



저작자표시-비영리-변경금지 2.0 대한민국

이용자는 아래의 조건을 따르는 경우에 한하여 자유롭게

- 이 저작물을 복제, 배포, 전송, 전시, 공연 및 방송할 수 있습니다.

다음과 같은 조건을 따라야 합니다:



저작자표시. 귀하는 원저작자를 표시하여야 합니다.



비영리. 귀하는 이 저작물을 영리 목적으로 이용할 수 없습니다.



변경금지. 귀하는 이 저작물을 개작, 변형 또는 가공할 수 없습니다.

- 귀하는, 이 저작물의 재이용이나 배포의 경우, 이 저작물에 적용된 이용허락조건을 명확하게 나타내어야 합니다.
- 저작권자로부터 별도의 허가를 받으면 이러한 조건들은 적용되지 않습니다.

저작권법에 따른 이용자의 권리는 위의 내용에 의하여 영향을 받지 않습니다.

이것은 [이용허락규약\(Legal Code\)](#)을 이해하기 쉽게 요약한 것입니다.

[Disclaimer](#)

공학석사 학위논문

Deep fading 채널을 고려한
블록단위 SNR-scalable coding

Block based SNR-scalable coding
for deep fading channels

2017 년 8 월

서울대학교 대학원

전기 정보 공학부

홍 성 진

Deep fading 채널을 고려한 블록단위 SNR-scalable coding

지도 교수 채 수 익

이 논문을 공학석사 학위논문으로 제출함
2016 년 12 월

서울대학교 대학원
전기 정보 공학부
홍 성 진

홍성진의 공학석사 학위논문을 인준함
2016 년 12 월

위 원 장 최 기 영 (인)

부위원장 채 수 익 (인)

위 원 이 혁 재 (인)

초 록

본 논문은 새로운 multiple-layer SNR-scalable coding을 제안하고, 무선통신망에서 fading 때문에 채널 용량이 급격하게 변하는 경우에도 비디오 영상을 끊김 없이 전송하기 위하여 제안한 video encoding을 최적화하는 문제를 다룬다. 그리고 제안한 multi-layer SNR-scalable coding 알고리즘은 블록단위로 동작하도록 HEVC 표준을 수정 확장하여 H/W구현에 적합하다.

실험에 이용한 시나리오에서는 fading 채널 용량의 변화를 구현하기 위해 이를 결정하는 3가지 파라미터(평균 SNR과 line-of-sight(K) 비중, maximum Doppler shift)를 조절하였다. 각 시나리오에서는 level cross rate(LCR), average fade duration(AFD), average inter-fade duration(AIFD)를 측정하였다.

본 논문에서는 시가변 채널의 SNR 정보를 일정한 지연시간 후에 매 packet마다 feedback 받지만, 구현은 picture 단위로 MCS 적용하였다. 채널 용량은 slice, picture, 2-GOP 단위마다 추정하였다. 이를 이용하여 coding layer 수를 결정하고, decoding distortion 기반 cost을 최소화하도록 각 layer별 평균 target bitrate을 정하였다.

Coding layer 수는 2-GOP 단위로 fading이 일어나는 횟수와 fading depth를 이용하여 채널 상태를 추정한 후, 이전 2-GOP 단위에서의 전송 layer 수 (K)의 평균 값에 이 추정 결과를 활용하여 결정하였다. Picture 단위 bit allocation에서는 decoding distortion 기반 cost를 최소화하도록 모든 경우의 bit allocation을 비교해 layer별 target rate를 할당하였다.

마지막으로, 전송 layer 수 K 는 인코딩 결과 실제 사용된 layer별 bitrate을 예측한 채널 용량으로 나눈 값을 이용하여 결정하였다. 이 때 각 layer별로 계산된 값을 본 논문에서 이용한 MCS code rate의 최대 값인 13/16과 비교하여, 이 값보다 작은 값을 가지는 layer까지 전송할 수 있는 것으로 가정하였다.

실험에는 2초 분량의 Ricean 채널을 이용하였다. 실험 시나리오는 송신단과 수신단이 2m 떨어진 상황에서 수신단이 송신단으로부터 1m/s로 멀어지며, 1초에서 1.5초 사이에 Ricean factor 값이 0.1인 장애물이 출현하는 상황을 가정하였다. 실험은 coding layer 수를 4개로 고정하고, bit을 layer별로 동일하게 할당할 것을 anchor로 사용하였다. 실험 결과, 수신단에서 측정된 Y-PSNR은 본 논문 실험 환경에서 평균 0.768dB 개선된 것을 확인하였다.

주요어 : high efficiency video coding(HEVC), SNR-scalable coding, rate control, channel capacity, fading channel

학 번 : 2015-21009

목 차

제 1 장 서 론	1
1.1 연구의 배경.....	1
1.2 관련 연구.....	3
1.3 논문의 구성.....	4
제 2 장 Rate control 알고리즘과 IEEE 802.11ad 프로토콜 설명	5
2.1 Rate control 알고리즘.....	5
2.1.1 기존 HM의 rate control 알고리즘.....	6
2.1.1.1 Picture / CTU bit allocation	7
2.1.1.2 λ estimation.....	9
2.1.1.3 QP refinement.....	11
2.1.2 제안하는 rate control 알고리즘	12
2.1.2.1 제안하는 picture bit allocation	12
2.1.2.2 제안하는 누적 rate model	12
2.2 IEEE 802.11ad 프로토콜 설명.....	19
2.2.1 Preamble.....	20
2.2.2 Header / Data	20
2.2.3 Low Density Parity Check(LDPC)	23
제 3 장 SNR-scalable coding 알고리즘 설명	24
3.1 블록(CTU) 단위 동작.....	25
3.2 Prediction 단계에 적용된 fast 알고리즘	25
3.3 Layer selector를 위한 bitstream에서의 layer 구분 ...	26
3.4 Decoding 성능을 높이기 위한 error concealment	

