

論文 2002-39SP-6-16

# 레일레이 채널에서 성능 향상을 위한 DVB-T용 비터비 디코더의 아키텍처

## (Novel Viterbi Decoding Architecture for DVB-T with Improved Performance in Rayleigh Channels)

吳姬妍\*, 朴奎鉉\*\*, 李丞峻\*

(Jung Youn Oh, Kyu Hyun Park, and Seung Jun Lee)

### 요약

본 논문은 유럽형 디지털 TV의 표준인 Digital Video Broadcasting(DVB) 수신기에 사용되는 비터비 디코더의 새로운 아키텍처를 제안하였다. OFDM system의 multiple carrier에서 각각의 sub-carrier에 Channel State Information(CSI)를 추정하여, 각 symbol의 bit-metric을 구할 때 가중치를 주는 방식으로 적용하였다. 가중치를 적용한 bit-metric은 심볼 디인터리버와 비트 디인터리버를 통과한 후에 비터비 디코더에 전달된다. 즉, CSI는 두개의 인터리버를 거친 후에도 그에 상응하는 비트에 정확히 적용될 수 있음을 말한다. 본 논문에서 제안한 새로운 아키텍처는 Rayleigh fading 채널에서 6~13dB의 상당한 성능의 향상을 보임을 변조 방식에 따라 시뮬레이션을 통해 보였다. 이 결과는 기존의 방식에 비하여 3.7~10.3dB 더 나은 성능이다.

### Abstract

This paper presents a novel Viterbi decoding architecture for European Digital Video Broadcasting (DVB) receiver. The channel state information (CSI) of each sub-carrier is used to weight the bit-metric of each symbol. The weighted bit-metric is delivered to Viterbi decoder after going through the symbol deinterleaver and bit deinterleaver, such that the CSI can be correctly applied to corresponding bits even after the two interleavings. Simulation shows that the new architecture gives significant performance enhancement of 6~13dB in Rayleigh fading channels depending on the modulation types. This results is also better than previous results by 3.7~10.3dB.

**KeyWords** : Viterbi decoder, CSI, DVB-T, Rayleigh, deinterleaver

### I. 서론

DVB(Digital Video Broadcasting)는 유럽형 디지털

\* 正會員, 梨花女子大學校 情報通信學科 CAD&VLSI 研究室

(Department of Information Electronics Engineering Ewha Womans University)

\*\* 正會員, (株)오픈솔루션

(Open Solution Inc.)

接受日字:2002年7月15日, 수정완료일:2002年10月21日

방송의 표준으로 기존의 위성(DVB-S), 유선(DVB-C), 지상파(DVB-T) 인프라를 이용하여 MPEG2 포맷의 디지털 TV 서비스를 위한 틀을 제공하였다. 현재는 25개 나라에서 200여 단체들이 이에 대한 연구를 하고 있고, 많은 나라들에서 이미 DVB 방송 서비스가 이뤄지고 있다.

DVB-T는 지상파 방송에 적합한 Orthogonal Frequency Division Multiplex (OFDM) 방식을 사용한다.<sup>[1]</sup> OFDM은 단일 반송파에 변조하는 대신에, 다수의 반송파에 변조된 심볼들을 실어서 전송하는 방식이다. 가용 주파수 대역이 서로 겹치지 않도록 N개의 독립적인 채널

널로 나누어 각각의 채널에 변조된 심볼들을 실어서 전송하게 된다. 이때 각 채널 간의 간섭이 발생하지 않도록 각 채널은 다른 채널에 대해서 orthogonal을 유지한다. OFDM은 Single Frequency Network (SFN)을 위해 매우 효과적이지만, 채널간의 간섭을 피하기 위해 보호 구간(guard interval)이 붙게 되어 데이터 전송 속도 측면에서 단점이 있다. DVB-T는 네트워크 성격에 따라서 2K/8K 모드를 지원하는데, 2K는 1705개의 서브 채널에 심볼을 나누어 전송하고, 8K는 6817개의 서브 채널에 심볼을 나누어 전송한다.

DVB-T는 VHF/UHF 대역에서 동작하므로, 지상파 환경에서 발생하는 fading, multipath propagation, 채널의 불완전한 특성과 여러 가지 장애로 인한 Co-Channel Interference (CCI)와 Adjacent Channel Interference (ACI)에 대한 충분한 보호가 필요하다. 채널 상의 에러를 정정하기 위한 오류 정정 방법으로 연접 오류 정정 코드 (Concatenated Forward Error Correction)를 사용하는데, 이것은 블록 부호(block code)와 길쌈 부호(convolutional code)를 직렬로 연결한 것이다. 수신 단에서는 이러한 오류를 정정하기 위해서 외부 부호화 (outer coding)로 RS(Reed-Solomon) coding을, 내부 부호화 (inner coding)로 길쌈 부호화(convolutional coding)를 사용한다.

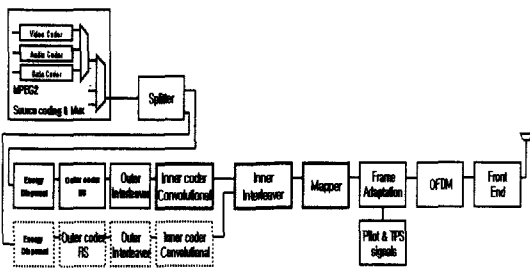


그림 1. DVB-T baseline system 블록도  
Fig. 1. DVB-T Baseline System Blockdiagram.

