

# 비터비 복호기의 최적 메모리 제어

조영규, 정차근

호서대학교 전기정보통신공학부 정보제어전공

## Optimal Memory Management of Viterbi Decoder

Young-Kyu Cho, Cha-Keon Cheong

The School of Electrical Engineering, Hoseo University

### 요약

본 논문은 이동 통신 및 IEEE 802.11a WLAN에서 사용하고 있는 컨벌루션 부호의 복호기인 비터비 복호기의 SMU(Survivor Metric Unit)의 최적 메모리 제어에 관한 연구이다. 비터비 복호기 구조는 크게 BMU, ACSU, SMU부로 구성된다. 이때 SMU부는 최적의 경로를 역추적 하여 최종 복호 데이터를 출력해 주는 블록으로, 역추적 길이에 따라 메모리 사용 양과 복호 성능이 좌우 된다. 따라서 본 논문에서는 최적 메모리 제어 알고리즘을 제안함으로써 복호 속도의 향상과 메모리 사용 양을 줄이는 방법을 제안한다. 제안 알고리즘의 성능을 검증하기 위해 기존의 비터비 복호기와 역추적 길이에 따른 비터비 복호기의 성능을 실험을 통해 분석함으로써 제안 방법의 객관적인 성능을 분석한다.

### I. 서론

정보통신기기의 이동성 및 개인화가 빠르게 진전되고, 휴대용 컴퓨터 및 PDA의 보급이 확산됨에 따라 장소에 상관없이 네트워크 망에 접속하는 수단으로 무선 LAN의 필요성이 증대되고, 그 보급이 빠르게 진전되고 있다. 또한 최근 OFDM 기술의 적용으로 6~54Mbps까지의 다양한 데이터 전송 속도를 갖는 IEEE 802.11a WLAN 표준안이 확정되어[1] 현재 선진 각국의 주요 칩 메이커를 중심으로 IEEE 802.11a WLAN 칩 개발이 진행되고 있다. 이와 같이 점차 고속 무선 통신 시스템으로 변해감에 따라 무선 채널 상에서의 채널 에러에 대한 정보 보호를 위한 채널 부호화의 연구가 필요하다.

컨벌루션 부호 방식은 이전의 데이터들이 일정한 규칙성을 가지고 현재의 정보에 영향을 미치는 구조로 랜덤 오류에 강한 특성을 보이며 연판정 복호를 할 경우 경관 정보보다 약 2dB의 부호 이득을 추가로 얻을 수 있는 장점을 가지고 있다.[2] 비터비 알고리즘은 Maximum Likelihood 방식을 이용하여 수신된 심볼을 트렐리스도 상에서 가능한 여러 경로에 대해 확률을 계산하여 가장 큰 확률을 갖는 경로를 선택하는 방법이다.[3][4]

비터비 복호기의 구조는 BMU, ACSU, SMU부로 구성된다. BMU부는 상태천이도에 의해 생성된 가지 부호어와 송신된 부호어와의 차를 구하고, ACSU부는 이전 상태까지 누적된 경로 메트릭과 수신 부호의 가지 메트릭을 이

용해 각 상태로 천이되는 경로 중 경로 메트릭이 적은 경로 즉 오류가 가장 적은 생존 메트릭을 결정한다. SMU부에서는 ACSU부에서 결정된 생존 메트릭에 대한 정보를 저장하며 동시에 이전 SMU부에 저장된 각 상태의 정보를 갱신해 준다. 이와 같은 과정이 역추적 깊이만큼 수행된 후 역추적 과정을 통하여 복호 데이터가 출력된다. 또한 비터비 복호기는 역추적 길이에 따라 메모리 사용 양과 복호 성능이 좌우되며 역추적 길이가 커질수록 많은 지연시간을 필요로 하게된다.[5][6] 이와같은 문제를 해결하고 높은 전송 속도를 요구하는 고속 무선 통신 시스템에 대응하기 위한 비터비 복호기 SMU부의 효율적인 메모리 제어 방법이 요구된다.

따라서 본 논문에서는 이를 위해 에러 발생 시점을 선택적으로 역추적 과정을 통해 상태 정보를 갱신함으로써 복호 데이터를 출력하는 효율적인 최적 메모리 제어 알고리즘을 제안하고, 이를 역추적 길이에 따른 성능을 분석하고 실험을 통하여 확인한다.

