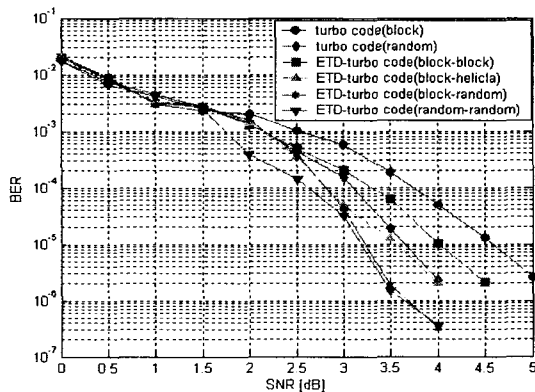


그림 7. 인터리버에 대한 BER
Fig. 7 BER due to the interleaver

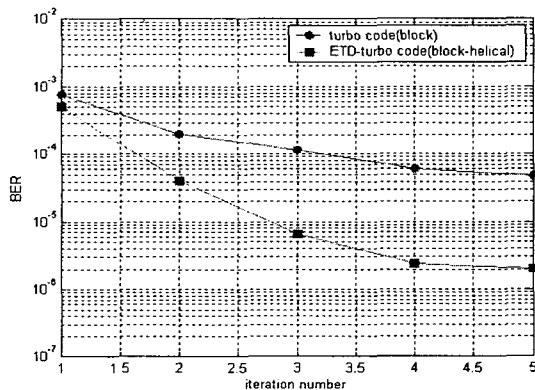


그림 8. 반복 복호 횟수에 따른 BER
Fig. 8 BER due to iteration numbers

그림 7은 인터리버에 대한 터보 코드와 ETD-터보 코

드의 BER 특성을 나타낸 것이다. 이때 복호 과정에서의 반복 복호 횟수는 5번으로 하였다. 그림에서 블록 인터리버를 사용한 터보 코드의 경우 BER 특성이 가장 낮게 나타났고, ETD-터보 코드에서 두 개의 인터리버 모두 랜덤 인터리버를 사용하였을 경우 가장 우수한 BER 특성을 나타내고 있다. 특히 랜덤 인터리버를 사용한 터보 코드 특성보다 블록 인터리버와 헤리컬 인터리버를 사용한 ETD-turbo 코드의 특성이 더 우수함을 알 수 있다. 앞서 언급한바와 같이 랜덤 인터리버의 경우 실시간 형태에서는 구현이 어려운 것으로 알려져 있고, 블록 인터리버나 헤리컬 인터리버의 경우 상대적으로 구성이 매우 간단하지만 랜덤 인터리버에 비해 그 성능이 떨어지는 것으로 알려져 있다. 그러나 ETD-터보 코드의 경우 블록 인터리버와 헤리컬 인터리버를 사용하여 랜덤 인터리버를 사용한 터보 코드보다 더 우수한 BER 특성을 나타내는 것을 그림을 통해 알 수 있다.

그림 8은 반복 복호 횟수에 대한 터보 코드와 ETD-터보 코드의 BER을 비교한 결과이다. 이때 사용된 인터리버는 터보 코드의 경우 블록 인터리버, ETD-터보 코드의 경우 블록 인터리버와 헤리컬 인터리버를 사용한 경우이다. 그리고 송신전력은 4[dB]로 고정하였다. 블록 인터리버를 이용한 기존의 터보 코드가 5회 반복에서 얻을 수 있는 BER을 블록 인터리버와 헤리컬 인터리버를 사용한 ETD-터보 코드는 2회의 반복 복호에서 얻을 수 있다. 이러한 결과로부터 ETD-터보 코드는 반복 복호 횟수를 줄임으로써 터보 코드의 복호과정에서 발생하는 지연시간을 경감시킬 수 있음을 알 수 있다.

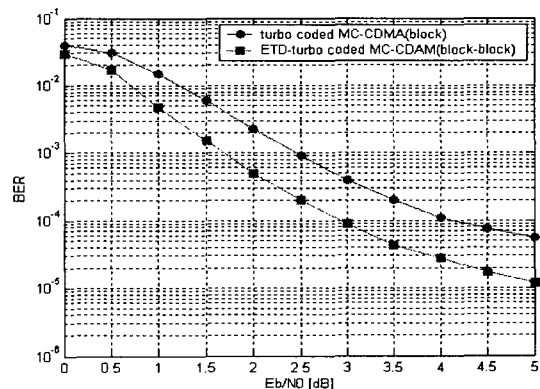


그림 9. 터보 코드와 ETD-터보 코드를 적용한 MC-CDMA 시스템의 BER
Fig. 9 BER of the turbo coded MC-CDMA system and ETD-turbo coded MC-CDMA system