<그림 1>은 DVB-T 시스템의 송신부의 전체적인 블록도 이다. 두 가지 계층 구조(hierarchical mode)로 전송하기 위해 Splitter를 거친 MPEG-2 데이터는 독립적인 오류 정정 부호화기(FEC: Forward Error Correction)를 거친다. 먼저 에너지의 분산을 목적으로 난수화(randomize) 된 데이터를 외부 RS 부호화기를 거친 다음 내부 길쌈 부호화를 한다. 각각의 채널 부호

화 후에는 군집 에러(burst error)가 발생할 때를 대비하여 인터리빙(interleaving)을 하게 된다. 인터리빙 된 데이터는 QPSK, different level 16QAM, 64QAM 등으로 변조를 한 후, 2K 모드일 때 1705, 8K 모드일 때 6817개의 채널에 변조된 심볼을 나누어 OFDM 방식으로 전송된다.

본 논문은 <그림 1>의 수신 단에 해당하는 내부 디인터리버(de-interleaver)와 Viterbi 복호기에 관한 것으로, Channel State Information (CSI)을 Viterbi 복호기에 적용했을 때, multi-path fading이 발생하는 Rayleigh 채널에서의 성능 측정에 관해 주로 다룬다.

CSI는 채널 상태에 따라 데이터가 얼마만큼 믿을만한지를 나타내는 정보로써, 이 CSI의 제곱을 bit-metric에 곱하여 가중치를 주는 방법으로 적용한다. CSI의 제곱을 곱하여 성능의 향상을 보인 경우는 기존의 논문<sup>[2]</sup>에도 있었으나, 디인터리빙을 하는 방식의 특성상 크게 성능이 향상되지 않았다.

본 논문에서는 CSI를 적용하는 새로운 방법을 제시하였으며, AWGN과 Rayleigh 채널에서의 성능을 시뮬레이션을 통해 측정하였다.

## II. CSI의 적용 방법

OFDM 시스템은 다수의 반송파에 변조된 심볼을 실어서 전송하기 때문에 각 반송파의 신호 대 잡음비(SNR)는 다르게 되고, SNR이 큰 반송파에 실린 데이터일수록 신뢰할 만 하다(reliability가 크다). 이러한 정보는 데이터와 함께 전송되는 파일럿 신호로부터 뽑아낸 CSI를 통해 나타난다. 이는 Viterbi 복호기에서 '0' / '1' 중 어떤 입력을 받았는지를 확률적으로 나타내는 branch-metric에 제공으로 곱해짐으로써 반영되어 채널의 신뢰도에 따라 복호기의 입력에 가중치를 줄 수 있다. Fading이 없는 AWGN 채널에서는 CSI가 일정한 상수 값을 갖게 되지만, multi-path fading이 존재하는 Rayleigh 채널에서는 각 변조된 심볼마다 다른 CSI를 갖게 된다.

CSI는 데이터가 실린 반송파가 전송되면서 거쳐 온 채널의 상태를 나타내기 때문에 올바른 적용을 위해서는 변조된 심볼별로 적용이 되어야 한다. CSI를 적용하는 방법에 따라 성능 향상의 정도가 달라짐을 기존 논문<sup>[2]</sup>을 통해 알 수 있다. DVB-T 시스템에서는 채널

오류 정정 부호화에 앞서 디인터리버(deinterleaver)를 통과해야 하는데, 이것은 심볼 디인터리버(Symbol deinterleaver)와 비트 디인터리버(Bit deinterleaver)로 구성되어 있다. 심볼 디인터리버는 변조된 심볼 단위로 디인터리빙 하는 것이고, 비트 디인터리버는 변조된 심볼을 비트 단위로 나누어서 각각을 디인터리빙 하는 것이다. 디인터리버 이후에 CSI를 적용하게 되면, 비트 디인터리버로 인하여 CSI의 값이 제대로 유지될 수 없기 때문에 성능의 차이가 발생하는 것이다.

기존의 연구<sup>[2]</sup>에서는 CSI를 심볼과 비트 모두 디인터리빙 시킨 방법과 심볼 디인터리빙만 시킨 방법을 사용하여 성능의 향상을 보였다. 먼저 CSI를 심볼과 비트 디인터리버를 모두 거치는 방법이 있다. 비트 디인터리버를 통과할 때 데이터는 변조된 심볼을 구성하는 각각의 비트로 쪼개져서 디인터리빙 되는 반면에, CSI는 데이터와 같이 비트로 쪼개질 수 없다. 그렇기 때문에 각 비트 디인터리버를 거친 평균값을 사용하게 된다. CSI가 비트 디인터리버를 거치는 동안 원래의 값을 유지하지 못하기 때문에 이를 적용하여도 성능의 향상은 거의 없다. 그래서 두 번째로 원래의 CSI를 유지하도록 심볼 디인터리버만 거치는 방법이 있다. 심볼 디인터리버를 거친 데이터는 비트 디인터리빙을 하지만, CSI는 이를 통과하지 않은 상태로 branch-metric을 구할 때 적용하는 것이다. 이러한 방법은 첫 번째 방법처럼 심볼과 비트 모두 디인터리빙 했을 때보다 성능은 많이 향상이 되지만, 비트 디인터리빙이 되지 않았기 때문에 정확한 CSI가 적용될 수 없다.

기존의 CSI를 적용하는 방법의 문제는 DVB-T 시스템 구성 상 오류 정정 부호화의 앞단에 디인터리버가 존재하고, 디인터리빙을 하는 동안 CSI가 정확한 값을 유지할 수 없다는 데에 있다. 변조된 심볼과 CSI를 먼저 디인터리빙한 후, bit-metric을 계산하였기 때문에 디인터리빙을 하는 동안 정확한 CSI가 적용되지 않아 성능의 향상이 거의 없었다.

