

# 짧은 블록 길이를 갖는 터보부호의 BER 상한값 성능 평가

준회원 이 은 진\*, 김 용\*, 정회원 이 필 중\*

## BER Performance Evaluation of Turbo Codes with Short Block Length Using Upper Bound Technique

Eun jin Lee\*, Young Kim\* Associate Members, Pil Joong Lee\* Regular Member

### 요 약

부호화 시스템의 성능 분석은 일반적으로 BER(Bit Error Rate)을 이용하여 이루어지는데, 많은 시간을 요하는 정확한 BER을 구하는 대신 근사치인 상한값(upper bound)을 이용하는 경우도 많다. 그러므로 실제 값과 가까운 BER 상한값을 구하는 것은 중요한 일이다. 터보부호의 경우 테일 비트(tail bit)에 의한 영향이 무시되는 블록 길이가 긴 경우[1]에 BER 상한값이 보고 되었다. 그러나 테일 달기(tail terminating) 기법을 사용하면서 블록 길이가 짧을 경우 추가되는 테일 비트가 BER 상한값에 미치는 영향은 무시할 수 없게 된다. 따라서 본 논문에서는 테일 비트를 고려한 터보부호의 BER 상한값 계산 방법을 제시한다. 그리고 테일 달기 기법보다 복잡하지만 테일 비트가 없으므로 블록 길이가 짧은 경우 효율적으로 사용될 수 있는 테일 물기(tail-biting) 기법을 이용하는 터보부호의 BER 상한값 계산 방법도 제시한다.

### ABSTRACT

An average bit error rate (BER) is used as the measure of performance of a coded system. One often use the upper bound on BER instead of the exact BER that takes much longer time to get. Therefore it is important to calculate the BER upper bound close to the exact value. For turbo code, [1] provides a BER upper bound when the block length is long enough so that the effect of tail bits can be neglected. When the tail terminating scheme is used and the block length is short, the effect of tail bits cannot be ignored. Thus, this paper suggests the calculation method of the BER upper bound of turbo code using tail terminating scheme, considering tail bits. Also, it suggests the claulation method of BER upper bound of turbo code using tail-biting scheme, which is more complicated but more efficient than tail terminating scheme.

### I. 서론

터보부호는 둘 이상의 재귀 조직 길쌈 부호(recursive systematic convolutional code, RSC)를 구성 부호로 사용하여 인터리버(inter-leaver)와 함께 병렬 연결한 블록 부호이다<sup>[2]</sup>. 그러므로 그 성능 분석 시에는 구성 부호와 인터리버 둘 다를 고려하여야 한다. 터보부호의 인터리버는 블록 길이의 계

승(factorial)만큼 존재하므로 블록 길이가 길어짐에 따라 이를 모두 고려하여 성능 평가한다는 것은 매우 복잡한 일이 된다. 이에 Benedetto는 유니폼 인터리버(uniform interleaver)를 고안하여 모든 인터리버를 사용한 경우들의 평균 오류 성능 상한값(average error performance bound)을 계산하는 방법을 제시하였다<sup>[1]</sup>. 이와 같은 평균 오류 성능 상한값은 터보부호를 설계할 때 인터리버와 무관하게

\* 포항공과대학교 전자컴퓨터공학부 정보보안 및 통신 연구실(ejinlee@oberon.postech.ac.kr)  
 논문번호 : 010108-0521, 접수일자 : 2001년 5월 21일

구성 부호를 설계할 수 있도록 해주며 평균 오류 성능 상한값 보다 우수한 성능을 나타내는 인터리버가 적어도 한 개 이상 존재함을 알려준다.

터보 부호의 평균 오류 성능 상한값은 최대 우도 복호(maximum likelihood decoding)하는 경우 터보 부호의 IRWEF(Input Redundant Weight Enumerating Function)로부터 구할 수 있다. 터보부호는 병렬 연결 부호이므로 그것의 IRWEF는 구성 부호의 IRWEF를 이용하여 구할 수 있다<sup>[1],[3]</sup>. 그러므로 구성 부호의 IRWEF를 구하는 것은 터보부호의 평균 오류 성능 상한값을 구하기 위한 중요한 작업이 된다.