알고리즘 28

제 4 장 Wireless 채널 용량 모델.....	30
4.1 Fading 채널 모델을 이용한 feedback 채널 정보의 추정	31
4.2 Slice/picture 정보를 이용한 picture/2-GOP 단위 채널 용량 추정	36
4.3 SNR과 packet error rate의 관계	37
제 5 장 Layer 수 결정과 layer별 bit allocation.....	38
5.1 k-GOP 단위 평균 layer bit 할당과 코딩 layer 수 결정	38
5.2 Picture 단위 layer별 bit 할당	40
5.3 Picture 단위 전송 layer 수 결정	43
5.4 Slice 단위 layer별 bit 할당.....	45
5.5 MCS index 결정	47
제 6 장 Fading channel 환경의 실험 결과	48
6.1 채널 용량 추정 관련 실험.....	48
6.1.1 Fading channel 시나리오 구성	48
6.1.2 채널 특성 파라미터 측정	54
6.1.3 Regression을 이용한 채널 용량 추정.....	56
6.2 Layer 수 결정 및 layer별 bit allocation 관련 실험....	60
6.2.1 Coding layer 수와 전송 layer 수의 비교	60
6.2.2 Allocation에 따른 성능 평가.....	63
제 7 장 결 론.....	67

표 목차

표 1. LOW-DELAY B에서의 ω_{pic}	7
표 2. RANDOM ACCESS에서의 ω_{pic}	8
표 3. BPP와 δ 의 관계.....	11
표 4. BASE LAYER (L0)에서의 BD-RATE	16
표 5. ENHANCEMENT LAYER (L1)에서의 BD-RATE.....	17
표 6. ENHANCEMENT LAYER (L2)에서의 BD-RATE.....	17
표 7. BASE LAYER (BL)에서의 BD-RATE.....	18
표 8. MODULATION AND CODING SCHEMES (MCS).....	21
표 9. ERROR CONCEALMENT 알고리즘의 평균 성능	30
표 10. MODULATION과 BIT ERROR PROBABILITY의 근사 관계	37
표 11. 사용한 MCS TABLE.....	44
표 12. 사용한 MCS TABLE (PICTURE 단위)	44
표 13. $f_D = 100\text{HZ}$ 일 때의 채널 특성 파라미터.....	54
표 14. $f_D = 200\text{HZ}$ 일 때의 채널 특성 파라미터.....	55
표 15. $f_D = 400\text{HZ}$ 일 때의 채널 특성 파라미터.....	55
표 16. $f_D = 800\text{HZ}$ 일 때의 채널 특성 파라미터.....	55
표 17. 2-GOP 단위 N과 K 평균 값의 비교 (BASKETBALLDRIVE, 100:1)	61
표 18. 2-GOP 단위 N과 K 평균 값의 비교 (BQTERRACE, 100:1)	61
표 19. 2-GOP 단위 N과 K 평균 값의 비교 (CACTUS, 100:1)	62
표 20. 2-GOP 단위 N과 K 평균 값의 비교 (KIMONO, 100:1)	62

표 21. 2-GOP 단위 N과 K 평균 값의 비교 (PARKSCENE, 100:1)	63
표 22. 수신단 기준 DEFAULT와 ADAPTIVE의 Y-PSNR 비교 (CLASS B, 100:1)	64

그림 목차

그림 1. HM의 RATE CONTROL 알고리즘	6
그림 2. BASE LAYER에서의 R-D CURVE (BPP-MSE)	13
그림 3. ENHANCEMENT LAYER에서의 R-D CURVE (BPP-MSE).....	14
그림 4. 누적 ENHANCEMENT LAYER에서의 R-D CURVE (BPP-MSE)	15
그림 5. PHY MODES (PACKET OVERVIEW)	19
그림 6. CONSTELLATION DIAGRAM BPSK/QPSK	22
그림 7. RECTANGULAR 16-QAM CONSTELLATION DIAGRAM.....	23
그림 8. 블록 단위로 동작하는 SNR-SCALABLE ENCODER 구조	25
그림 9. FAST 알고리즘이 적용된 PREDICTION 단계.....	26
그림 10. BITSTREAM에서의 LAYER 구분 (SLICE 단위)	27
그림 11. LAYER 개수 SELECTOR	27
그림 12. ERROR CONCEALMENT 알고리즘.....	29
그림 13. RICEAN FADING CHANNEL 모델의 예시	33
그림 14. 가정된 SNR-SCALABLE CODING과 WIRELESS MODEM의 관계	34
그림 15. PICTURE내 SLICE와 PACKET의 관계	35
그림 16. LAYER 결정 전 RATE ALLOCATION 문제.....	38
그림 17. $f_D = 100\text{HZ}$ 일 때의 채널 상태	50
그림 18. $f_D = 200\text{HZ}$ 일 때의 채널 상태	51

그림 19. $f_D=400\text{HZ}$ 일 때의 채널 상태	52
그림 20. $f_D=800\text{HZ}$ 일 때의 채널 상태	53
그림 21. SLICE 단위 채널 용량 추정.....	57
그림 22. PICTURE 단위 채널 용량 추정.....	58
그림 23. 2-GOP 단위 채널 용량 추정	59
그림 24. 수신단에서의 Y-PSNR (BASKETBALLDRIVE, 100:1)	64
그림 25. 수신단에서의 Y-PSNR (BQTERRACE, 100:1)	65
그림 26. 수신단에서의 Y-PSNR (CACTUS, 100:1)	65
그림 27. 수신단에서의 Y-PSNR (KIMONO, 100:1)	66
그림 28. 수신단에서의 Y-PSNR (PARKSCENE, 100:1)	66

