

論文94-31A-9-3

# 가우스성 잡음과 임펄스성 잡음이 혼재하는 통신 채널 상에서 8 PSK, 8 PAM, 16 QAM 에 대한 TCM 복호기 성능에 관한 연구 (A Study on TCM Decoder Performance for 8 PSK, 8 PAM, 16 QAM in the Communication Channel with additive Mixture Gaussian and Impulsive Noise)

鄭智元\*, 金庚信\*, 元東豪\*

(Ji-Won Chung, Kyung-Sin Kim and Dong-Ho Won)

## 要約

이전의 TCM 복호기는 복호 과정이 복잡한 Viterbi 복호 알고리즘을 사용하였으나, 본 논문에서는 복호기의 복잡성을 줄이기 위한 방안으로 convolution 부호기의 복호 알고리즘인 path back 방식을 이용한 복호 알고리즘을 더욱 확장하여 TCM의 병렬 천이 복호기에 적용 가능하게끔 변형된 path back 방식을 이용한 복호 알고리즘을 제안하였다.

가우스성 잡음과 임펄스성 잡음이 혼재하는 채널상에서 Monte Carlo 시뮬레이션 기법을 이용하여 8 PAM, 8 PSK, 16 QAM 등의 각 변조 기법에 대한 성능과 그에 따른 제안한 복호 알고리즘의 효율성을 분석하였다.

## Abstract

Previously, TCM decoder has been using the Viterbi algorithm that was complex. In order to reduce the complexity of decoder, we proposed the "modified decoding algorithm using the path back method" and the modified decoding algorithm which extends the previous decoding algorithm to be able to applied to TCM decoder in this paper. On the gaussian and impulsive noise channel, Monte-Carlo simulation is used for analyzing the TCM performance and proving the efficiency of proposed decoding algorithm.

## 1. 서론

대역폭이 제한된 채널에서 대역폭의 확장없이 코딩 기법을 이용하여 오류률을 개선시킬 수있는 TCM (Trellis Coded Modulation, 격자 부호 변조)이

1982년 Ungerboeck에 의해 제안되었다.<sup>1)</sup> TCM 은 채널 코딩과 변조가 결합된 redundant non-binary modulation 방식으로 대역이 제한된 채널에서 각종 통신 시스템에 널리 적용되고 있다.<sup>2) 3)</sup> 그러나, TCM이 각종 통신 시스템에 널리 적용되고 있지만 이의 복호기는 여전히 연산 수가 많고 복호 속도가 느리며, 복호 회로가 복잡한 Viterbi 복호 알고리즘을 이용하여 복호를 하고 있는 실정이다. 이러한 문제점의 해결을 위하여 Yamashita는 path back

\*正會員, 成均館大學校 情報工學科  
(Dept. of Info. Eng., Sungkyunkwan Univ.)  
接受日字 : 1993年 6月 21日

방식을 이용한 복호 알고리즘을 제안하였으며<sup>[7]</sup>, 이 알고리즘을 convolution 부호의 복호 알고리즘으로 적용시켜 본 결과, 이전의 Viterbi 복호 알고리즘보다 회로가 간단하고, 연산 수가 적고 복호 속도가 더 빠르므로 여러가지 측면에서 효율성이 높음을 증명하였다. 그러나, convolution 부호기의 복호 알고리즘으로 적용되는 path back 방식을 이용한 복호 알고리즘을 TCM에 적용하면 TCM과 convolution 부호와의 송,수신단의 구성 차이 때문에 적용의 어려움이 있다. 그 이유는 path back 방식을 이용한 복호 알고리즘은 격자 상에 존재하는 신호가 binary 이며, 한 상태에서 다음 상태로의 천이가 비 병렬 천이인 데 반해 TCM은 nonbinary이며, 병렬 천이 혹은 비병렬 천이이기 때문이다. 따라서 본 논문에서는 path back 방식을 이용한 복호 알고리즘을 TCM의 복호 알고리즘으로 적용시킬 수 있도록 기존의 알고리즘을 더욱 더 확장시켜 변형된 path back 방식을 이용한 복호 알고리즘을 제안하였다.