터보부호의 이러한 상한값을 구하기 위한 기존의 방법<sup>[1]</sup>에서는 블록 길이가 긴 경우만을 고려하여 구성 부호의 IRWEF를 구할 때 테일 달기 기법을 사용하면서도 테일 비트의 영향은 무시하고 있다. 블록 길이가 짧은 경우 테일 비트에 의한 효과는 무시할 수 없게 되므로 기존의 방법들을 그대로 사용하면 잘못된 결과를 얻을 수 있다.

그리고 부복호기를 설계할 때 블록 길이는 전송할 데이터량을 고려하여 적절하게 선택한다. 그러므로 블록 길이가 짧은 터보부호 역시 관심의 대상이 될 수 있으며 실제로 [4], [5]의 연구 결과에 따르면 이동통신의 스피치 전송(speech transmission)의 경우 비교적 짧은 블록 길이의 터보부호를 사용하는 것이 효과적이다<sup>[6]</sup>.

따라서 본 논문에서는 짧은 블록 길이를 갖는 경우에도 구성 부호의 정확한 무게 분포(weight distribution)를 구할 수 있는 다음의 두 가지 방법을 제시한다. 첫 번째는 테일 달기 기법에서 테일 비트를 고려한 IRWEF를 구하는 방법이고, 두 번째는 테일 비트가 없는 테일 물기 기법을 사용하는 경우의 IRWEF를 구하는 방법이다.

그리고 본 논문에서는 기존에 제안된 방법으로 구해진 터보부호의 평균 오류 성능 상한값과 본 논문에서 제시하는 방법으로 구해진 터보부호의 평균 오류 성능 상한값을 비교 분석하였으며 이해를 돕기 위한 예제를 제시하였다.

## II. 터보부호의 BER 상한값

그림 1은 길이가  $N$ 인 유니폼 인터리버를 사용하여 두 개의 구성 부호기를 병렬 연결한 터보부호기이다. 그림 1의 병렬 연결 길쌈 부호(Parallel Concatenated Convolutional Code, PCCC)는 코드

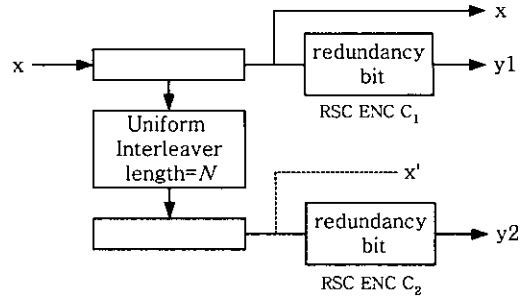


그림 1. 유니폼 인터리버를 사용한 터보부호기

워드(codeword)의 길이가  $3N$ 인 병렬 연결 블록 부호(Parallel Concatenated Block Code, PCBC)와 동일하게 생각할 수 있다<sup>[7]</sup>. 이러한 블록 부호의 IRWEF는 다음과 같은 형태로 표현된다<sup>[1]</sup>.

$$A(W, Z) = \sum_i \sum_j A_{i,j} W^i Z^j \quad (1)$$

여기서  $A_{i,j}$ 는 입력 무게가  $i$  이고 패리티 비트 (parity bit)의 무게가  $j$  인 터보부호의 코드워드 수이다. 이 경우 코드워드의 해밍 무게(Hamming weight)는  $i+j$  가 된다.