제 1 장 서 론

1.1 연구의 배경

High Efficiency Video Coding(HEVC)은 비디오 압축 표준으로써 기존 표준인 H.264/AVC에서 HD 영상을 압축한 크기와 HEVC에서 UHD 영상을 압축했을 때의 크기를 동일하게 하는 것을 목적으로 한다 [1]. 따라서 HEVC에는 목표로 하는 영상의 크기가 커진 만큼 H.264/AVC보다 복잡한 알고리즘이 적용되었다. 먼저 coding의 단위인 block의 크기가 H.264/AVC에서는 16x16 pixel인 것에 비해 HEVC에서는 최대 64x64 pixel까지 지원하고, 이에 맞춰 transform unit 크기도 최대 32x32 pixel까지 지원한다. 또한, motion vector search를 정밀하게 하기 위해 interpolation filter를 quarter-pel 단위까지 지원하며 in-loop filter에는 블록 경계의 계단 현상을 없애는 deblocking filter (DF)와 에러를 보정하는 sample adaptive offset(SAO)기술이 포함되었다.

HEVC에서는 네트워크 전송을 본격적으로 지원하여 NAL 구조를 적용했으며, 비트율 제어를 위한 알고리즘이 적용되었다. 네트워크를 통해 영상을 전송하는 경우에는 출력 비트율을 제한하는 rate control 알고리즘이 중요하다. 해당 알고리즘에는 buffer 제어를 위한 hypothesis model이 적용되었다.

무선통신망을 통해 실시간 스트리밍 서비스를 제공할 때, 급격한 채널 용량 변화에도 영상을 끊김 없이 재생하기 위해 채널 용량에 따라 데

이더 전송률을 조절하는 기술이 필요하다. 기존 rate control 알고리즘은 encoder의 출력 비트율을 평균 데이터 전송률에 맞추지만, 무선 채널에 음영/간섭이 생겨 채널 용량이 급격하게 떨어지는 상황에서는 실시간 대응을 하지 못하는 문제가 있다. 이는 bit을 encoding 전에 할당함에 따라 encoding time만큼의 delay가 발생하기 때문이다.

이를 해결하기 위해 scalable coding 기술이 도입되었다[2]. Scalable coding에서는 encoder에서 미리 비디오 영상의 출력 비트율을 여러 세트로 나누어 coding하여, 채널 용량이 부족한 경우 encoding 후에도 비트율을 조절할 수 있도록 하였다. Scalable coding은 비디오 영상을 layer로 나누는 방식에 따라 spatial, temporal, quality(SNR) scalable coding으로 구분된다. Spatial scalable coding은 영상의 크기에 따라, temporal scalable coding은 영상의 빈도에 따라, quality(SNR) scalable coding은 영상의 화질에 따라 layer를 나눈다. 본 논문에서는 SNR-scalable coding에 대해 다룬다.

SNR-scalable coding은 화질에 따른 layer로 영상을 구분하며 layer가 쌓일수록 비트율이 높아져서 화질이 점차 좋아진다. 하지만 layer를 나눌 때마다 부가적인 bit이 발생하기 때문에 압축 효율의 측면에서는 하나의 layer로 encoding하는 것이 유리하다. 그러나 현재의 채널 정보를 feedback 받을 때 delay가 발생하고, encoding 도중에 채널 용량이 변할 수 있기 때문에 정확한 채널 용량을 알 수 없다. 따라서 encoder에서는 채널 용량 추정을 통해 encoding을 수행해야 하고, 이를 위해 layer를 여러 개로 나누어 전송 데이터 비트율의 이득을 목표로 한다. 본 논문에서는 feedback으로 받은 receiver의 SNR을 이용하여 채널 용량을 추정하고, 이를 통해 encoder에서 사용할 수 있는 total

target rate 계산 후 decoder 성능을 최대로 하는 layer 개수와 각 layer에 할당할 bit을 결정하는 알고리즘을 제안한다.

1.2 관련 연구

Giuseppe Bianchi는 채널 상황에 adaptive한 bit allocation 방법으로 utility-fair 개념을 제안했다[3]. 여기서 utility는 주관적 화질 측정 방법인 mean-opinion score(MOS)로, 임의로 얻은 utility curve를 통해 다른 rate 환경에서도 이를 일반화해 적용하는 방식이다. 이 방식은 채널을 공유하는 개별 어플리케이션에 대역폭을 할당하는 방식이므로, layer간 dependency를 고려해야 하는 SNR-scalable coding에는 적합하지 않다. 하지만 사용자의 관점에서 채널 성능을 극대화하고, 서로 다른 rate의 어플리케이션 성능 곡선을 단일화하는 것은 해당 주제에 대한 이해를 용이하게 한다는 장점이 있다.

Lisimachos P. Kondi는 channel coding과 source coding을 모두 고려하는 방법을 제안했다[4]. 이는 Giuseppe Bianchi가 제안한 방법과 마찬가지로 각 layer의 R-D curve를 임의로 생성한 후, 이를 아우르는 하나의 R-D curve를 만들고, 채널 모델로 얻은 채널 용량을 cost 계산에 사용한다. 하지만 본 논문의 SNR-scalable coding에서는 별도의 R-D 관계를 이용해 각 layer를 구분 했으므로 하나의 정해진 distortion이 아닌, decoding 화질 추정 시 layer가 전송될 확률을 구하고 distortion의 기댓값을 사용한다.