본 논문에서의 결과 분석은 첫째로, 가우스성 잡음과 임펄스성 잡음이 혼재하는 채널하에서 각 변조 기법에 대한 오류률을 Monte Carlo 시뮬레이션을 이용하여 분석하였다.<sup>[10]</sup> 오류를 분석을 통하여 여러가지 통신 채널 환경에서 TCM에 적용된 변조 기법의 성능 분석을 하였다. 둘째로, 제안한 복호 알고리즘의 복호 과정을 분석하고 기존의 알고리즘과 차이점을 수식을 통하여 비교하였으며, 복호시 효율성을 컴퓨터 수행 시간, 복호기 메모리의 복잡도 그리고 입력 비트를 복호하기 위한 평균 추적 횟수 관점으로 Monte Carlo 시뮬레이션을 이용하여 분석하였다.

II. 시스템 모델과 신호 매핑

II 장에서는 그림 1.에 나타난 TCM 전체 구성도를 기초로 하여 각 블록에서의 이론적 배경을 논한다.

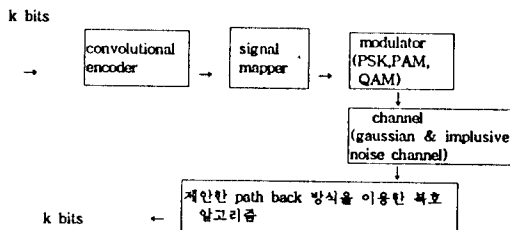


그림 1. TCM의 전체 구성도  
Fig. 1. Blockdiagram of TCM.

TCM의 송신단 모델은 크게 나누어 convolution

부호기 그리고 부호화된 비트를 격자 상으로 재 배열하는 signal mapper와 채널로 신호를 전송하기 위한 변조기로 나뉘어 있으며, 세부적인 변조기의 모델은 PSK, PAM, QAM 변조 기법으로 모델링 하였다. 부호기의 부호화율은 8 PAM, 8 PSK인 경우 2/3 이며, 16 QAM인 경우 3/4 이며 부호화율이 2/3 일 때의 부호기와 격자 상태도는 그림 2와 같다.

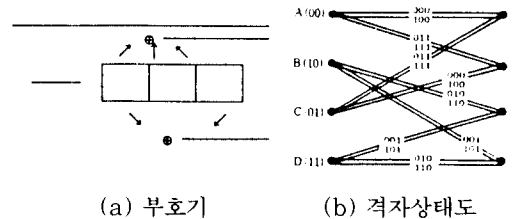


그림 2. 부호화율이 2/3 일 때의 부호기와 격자 상태도

Fig. 2. Encoder and state diagram for 2/3 encoder rate.

부호기의 출력을 채널 신호에 적절히 매핑시킴으로써 주어진 시스템에 대한 euclidean distance를 최대화 할 수 있다. euclidean distance를 최대화 하기 위한 매핑 규칙이 Ungerboeck에 의해 제안된 set partition 규칙이다. Set partition 규칙을 정리 하면 다음과 같다.

- 1) 같은 상태에서 발생하는 천이는 부분 집합 B0, B1의 waveform에서 할당된다.
- 2) 한 상태에서 결합되는 천이는 부분 집합 B0, B1의 waveform에서 할당된다.
- 3) 병렬 천이는 부분 집합 C0, C1, C2, C4로부터 할당된다.

그림 2를 기초로 하여 매핑 규칙을 살펴본다. 동일한 상태 노드에서 입,출력 되는 4개의 branch상에 있는 부호열은 다음의 두 부집합 중 하나에 속하게 된다.

$$B0 = \{ 000, 100, 011, 111 \} \quad B1 = \{ 010, 110, 001, 101 \}$$

병렬 천이 branch상에 있는 부호열은 다음 4개의 부집합 중 하나에 속하게 된다.

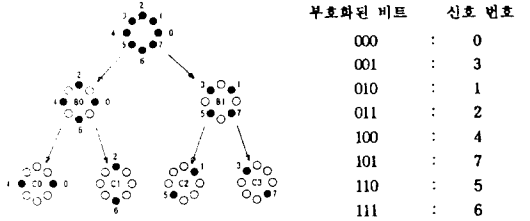
$$C0 = \{ 000, 100 \} \quad C1 = \{ 011, 111 \}$$

$$C2 = \{ 010, 110 \} \quad C3 = \{ 001, 101 \}$$

따라서 euclidean distance를 증가시키는 방향으로 신호 집합을 세분화 하면 각 부호열에 대한 채널 신호가 일대일로 매핑된다.(그림 3 참조)

채널 모델은 가우스성 잡음만 존재하는 채널 및 가

우스성 잡음과 임펄스성 잡음이 혼재하는 채널로서 모델링 하였으며 상세한 내용은 III장에 기술하였다.<sup>[11]</sup>



(a) Set partition (b) 부호열에 대한 신호 할당

그림 3. 8 PSK 신호에 대한 채널 신호 매핑  
Fig. 3. Channel signal mapping for 8 PSK.