식 (1)의 IRWEF를 이용하여 AWGN 채널에서 최대 우도 연성 복호에 의한 터보부호의 BER 상한값은 다음과 같이 구할 수 있다<sup>[1]</sup>.

$$P_b \leq \sum_i \sum_j \frac{i}{N} A_{i,j} Q\left(\sqrt{(i+j) \frac{2R_{TC}E_b}{N_0}}\right) = \sum_d A_d^{TC} Q\left(\sqrt{d \frac{2R_{TC}E_b}{N_0}}\right) \quad (2)$$

여기서  $R_{TC}$ 는 터보부호의 부호율을 의미한다. 그리고  $A_d^{TC}$ 는 다음과 같이 정의된다.

$$A_d^{TC} = \sum_{i+j=d} \frac{i}{N} A_{i,j} \quad (3)$$

이때 그림 1에서와 같이 유니폼 인터리버를 갖는 병렬 연결 부호의 IRWEF는 단순히 구성 부호인 RSC와 동일한 블록 부호  $C_1, C_2$ 의 IRWEF  $A^{C_i}(W, Z)$ 로부터 다음과 같이 유도된다<sup>[1]</sup>.

$$A(W, Z) = \sum_i \sum_j \sum_k \frac{1}{\binom{N}{i}} A_{i,j}^{C_1} A_{i,k}^{C_2} W^{i+j+k} \quad (4)$$

$$A^{C_i}(W, Z) = \sum_i \sum_j A_{i,j}^{C_i} W^i Z^j$$

여기서  $A_{i,j}^C$ 은 입력 무게  $i$ , 패리티 무게  $j$ 인  $n$  번째 구성 부호의 코드워드 수이다.

유니폼 인터리버의 정의<sup>1)</sup>에 의해 식 (4)로부터 구해진 터보부호의 IRWEF를 이용해서 구한  $P_b$ 는 모든 인터리버를 사용한 경우들의  $P_b$ 를 평균한 것과 같은 의미를 갖게 된다<sup>1)</sup>.

### III. 구성부호의 IRWEF와 부호율

길쌈 부호를 블록 부호로 처리하는 방법에는 테일 비트를 첨가하여 매 블록마다 시작 상태(starting state)와 마지막 상태(ending state)를 0으로 만들어 주는 테일 달기 기법과 테일 비트를 사용하지 않고 시작 상태와 마지막 상태를 같도록 만들어주는 테일 물기 기법이 있다<sup>9)</sup>.

테일 달기 기법은 추가되는 테일 비트에 의해 부호율이 감소하는 단점이 있지만 블록 길이가 큰 경우에는 테일 비트의 영향을 무시할 수 있다. 따라서 터보부호의 성능평가를 위한 기존의 방법에서는 큰 블록 길이를 갖는 경우만을 고려하여 테일 비트의 영향은 무시한 방법으로 구성 부호의 IRWEF를 구하였다<sup>1),[3],[7],[8]</sup>.

짧은 블록 길이를 갖는 터보부호의 경우 이러한 방법으로 오류 성능을 구하게 되면 테일 비트의 영향으로 잘못된 결과를 얻게 될 수 있다. 그러므로 테일 달기 기법을 사용한 경우 테일 비트를 고려한 구성 부호의 IRWEF를 구하는 방법을 제시한다. 그리고 테일 비트가 없는 테일 물기 기법을 사용하는 경우의 IRWEF를 구하는 방법을 제시한다. 테일 물기 기법은 복호기가 복잡해지는 단점이 있지만 부호율의 감소가 없기 때문에 테일 비트의 영향이 크게 나타나는 짧은 블록 길이를 갖는 경우 유용하다.

#### 1. Notations

우선 표현을 간단히 하기위해 구성 부호는 부호율이 1/2인 동일한 두 개의 RSC를 고려한다. 그러면 식 (4)는 다음과 같이 표현된다.

$$A(W,Z) = \sum_i \frac{1}{\binom{N}{i}} \left( \sum_j A_{i,j}^C \right)^2 W^i \quad (5)$$

1) [1],[3],[8]에서 길이가  $N$ 인 유니폼 인터리버는 무게가  $i$ 이고 길이가  $N$ 인 입력신호를 동일한 확률  $1/\binom{N}{i}$ 로 전치시키는 확률적인 장치로 정의하고 있다.