### II. 최적 메모리 제어에 의한 SMU 구조

그림 1은 최적 메모리 제어에 의한 비터비 복호기의 구조를 나타내고 있다. 먼저 채널을 통해 수신된 심볼은 BMU에 의해 가지 메트릭이 계산되고 ACSU로 입력된다. ACSU는 현재 시간에 계산된 가지 메트릭과 이전 생존 메트릭을 이용하여 새로운 생존 메트릭  $SM_t$ (Survivor

metric)와 생존 매트릭 정보  $SMI_t$ (Survivor metric information)를 출력한다. 모든 상태에 대한 생존 매트릭은 최소 경로 검파기에 의해 현재 시간에서의 모든 상태의 생존 매트릭중 최소 생존 경로  $MSM_t$ (Minimum Survivor Metric)와 최소 생존 경로 정보  $MSMI_t$ (Minimum Survivor Metric Information)가 출력된다. 최소 생존 경로 매트릭  $MSM_t$ 는 ACSU에서 누적 경로 값을 더하는 과정에서 오버플로우로 인한 연산 에러 발생을 방지하기 위해 정규화 블록으로 입력되고, 최소 생존 경로 정보  $MSMI_t$ 는 상태 에러 검출기에 입력되어 이전 시간에 저장된 생존 경로 정보  $SMI_{t-1}$ 를 이용하여 에러가 발생하지 않을 경우는 이전의 경로와  $MSMI_t$ 에 의해 최적 경로를 형성하게 되고, 만약 에러가 발생하였을 경우는 현재 시간에 발생된  $MSMI_t$ 에 의해 새로운 최적 경로를 형성하게 된다.

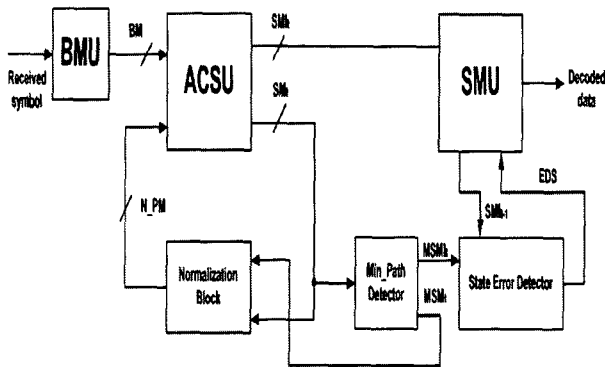


그림 1. 최적 메모리 제어에 의한 비터비 복호기 구조

### III. 최적 메모리 제어에 의한 역추적 과정

기존의 역추적에 의한 복호 방법은 역추적 깊이 만큼 상태 정보를 저장하고, 데이터를 복호한다. 그러나 역추적 깊이 이후 연속적인 역추적 과정에서 에러가 발생하지 않은 스테이지에 대해서도 반복적으로 역추적 과정을 수행하게 된다. 따라서 역추적 과정을 통하여 복호 데이터가 결정되기 전까지 다음 복호 과정을 실행할 수 없게 되고, 생존 경로 정보를 저장하기 위해서는  $traceback\ depth \times total\ state$ 만큼의 저장 공간이 필요 하게 된다.[7]

본 논문에서 제안한 SMU의 최적 메모리 제어에 의한 데이터 복호 과정은 다음과 같이 두 단계로 나누어 수행 된다.

첫째, 각각의 스테이지의 모든 상태 값에 대한 생존 경로 값은 상태천이 과정에서 컨벌루션 부호기 상태의 유사성의 가중치를 나타낸다. 따라서 임의의 시간  $t$  스테이지에 대해 고려해 보면, 모든 상태에서의 생존 경로 값 중 최소 생존 경로 값이 시간  $t$ 에서 입력 값에 대한 컨벌루

션 부호기의 상태와 가장 유사하다 볼 수 있다. 따라서 시간  $t$ 에서의 복호 데이터는 최소 생존 경로 값을 갖는 상태의 입력 데이터가 된다.