수신단에서는 본 논문에서 제안한 변형된 path back 방식을 이용한 복호 알고리즘을 적용하였으며 전체적인 성능 평가 및 복호 알고리즘의 효율성 분석은 입력되는 K 비트와 출력되는 K' 비트를 비교하여 에러의 갯수를 counting 함으로써 분석하였다.

III. 채널

Monte Carlo 시뮬레이션 기법에 있어서 잡음의 처리는 시스템의 전체 성능 평가에 대단히 큰 영향을 미치기 때문에 신중한 잡음 처리가 요구된다.

I, Q 채널에 대한 각각의 잡음 전력은 신호 전력을 1로 정규화 했을 때 입력된 SNR [dB]로부터 다음과 같이 계산된다.

$$PN = 0.5 * 10^{(-SNR/10)} \tag{1}$$

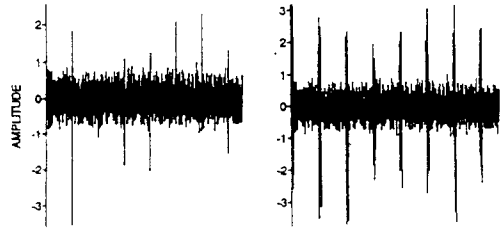
따라서 표준 편차를  $\sqrt{PN}$ 으로 하여 I, Q채널에 대하여 각각 독립적인 가우스 잡음 생성 루틴을 이용하여 두 잡음 성분을 얻어낼 수 있다.

한편 임펄스 잡음은 참고문헌 [11]에 주어진 확률 분포를 이용하여 모델링할 수 있으며 임펄스 잡음을 I 채널과 Q 채널에 대한 joint probability density로 나타내면 식 (2)와 같다.

$$PI(N) = \frac{\text{Exp}(-A)}{W} \sum_{j=0}^{\infty} [A_j * N] / [j! * \sigma_j] \text{Exp}[-N^2 / (2W\sigma_j)] \tag{2}$$

- W : 가우스성 잡음 전력 + 임펄스성 잡음 전력
- A : 단위 시간당 존재하는 임펄스성 잡음의 시간 비
- $\sigma_j$  :  $(j/A + \Gamma) / (1 + \Gamma)$
- $\Gamma$  : 임펄스성 잡음에 대한 가우스성 잡음의 전력비

그림 4는 SNR이 10 dB일때 A= 0.01,  $\Gamma$ 는 0.1인 경우와 A=0.1,  $\Gamma=0.1$ 인 임펄스성 잡음을 시간 영역에서 도시한 것이다. 한편 가우스성 잡음과 임펄스성 잡음이 혼재하는 경우에 대한 확률 분포는 그림 5와 같다. V장에서는 임펄스성 잡음(A= 0.01,  $\Gamma=0.0.1$  & A= 0.1,  $\Gamma= 0.1$ )과 가우스성 잡음이 혼재하는 채널에서 시뮬레이션 결과를 제시하였다.



(a) (b)

그림 4. 가우스 및 임펄스 잡음에 대한 파형  
(a) A = 0.01,  $\Gamma$ 는 0.1인 임펄스성 잡음  
(b) A = 0.1,  $\Gamma$ 는 0.1인 임펄스성 잡음  
Fig. 4. Waveform of gaussian and impulsive noise.

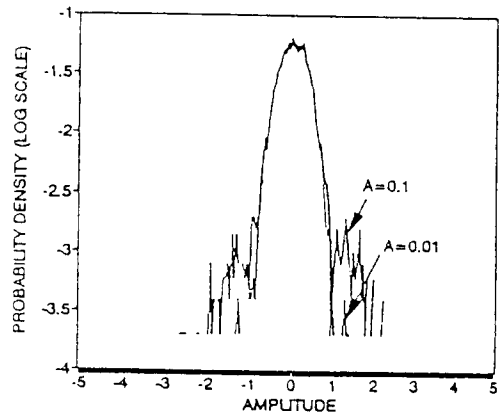


그림 5. Log scale 로 나타낸 잡음의 확률 분포  
Fig. 5. Log scale probability density of noise.