그러나 본 논문에서는 bit-metric을 미리 구한 후에 디인터리빙을 하는 새로운 방법을 제안하였다. 기존 방법의 문제점이 비트 디인터리버에서 CSI의 값이 정확히 유지되지 않는다는 것이기 때문에, 비트 디인터리버를 거치기 이전에 CSI를 적용하였다. 변조된 심볼 데이터를 심볼 디인터리빙 한 후 비트 단위로 쪼개지기 전에 bit-metric을 구하여 CSI를 적용 시킨다. 그리고 CSI가 적용된 bit-metric을 비트 디인터리빙 한다. 변

조된 데이터와 CSI를 비트 디인터리빙 하는 기존의 방법과는 달리, bit-metric을 미리 구한 후 비트 디인터리빙 하는 것이 가장 큰 차이점이다. 이러한 방법은 CSI가 정확히 적용될 수 있으므로, 기존의 방법보다 성능이 크게 향상하는 것을 볼 수 있다. CSI를 적용하는 새로운 방법을 제안한 구조는 3장에서, 그 방법을 이용한 Rayleigh 채널에서의 성능 측정 결과는 4장에서 보인다.

III. 세부 블록 기능과 설계

<그림 2>는 본 논문에서 제안한 복호기의 전체 블록도이며, 크게 Bit metric calculation과 내부 디인터리버, 그리고 Viterbi 복호기로 구성된다.

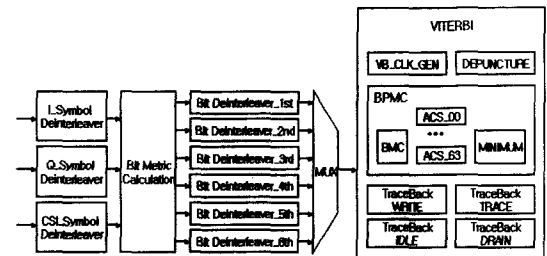


그림 2. Viterbi decoder, Inner De-interleaver의 블록도  
Fig. 2. Viterbi Decoder and Deinterleaver.

(1) De-Interleaver

채널의 불완전한 특성 때문에 전송 중 군집 오류가 발생할 때, 성능의 저하를 막기 위해 각 채널의 오류 정정 부호화 후에는 인터리빙을 통해 데이터를 흩어서 전송하고, 수신 단에서는 이를 원래의 데이터 순서대로 디인터리빙 시킨다. 내부 디인터리버는 <그림 3>에서와 같이 크게 심볼 디인터리버(Symbol deinterleaver), 비트 디인터리버Bit deinterleaver), 멀티플렉스(Mux)로 구성이 된다.

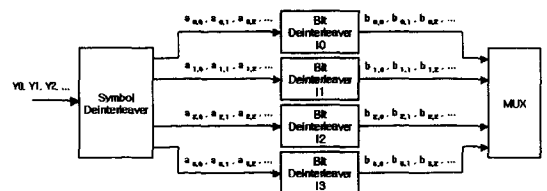


그림 3. 16QAM, Non-Hierarchical mode Deinterleaver 블록도  
Fig. 3. 16QAM Non-Hierarchical Mode Deinterleaver.

심볼 디인터리버는 1705(2K) / 6817(8K) 개의 OFDM 반송파에  $v$  bit 으로 이뤄진 심볼을 매핑(mapping) 하기 위한 목적으로 사용된다. OFDM에서 사용하는 프레임에는 보호 구간과 복조와 복호에 관한 여러 가지 정보를 가지고 있는 파일럿 신호가 존재하기 때문에 이를 제외한 1512(2K) / 6048(8K) 개의 실제 의미 있는 데이터를 단위로 디인터리빙 한다.

비트 디인터리버는 하나의 변조된 심볼을 이루는  $v$ -bit을 각 비트 단위로 쪼개고, 이를 126개를 하나의 블록으로 묶어서 디인터리빙 한다. 즉, 같은 채널을 통과하여 같은 잡음 환경을 거친 비트끼리도 다시 섞이게 된다.

$$\begin{aligned} y'q &= y H(q) && \text{for even OFDM symbol} \\ y' H(q) &= y q && \text{for odd OFDM symbol} \\ y' w &= (a_{0,w}, a_{1,w}, \dots, a_{v-1,w}) \end{aligned} \quad (1)$$

$$\text{out\_bit}_{e,w} = \text{in\_bit}_{e,He(w)} : a_{e,w} = b_{e,He(w)}$$

$$\begin{aligned} \text{bit A} : H0(w) &= w \\ \text{bit B} : H1(w) &= (w+63) \% 126 \\ \text{bit C} : H2(w) &= (w+21) \% 126 \\ \text{bit D} : H3(w) &= (w+84) \% 126 \\ \text{bit E} : H4(w) &= (w+105) \% 126 \\ \text{bit F} : H5(w) &= (w+42) \% 126 \end{aligned} \quad (2)$$

식 (1)은 심볼 디인터리버의 방식으로,  $y(x)$ 가 디인터리버의 입력이고,  $y'(x)$ 가 출력이 된다.  $H(q)$ 는 1512/6048 개의 반송파에 실린 변조된 심볼의 정렬 순서를 나타내는 함수로써, 이는 데이터에 상관없이 일정한 것이므로 ROM을 통해 구현하였고, 수식은 생략한다.

식 (2)는 비트 디인터리버의 방식으로써, 변조된 심볼을 구성하는  $v$ -bit를 비트 단위로 쪼개서 각각을 섞는다. 예를 들어 QPSK 경우는, 하나의 변조된 심볼이 2-bit으로 구성되므로 bit A와 bit B의 방식으로 각각 비트 디인터리빙 된다. 또 64QAM의 경우는, 하나의 변조된 심볼이 6-bit으로 구성되므로 6-bit이 각각 bit A~bit F의 방식으로 디인터리빙 된다.