둘째, 역추적에 의한 상태 정보의 갱신 위치는 에러 발생 시점으로 한다. 즉 역추적 깊이 이후 연속적인 역추적 과정을 통해 상태 값을 갱신하지 않고, 각 스테이지에서의 에러 유무를 판단하여 상태 값을 갱신해 주는 것이다. 구속장  $K=3$ 이고 2-point 알고리즘을 적용할 경우 발생할 수 있는 상태 천이 및 상태 천이 정보는 그림 2와 같다. 실선은 입력으로 1이 들어왔을 경우이며 점선은 입력으로 0이 들어왔을 경우를 의미한다. 임의의 시간  $t$ 의 상태에서 입력으로 0이 들어왔을 경우 시간  $t+1$ 에서의 천이 될 수 있는 상태는 00, 01이고,  $t$ 의 상태에서 입력으로 1이 들어왔을 경우 시간  $t+1$ 에서의 천이 될 수 있는 상태는 10, 11이다. 따라서 이와 같은 상태 천이 및 상태 천이 정보를 이용하면 에러 발생 유무를 판별할 수 있다.

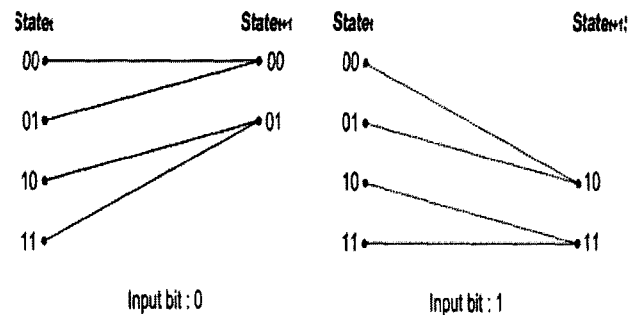


그림 2. 상태 천이 및 상태 천이 정보

그림 3은 기존의 역추적 깊이 이후 역추적 과정을 통해 상태 값을 갱신하는 과정과, 본 논문에서 제안한 에러 발생 시점에서만 역추적 과정을 통해 상태 값을 갱신하는 과정을 비교하여 나타내고 있다.

SMU의 메모리 제어에 의한 역추적 방법은 에러의 유무를 판별해 에러가 발생된 시간에서 역추적 과정을 통해 상태 값을 갱신 시켜주는 것이다. 즉 역추적 깊이 이후 연속적인 역추적 과정을 통하여 데이터를 복호 하는 기존의 방법과는 달리 메모리 제어 방법은 ACSU에서 생존 경로 값이 결정되는 순간 에러 발생 유무에 따라 복호 과정을 수행하므로 복호 속도가 향상 된다. 또한 상태 정보를 갱신 시켜주기 위한 메모리의 크기는 에러가 발생된 시점에서만 역추적 과정을 통해 상태 정보를 갱신 시켜 주면되므로 기존의 역추적 방법에 필요한 메모리의 크기보다 작은 메모리를 필요로 하게 된다. 따라서 에러 발생 시점을 선택적으로 역추적 과정을 통해 상태 정보를 갱신함으로써 기존 방법에 비해 적은 메모리 양으로도 비교적 정확한 추정이 가능하며 복호 속도 또한 향상 시킬 수 있다.

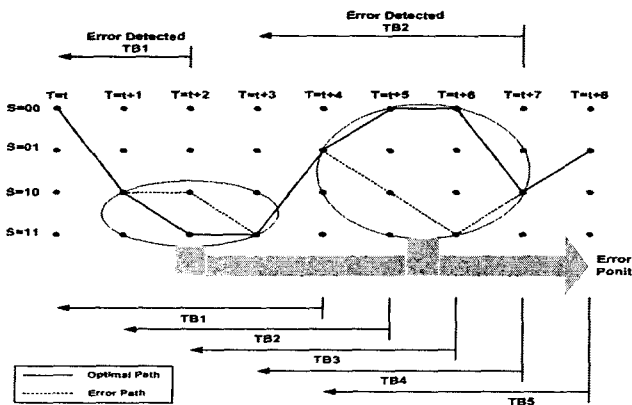


그림 3. 역추적 과정 비교

### IV. 실험 및 결과

본 논문에서는 구속장  $K=3,5,7$ 에 대해 부호율 1/2인 컨벌루션 부호화 과정을 거친 부호어 BPSK 변조한다는 가정하에 부가 백색 가우시안 잡음(AWGN)이 존재하는 채널로 모델링 하였다.

#### 1. 시스템 구성

##### (1) 컨벌루션 부호기

구속장  $K=3,5,7$ 에 대한 부호율 1/2인 컨벌루션 부호기의 생성 다항식은 다음과 같다.