A. Goldsmith의 Wireless communications[5]에서는 시간에 따라 채널이 변하는 것을 Doppler effect를 이용해 모델링 했다. 이 때,

Doppler effect의 Doppler frequency는 수신기가 움직이는 속도에 따라 변하고, 간섭이 변함에 따라 신호의 세기도 변하는 것을 반영한 값이다. J. B. Andersen[6]은 주변 환경이 움직이는 상태에서 Doppler spectrum을 관찰했는데, 송수신기가 움직일 때와 달리 zero frequency가 높아 신호 세기가 많이 변하지 않았다. 하지만 [6]에는 수식적으로 전개한 내용이 없어 실제 실험에는 송수신기가 움직이는 환경에서의 Doppler frequency 분포[7]를 사용했다.

이처럼 채널 용량이 시간에 따라 변하기 때문에 전송 layer가 선택될 확률은 알 수 없다. 따라서 본 논문에서는 이전 채널 용량의 분포를 이용해 이후의 채널 용량을 추정하고 이 때 발생한 추정 에러는 feedback을 받아 분포를 update하는 방법으로 보정했다.

1.3 논문의 구성

2.1에서는 누적 layer rate을 사용하여 채널 용량이 변할 때 enhancement layer에서 HM의 rate control 알고리즘이 제대로 동작하지 않는 현상을 해결하는 방안을 서술하였다. 2.2에서는 본 논문의 무선 채널 기준으로 이용하는 IEEE 802.11ad 프로토콜에 대해 설명하였다. 3장에서는 본 논문의 SNR-scalable coding 알고리즘을 설명하고, 기존 HEVC scalable extension의 reference S/W인 SHVC와 달리 layer를 CTU 단위로 구분하며 달라진 점을 드러내었다. 4장에서는 무선 채널의 용량을 추정하기 위한 방법을 기술하였으며, 5장에서는 이를 이용해 decoding 성능을 최대로 하는 SNR-scalable coding layer 수 결정과 layer별 bit 할당 알고리즘을 설명하였다.

4장은 fading 채널 모델을 이용해 feedback 채널 정보를 추정하는 내용부터 slice, picture 정보를 이용해 picture, k-GOP 단위 채널 용량을 추정하는 내용까지 상술하였다. 5장은 4장의 역순으로 bit allocation과 코딩 layer 수 및 전송 layer 수를 결정하는 알고리즘에 대해 서술하였다.

6장은 4장과 5장 내용에 대한 실험 결과를 포함하고 있고, 마지막 7장에서는 본 논문의 결론을 기술하였다.

제 2 장 Rate control 알고리즘과 IEEE 802.11ad 프로토콜 설명

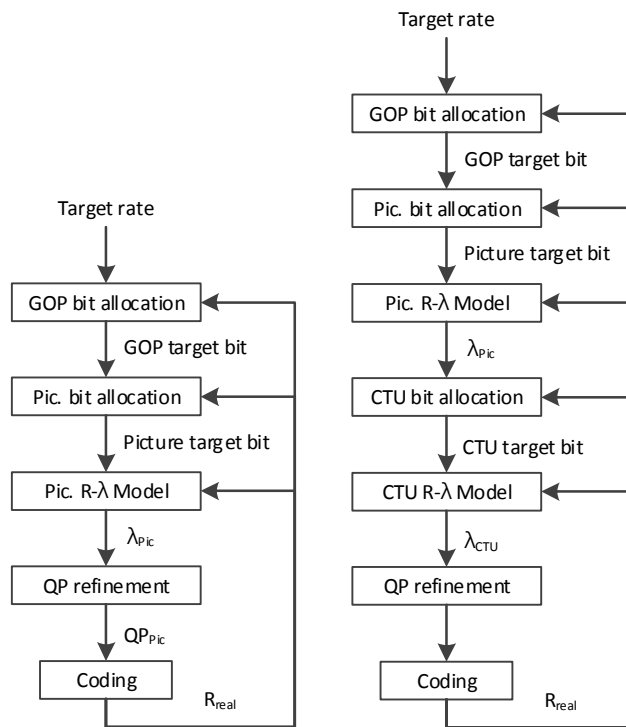
2.1 Rate control 알고리즘

기존 SHVC에는 HM에 적용된 Bin Li의 rate control 알고리즘[9]이 들어가 있다. Scalable coding을 위한 layer 분리 시에는 파라미터 값을 기존 HM과 다르게 사용해야 하지만, 현재 SHVC는 HM의 파라미터 값을 그대로 사용하고 있다.

이와 관련해 Li Li는 첫번째 frame의 초기값을 조절하는 방법을 제안했다[10]. 이는 현재 frame과 다음 frame 복잡도의 비를 이용해 target bpp를 scaling하는 방식으로 구현했기 때문에, 모델 파라미터 값 자체를 변경하는 방법이 아니다. 그러나 다음 frame을 미리 buffering하는 것은 CTU 단위 SNR-scalable coding의 목적에 맞지 않으므로 파라미터 조절은 본 논문에서 제외했다.

2.1.1 기존 HM의 rate control 알고리즘

기존 HM의 rate control 알고리즘은 각 layer에 할당된 bitrate을 해당 layer의 rate model에 사용한다. 다음은 HM의 rate control 알고리즘을 bit allocation, rate model, QP refinement 단계로 순서대로 나타낸 것이다.



(a) GOP, picture 단위 (b) GOP, picture, CTU 단위

그림 1. HM의 rate control 알고리즘

우선, target rate를 입력 받은 후 picture/CTU 단위 bit allocation을 수행한다. 이후 rate model($R-\lambda$ model)에 rate을 대입해 λ 값을 계산하고, QP refinement 과정을 통해 실제 coding에 사용하는

quantization parameter(QP)를 얻는다.

