

## 터보 부호를 위한 MAP 복호기의 구현

이정원\*, 김종태  
성균관대학교 전기.전자.컴퓨터공학부

### The Implementation of MAP decoder for Turbo codes

Jung won Lee\*, Jong Tae Kim  
School of Electrical&Computer Engineering Sung Kyun Kwan University

**Abstract** - Turbo codes that have attracted a great attention in recent years are applied to wireless communication networks that require variable quality of service and transmit over unknown fading channel. A MAP decoder is the constituent of turbo decoder. In this paper, we propose a high speed architecture of MAP decoder and a new normalization technique. In conclusion, this paper presents the efficient implementation of serial block MAP decoder for turbo codes.

## 1. 서 론

차세대 이동통신시스템에서는 음성, 고속 데이터 및 동영상 등의 서비스를 제공하여야 하기 때문에 신뢰도가 높은 데이터 전송이 필수적이다. 따라서 그에 따른 우수한 오류 정정 부호가 필요하다. 이를 위해 현재 가장 각광 받고 있는 오류정정 부호가 터보 부호이다.

터보 부호는 특성상 반복 부호가 가능해야 하기 때문에 내부 복호기는 연판정 복호를 수행하여야 한다. 이러한 연판정 복호기는 복호화 기법에 따라 크게 두 가지로 나뉘는데, MAP(Maximum A Posteriori) 복호기와 SOVA (Soft-Output Viterbi Algorithm)복호기로 구분 할 수 있다.

연판정 복호기인 SOVA 복호기는 Viterbi 복호기를 soft-out.으로 확장한 것으로 구현은 간단하나 MAP 복호기에 비해 BER 성능이 떨어진다[1]. 이러한 이유로 하드웨어 구현은 복잡하지만 BER 성능이 좋은 MAP 알고리즘이 사용되고 있는 추세이다[2][5]. 그러나 MAP 알고리즘을 적용하게 되면 하드웨어의 구조가 복잡해지고 특히 그 크기가 SOVA에 비해 병렬형으로 구현하였을 때 약 3배, 직렬형으로 구현하였을 때 약 1.3배 가량 늘어나는 경향을 보인다[3]. 뿐만 아니라 MAP 알고리즘은 복호시 한 프레임의 정보 비트를 모두 수신 후 복호가 가능하므로 SOVA복호시 필요한 구축장의 7~9배의 시간 지연보다 더 큰 시간 지연을 가지게 된다. 따라서 SOVA에 비해 성능이 우수한 MAP 알고리즘을 적용하기 위해서는 그 크기를 줄이는 것이 무엇보다 우선이며 그 이외에 복호시 시간 지연을 최소로 하는 것이 중요하다.

본 논문에서는 MAP 복호기의 새로운 구조를 제안하며 복호를 효율적으로 하여 시간 지연을 최소화하는 방법을 제시하였다. 또한 알고리즘상의 연산을 줄여 저장공간의 효율적 이용도 고려하였다. 그 외에 연산시 발생 할 수 있는 overflow문제에 대처하는 평준화 문제를 기존의 방식 이외에 다른 방식을 제안하여 실질적인 MAP 복호기의 구현을 목적으로 하고 있다. 본 연구는 cdma-2000에 제안된 프레임 크기가 378bits(6 tail bits), 20ms인 규격을 구현한다. 부호기는 구축장 K=4, 생성 다항식이 (13,15)<sub>8</sub>이며 부호화율 1/3인 구조이다.

## 2. 본 론

### 2.1 터보 부호

터보 부호에서 사용되는 일반적 터보 복호기의 구성은 다음과 같다.