· 구속장  $K=3$

$$\begin{aligned} g_1 &= 111_2 = 1 + x^1 + x^2 \\ g_2 &= 101_2 = 1 + x^2 \end{aligned} \quad (1)$$

· 구속장  $K=5$

$$\begin{aligned} g_1 &= 11101_2 = 1 + x + x^2 + x^4 \\ g_2 &= 10011_2 = 1 + x^3 + x^4 \end{aligned} \quad (2)$$

· 구속장  $K=7$

$$\begin{aligned} g_1 &= 1011011_2 = 1 + x^2 + x^3 + x^5 + x^6 \\ g_2 &= 1111001_2 = 1 + x + x^2 + x^3 + x^6 \end{aligned} \quad (3)$$

##### (2) 비터비 복호기

###### a. BMU

가지 매트릭은 경판정일 경우는 해밍 거리, 연판정일 경우는 유클리디언 거리로 정의된다. 유클리디언 거리는 가지 부호어와 송신된 부호어와의 내적을 구하는 것으로 유클리디언 거리, 자승 유클리디언 거리, 근사화 유클리디언 거리 3가지의 방법으로 계산할 수 있다. 본 논문에서는 근

사화 유클리디언 거리를 이용하여 가지 매트릭을 계산하였고 식 (4)와 같이 정의 된다.

$$AED = |r - v| \quad (4)$$

###### b. ACSU

ACSU부는 구속장  $K=7$ 은 32조,  $K=5$ 는 8조,  $K=3$ 은 2조의 ACSU 연산기를 필요로 한다. 그림 4는 1조의 버터플라이 구조의 ACSU 연산기의 구조를 나타낸다.

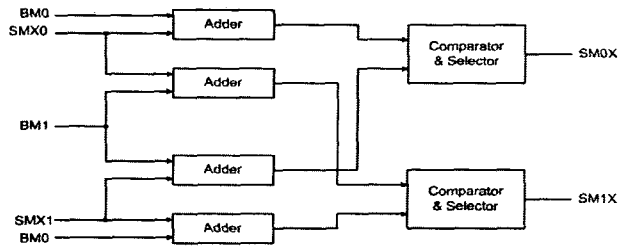


그림 4. 버터플라이 구조의 ACS 연산기

###### c. SMU

SMU부는 그림 1에 보여진 것처럼 ACSU부에서 계산된 생존 매트릭과 생존 매트릭 정보에 의해 에러가 발생한 곳에만 역추적 과정을 통해 경로를 갱신하고 역추적 값이 이후 복호 데이터를 출력 한다.

#### 2. 실험 결과

그림 5, 그림 6, 그림 7은 구속장  $K=3,5,7$ 에서 역추적 길이에 따른 비터비 복호기의 성능을 나타낸 것이다. 이 결과로부터 역추적 길이가 같을 때는 SMU 메모리 제어에 의한 비터비 복호기의 성능이 개선됨을 알 수 있다. 역추적 길이를 작게 하였을 경우 SMU 메모리 제어에 의한 비터비 복호기는 신호 대 잡음비가 커질수록 성능이 좋아짐을 확인 할 수 있으며, 역추적 길이를 구속장에 5배로 한 일반적인 비터비 복호기와 유사한 성능을 보임을 알 수 있다.

### V. 결론

메모리 양을 줄이고 복호 속도를 향상 시키기 위한 SMU 메모리 제어알고리즘을 제안하고 실험으로 성능을 분석하였다.

일반적인 비터비 복호기는 구속장과 부호율이 커질수록 역추적 길이가 커져야 하므로 많은 메모리 사용 양과 복호 시간이 지연된다. 반면에 SMU 메모리 제어 방법을 사용할 경우 복호 속도를 향상 시킬 수 있으며 메모리 사용 양을 줄일 수 있으므로 전력소모를 줄일 수 있다.