위의 수식을 보면, 같은 잡음 환경을 거쳐 온 데이터라도 비트 디인터리빙을 통해서 쪼개지고 다른 방식으로 섞인다는 것을 알 수 있다. 하나의 변조된 심볼은 고유한 CSI를 갖게 되는데, 이는 심볼과 같이  $v$ -bit로

쪼개질 수 없기 때문에 비트 디인터리버를 통하는 과정에서 정확히 적용되기가 어렵게 된다.

본 논문에서 제안하는 구조는 비트 디인터리빙을 하기 전에 CSI를 적용하여 미리 bit-metric을 구하는 방식이다. 그러므로 실제로는 비트 디인터리버가 아닌 bit-metric 디인터리버로 동작하게 된다.

멀티플렉스는 비트 디인터리버를 통해서 비트 단위로 쪼개진 데이터들을 Hierarchical mode, 변조 방식에 따라 정해진 순서에 맞게 조합하여 멀티플렉싱 한다.

## (2) Bit-Metric Calculation

'Bit-Metric Calculation' 블록은 수신된 심볼과 성상(constellation) 위의 기준 신호 사이의 거리를 구하고, CSI를 적용하는 블록이다. Bit-metric은 수신된 심볼이 성상 위의 기준 신호 중 어느 것과 가까운지를 나타내어 '0'에 가까운 값인지 '1'에 가까운 값인지를 나타낸다. 경판정(hard-decision)의 경우는 모든 기준 신호와의 거리가 가장 가까운 심볼을 찾아서 판단하면 되지만, 연판정(soft-decision)의 경우는 '0'과 '1'의 거리 중 어느 쪽으로 더 가까운지를 확률적으로 나타내는 bit-metric을 계산하여 Viterbi 복호에 이용한다.

Bit-metric을 계산할 때, 모든 기준 신호와의 거리를 구하게 되면 64QAM의 경우는 64개의 bit-metric을 구해야 하므로 하드웨어 면적이 커지게 된다. 또 비트 디인터리버를 거치기 위해서 <그림 4>와 같이 변조된 심볼을 이루는 각  $v$ -bit에 대해서 거리를 구하는 방법을 사용하였다.<sup>[4]</sup> 기준 신호들의 각 bit에 대해서 '0'과 '1'의 집합으로 나눈 뒤, '0'의 집합 중 가장 bit-metric이 작은 것과 '1'의 집합 중 가장 작은 bit-metric을 선택한다.<sup>[4]</sup> 이때 각 집합의 선택된 가장 작은 bit-metric 값의 절대적인 값이 중요한 것이 아니라 어느 쪽에 더 가까운 지가 중요하기 때문에 '0'과의 거리와 '1'과의 거리의 상대적인 차이만을 취한다. 이렇게 하면, 이후에 거치는 비트 디인터리버의 개수를 12 개에서 6개로 줄일 수 있다.

아래 <그림 4>는 16QAM의 성상을 나타내며, '1'이란 신호를 받았을 때 변조된 심볼을 구성하는 4bit 중 에서 첫 번째 bit에 대해서 bit-metric을 구하는 예이다. 'distance\_1st\_0'은 'S3'과의 거리를 계산한 것이 되고, 'distance\_1st\_1'은 'S11'과의 거리를 계산한 것이 되는데, I-phase의 거리를 구할 때는 Q-phase의 거리는 같기 때문에 계산하지 않아도 된다.

각 비트 별 비트 디인터리버는 아래의 수식에 따라 계산된다.

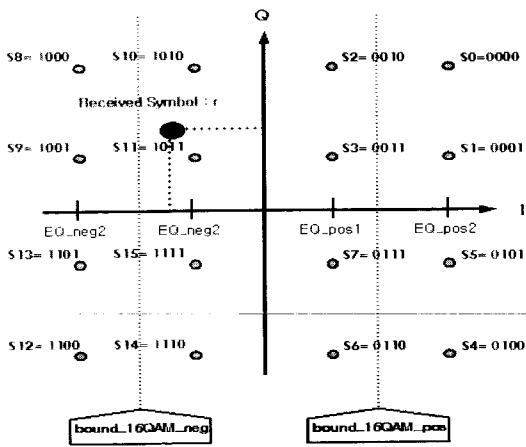


그림 4. Bit-metric calculation-16QAM, alpha=1의 예  
Fig. 4. Bit-metric calculation in 16QAM with alpha=1.

- modulation symbol을 이루는 첫 번째 bit에 대해서,
- \* 수신된 신호를 좌표에서  $r [i,q]$  라고 표시한다.
- \* distance\_1st\_0 : min[distance from received symbol to reference symbol of n-th bit '0' group]
- \* distance\_1st\_1 : min[distance from received symbol to reference symbol of n-th bit '1' group]
- \* ref\_q : I-phase 거리를 구할 때, Q-phase reference가 되는 기준값

```

if ( r[i] > bound_16QAM_neg )
distance_1st_1 = (r[i] - EQ_neg1)2 + (r[q]-ref_q)2
else
distance_1st_1 = (r[i] - EQ_neg2)2 + (r[q]-ref_q)2

if ( r[i] > bound_16QAM_pos )
distance_1st_0 = (r[i] - EQ_pos2)2 + (r[q]-ref_q)2
else
distance_1st_0 = (r[i] - EQ_pos1)2 + (r[q]-ref_q)2
    
```

subtract\_1st = distance\_1st\_0 = distance\_1st\_1

$BM = |Hk|2 | bn |2$

$BM_{1st} = distance_{1st_0} \times (CSD)2 - distance_{1st_1} \times (CSD)2$   
 $= subtract_{1st} \times (CSD)2$  (3)