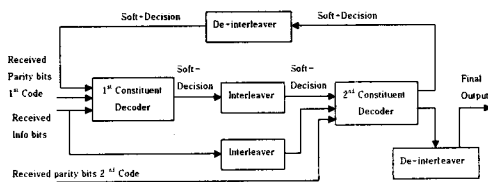


그림 1 터보 복호기

터보 부호에서 복호기는 위와 같이 내부 복호기 (Constituent decoder)와 interleaver 그리고 De-interleaver로 구성되어 있다. 복호과정은 전송된 신호를 첫 번째 복호기에서 그 확률 값을 계산하고 그 값을 interleaver를 통해 순서를 바꾼 후 그 확률 값을 이용하여 두 번째 복호기에서 복호한다. 다시 그 값을 De-interleaver를 이용해 원 순서로 재배열하고 그 확률 값을 이용하여 복호를 재차 시행한다. 이러한 반복과정을 통해 그 확률의 정확성을 높이고 결국 최종적으로는 가장 확률이 높은 값을 선택하여 복호를 끝마치게 된다. 그럼에서 1<sup>st</sup> constituent decoder와 2<sup>nd</sup> constituent decoder는 모두 연입출력 복호기이다. 연입출력 복호기란 그 입력과 출력이 모두 두 개의 논리 값(0 또는 1)을 갖지 않고 실수범위의 값을 갖는다는 것을 의미한다. 이와 같은 연입출력 복호기의 구현 기법으로 사용되는 것이 MAP(Maximum A posteriori) 알고리즘이다.

### 2.2 MAP 알고리즘

MAP 알고리즘에 대한 최초의 제안은 Chang 과 Hancock에 의해 이루어 졌다[4]. 그와 비슷한 시기에 채널부호화에 대한 알고리즘들이 개발되었으나 이러한 알고리즘은 그 구조상 매우 복잡하여 실제적으로 사용하기 위해서는 많은 문제점을 가지고 있었다[5]. 그후에 Berrou등이 터보코드를 제안하면서 이 알고리즘은 다시 한번 주목받게 된다. 기본적으로 MAP알고리즘은 확률 연산이기 때문에 많은 곱셈과 나눗셈 연산을 포함하고 있다. 따라서 이를 구현하기 위해서는 매우 많은 곱셈기를 필요로 한다. 또한 나눗셈을 위해서는 곱셈기보다 큰 하드웨어를 요구한다. 이렇듯 엄청난 설계비용과 면적을 가지는 MAP복호기는 현실적으로 구현 가치가 떨어진다. 그에 따라 이를 보완한 log-MAP 알고리즘이 선보이게 된다. 이는 MAP 알고리즘의 확률연산에 log를 취한 형태이다. 본 논문에서는 이러한 log-MAP알고리즘[5]을 적용하여 MAP복호기를 구현한다.

## 2.3 일반적인 MAP 복호기

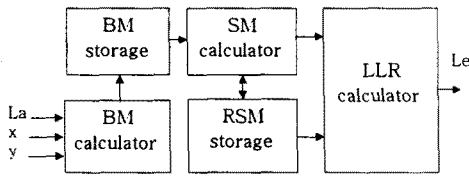


그림 2 일반적인 MAP복호기의 구조(6)

그림2는 log-MAP알고리즘을 이용한 MAP복호기의 일반적인 구조를 보여주고 있다(6). 복호 동작은 입력받은 순서대로 가지 metric(BM, Branch Metric)을 계산, 저장한다. 그 다음에 저장된 값을 이용하여 역순으로 역방향 상태 metric(RSM, Reverse State Metric)을 계산하여 저장한다. 그 뒤 다시 원래 입력 순서대로 순방향 상태 metric(FSM, Forward State Metric)을 계산하면서 이 값과 저장되어 있던 역방향 상태 metric을 이용하여 LLR(Log Likelihood Ratios)를 계산하여 출력한다. 이와 같은 과정으로 복호를 수행하게 되는데 만약 하나의 프레임의 크기가 N 이라고 가정하면 하나의 프레임 전체를 복호하는데 필요한 연산시간은 총 3N이 된다.