IV. 제안한 TCM에서의 path back method 복호 알고리즘

참고문헌 [7]에서 제시한 convolution 부호에서 path back 방식을 이용한 복호 알고리즘의 전체 구

성도는 그림 6과 같으며 복호 과정은 단계 1~ 단계 5와 같다.

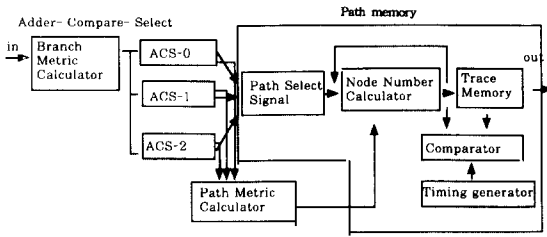


그림 6. path back 방식에 의한 복호기 구성도  
Fig. 6. Blockdiagram of decoder using the path back method.

단계 1. Branch matrices를 구한다. (branch signal과 received signal 과의 거리 (Hamming distance)를 구한다)

단계 2. ACS(Adder-Compare-Selector) : 한 node로 들어오는 것 중에서 branch matrices가 가장 작은 path를 선택함으로써 path select signal을 결정한다. 한 node로 들어오는 것 중 위로부터 들어오는 path를 "0"으로 보고 아래서 들어오는 path를 "1"로 본다.

단계 3. 행렬  $2^{k^1} \times T$ 의 원소인 path select signal을 path memory에 저장한다. (T는 구속장 수의 4배 혹은 5배)

단계 4. Node number calculator에 의해 T만큼 trace back한 뒤에 한 bit를 복호 시킨 다음 shift 시킨다.

Next node numner calaulator 공식은 식 (3)과 같다.

$$N_i = 2^{k-2} x PS_i + \lfloor N_{i-1} / 2 \rfloor$$

$$N_i = \text{Node number at } i \text{ step}$$

$$PS_i = \text{Path select signal at } i \text{ step} \quad (3)$$

$$k = \text{cos train length}$$

$$\lfloor x \rfloor = \text{maximum integer less than } x$$

단계 5. Trace memory에 저장되어 있던 node number와 node number calculator에서 출력되는 현재의 most likely node number를 comparator에서 비교하여 일치하면 더 이상 trace back 하지않고 바로 복호한다.

본 논문에서 제안한 TCM에서의 path back 방식을 이용한 복호 알고리즘의 ACS, comparator 작용은 기존의 참고문헌 [7]에서 기술한 path back method 복호 알고리즘과 동일하나 path select

signal 과 node number calculator의 구성은 다르다. state 수의 증가에 따른 path select signal과 node number calculator의 변화를 살펴보고 8 state에서 복호하는 과정을 살펴보자.

node number는 기존의 convolution coding과 동일하나 node로 들어 오는 path select signal을 본 논문에서는  $0, 1, \dots, 2^k$ 을 할당하여 그에 따른 node number calculator의 공식을 제안한다.

TCM에서 node number calculator는 다음과 같은 공식에 의하여 다음 node를 결정한다.

$$N_i = 2x \lfloor PS_i / 2 \rfloor + \lfloor N_{i-1} / 2 \rfloor$$

$$N_i = \text{Node number at } i \text{ step}$$

$$PS_i = \text{Path select signal at } i \text{ step} \quad (4)$$

$$k = \text{cos train length}$$

$$\lfloor x \rfloor = \text{maximum integer less than } x$$

위에서 제안한 공식을 적용하면 TCM에서 구속장 수가 증가하여도 path select signal과 이전의 node number를 이용하여 다음 node number를 추적할 수 있다. Path memory operation을 통하여 8 state에서 복호하는 과정은 그림 7과 같다.