블록 길이는 그림 1에서와 같이  $N$ 으로 하고 구성 부호의 메모리 길이는  $m$  이라고 두자. 이 경우 구성 부호의 상태 수는  $2^m$ 이 된다.

그리고 구성 부호의 상태 전이 행렬을  $S(W,Z)$ 라고 한다. 이때  $S(W,Z)$ 의 각 원소  $s_{k,l}$ 는 다음과 같이 쓸 수 있다.

$$s_{k,l} = t_{k,l} W Z^l \quad k, l = 0, 1, \dots, 2^m - 1$$

여기서  $t_{k,l}$ 은 상태  $k$ 에서 상태  $l$ 로 가는 가지(branch)가 존재할 때 입력 무게가  $i$ 이고 패리티 무게가  $j$ 인 가지의 수를 의미한다. 상태 수가  $2^m$ 이므로  $S(W,Z)$ 는  $2^m \times 2^m$ 행렬이 되고 부호율이 1/2인 경우  $S(W,Z)$ 의 각 행의 원소는 두 개만 1을 가지고 나머지는 모두 0이 된다. 예를 들어 그림 2의  $m=2$ 인 간단한 RSC의 경우 상태 전이 행렬은 다음과 같이 된다.

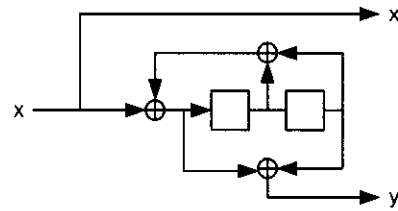


그림 2.  $m=2$ 인 RSC

$$S(W,Z) = \begin{bmatrix} 1 & 0 & WZ & 0 \\ WZ & 0 & 1 & 0 \\ 0 & W & 0 & Z \\ 0 & Z & 0 & W \end{bmatrix}$$

그리고  $S(W,Z)$ 의  $i+1$  번째 행을 행 벡터  $r_i(W,Z)$ 라고 하고  $i+1$  번째 열을 열 벡터  $c_i(W,Z)$ 라고 정의한다.

마지막으로 길쌈부호를 블록 부호로 취급하기 위해 테일 달기 기법을 사용하는 경우 테일 달기 길쌈부호, TTCC라고 부르고 테일 물기 기법을 사용하는 경우 테일 물기 길쌈 부호, TBCC라고 부르는 것이다. 그리고 TTCC, TBCC를 구성 부호로 하는 터보부호를 각각 테일 달기 터보부호, TTTC와 테일 물기 터보부호, TBTC라고 부를 것이다.

#### 2. TTCC의 IRWEF

TTCC는 시작 상태를 0으로 초기화 한 후  $N$  비트의 입력을 부호화하고  $m$  비트의 테일 비트를 추가로 부호화하여 마지막 상태를 0으로 만든다. 이

경우 추가되는 테일 비트도 패리티 비트가 되므로 TTCC는  $(2N+2m, N)$  블록 부호가 된다. 따라서 TTCC의 IRWEF는 다음과 같은 수식으로 얻어진다.

$$A^{TTCC}(W, Z) = r_o(W, Z)S(W, Z)^{N-1} S(Z, Z)^{m-1}c_o(Z, Z) \quad (6)$$

이 식에서 우변의  $r_o(W, Z)S(W, Z)^{N-1}$ 는 입력  $N$  비트를 부호화하는 것을 의미하고  $S(Z, Z)^{m-1}c_o(Z, Z)$ 은 추가되는  $m$  비트의 테일 비트를 부호화하는 것을 의미한다. 이때 테일 비트의 부호화는 입력을 패리티로 처리하므로  $W$ 를  $Z$ 로 쓴다.  $r_o(W, Z)$ 와  $c_o(Z, Z)$ 는 시작 상태와 마지막 상태가 모두 0임을 의미한다.