## 2.4 제안된 MAP 복호기

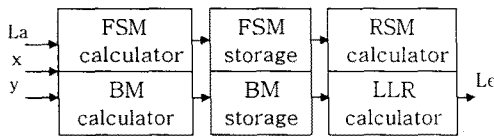


그림 3 제안된 MAP복호기의 구조

그림3은 본 논문에서 제시하는 터보 부호를 위한 MAP복호기의 전체적인 구조를 나타낸 그림이다. 일반적인 MAP복호기와 그 연산부의 역할을 다르게 조정하였고 그 연산순서 역시 다르게 하였다. 그 동작을 살펴 보게 되면 입력된 데이터를 이용하여 입력순서와 동일하게 가지 metric과 순방향 상태 metric을 동시에 계산하고 저장하게 된다. 이 저장된 가지 metric과 순방향 상태 metric을 이용하여 입력순서의 역순으로 역방향 상태 metric과 FSM을 계산하여 출력하게 된다. 그렇게 되면 출력은 입력의 역순이 되어 만약 프레임의 크기가 N일 경우 1부터 N까지의 데이터를 복호하게 되면 N부터 결과가 출력된다. 따라서 이 MAP 복호기를 단독으로 사용할 경우 이 출력된 LLR의 순서를 다시 입력순서대로 바꿔주는 저장공간이 있어야 한다. 그에 따라 모든 연산은 프레임의 크기가 N인 경우 총 2N이면 끝나지만 그 순서를 바꾸는데 N만큼의 시간이 소요된다. 따라서 이 복호기 역시 기존의 MAP 복호기와 동일하게 전체 연산시간은 3N이 되며 반면 그 크기는 불필요한 저장공간이 필요하게 되므로 기존의 MAP 복호기에 비해 커지게 된다. 이는 이 MAP 복호기가 기존의 MAP 복호기에 대해 연산시간은 동일하고 오히려 그 크기는 더 큰 불합리한 구조로 되어 있음을 의미한다. 그러나 만약 이 MAP 복호기가 터보 복호기내의 연입출력 복호기로 사용된다면 터보 복호기 내의 interleaver 혹은 De-interleaver를 적절히 설계함으로써 불필요한 저장

공간은 없앨 수 있다. 즉, interleaver라는 것이 어느 순서로 배열된 값을 다시 재배열하는 것이므로 재배열 시 입력의 역순으로 재배열하게 되면 굳이 저장공간을 만들어서 그 순서를 입력순서로 만들 필요가 없다. 따라서 전체 연산시간은 2N으로 기존의 MAP 복호기에 비해 N만큼의 시간 이득을 볼 수 있다. 이를 터보 복호기에 적용할 경우 한 번의 반복 복호만을 수행한다고 가정하면, 기존의 MAP 복호기는 총 6N의 시간이 필요한 반면 제안된 MAP 복호기는 총 2N의 연산시간이 필요하다. 이러한 차이는 반복횟수가 증가할수록 더욱 커지게 된다.