Node number	Input path select signal	path select signal
0	0	0 2 7 3 5 0 0
1	2	3 4 6/3 2 1 5 3
2	4	5 3 2/7 5 2 4 1
3	5	4 2 3 6 3 3 3
4	7	7 2 1 5 2 1 1
5	1	5 2 1 3 7 2 5 3
6	3	3 2 1 4 5 7 6
7	6	1 4 5 5 3 7

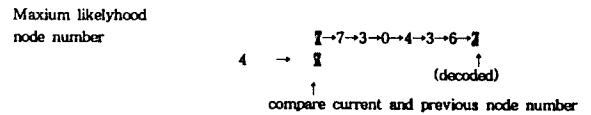


그림 7. 8 state에서 path memory operation ( trace number = 1 )

Fig. 7. Path memory operation of TCM for 8 state( trace number = 1 ).

그림 7의 복호 과정의 순서는 단계 1 ~ 단계 7과 같다.

단계. 1. 이전의 path select signal과 node number를 path memory ( 행렬  $2^{k^1} \times T$  )에 기억시킨다.

단계. 2. 현재 입력되는 node number와 path select signal을 node number calculator에 입력 시킨다. ( 그림 6 참조 )

단계 3. Node number calculator의 출력 node number와 path memory에 저장된 node number를 comparator에서 비교한다.

단계 4. 비교한 결과 먼저 저장되어 있던 node number와 일치하는 경우 더 이상 trace back 하지 않고 입력 비트를 복호한다.

단계 5. 다음 path memory에 있는 node number와 path select signal을 한 bit shift 시킨다.

단계 6. Next node number와 path select signal을 path memory에 입력한다.

단계 7. 단계 1 ~ 단계 6의 복호 과정을 전송된 비트수 만큼 반복하여 수신 신호와 송신 신호를 비교한 후 오류 비트수를 측정하여 변조 기법에 대한 성능을 분석한다.

위의 step을 통하여 T 만큼 trace back하지 않고도 2 bit, 3 bit를 복호할 수 있기 때문에 T - trace number 만큼의 복호 효율성이 향상됨을 알 수 있다.

Convolution coding과 제안한 TCM의 path back method 복호 알고리즘의 차이점은 표 1과 같다.

표 1. Path back method를 적용할 경우 TCM과 convolution coding의 복호 과정의 차이점

Table 1. Comparison of TCM and convolution coding for path back method de coding algorithm.

	Convolution coding	TCM(Proposed algorithm)
Node number calculator	$N_i = 2^{i-2} \times PS_i + \lfloor N_{i-1} / 2 \rfloor$	$N_i = 2 \times \lfloor PS_i / 2 \rfloor + \lfloor N_{i-1} / 2 \rfloor$
Node number	$N_i = 0, 1, 2, \dots, 2^{i-1} - 1$	$N_i = 0, 1, 2, \dots, 2^{i-1} - 1$
Path select signal	$PS_i = 0, 1$	$PS_i = 0, 1, 2, \dots, 2^{i-1} - 1$

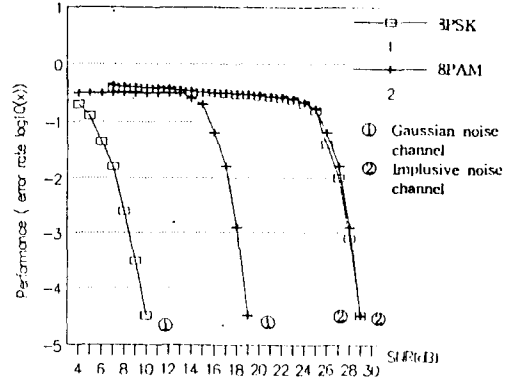
V. 결과 및 검토

가우스성 잡음 및 임펄스성 잡음 채널 환경에서 TCM에 적용된 각 변조 기법에 대한 BER 곡선을 그림 8에 나타내었다. BER = 10<sup>-4</sup>을 기준으로 rate 2/3일 때 가우스성 잡음 채널에서는 8 PSK가 8 PAM 보다 10 dB의 코딩 이득을 얻을 수 있음을 알 수 있으며 임펄스성 잡음 채널하에서는 비슷한 성능을 보였다.

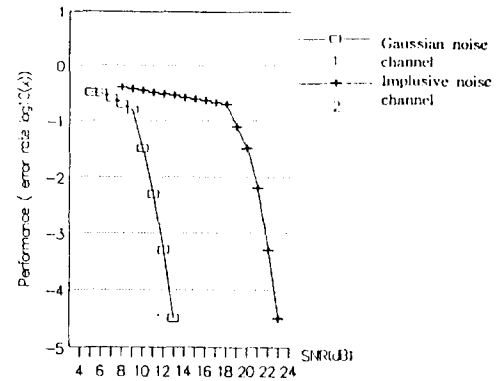
Rate 3/4 인 16 QAM인 경우 가우시안 잡음 채널과 임펄스성 잡음 채널에서의 코딩 이득의 차이는 10 dB의 차이를 나타냄을 보인다. 8 PSK 신호의 경우 임펄스성 잡음 채널에서 성능이 아주 저하됨을

알 수 있다.