### 3. TBCC의 IRWEF

TBCC는 추가되는 테일 비트 없이 시작 상태와 마지막 상태가 동일하다는 조건으로부터 IRWEF를 다음과 같이 얻는다.

$$A^{TBCC}(W, Z) = \sum_{i=0}^{2^m-1} r_i(W, Z)S(W, Z)^{N-2}c_i(W, Z) \quad (7)$$

여기서  $r_i(W, Z)$ 와  $c_i(W, Z)$ 는 시작 상태와 마지막 상태가 같음을 의미한다.  $S(W, Z)^{N-2}$ 는 추가되는 테일 비트가 없으므로  $S(W, Z)$ 를  $(N-2)$ 승하여 블록 길이가  $N$ 이 됨을 의미한다. 시작 상태와 마지막 상태는  $2^m$ 가지의 모든 상태가 가능하므로 모든 상태에 대한 합을 해 준다.

### 4. 부호율

2와 3에서 제시한 방법으로 구성 부호의 IRWEF를 구하면 유니폼 인터리버를 갖는 터보부호의 IRWEF를 식 (4), 식 (5)를 이용하여 구할 수 있다. 이러한 방법으로 구한 터보부호의 IRWEF를 이용하여 식 (2)로부터 AWGN 채널에서 터보부호의 BER 상한값을 계산할 수 있다. 이때 TBCC의 경우 추가되는 테일 비트가 없으므로 TBTC의 부호율은  $1/3$ 이 된다. 그러나 TTCC의 경우에는 추가되는 테일 비트가 있으므로 TTTC의 부호율은 TBTC와 다르다. TTTC는 구성 부호의 메모리 길이가  $m$ 이므로 각각의 RSC에서 테일 비트는  $m$  비트 만큼씩 필요하다. 그러므로 입력 길이  $N$ 에 대한 터보부호기의 출력 길이는  $3N+4m$ 으로 증가한다. 즉 TBCC를 이용한 경우 터보부호의 부호율은  $N/(3N+4m)$ 이 된다.

이상의 과정들에 대한 간단한 예제를 살펴보자.

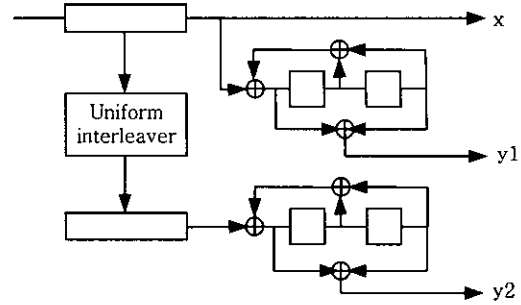


그림 3. (7,5) RSC를 갖는 터보부호기

그림 3은  $N=4, m=2$ 인 간단한 RSC를 구성 부호로 갖는 터보부호이다. 식 (6), 식 (7)을 이용하여 구성 부호의 IRWEF의 계수  $A_{i,j}^C$ 를 구하고 이 값으로 식 (5)를 이용하여 TTTC, TBTC 각각의 IRWEF를 구하면 다음과 같다.

$$A^{TTTC}(W, Z) = \frac{1}{4} W(Z^8 + 4Z^3 + 6Z^{10} + 4Z^{11} + Z^{12}) + \frac{1}{6} W^2(Z^6 + 4Z^7 + 10Z^8 + 12Z^9 + 9Z^{10}) + \frac{1}{4} W^3(4Z^4 + 4Z^5 + Z^6 + 4Z^7 + 2Z^8 + Z^{10}) + W^4 Z^{12}$$

$$A^{TBTC}(W, Z) = 4WZ^4 + \frac{1}{6} W^2(4 + 16Z^4 + 16Z^8) + 4W^3 Z^4 + W^4$$

TTCC, TBCC 각각의 경우 IRWEF의 계수는 표 1에서 보여준다.