### 2.4.1 평준화(Normalization)

일반적으로 대부분의 MAP 복호기에서는 상태 metric의 overflow를 방지하기 위해 별도의 장치를 두고 있다. 보통의 방법은 크게 3가지로 나뉘는데 최소 값 감산법, modulo 연산법, 상수 가감산법 등이 있다. 이중 최소 값 감산법은 매 단위시간에 가장 작은 상태 metric을 계산하여 그 값을 다음 단위시간에 감산해 주는 방식으로 지속적인 비교와 감산이 이루어져 불필요한 비교와 감산기가 사용되지 않는다. 따라서 전력소모가 많고 하드웨어적인 면적이 증가하게 된다. modulo 연산법은 특별한 연산이나 비교 없이 어느 일정 값 이상으로 상태 metric의 범위를 설정함으로써 마치 modulo 연산이 수행되는 것과 동일한 효과를 가진다. 이것은 modulo 연산을 통해 동일한 시간에서 각 상태 metric이 어느 일정 값 이하의 차이만을 가진다는 것에 착안한 방법이다. 그러나 이 방법은 MAP 알고리즘에 도입하는데 무리가 따른다. 즉 단순히 두 값의 차이만을 이용하는 경우는 상관없지만 크기를 비교하여 크거나 작은 값을 선택하는 경우는 그 선택이 어렵기 때문이다. 상수 가감산법은 매 연산마다 일정 값을 감산 혹은 가산함으로써 overflow를 방지하는 방법이다. 본 논문에서는 변형된 형태의 상수 가감산법을 사용한다. log-MAP 알고리즘의 연산과정에서 수행하게 되는 E함수가 두 개의 피연산자 중 작은 값을 선택하도록 되어 있으므로(5) 양수의 overflow는 문제가 되지 않는다. 따라서 상태 metric이 음수인 경우 특정 값 이하가 되는 경우에만 양의 상수를 더해 줌으로써 평준화를 수행하는 것이다. 이 방법이 최소 값 감산 법이나 상수 가감산법보다 면적과 전력소모 면에서 유리한 것은 직관적으로 알 수 있다. 따라서 제안된 방식이 일반적인 평준화 기법보다 면적과 전력소모 면에서 더 우위에 있다고 할 수 있다.

### 2.4.2 가지 metric(BM)의 연산과 저장

본 논문에서는 좀 더 효율적인 하드웨어의 활용을 위하여 가지 metric(BM) 연산에서 불필요한 부분을 제거하고 간략화 된 식의 특징을 이용하여 연산량과 저장공간을 줄였다.

$$D_k^i = -K_k - (z_k + Ax_k)j - Ay_k c^{i,m}$$

이 식은 MAP 알고리즘의 가지 metric에 대응하는 log-MAP 알고리즘상의 가지 metric 연산 식이다. 이 식에서  $K_k$ 는 평준화를 위한 상수이다. 따라서 본 논문에서 제시한 방법을 통한 평준화를 고려한다면 이 상수는 가지 metric 계산 시 0으로 처리할 수 있다. 그에 따라 가지 metric은  $j$ 와  $c^{i,m}$ 에 의해서만 그 값이 결정된다. 그런데  $i$ 와  $c^{i,m}$ 는 0혹은 1의 값만을 가진다. 따라서 모든 상태에 따른 가지 metric의 값은 총 4가지만 존재한다. 또한 두 값이 모두 0인 경우 가지 metric의 값은 0이므로 굳이 계산할 필요가 없다. 그에 따라 만약 구속장이 4인 경우 전체 상태는 8개이므로 기존의 연산 방법으로는 16번의 가지 metric연산과 매 단위시간당 16개의 저장공간이 필요하였으나 제안된 복호기에서는 매 단위시간당 3번의 연산과 그에 따른 3개의 저장공간

만을 필요로 한다. 결과적으로 8개의 상태를 가지는 터보 부호에서 제안된 MAP 복호기는 기존의 복호기에 비해 가지 metric의 연산과 저장에 있어서 3/16의 면적 감소 효과를 나타낸다.

### 2.5 모의실험 및 논리합성 결과

본 논문에서는 제안된 MAP 복호기의 구현을 위해 VHDL을 이용하여 설계하였다.

설계된 복호기는 크게 A-D calculator, SRAM, R-L calculator로 나누어진다. A-D calculator는 순방향 상태 metric과 가지 metric을 동시에 계산하는 모듈이고 R-L calculator는 역방향 상태 metric과 LLR을 계산하는 모듈이다. 또한 SRAM은 계산된 순방향 상태 metric과 가지 metric을 저장하는 공간으로 그 크기는 프레임의 크기에 따라 그 크기가 달라지게 된다. 본 논문에서는 프레임의 크기가 378이므로 SRAM의 크기는 5.3Kbyte 정도로 설계하여 시뮬레이션 하였다. 구현된 복호기의 functional level상의 검증은 위해서 Synopsys<sup>TM</sup>사의 VHDL System Simulator(VSS)를 사용하였고 그림4는 그 모의실험의 결과이다. 모의실험 결과 예상대로 한 프레임을 복호하는데 프레임 크기의 2배만큼의 시간이 소요되었다.