OFDM 시스템은 다수의 반송파를 사용하여, 하나의 OFDM 심볼을 이루는 1512/6048 개의 변조된 심볼들이 각기 다른 반송파에 실려 전송되기 때문에, 채널 상태에 따라서 각 심볼들이 받는 SNR이 다르게 되고 이는 CSI로 표현된다. 채널에서 전송되는 신호가 multipath fading에 의해 받게 되는 신호의 감쇠 정도가 채널 측정(channel estimation)에 의해 구해진 CSI와 비례하므로, 일정한 AWGN 잡음이 더해졌을 때, CSI와 SNR은 비례하게 된다. SNR이 높은 반송파에 실린 데이터는 낮은 SNR 에서 전송된 데이터보다 신뢰도가 높기 때문에, CSI도 큰 값을 갖게 된다. CSI는 이러한 신뢰의 정도를 나타내는 정보로써, 수신된 신호에 가중치를 줄 수 있다. Euclidian distance 방식으로 구한 bit-metric에 이 CSI의 제곱을 곱함으로써 복호할 때 가중치를 줄 수 있다. 식 (3)에서, Bit-metric BM은 연판정 된 수신 신호 'r'과 기준 신호 's11'과의 거리 'bn'에 CSI의 제곱을 가중치를 주어 구한다. 여기서  $|Hk|2$  는 CSI를 의미하며, k 번째 반송파의 채널 측정 결과 SNR이다.

multi-path fading이 없는 AWGN 채널에 대해서는 CSI는 일정한 상수 값을 유지하지만, Rayleigh 채널에서는 Equalizer의 채널 측정에 의해서 CSI를 추출할 수 있다.

(3) Viterbi Decoder

Viterbi 복호기는 <그림 2>와 같이 VB\_CLK\_GEN, Depuncture, BPMC, Trace-Back 블록으로 구성된다.

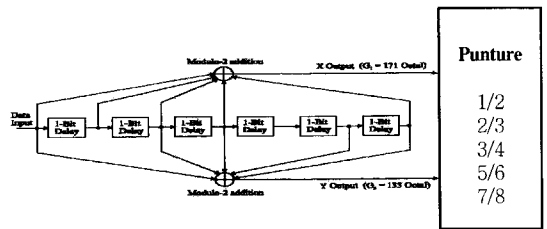


그림 5. Convolutional Encoder.  
Fig. 5. Convolutional Encoder.

Viterbi 복호기는 길쌈 부호화를 복호하기 위해 가장 보편적으로 사용하는 maximum-likelihood 방식의 알고리즘이다. 길쌈 부호화는 구속장 (constraint length) K 개의 입력을 조합하여 전송하는데, 기본적으로 1/2 부호률 (code rate)을 사용하고, 천공(puncturing) 방식을 이용하여 2/3, 3/4, 5/6, 7/8의 부호률을 지원하게 된다.

### ① VB\_CLK\_GEN, Depuncture

Viterbi 입력은  $10\text{MHz} \times 4 = 40\text{MHz}$  (64QAM 경우 내부 디인터리버의 입력 속도  $10\text{MHz}$ 의 세배의 속도로 출력되지만, 세배 빠른 클럭이 없기 때문에 가장 빠른 64QAM을 기준으로 4배 클럭을 사용하고 변조 방식에 따른 enable 신호를 만들어 Viterbi 복호기에 입력된다.)의 속도로 들어온다.

Depuncturing은 puncturing된 자리에 중간 값을 채워 넣게 되는데, 이때 데이터 전송 속도가 증가하게 된다. 가장 많이 데이터 전송 속도가 증가하는 7/8 부호율을 기준으로, Viterbi 복호기가 동작하는 클럭을 만들기 위해서 입력 클럭의 2배 클럭이 필요하게 된다. 그래서 기본적으로 Viterbi 복호기는  $40\text{MHz} \times 2 = 80\text{MHz}$  클럭으로 동작하고, VB\_CLK\_GEN 블록에서 각 부호를 별로 압축된 클럭을 위한 enable ('enable\_viterbi') 신호를 만들어 낸다.

### ② BPMC (Branch & Path Metric Calculation)

BPMC는 <그림 2>와 같이 Branch-Metric Calculation (BMC)과 64개의 Add-Compare and Selection (ACS) array, 그리고 MINIMUM 블록으로 구성되어 있다.

길쌈 부호화에서 하나의 상태가 가질 수 있는 출력의 조합은 4가지(codeword)가 존재한다. BMC는 이 4가지 codeword와 수신된 신호 간의 거리, 즉 branch-metric을 구하는 블록으로, 이미 구해진 Bit metric의 I/Q-phase의 bit-metric의 합으로 구할 수 있다.

BPMC 내에는 총 64개의 ACS가 있는데, 이는 길쌈 부호화에 존재할 수 있는 모든 상태의 개수이다. ACS는 22bit의 branch-metric을 입력으로 받게 되는데, 정상 기준 신호의 신호 크기를 볼 때, branch-metric은 제공된 값이기 때문에 항상 적어도 32 이상의 값을 갖게 된다. 그래서 path-metric (누적된 branch-metric)의 비트를 많이 할당하지 않기 위해 LSB 4bit를 truncation 하였다. path-metric은 reset이 되지 않는 한 계속 누적되기 때문에 22bit으로 표현될 수 있는 가장 큰 값을 넘지 않도록 re-scaling이 필요하다. 하나의 path-metric이라도 22bit으로 표현되는 가장 큰 값의 1/2를 넘을 경우 현재의 가장 작은 path-metric 값을 빼주고, 적어도 144 traceback depth마다 한번씩 가장 작은 path-metric을 빼줌으로써 오버플로우(overflow)가 발생하지 않도록 한다. 두 가지 경로 중 선택된 경로가 무엇인지를 결정하는 decision 값은 Trace-Back

블록에 저장되어 이로부터 Trellis 도를 따라 복호하게 된다.