2.1.1.1 Picture / CTU bit allocation

Target rate은 해당 layer의 평균 할당 bitrate을 나타내는 값이며, 우선 GOP 내부의 level에 따라 picture별 bit을 할당한다. FR 은 frame rate, N_{coded} 는 coding한 picture의 개수, R_{coded} 는 coding에 사용된 rate를 의미한다.

$$R_{PicAvg} = R_{tar} / FR \quad (1)$$

$$Target_{GOP} = \frac{R_{PicAvg} \times (N_{coded} + SW) - R_{coded}}{SW} \times N_{GOP} \quad (2)$$

SW는 smoothing window로 HM에서는 실험값인 40을 사용한다.

$$Target_{Pic} = \frac{Target_{GOP} - Coded_{GOP}}{\sum_{\{AllNotCodedPictures\}} \omega_{Pic}} \times \omega_{PicCurr} \quad (3)$$

ω 는 GOP 내부의 level에 따라 달라지는 hierarchical bit allocation을 위한 weight로, low-delay B에서의 값은 다음과 같다.

표 1. Low-delay B에서의 ω_{Pic}

Position	bpp			
	(0, 0.05]	(0.05, 0.1]	(0.1, 0.2]	(0.2, +∞]
f_{4n}	14	12	10	6
f_{4n+2}	3	3	3	3
f_{2n+1}	2	2	2	2

표 1에서 position의 $4n$ 은 GOP size=4에서 GOP 내 첫번째 frame(level 1)을 의미하고, $4n+2$ 는 3번째 frame(level 2), $2n+1$ 은 앞서 언급한 것을 제외한 홀수 번째 frame(level 3)을 의미한다.[9]

Random access에서의 ω 값은 표 2와 같다.

표 2. Random Access에서의 ω_{Pic}

Position	bpp			
	(0, 0.05]	(0.05, 0.1]	(0.1, 0.2]	(0.2, +∞]
f_{8n}	30	25	20	15
f_{8n+4}	8	7	6	5
f_{4n+2}	4	4	4	4
f_{2n+1}	1	1	1	1

CTU level rate control을 수행하는 경우에는 picture level보다 작은 단위인 basic unit bit allocation을 수행한다.

$$Target_{BU} = \frac{Target_{Pic} - Bit_H - Coded_{Pic}}{\sum_{\{AllNotCodedBUs\}} \omega_{BU}} \times \omega_{BUCurr} \quad (4)$$

Bit_H 는 header bit을 의미하며, bit allocation을 하는 순간에는 알 수 없으므로 같은 level의 이전 picture 값을 그대로 사용한다. CTU level bit allocation에서의 ω 값은 mean absolute difference(MAD)에 따라 결정된다. 이 때 MAD는 같은 level의 collocated picture prediction 차의 평균을 의미한다.

$$MAD_{BU} = \frac{1}{N_{pixels}} \sum_{\{AllPixelsInBU\}} |P_{org} - P_{pred}| \quad (5)$$

$$\omega_{BU} = MAD_{BU}^2 \quad (6)$$

2.1.1.2 λ estimation

Picture/CTU 단위 bit allocation이 끝나면 λ estimation을 수행한다. 이 과정에서 rate model을 세워 rate과 λ 의 관계를 이용하는데, HM에서는 Bin Li[9]의 hyperbolic model을 사용한다.

Hyperbolic model은 rate과 distortion의 관계가 hyperbolic이라고 가정한 후, rate과 λ 의 관계식을 도출한다.

$$D(R) = C \cdot R^{-K} \quad (7)$$

$$\lambda = -\frac{\partial D}{\partial R} = CK \cdot R^{-K-1} = \alpha R^\beta \quad (8)$$

$$\lambda = \alpha \cdot bpp^\beta \quad (9)$$

실제 구현에는 rate으로 bit per pixel(bpp)를, distortion으로는 pixel단위 mean square error(MSE)를 이용한다.

λ 는 target bpp로 도출하는데 이는 estimation값이므로 실제 coding 후 파라미터 update가 필요하다.

$$\lambda_{comp} = \alpha_{old} \cdot bpp_{real}^{\beta_{old}} \quad (10)$$

$$\alpha_{new} = \alpha_{old} + \delta_\alpha \cdot (\ln \lambda_{real} - \ln \lambda_{comp}) \cdot \alpha_{old} \quad (11)$$

$$\beta_{new} = \beta_{old} + \delta_\beta \cdot (\ln \lambda_{real} - \ln \lambda_{comp}) \cdot \ln bpp_{real} \quad (12)$$

λ_{real} 은 bit allocation을 통해 나온 bpp_{target} 에 의해 결정된다.

$$\lambda_{real} = \alpha \cdot bpp_{target}^\beta \quad (13)$$

후에, 실제 발생한 bpp_{real} 와 α , β 를 R- λ model에 적용해 λ_{comp}

를 계산하고, 이를 새로운 α , β 를 얻기 위한 목표치로 정한다.

$$\lambda_{comp} = \alpha \cdot bpp_{real}^{\beta} \quad (14)$$

지수 형태의 식을 linear한 관계로 만들기 위해 양 변에 log를 취하면 (15), (16)과 같으며, adaptive least mean square를 적용해 error term을 (17)과 같이 정의했다.

$$\ln \lambda_{real} = \ln \alpha + \beta \cdot \ln bpp_{target} \equiv \alpha' + \beta \cdot \ln bpp_{target} \quad (15)$$

$$\ln \lambda_{comp} = \ln \alpha + \beta \cdot \ln bpp_{real} \equiv \alpha' + \beta \cdot \ln bpp_{real} \quad (16)$$

$$e^2 = (\ln \lambda_{real} - \ln \lambda_{comp})^2 \quad (17)$$

$$\frac{\partial e^2}{\partial \alpha'} = \frac{\partial e^2}{\partial \ln \lambda_{comp}} \cdot \frac{\partial \ln \lambda_{comp}}{\partial \alpha'} = -2(\ln \lambda_{real} - \ln \lambda_{comp}) \quad (18)$$

$$\frac{\partial e^2}{\partial \beta} = \frac{\partial e^2}{\partial \ln \lambda_{comp}} \cdot \frac{\partial \ln \lambda_{comp}}{\partial \beta} = -2(\ln \lambda_{real} - \ln \lambda_{comp}) \cdot \ln bpp_{real} \quad (19)$$