표 1. (7,5) RSC의 TTCC, TBCC 각각의 경우 IRWEF의 계수

		$A_{i,j}^{TTCC}$						$A_{i,j}^{TBCC}$						
$i \setminus j$		0	1	2	3	4	5	6	$i \setminus j$	0	1	2	3	4
0	1								0	1				
1					1	2	1		1			4		
2				1	2	3			2	2				4
3			2	1		1			3			4		
4							1		4	1				

부호율은 TBTC의 경우 추가되는 비트가 없으므로  $R_{TBTC}=1/3$  인 반면 TTTC의 경우 테일 비트의

영향으로  $R_{TTC}=1/5$  이 된다. 이제 식 (2)를 이용하여 BER 상한값을 구하면 다음과 같은 결과를 얻게 된다.

$$\begin{aligned}
 P_b^{TTC} &\leq Q(\sqrt{7 \cdot 2E_b/5N_o}) + \frac{7}{6} Q(\sqrt{8 \cdot 2E_b/5N_o}) \\
 &+ \frac{7}{6} Q(\sqrt{9 \cdot 2E_b/5N_o}) + \frac{11}{3} Q(\sqrt{10 \cdot 2E_b/5N_o}) \\
 &+ 4Q(\sqrt{11 \cdot 2E_b/5N_o}) + \frac{11}{4} Q(\sqrt{12 \cdot 2E_b/5N_o}) \\
 &+ \frac{1}{4} Q(\sqrt{13 \cdot 2E_b/5N_o}) + Q(\sqrt{16 \cdot 2E_b/5N_o}) \\
 P_b^{TBTC} &\leq \frac{2}{3} Q(\sqrt{2 \cdot 2E_b/3N_o}) + Q(\sqrt{4 \cdot 2E_b/3N_o}) \\
 &+ 4Q(\sqrt{5 \cdot 2E_b/3N_o}) + \frac{8}{3} Q(\sqrt{6 \cdot 2E_b/3N_o}) \\
 &+ 4Q(\sqrt{7 \cdot 2E_b/3N_o}) + \frac{8}{3} Q(\sqrt{10 \cdot 2E_b/3N_o})
 \end{aligned}$$

#### IV. 분석과 전산 실험

그림 4는 그림 3의 부호율이 1/2인 (7,5) RSC를 갖는 터보부호기를 이용하여  $N=16$ 인 경우, TTTC와 TBTC의 IRWEF 계수  $A_d^{TTC}$ ,  $A_d^{TBTC}$ 와 테일 비트를 무시한 Benedetto의 방법<sup>[1]</sup>으로 얻은 터보부호의 IRWEF의 계수  $A_d^{BEN}$ 를 나타낸 것이다. 이때 테일 물기 기법을 사용한 TBTC의 부호율은 1/3(=7/21)이고 테일 달기 기법을 사용한 TBTC와 BEN<sup>2)</sup>의

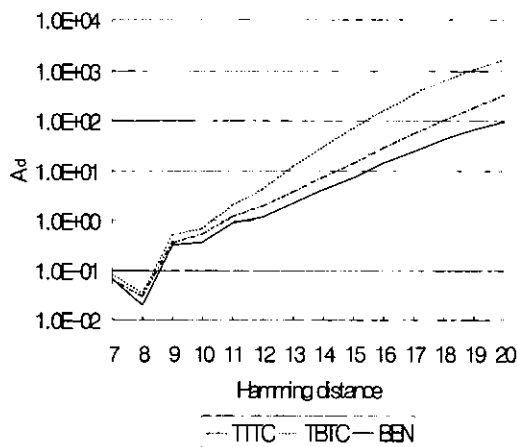


그림 4.  $N=16$ , (7,5) RSC를 갖는 터보부호의  $A_d^{TTC}$ ,  $A_d^{TBTC}$ ,  $A_d^{BEN}$

2. Benedetto가 제시한 [1]에서의 방법을 이용한 경우 BEN이라 표기한다.

부호율은 2/7(=6/21)이다. 해밍 거리가 먼 경우 BER 상한값에 미치는 영향이 작아지므로 그림 4에서는 해밍 거리가 20일 때까지의 계수들만 나타내었다. 그림 4의 값을 살펴보면 세 경우의 최소거리는 동일하지만 해밍 거리가 짧은 경우  $A_d$  값은 TBTC에서 가장 큰 것을 알 수 있다. 그리고 테일 달기 기법을 사용한 경우에는 BEN의 경우 TTTC의 경우보다 더 작다. 이는 테일 비트를 무시한 결과이다.