MINIMUM 블록은 64개의 상태가 누적하고 있는 path-metric 중에서 가장 작은 path-metric을 갖는 상태를 찾아내는 블록이다.

### ③ Trace-Back

Trellis 도를 보면 알 수 있듯이, Viterbi는 하나의 상태 변이를 보고 복호 하는 것이 아니라 일정 traceback depth를 지난 후에 복호를 시작하는데, 본 설계에서는 144를 traceback depth로 하였다. BPMC로부터의 decision 값들은 이 traceback depth만큼 메모리에 저장되었다가 복호가 시작된다.

Trellis 도의 복호 방식과 같이 역추적은 에러가 발생할 확률이 가장 작은, 즉 path-metric이 가장 작은 상태를 찾아서 이로부터 시작한다. 만약 이 시작 상태를 잘못 찾을 경우 올바른 경로로 수렴하기까지 잘못된 복호를 하게 된다. 그렇기 때문에 이후 데이터를 보고 이로부터 수렴된 경로를 찾는 방법을 사용 하였다. 그렇기 때문에 3-cycle이면 충분한 역추적이 수렴된 경로를 찾는 IDLE cycle이 추가되어 WRITE/ TRACE/ IDLE/ DRAIN의 4-cycle이 필요하게 되고, 4개의 메모리가 소요된다.

Trace-Back 블록은 LIFO (last-input, first-output) 블록이 포함되어 있다. 역추적은 나중 data를 먼저 decoding 하기 때문에 이를 원래의 순서대로 재정렬하는 LIFO가 필요하다.

## IV. Simulation 결과

본 논문에서는 CSI를 적용한 Viterbi 복호기에 대해서 fixed-point 시뮬레이션을 수행하여 AWGN과 Rayleigh 채널에서 C/N에 대한 BER 성능을 측정하였다. Fading이 존재하는 Rayleigh 채널에서는 CSI를 적용할 때 매우 큰 성능의 향상을 볼 수 있었다. 또, 비트 디인터리버 전에 먼저 bit-metric을 구함으로써 CSI를 적용할 때, CSI를 심볼과 비트 디인터리빙을 하거나 심볼 디인터리버만을 하는 기존의 방법보다 성능이 향상됨을 확인하였다. DVB-T spec.을 제시한 ETSI EN 300 744 V1.3.1에서 요구한 BER과 시뮬레이션 결과와 비교하였다.

먼저 <표 1>은 설계 사양과 시뮬레이션 조건을 요약한 것이다.

표 1. 설계 사양과 simulation 조건

Table 1. System Specification and Simulation Condition.

설계 사양			Simulation 조건	
Input bit	L/Q	7bit	Hierarchical	Non (alpha=1)
	CSI	4bit		
Bit Metric bit	22bit		FFT mode	2K
Path Metric bit	22bit		# of source	106
Symbol 입력 clock	10MHz			
Viterbi 동작 clock	80MHz			
Tracback Depth	144			

(1) AWGN channel

<그림 7, 8, 9>는 AWGN 채널에서 QPSK, 16QAM, 64QAM 등의 세 가지 변조 방식에 따라서 각 부호를 별로 SNR에 대한 BER을 측정 한 것이다.

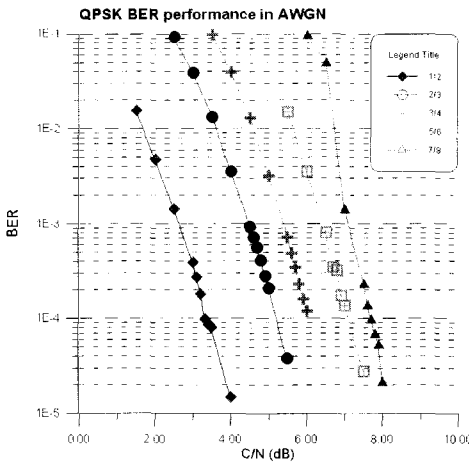


그림 6. AWGN, QPSK

Fig. 6. QPSK in AWGN Channel.

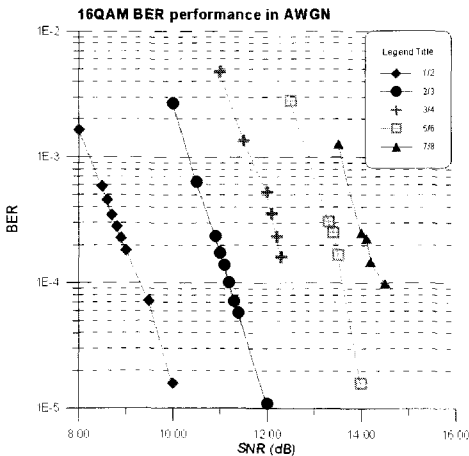


그림 7. AWGN, 16QAM

Fig. 7. 16QAM in AWGN Channel.

<표 2>는 ETSI EN 300 744 V1.3.1에서 제시한  $2 \times 10^{-4}$ 의 BER을 만족하는 SNR을 비교한 것이다.

표 2. AWGN채널에서 ETSI 300 744 V1.3.1과 simulation 결과 비교

Table 2. Simulation Results in AWGN Channel.