Parameter α 와 β 의 update 식은 다음과 같다. ($\alpha' = \ln \alpha$)

$$\begin{aligned} \alpha'_{new} &= \alpha'_{old} - \delta \cdot (-2(\ln \lambda_{real} - \ln \lambda_{comp})) \\ &= \alpha'_{old} + 2\delta \cdot (\ln \lambda_{real} - \ln \lambda_{comp}) \end{aligned} \quad (20)$$

지수 형태의 식을 linear 관계로 만들기 위해 log를 취하면 다음과 같다.

$$\ln \alpha_{new} = \ln \alpha_{old} + 2\delta \cdot (\ln \lambda_{real} - \ln \lambda_{comp}) \quad (21)$$

$$\begin{aligned} \alpha_{new} &= \alpha_{old} \cdot \exp(2\delta \cdot (\ln \lambda_{real} - \ln \lambda_{comp})) \\ &\approx \alpha_{old} \cdot (1 + 2\delta \cdot (\ln \lambda_{real} - \ln \lambda_{comp})) \end{aligned} \quad (22)$$

$$\alpha_{new} = \alpha_{old} + \delta_{\alpha} \cdot (\ln \lambda_{real} - \ln \lambda_{comp}) \cdot \alpha_{old} \quad (23)$$

$$\begin{aligned} \beta_{new} &= \beta_{old} - \delta \left(-2 (\ln \lambda_{real} - \ln \lambda_{comp}) \cdot \ln bpp \right) \\ &= \beta_{old} + 2\delta \cdot (\ln \lambda_{real} - \ln \lambda_{comp}) \cdot \ln bpp_{real} \end{aligned} \quad (24)$$

$$\beta_{new} = \beta_{old} + \delta_{\beta} \cdot (\ln \lambda_{real} - \ln \lambda_{comp}) \cdot \ln bpp_{real} \quad (25)$$

이 때 bpp가 클수록 parameter adaptation factor(δ)도 함께 증가한다.

표 3. Bpp와 δ 의 관계

Bpp range	δ_{α}	δ_{β}
$0 < bpp_{target} < 0.03$	0.01	0.005
$0.03 \leq bpp_{target} < 0.08$	0.05	0.025
$0.08 \leq bpp_{target} < 0.2$	0.1	0.025
$0.2 \leq bpp_{target} < 0.5$	0.2	0.1
$0.5 \leq bpp_{target}$	0.4	0.2

2.1.1.3 QP refinement

마지막으로 estimation된 λ 에 따라 quantization parameter(QP)를 결정하기 위해, multiple QP coding 방식으로 정해진 범위 내에서 search하는 방식과 Bin Li[11]의 관계식을 이용한다. HM에서는 다음 수식을 사용하며 [11], 본 논문에서도 이를 이용해 실험을 진행했다.

$$QP = 4.2005 \cdot \ln \lambda + 13.71222 \quad (26)$$

2.1.2 제안하는 rate control 알고리즘

2.1.2.1 제안하는 picture bit allocation

본 논문에서 사용한 블록단위 SNR-scalable coding은 채널 용량의 변화에 따라 target bitrate을 변화시킨다. 하지만 기존 HM의 rate control 알고리즘은 평균 target bitrate을 입력 받아 picture단위로 bit을 할당하고 있으므로 수정이 필요하다. 본 논문에서는 feedback 받은 채널 용량에 맞춘 target bitrate를 이용해 매 picture마다 bit allocation을 수행한다.

2.1.2.2 제안하는 누적 rate model

2.1.1의 기존 HM rate control 알고리즘을 SNR-scalable coding에 그대로 적용하면 rate model에 문제가 발생한다. 기존 HM rate control 알고리즘의 R-D curve는 base layer에서는 hyperbolic 관계를 보이지만, enhancement layer에서는 hyperbolic 관계를 이루지 않았다. 다음은 이에 대한 예시로, BasketballDrive와 Cactus에서 R-D curve를 layer별로 나타낸 그래프이다.

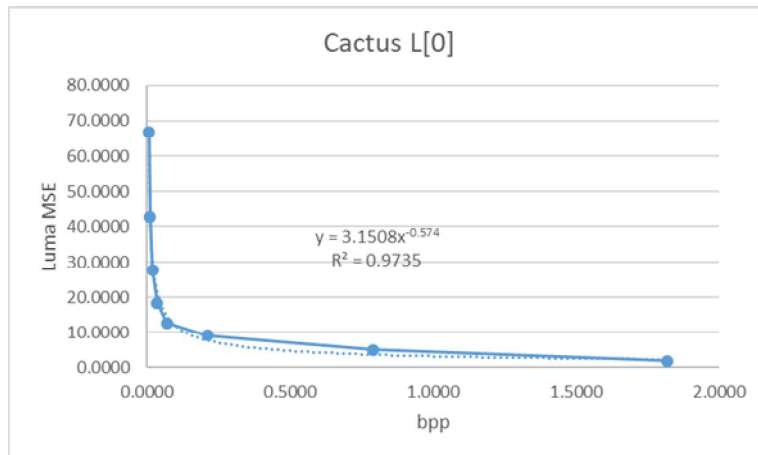
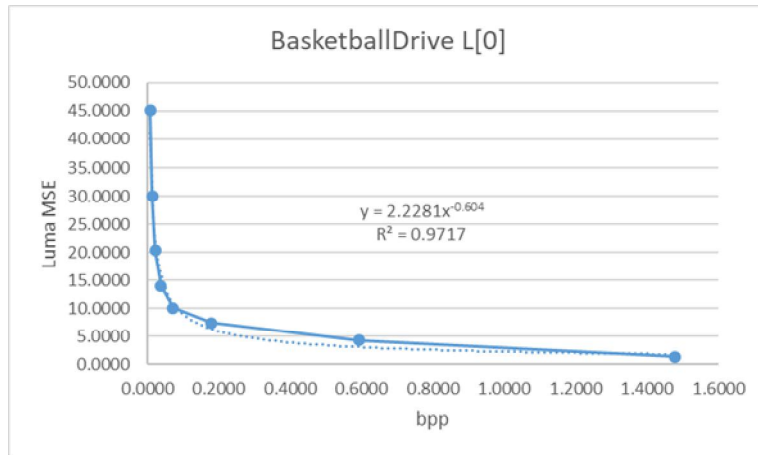