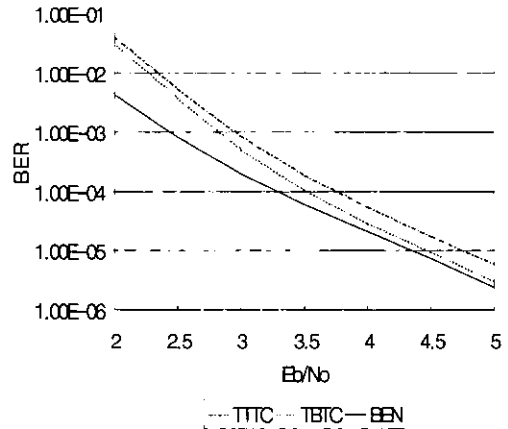


그림 5.  $N=32$ , (7,5) RSC를 갖는 터보부호의 BER 상한값

그림 5는 그림 3의 부호율이 1/2인 (7,5) RSC를 갖는 터보부호기를 이용하여  $N=32$ 인 경우에 대한 BER 상한값을 나타낸 그래프이다. 그림 5를 살펴보면 테일 물기 기법을 사용한 경우가 테일 달기 기법을 사용한 경우보다 더 좋은 성능을 나타냄을 확인할 수 있다. 테일 달기 기법에서는 추가되는 테일 비트로 인해 부호율이 감소하므로 부호율의 감소가 없는 테일 물기 기법을 사용한 경우보다 더 나쁜 성능을 나타낸다. 즉, TBTC의 부호율은 1/3(=13/39)인 반면 TTTC의 부호율은 4/13(=12/39)이다. 그러나 그림 5의 그래프에서 Benedetto의 방법은 테일 달기 기법을 사용하였음에도 불구하고 테일 물기 기법을 사용한 경우보다 더 좋은 성능을 보여주고 있다. 이는 Benedetto가 구한 BER의 상한값은 테일 비트의 영향을 무시하였으므로 나타나는 잘못된 결과이다.

그림 6은 그림 5에서와 동일한 조건에서 테일 달기 기법과 테일 물기 기법을 사용한 경우에 대한 전산 실험 결과를 나타낸 것이다. 그림 6의 전산 실험에서는 임의의 인터리버를 사용하였다. 그림 6에