Modulation Type	Code rate	Required C/N for BER = $2 \times 10^{-4}$ After Viterbi in AWGN channel	
		ETSI EN 300 744 V1.3.1	Simulation-결과
QPSK	1/2	3.1 dB	3.2 dB
	2/3	4.9 dB	5 dB
	3/4	5.9 dB	5.8 dB
	5/6	6.9 dB	6.9 dB
	7/8	7.7 dB	7.5 dB
16QAM	1/2	8.8 dB	9.0 dB
	2/3	11.1 dB	11 dB
	3/4	12.5 dB	12.2 dB
	5/6	13.5 dB	13.5 dB
	7/8	13.9 dB	14.1 dB
64QAM	1/2	14.4 dB	13.5 dB
	2/3	16.5 dB	16.4 dB
	3/4	18.0 dB	18.2 dB
	5/6	19.3 dB	19.1 dB
	7/8	20.1 dB	19.8 dB

<그림 6, 7, 8> code rate 별 C/N에 따른 BER

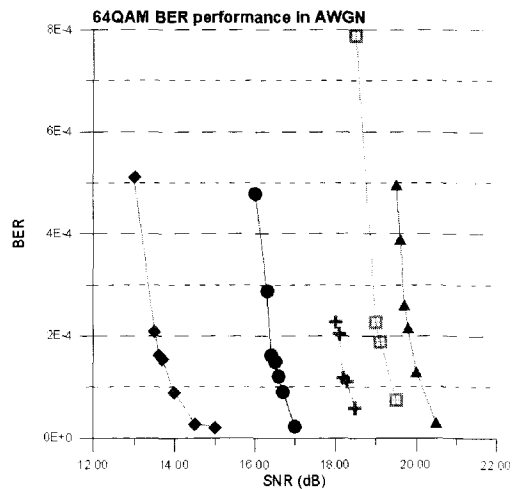


그림 8. AWGN, 64QAM

Fig. 8. 64QM in AWGN Channel.

(2) Rayleigh channel

<그림 9, 10, 11>은 Rayleigh 채널, 1/2 code rate에 대해서 QPSK, 16QAM, 64QAM 경우, CSI를 사용하

지 않았을 때와 논문에서 제시한 방법으로 CSI를 사용한 경우에 SNR에 대한 BER을 측정한 결과이다.

표 3. Rayleigh : ETSI EN 300 744 V1.3.1과 simulation 결과 비교

Table 3. Simulation Results in Rayleigh Channel.

Modulation Type	Code Rate	BER	SNR (dB)	Required dB	
QPSK	1/2	2.55 e-4	5.5	5.4	
		1.99 e-4	5.7		
		1.22 e-4	5.9		
	3/4	1.01 e-4	6.1		
		2.75 e-4	10.7		10.7
		1.61 e-4	10.9		
	7.5 e-5	11.1			
	7/8	7.5 e-5	16.5		16.3
		6.6 e-5	16.5		
6.3 e-5		16.7			
16QAM	1/2	2.46 e-4	11.2	11.2	
		1.73 e-4	11.4		
		1.09 e-4	11.6		
	3/4	2.14 e-4	16.7		16.7
		1.89 e-4	16.8		
	7/8	4.3 e-5	22.8		22.8
3.1 e-5		23.0			
64QAM	1/2	1.08 e-4	16.0	16.0	
		7.7 e-5	16.2		
		5.6 e-5	16.4		
	3/4	2.81 e-4	21.7		21.7
		2.24 e-4	21.9		
		1.88 e-4	22.1		
	7/8	1.06 e-4	27.5		27.9
		9.5 e-5	27.7		
		5.1 e-5	27.9		

<표 3>은 ETSI EN 300 744 V1.3.1에서 제시한  $2 \times 10^{-4}$ 의 BER을 만족하는 SNR을 비교한 것이다.

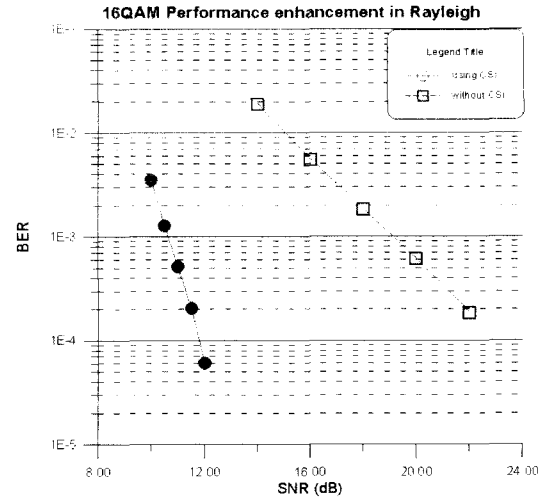


그림 10. Rayleigh, 16QAM, 1/2 -BER enhancement  
Fig. 10. BER in Rayleigh Channel - 16QAM.

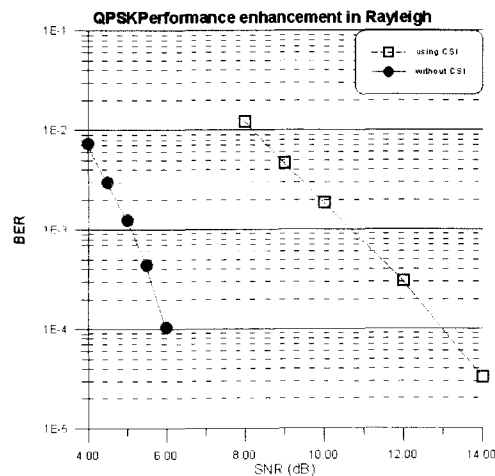


그림 9. Rayleigh, QPSK, 1/2 - BER enhancement  
Fig. 9. BER in Rayleigh Channel - QPSK.