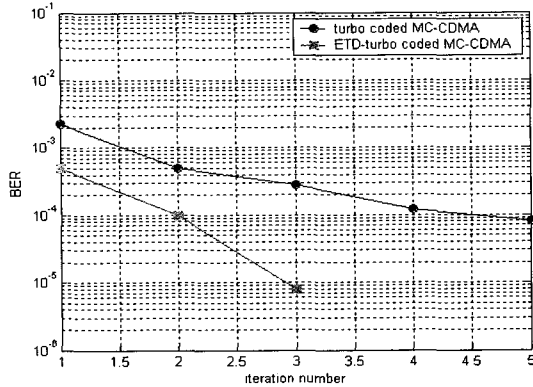


그림 10. 반복 복호 횟수에 따른 부호화 MC-CDMA 시스템의 BER

Fig. 10 BER of coded MC-CDMA system due to iteration number

그림 9와 그림 10은 Rayleigh 페이딩 채널에서 기존의 터보 코드를 적용한 MC-CDMA 시스템과 ETD-터보 코드를 적용한 MC-CDMA 시스템의 오율을 비교한 결과이다. 터보 코드는 블록 인터리버를 사용하였고 ETD-터보 코드도 두 개의 인터리버 모두 블록 인터리버를 사용하였다. 그림 9의 경우 반복 복호 횟수를 1회로 하고 E_b/N_0 [dB]에 대한 성능을 나타낸 것으로서, ETD-터보 코드를 적용한 MC-CDMA 시스템의 BER이 약 1[dB] 정도 개선됨을 알 수 있다. 그림 10은 E_b/N_0 를 2[dB]로 하였을 때 반복 복호 횟수에 대한 BER을 비교한 것이다. 반복 복호 횟수를 증가시키에 따라 두 시스템의 BER이 크게 차이가 남을 알 수 있다. 기존의 터보 코드를 적용한 MC-CDMA의 경우 5회 정도 반복 복호하였을 경우 BER이 10^{-3} 이하로 되지만, ETD-터보 코드를 적용한 MC-CDMA의 경우 3회 정도의 반복 복호로도 더 우수한 BER 특성을 나타냄을 알 수 있다.

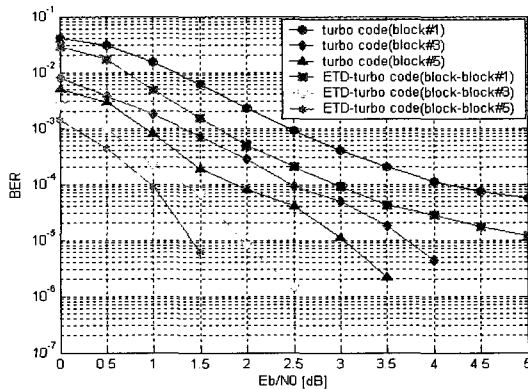


그림 11. 반복 복호 횟수와 SNR에 대한 부호화 MC-CDMA 시스템의 BER

Fig. 11 BER of coded MC-CDMA system due to iteration numbers and SNR

그림 11은 Rayleigh 페이딩 채널 환경에서 블록 인터리버를 사용한 기존의 터보 코드를 적용한 MC-CDMA 시스템과 두 개의 인터리버 모두 블록 인터리버를 사용한 ETD-터보 코드를 적용한 MC-CDMA 시스템에 대해 반복 복호 횟수와 SNR에 대한 BER성능을 비교한 결과이다. 그림에서 #N(단, N=1, 3, 5)는 반복 복호 횟수를 의미한다. 결과에서 반복 복호 횟수가 3회인 ETD-터보 코드를 적용한 MC-CDMA 시스템의 BER이 반복 복호 횟수를 5회로 한 기존의 터보 코드를 적용한 MC-CDMA 시스템의 BER보다 낮게 나타난다. 기존의 터보 코드 구성에서 인터리버를 추가하여 구성한 ETD-터보 코드는 시간 다이버시티의 효과가 증대됨으로써 이를 적용한 시스템은 적은 반복 복호 횟수로 고효율의 BER을 얻을 수 있음을 알 수 있다.