Monte-Carlo 시뮬레이션 기법에 있어서 10<sup>-4</sup>의 BER을 산출하기 위해 10<sup>6</sup>이상의 데이터를 처리해야만 신뢰할 수 있는 BER을 얻을 수 있다.



(a)



(b)

그림 8. 각 변조 기법에 대한 성능

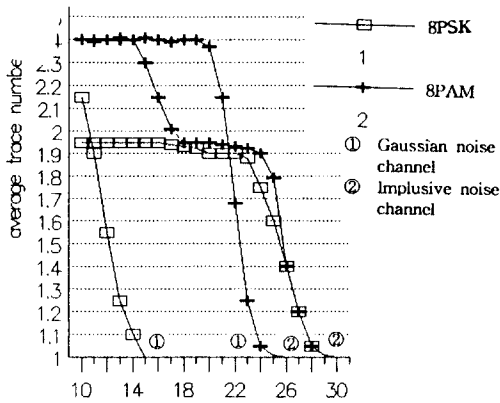
(a) 가우시안 잡음과 임펄스성 잡음 채널에서 PSK, PAM 성능 분석(rate = 2/3)

(b) 가우시안 잡음과 임펄스성 잡음 채널에서 QAM 성능 분석(rate = 3/4)

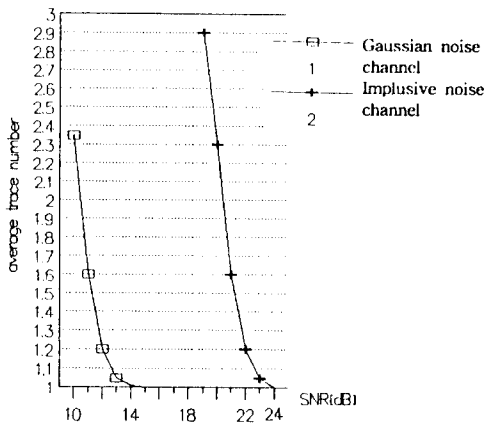
Fig. 8. Performance of each modulation.

그림 9에서는 각 변조 기법에 대해 제안한 복호기의 평균 추적 횟수를 나타내었다. 그림 9의 세로 축은 그림 7의 path memory에서 trace back하는 평균 횟수인 평균 추적 횟수를 나타낸다. 그림 9에서 가우스성 잡음 채널에서 평균 추적 횟수가 1에 근접하기 위해서는 8 PSK가 8 dB ~ 9 dB, 16 QAM

이 13 ~ 14 dB, 8 PAM이 19 ~ 20dB 정도의 SNR이 필요함을 알 수 있으며, 임펄스성 잡음 채널에서 평균 추적 횟수가 1 에 근접하기 위해서는 8 PSK, 8 PAM이 29 ~ 30 dB, 16 QAM이 23 ~ 24 dB 정도의 SNR이 필요함을 알 수 있다.



(a)



(b)

그림 9. 각 변조 기법에 대한 평균 추적 횟수 (a) 가우시안 잡음과 임펄스성 잡음 채널에서 PSK, PAM 신호에 대한 복호기 추적횟수(rate = 2/3)(b) 가우시안 잡음과 임펄스성 잡음 채널에서 QAM 신호에 대한 복호기 추적횟수(rate = 3/4)

Fig. 9. Average trace number for each modulation.

기존의 Viterbi 복호 알고리즘은 순차적으로 입력된 비트를 순차적으로 복호하기 위해 항상 T(구속장 수의

4 혹은 5배) 만큼 모든 가능한 경로를 추적해야 하나, 본 논문에서 제안한 복호 알고리즘을 이용하여 복호하면, high SNR일 때 주어진 node number calculator 공식 (4)에 의해 평균 1 회만 추적하면 입력된 비트를 복호할 수 있으므로 이전의 알고리즘에 비해 복호 효율성을 알 수 있다.