그림 2. Base layer에서의 R-D curve (bpp-MSE)

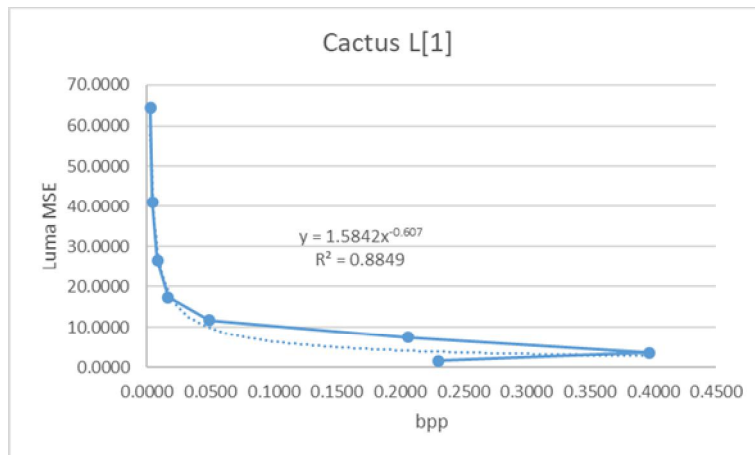
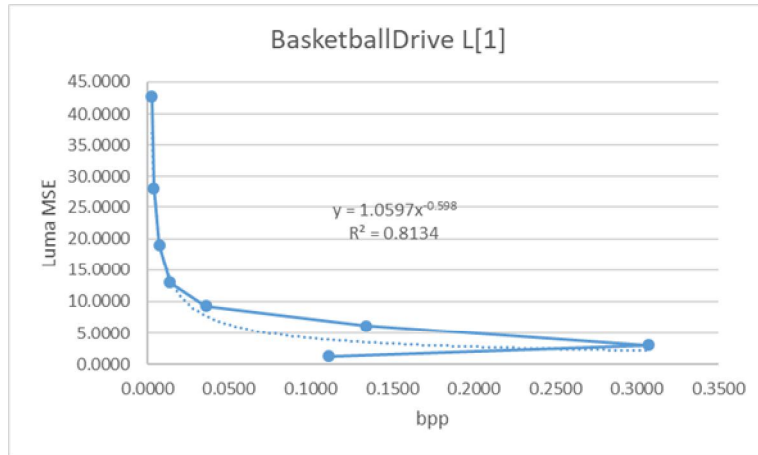


그림 3. Enhancement layer에서의 R-D curve (bpp-MSE)

Base layer는 hyperbolic 관계가 잘 맞지만, enhancement layer에서는 hyperbolic 관계가 잘 맞지 않음을 확인할 수 있다. 작은 QP 값에서 rate이 낮게 나온 것은 base layer의 QP가 낮을수록 inter-layer prediction의 효율이 높아지기 때문이다.

다음은 base layer와 enhancement layer의 누적 rate을 가로 축으로 한 그래프로, 이는 enhancement layer만으로 R-D curve를 그린 것 보다는 hyperbolic 관계에 근접하다.

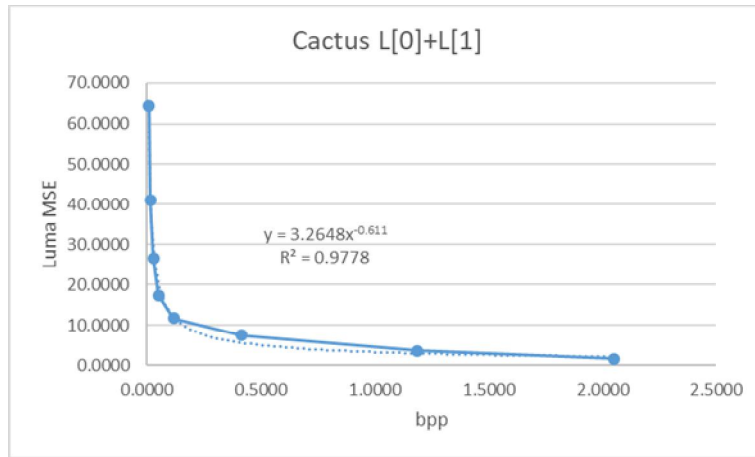
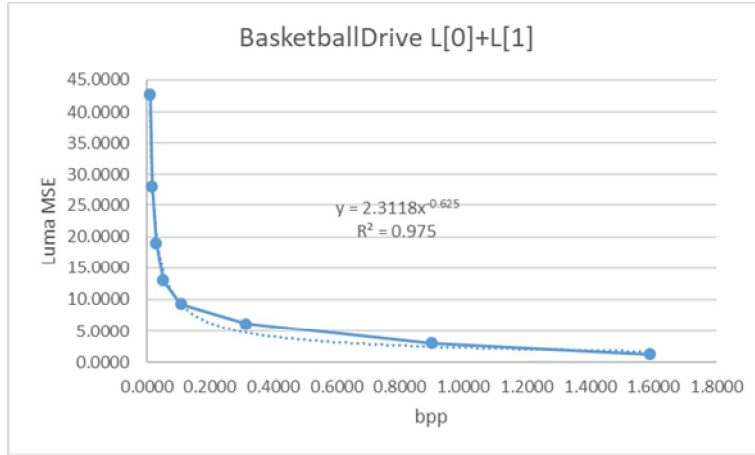