V. 결론

본 논문에서는 Shannon의 용량 한계에 근접하는 채널 코딩 방식으로 알려진 터보 코드의 성능을 향상시키고 단점을 보완하기 위해 ETD-터보 코드를 구성하고, 시물레이션을 통해 기존의 터보 코드와 ETD-터보 코드의 특성을 비교·분석하였다. ETD-터보 코드는 기존의 터보 코드구성에서 인터리버를 하나 더 추가하여 구성함으로써 시간 다이버시티 효과를 증대시키도록 하였다. 그 결과 ETD-터보 코드는 블록 인터리버나 해리컬 인터리버와 같은 단순한 구성의 인터리버 조합으로도 랜덤 인터리버를 적용한 터보 코드보다 더 좋은 BER 특성을 나타내었고, 반복 복호 횟수를 줄일 수 있었다. 랜덤 인터리버의 경우 고성능에 비해 실시간 처리를 위한 구현이 어려운 것으로 알려져 있다. 또한 Rayleigh 페이딩 채널에서 3GPP2의 순방향 채널 표준인 MC-CDMA 시스템에 채널코딩 방식으로 기존의 터보 코드와 ETD-터보 코드를 적용하여 BER 특성을 비교하였다. 그 결과 ETD-터보 코드를 적용한 MC-CDMA 시스템이 SNR 측면에서 뿐만 아니라 반복 복호 측면에서도 기존의 터보 코드를 적용한 MC-CDMA 시스템보다 고효율의 BER 특성을 내었다. 그러나 인터리버의 추가로 알고리즘의 복잡도가 증가하는 문제점이 있으나, 제안된 ETD-터보 코드는 간단한 구조의 인터리버를 적용으로도 시간 다이버시티 효과를 증대시킴으로써 시스템의 BER 성능이 개선되고, 이를 적용한 시스템은 반복 복호 횟수를 줄임으로써 복호 과정에서 발생하는 처리 지연 시간을 감소시킬 수 있다. 따라서 본 논문에서 제안한 ETD-터보 코드를 적용한 시스템은 최적의 인터리버 조합과 적절한 복호 알고리즘을 개발하면 다양한 시스템에 적용하였을 때 더 우수한 성능을 나타낼 것이며, 차세대 이동통신의 채널 코딩 기법으로 적용될 수 있을 것으로 기대된다.

참고문헌

[1] C. Berrou, A. Glavieux and P. Thitimajshima, "Near shannon limit error correcting coding and decoding: Turbo-codes(1)", in Proc., IEEE ICC'93, pp.1064-1070, May 1993.

[2] N. Yee, J. P. M. G. Linnartz and G. Fettweis, "Multi-carrier CDMA indoor wireless networks", in Proc. 4th Int. Symp. PIMRC, Pacific Yokohama, pp.D1.3.1-D1.3.5, Yokohama, Japan, 1993.

[3] L. R. Bahl, J. Cocke, F. Jelinek and J. Raviv, "Optimal decoding of linear codes for minimizing symbol error rate", *IEEE Trans. Information Theory*, vol. IT-20, pp.248-287, March 1974.

[4] 이문호, "IMT-2000에서 채널코드의 터보 인터리버", 전자진흥, 제 9권 6호, 1999, 12.

[5] C. Berrou, A. Glavieux, "Near optimum error correcting coding and decoding: Turbo-codes", *IEEE Transaction on communication*, vol. 44, pp. 1261-1271, Oct. 1996.

[6] Bernard Sklar, "A primer on turbo code concepts", *IEEE communications Magazine*, pp.94-102, 1997.

[7] P. Robertson, E. Villebrun, P. Hoeher, "A comparison of optimal and sub-optimal MAP decoding algorithms operating in the log domain", ICC 95 Seattle, Gateway to Globalization, 1995 IEEE International Conference, Vol. 2, pp.1009-1013, June, 1995.



고 연 화(Yeon-Hwa Ko)

準會員

1996년 2월: 동의대학교 전자통신공학과 (공학사)

1998년 2월: 동의대학교 전자공학과(공학석사)

2000년 3월~현재: 부경대학교 정보통신공학과 박사과정
관심분야 : 이동통신시스템, 채널코딩, 다중화방식



하 덕 호(Deock-Ho Ha)

正會員

1979년 2월: 한양대학교 전자공학과(공학사)

1984년 3월: 일본 Kyoto대학(공학석사)

1987년 3월: 일본 Kyoto대학(공학박사)

1978년 11월~1981년 1월: (주) 금성사 중앙연구소 연구원
1990년 8월~1991년 8월: 캘리포니아 대학(U.C., Davis) 방문 연구교수
2000년 2월~2001년 2월: 콜로라도 주립대학 연구교수
1987년 9월~현재: 부경대학교 정보통신공학과 정교수
[주관심분야] 실내의 전파특성 모델링, 다이버시티 시스템, 디지털이동통신방식, 채널코딩 등