표 2. 구속장 수에 따른 컴퓨터 시뮬레이션 시간  
Table 2. Computer simulation time according to constrain length.

입력비트수 (bits)	구속장 수	500	1K	10K	50K
기본 알고리즘	3	0.109 soc	0.274 soc	2.802 soc	15.38 soc
	5	0.989 soc	1.97 soc	19.6 soc	97.80 soc
	7	5.6 soc	11.48 soc	114 soc	572.36 soc
제안 알고리즘	3	almost 0 soc	0.109 soc	1.59 soc	8.35 soc
	5	0.33 soc	0.65 soc	7.47 soc	37.47 soc
	7	1.87 soc	3.79 soc	38 soc	193 soc

표 3. 메모리 복잡도 및 기타 관점에서의 비교  
Table 3. Comparison of memory complexity and other aspects.

	기본 알고리즘	제안 알고리즘
error correction 방식	forward	back
구속장 수에 따른 성능	향상	향상
구속장 수에 따른 메모리 복잡도	연산작용이 많아져 기하급수적으로 증가(수많은 D I/과 select)	path memory에서 ACS작용으로 인해 LSI RAM안 필요
복호시 저장해야 할 내용	각 node에서의 euclidean distance 및 survivor path( 구속장 수가 증가하면 저장해야 할 survivor path가 기하급수적으로 증가하여 복호시 비교 판단이 많아짐)	Node number register 및 path select signal memory( 구속장 수가 증가하더라도 node number calculator 증가하더라도 node number calculator 공식에 의해 자동적으로 복호 수행)

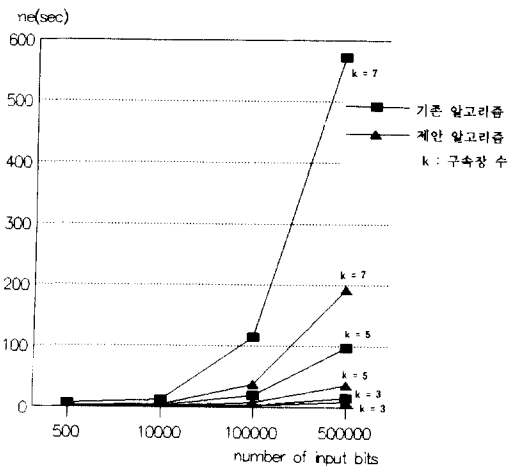


그림 10. 제안 알고리즘의 시뮬레이션 시간  
Fig. 10. Simulation time of proposed algorithm.

복호 효율성을 증명하기 위해 표2와 표3에서는 컴퓨터 시뮬레이션 시간 및 메모리의 복잡도면에서 기존의 방식과 비교하였다.

표 2에서 가로축을 입력 비트 수, 세로축을 시간으로 놓고 그래프를 그리면 그림 10과 같다.

위의 표 2, 3 그리고 그림 10에서 알 수 있듯이 제안한 알고리즘이 기존의 알고리즘보다 2배 이상의 속도 개선을 하였으며 메모리면에서도 효율성이 좋음을 알 수 있다. 시뮬레이션 환경은 IBM PC 486(50 MHz)에서 C 언어로 구현하였다.

## VI. 결론

TCM은 코딩과 변조가 결합되어 대역폭의 증가없이 코딩 이득을 향상시킬 수 있는 통신 방식으로 위성통신이나 음성통신과 같이 대역폭이 제한된 채널에서 널리 이용되고 있다. 본 논문에서는 첫째로 Monte Carlo 시뮬레이션 기법을 이용하여 TCM에 적용된 변조 기법인 PSK, PAM, QAM을 가우스성 잡음 및 임펄스성 잡음 채널 환경에서 성능 분석을 하였다. 가우스성 잡음 채널에서는 8 PSK가 10 dB 정도 8PAM에 비해 성능이 좋으며 임펄스성 잡음 채널에서는 비슷한 성능을 보였다. 16 QAM인 경우, 가우스성 잡음 채널에서 SNR값은 13 dB, 임펄스성 잡음 채널에서 22 dB임을 알 수 있다. PSK 신호의 경우 임펄스성 잡음 채널에서 성능이 저하됨을 알 수 있다.