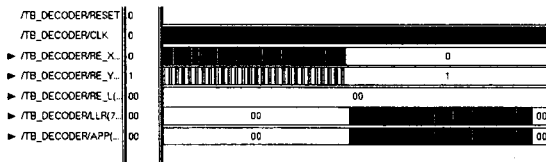


그림 4 모의실험 결과

그림 5, 6은 A-D calculator와 R-L calculator의 논리 합성 결과를 보여 주고 있다. 논리 합성은 Synopsys<sup>TM</sup>사의 Design Compiler를 사용하였으며 LSI logic사의 lsi\_10k 라이브러리를 사용하였다. 이러한 논리 합성된 결과를 토대로 각 부분의 크기를 표로 나타내었다(표 1).

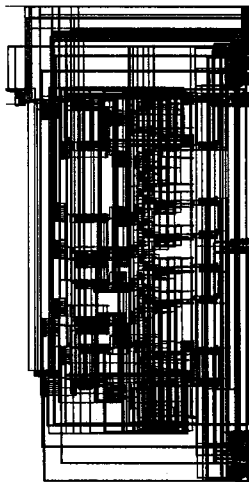


그림 5 A-D calculator

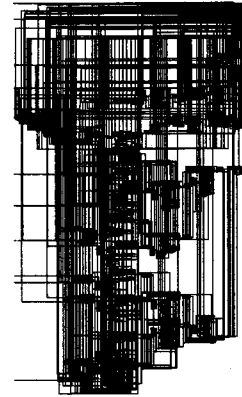


그림 6 R-L calculator

(단위 : Gate)

	Area
제어부	297
A-D cal.	13470
R-L cal.	30970
Total	44737

표 1 제안된 MAP복호기의 부분별 합성결과

### 3. 결 론

본 논문에서는 터보 부호에서 사용되는 MAP 복호기의 효율적 구현을 구체적으로 제안하였다. 우선 구조적으로 새로운 구조의 MAP 복호기를 제안하여 연산에 따른 시간지연을 줄였고 가지 metric 연산에 있어서 연산량과 저장공간을 줄였다. 뿐만 아니라 평균화에서도 새로운 기법을 도입하였다. 그러나 여전히 평균화 기법에 있어서는 제안된 기법이 최적의 기법이 아니므로 앞으로 그에 대한 추가적인 연구가 필요하다.

### (참 고 문 헌)

- [1] David Kwan and Samir Kallel, "A Rate-k/n Heuristic Soft-Output Viterbi Algorithm(SOVA) that is Postdetector-Compatible" IEEE Trans. on Communications, V.46 N.5, 1998.
- [2] Bahl, L., Cocke, J., Jelinek, F., and Raviv, J., "Optimal decoding of linear codes for minimizing symbol error rate," IEEE Trans. Inform. Theory, vol. IT-20, pp. 284-287, Mar. 1974.
- [3] Arun Raghupathy, K. J Ray Liu, "VLSI Implementation Considerations for Turbo Decoding Using A Low Latency Log-MAP" Proceedings of the IEEE 1999 International Conference on Consumer Electronic, 1999.
- [4] R. W. Chan and J. C. Hancock, "On receiver structures for channels having memory" IEEE Trans. Inform. Theory, vol. IT-12, pp. 463-468, oct. 1966.
- [5] Steven S. Pietrobon, "Implementation and Performance of a Turbo/MAP Decoder" International Journal of Satellite Communication, V.16.N.1 1998.
- [6] "Product Databook of MAP Decoder (MAP\_TW1)" ,AnsLab Co., Ltd.