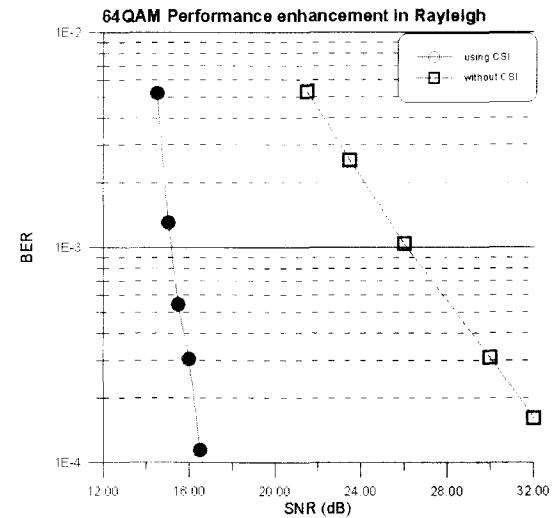


그림 11. Rayleigh, 64QAM, 1/2 -BER enhancement  
Fig. 11. BER in Rayleigh Channel - 64QAM.

<표 4>는 CSI 적용 방식에 따라 성능의 향상 정도를 비교한 것이다. 본 논문에서 제안한 방법으로 CSI를 적용했을 경우, 디인터리버를 모두 통과한 경우나 심볼 디인터리버만을 통과한 CSI를 적용했을 경우보다 BER 성능이 한층 향상되었음을 실험을 통해 확인할 수 있었다. 이러한 성능의 향상은 기존의 방식이 정확한 CSI를 유지하지 못했던 반면, 본 논문에서는 정확한 CSI를 적용할 수 있도록 bit-metric을 먼저 구해 CSI를 적용



삭히고, bit-metric을 비트 디인터리빙 하는 구조를 제안했기 때문이다.

표 4. CSI 적용 방식에 따른 성능 향상의 정도 비교

Table 4. Performance Comparison in Rayleigh Chnnal.

Modulation Type	제안한 방법	*기존의 방법 [2]	
		Full deinterleaver	No bit deinterleaver
QPSK	6.6 dB	*0.7 dB	*2.9 dB
16QAM	13 dB	*0.65 dB	*2.7 dB
64QAM	10.5 dB	*0.15 dB	*2.2 dB

\*디인터리버를 모두 거친 경우와 심볼 디인터리버만 거친 경우의 성능 향상은 기존의 논문 [2],[3]을 참조한 것이다.

### V. 결 론

지상파를 통한 TV 방송의 경우 multipath 환경에서의 fading으로 인해 Rayleigh channel에서 성능의 저하가 나타나게 된다. 본 논문에서는 DVB-T 수신 단에 사용되는 Viterbi 복호기와 이에 따른 내부 디인터리버의 구조에 CSI정보를 통한 가중치를 적용함으로써 Rayleigh fading channel에서 보다 향상된 성능을 보이는 복호기 구조를 제안하였다. 또 기존의 보고되어진 방식들과는 달리 비트 디인터리버 이전 단계 CSI 정보를 적용하여 bit-metric을 구함으로써 CSI 정보가 해당

되는 비트에 정확히 이용될 수 있도록 하였다. 시뮬레이션 결과를 분석하여 보면 Rayleigh 채널에서 CSI를 적용하지 않았을 때 보다 약 6~13dB 정도의 성능 향상을 나타냈고, 기존의 CSI를 적용한 방법보다 약 3~10dB의 성능 향상을 확인하였다.

### 참 고 문 헌

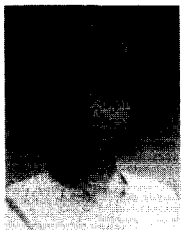
[1] ETS EN 300 744 : "Digital broadcasting systems for television, sound and data services: framing structure, channel coding and modulation for digital Terrestrial television"

[2] Weon-cheol Lee, Hyung-Mo Park, Kyung-jin Kang and Kuen-bae Kim "Performance Analysis of Viterbi Decoder Using Channel State Information in COFDM System" *IEEE Trans. Broadcasting*, Vol. 44, No. 4, December 1998.

[3] Weon-cheol Lee, Hyng-Mo Park and Kong Seok Park "Viterbi Decoding Methods using Channel State Information in COFDM System" *IEEE Trans. Consumer Electronics*, Vol. 45, No. 3, AUGUST 1999.

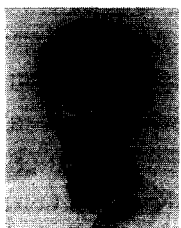
[4] Kyu-Man Lee, Dong-Seog Han and Ki-Bum Kim "Performance of the Viterbi Decoder for DVB-T in Rayleigh Fading Channels" *IEEE Trans. Consumer Electronics*, Vol. 44, No 3, August 1998.

### 저 자 소 개



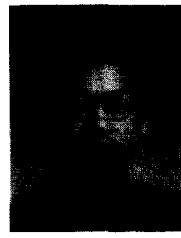
吳 姬 妍(正會員)

2000년 이화여자대학교 전자공학과 학사. 2002년 이화여자대학교 정보통신학과 석사. 2002년 2월~현재 (주)오픈솔루션 SOC팀 연구원



朴 奎 鉉(正會員)

1988년 공과대학 전기공학 학사. 1991년 Rensselaer Ploytechnic Institute, 전기공학 석사. 1995년 Rensselaer Ploytechnic Institute, 정보통신 박사. 1997년 현대전자 선임연구원. 1998년~현재 (주)오픈솔루션 수석연구원



李 丞 晔(正會員)

1986년 서울대학교 전자공학과 학사. 1989년 University of California, Berkeley 전자공학과 석사. 1993년 University of California, Berkeley 전자공학과 박사. 1992년 1월~1998년 10월 현대전자 시스템IC 연구소. 1999년 3월~현재 이화여자대학교 전자공학과 조교수. <주관심분야 : 통신용 반도체 설계, CAD>