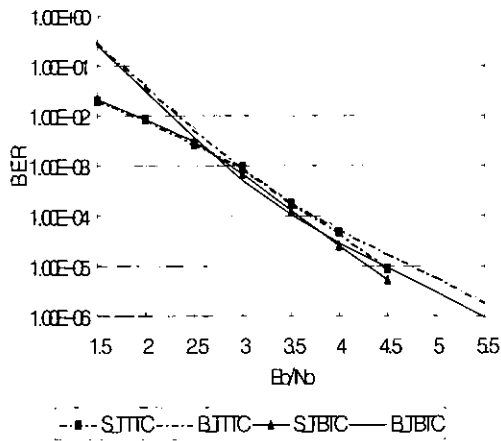


그림 6. N=32, (7,5)RSC인 TTTC와 TBTC의 BER 상한값과 전산 실험 결과

서 BER이  $10^{-4}$ 이하인 경우 각 그래프의 기울기를 비교해 보면 전산 실험 결과가 더 급한 기울기를 나타냄을 볼 수 있다. 이는 평균 상한 값에는 모든 인터리버를 사용하였을 때 얻어지는 해밍거리 중 가장 작은 거리가 포함되어 있기 때문이다.  $E_b/N_0$ 가 클 때 최소 거리(minimum distance)가 짧을수록 BER 그래프의 기울기는 급해진다. BER이  $10^{-4}$ 인 경우 TTTC와 TBTC의 부호이득은 상한값에서는 0.2dB 차이를 보이지만 실험 결과에서는 0.12dB 정도의 차이를 보인다. 이것은 임의의 인터리버를 사용한 TTTC와 TBTC의 차이 때문이기도 하지만 TBTC는 복호 복잡도를 낮추기 위해 준 최적 복호(sub optimal decoding)를 사용한 것도 영향을 준다. 그리고 BER이  $10^{-3} \sim 10^{-4}$  정도에서 실험 결과가 유니폼 인터리버를 사용한 상한값보다 나쁜 성능을 보여준다. 이것은 본 실험에서 사용한 인터리버의 특성이 유니폼 인터리버를 이용해 얻은 평균값보다 나쁘기 때문이 아니라 전산 실험에서는 반복 복호하였지만 상한 값의 계산에서는 최대 우도 복호 한 것에 기인한다.

### V. 결론

길쌈 부호를 블록 부호로 고려할 때 블록 길이가 긴 경우 테일 달기 기법을 이용하더라도 추가되는 테일 비트의 영향은 거의 없다. 그러나 블록 길이가 짧은 경우는 추가되는 테일 비트가 BER 성능에 큰 영향을 미치므로 이를 고려한 성능 평가가 이루어져야 한다. 따라서 본 논문에서는 테일 달기 기법을

이용한 짧은 블록 길이를 갖는 터보부호의 BER 상한값 계산 방법을 제시하였다. 그리고 테일 물기 기법을 이용한 터보부호의 BER 상한값 계산 방법도 제시하였다.

본 논문에서 제시한 방법을 이용하여 터보부호의 성능을 평가하게 되면 기존에 제시된 테일 비트를 무시한 방법보다 정확한 성능 평가가 가능하게 된다.

이러한 방법의 성능 평가는 블록 길이가 짧은 터보부호를 설계할 때 구성 부호의 결정에 유용하게 쓰일 수 있다. 그리고 테일 달기 기법을 사용하는 경우와 테일 물기 기법을 사용하는 경우의 성능 비교가 가능하므로 어느 기법을 적용할 것인지를 결정하는데 이용될 수 있다.

### 참고 문헌

- [1] S. Benedetto, "Unveiling Turbo Codes: Some Results on Parallel Concatenated Coding Schemes," *IEEE Transactions on Information Theory*, vol. 42, no. 2, pp. 409-428, Mar 1996.
- [2] C. Berrou and A. Clavier and P. Thitimajshima, "Near Shannon Limit Error-Correcting Coding and Decoding: Turbo Codes," *Proceeding of the IEEE International Conference on Communications*, vol. 2, pp. 1064-1070, 1993.
- [3] B. Vucetic and J. Yuan, *Turbo Codes: Principles and Applications*, Boston: Kluwer Academic Publishers, 2000.
- [4] ETSI/TC GSM Recommendations. Series 01-12, 1988.
- [5] J. Blanz, A. Klein, M. Nasshan and A. Steil, "Performance of a Cellular Hybrid C/CDMA Mobile Radio System Applying Joint Detection and Coherent Receiver Antenna Diversity," *IEEE Journal on Selected Areas in Communications*, vol. 12, no. 4, pp. 568- 579, May 1994.
- [6] P. Jung and M. Nasshan, "Performance Evaluation of Turbo Codes for Short Frame Transmission Systems," *Electronics Letters*, vol. 30, no. 2, pp. 111-113, Jan 1994.
- [7] S. Benedetto, "Design of Parallel Concatenated Convolutional Codes," *IEEE Trans- action*