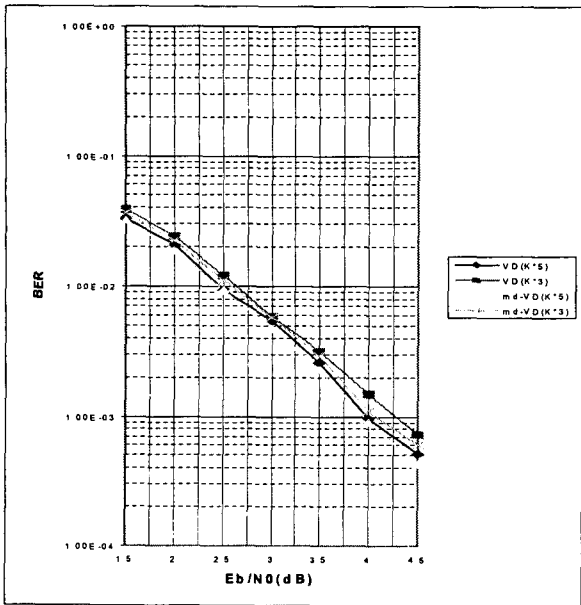


그림 5. 구속장  $K=3$ 일때 역추적 길이에 따른 비터비 복호기능 성능

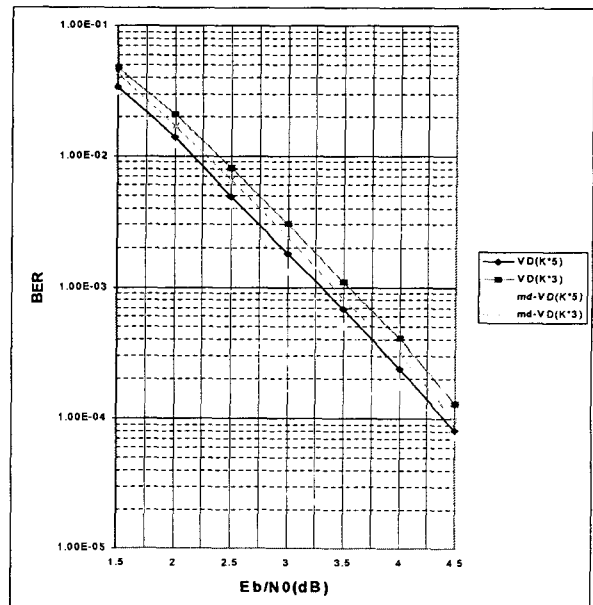


그림 7. 구속장  $K=7$ 일때 역추적 길이에 따른 비터비 복호기능 성능

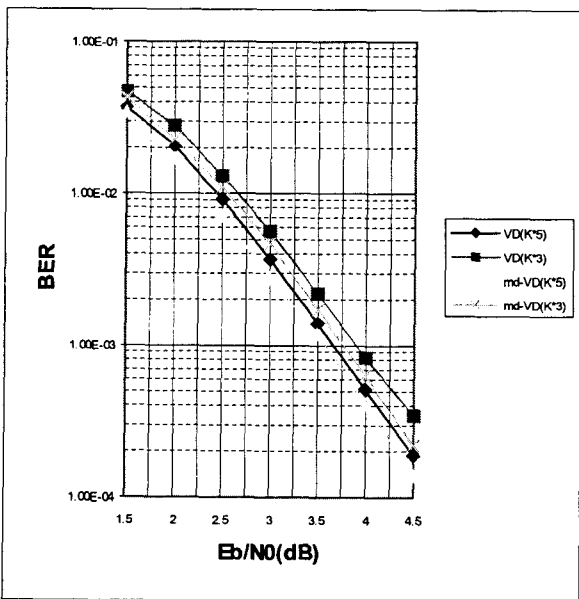


그림 6. 구속장  $K=5$ 일때 역추적 길이에 따른 비터비 복호기능 성능

### 참고문헌

- [1] IEEE std 802.11a Wireless LAN Medium Access control(MAC) and physical Layer(PHY) Specifications : Higher-Speed Physical Layer Extension in the 5 GHz Band, 1999
- [2] Sho Lin, Daniel J. Costello, Jr., Error control coding : Fundamentals and Application, Engelwood Cliffs : Prentice Hall, Inc, 1998
- [3] Heller, J. and Jacobs, I.W, "Viterbi Decoding for Satellite and Space Communication" IEEE trans. Commun. Technol. COM19, no5, October 1971
- [4] Forney, G . D. Jr, "Convolutional Codes : I. Algebraic Structure," IEEE Trans. Infor. Theory, vol IT 16, NO. 3, Nov. 1970, pp. 702~738
- [5] G. Fettweis and H. Meyr. "High-Rate Viterbi Processor : A Systolic Array Solution." IEEE Journal on Selected Areas in Commun : 1520~1534 , October 1990.
- [6] C. B. Shung, H. D. Lin, R. Cypher, P. H. Siegel, and H. K. Thapar. "Area-efficient architecture for the Viterbi algorithm I. Theory." IEEE Trans on Commun, 41(4) : 636-44, April 1993
- [7] Herbert Dawind, "Viterbi Decoder : High Performance Algorithm and Architectures", DSP Solutionas Group, Synopsys, Inc