둘째로 TCM 복호 알고리즘을 제안하였다. 기존의 TCM 복호 방식으로는 Viterbi 복호 알고리즘을 사용하고 있으나 본 논문에서는 path back method 복호 알고리즘을 확장시켜 변형된 path back 방식을 이용한 복호 알고리즘을 제안하였으며 이를 병렬 천이 TCM 복호기에 적용시킴으로써 Viterbi 복호 알고리즘의 단점인 복호 회로의 복잡성과 복호 속도가 개선됨을 컴퓨터 시뮬레이션 수행 시간 및 여러 관점에서 비교하여 보였다.

Path back method 복호 알고리즘을 TCM에 적용시키기 위해서 path select signal과 다음 node number를 결정하는 node number calculator의 공식을 제안하였으며, 이를 TCM에 실제로 적용시켜 본 결과 복호 cycle에서 평균 추적 횟수가 1에 근접함을 시뮬레이션을 통하여 확인하였다. 이는 종전의 Viterbi 복호 알고리즘이 입력된 비트를 복호하기 위해 항상 T(구속장 수의 4 혹은 5 배) 만큼 모든 가능한 경로를

추적해야 하나, 본 논문에서 제안한 복호 알고리즘을 이용하여 복호하면, path select signal과 현재의 node number만 알면 node number calculator 공식(4)에 의해 입력된 비트를 쉽게 복호할 수 있다.

본 논문에서 path back method 복호 알고리즘을 적용시켜 구성한 TCM 복호 방식은 대역폭이 제한된 통신 방식에 유용하리라 사료된다.

## 參考文獻

- [1] Sklar, B. "Digital Communications : Fundamentals and Applications", Prentice Hall, pp.333-428, 1988.
- [2] Scalar, "Trellis Coded Modulation", pp. 35.1.1-35.1.9, 1988.
- [3] Ungerboeck, G. "Channel Coding with Multilevel/Phase Signals", IEEE Trans. Inform. Theory, Vol. IT-28, pp.55-67, January 1982.
- [4] Shu Lin, "Error Control Coding", Prentice Hall, pp.287-349, 1983.
- [5] 이 만영, "符號理論", 회중당, pp.277-353, 1884
- [6] Hemant K, "Real Time Application of Trellis Coding to High Speed Voice Band Data Transmission", *IEEE Journal Select Area in Commun.* Vol. 2, pp. 648-658, Sept 1984.
- [7] Atsushi Yamashita, "A New Path Memory for Viterbi decoders", *FUJITSU Sci. Tech. J.*, pp.37-42, Mrch 1989.
- [8] G. D. Forney, Jr., 'The Viterbi Algorithm', *Proc. IEEE*, Vol 61, pp.268-278 Mar, 1973.
- [9] Rader, C. M "Memory Management in a Viterbi Decoder" *IEEE Trans. Commun* Vol 9, pp.1399-1401, 1981.
- [10] S. Joseph Campanella, Sayegh, "Onboard multichannel demultiplexer/ demodulator study," *Comsat Laboratories*, Clarksburg, Maryland, Dec. 1987.
- [11] Coates, 'Monte Carlo Simulation' *IEEE Journal on Select* Vol 6, pp.58-66, 1988.

著 者 紹 介



鄭智元(正會員)

1989年 성균관대학교 전자공학 졸업(공학사). 1991年 성균관대 대학원 전자공학과 졸업(공학 석사). 1992年 ~ 현재 성균관대학교 정보공학과 박사과정 재학중. 1990年 ~ 1992年 금성정보통신(주)

연구소 연구원



金庚信(正會員)

1983年 성균관대학교 전자공학 졸업(공학사). 1985年 성균관대 대학원 전자공학과 졸업(공학 석사). 1993年 ~ 현재 성균관대학교 정보공학과 박사과정 재학중. 1984年 ~ 1991年 삼성 전자(주)

컴퓨터 부문 선임 연구원. 1991年 ~ 현재 연암공업 전문대 전산과 조교수



元東豪(正會員)

1976年 성균관대학교 전자공학 졸업(공학사). 1978年 성균관대 대학원 전자공학과 졸업(공학 석사). 1988年 성균관대학교 대학원 전자공학과 졸업(공학 박사). 1978年 ~ 1980年 한국전자통신 연구소

연구원. 1985年 ~ 1986年 일본 동경 공대 객원 연구원. 1982年 ~ 현재 성균관대학교 정보공학과 조교수, 부교수, 교수