그림 4. 누적 enhancement layer에서의 R-D curve (bpp-MSE)

따라서 enhancement layer의 rate model은 하위 layer의 rate을 모두 누적인 값을 target rate으로 하여 λ 를 추정하는 것이 효율적이다. 기존 HM 알고리즘의 λ estimation 수식은 다음과 같다.

$$\lambda_{base.est} = \alpha_{base} \cdot bpp_{base}^{\beta_{base}} \quad (27)$$

$$\lambda_{enhance.est} = \alpha_{enhance} \cdot bpp_{enhance}^{\beta_{enhance}} \quad (28)$$

다음은 SNR-scalable coding을 위한 rate을 누적 rate으로 수정한 λ estimation 수식이다.

$$\lambda_{base.est} = \alpha_{bas} \cdot bpp_{base}^{\beta_{base}} \quad (29)$$

$$\lambda_{enhance.est} = \alpha_{enhance} \cdot (bpp_{base} + bpp_{enhance})^{\beta_{enhance}} \quad (30)$$

누적 rate를 사용하도록 수정한 rate model의 layer별 BD-rate은 다음과 같다. Anchor는 기존 HM의 rate model을 이용하였다.

표 4. Base layer(L0)에서의 BD-rate

Seq.	압축률	Anchor		Acc. Rate model		BD-rate
		Kbps	Ypsnr	kbps	Ypsnr	
Kimono	50:1	2986.00	39.46	2986.01	39.46	-0.49%
	100:1	1493.03	36.69	1493.00	36.72	
	150:1	995.36	35.12	995.36	35.14	
	200:1	746.52	33.98	746.53	34.00	
ParkScene	50:1	2985.76	36.56	2985.76	36.56	0.13%
	100:1	1492.84	34.29	1492.84	34.28	
	150:1	995.18	33.06	995.14	33.05	
	200:1	746.37	32.18	746.34	32.17	
Cactus	50:1	6220.23	36.71	6220.14	36.70	0.17%
	100:1	3110.36	35.01	3110.35	35.01	
	150:1	2073.62	33.79	2073.62	33.78	
	200:1	1555.23	32.90	1555.21	32.90	
BasketballDrive	50:1	6220.80	37.10	6220.80	37.10	-0.17%
	100:1	3110.41	35.21	3110.41	35.21	
	150:1	2073.60	33.94	2073.59	33.94	
	200:1	1555.20	32.95	1555.20	32.96	
BQTerrace	50:1	7465.01	35.20	7464.98	35.19	0.41%
	100:1	3732.51	34.40	3732.61	34.39	
	150:1	2488.36	33.74	2488.38	33.74	
	200:1	1866.29	33.22	1866.29	33.21	

표 5. Enhancement layer (L1)에서의 BD-rate

Seq.	압축률	Anchor		Acc. Rate model		BD-rate
		Kbps	Ypsnr	kbps	Ypsnr	
Kimono	50:1	5972.00	40.77	5975.98	40.75	0.78%
	100:1	2986.03	38.31	3011.68	38.30	
	150:1	1990.71	36.63	1992.33	36.62	
	200:1	1493.02	35.44	1495.24	35.45	
ParkScene	50:1	5971.46	37.62	5973.64	37.68	-2.35%
	100:1	2985.79	35.27	3001.28	35.37	
	150:1	1990.17	33.93	1991.22	34.02	
	200:1	1492.76	33.04	1493.39	33.09	
Cactus	50:1	12440.88	37.40	12456.79	37.44	-1.85%
	100:1	6220.76	35.88	6241.12	35.91	
	150:1	4146.95	34.66	4151.22	34.76	
	200:1	3110.44	33.78	3115.64	33.89	
BasketballDrive	50:1	12441.59	37.95	12503.89	38.03	-2.67%
	100:1	6220.76	36.24	6272.46	36.33	
	150:1	4147.04	34.99	4207.77	35.13	
	200:1	3110.31	34.04	3153.40	34.19	
BQTerrace	50:1	14930.04	35.68	14984.27	35.74	-5.47%
	100:1	7465.14	34.85	7484.59	34.93	
	150:1	4976.68	34.25	4980.45	34.34	
	200:1	3732.59	33.78	3735.51	33.86	

표 6. Enhancement layer (L2)에서의 BD-rate

Seq.	압축률	Anchor		Acc. Rate model		BD-rate
		Kbps	Ypsnr	kbps	Ypsnr	
Kimono	50:1	8958.04	41.41	9188.42	41.43	3.71%
	100:1	4479.05	39.22	4686.69	39.21	
	150:1	2986.08	37.62	3015.16	37.54	
	200:1	2239.52	36.42	2283.01	36.36	
ParkScene	50:1	8957.28	38.35	9170.68	38.40	1.20%
	100:1	4478.72	36.02	4604.93	36.04	
	150:1	2984.95	34.59	3040.96	34.69	
	200:1	2238.96	33.67	2288.20	33.73	
Cactus	50:1	18661.66	37.82	18873.06	37.85	-0.91%
	100:1	9331.10	36.37	9465.92	36.40	
	150:1	6220.53	35.21	6303.68	35.31	
	200:1	4665.63	34.32	4735.11	34.46	
BasketballDrive	50:1	18662.34	38.49	18939.63	38.56	-1.49%
	100:1	9331.15	36.85	9503.61	36.92	
	150:1	6220.61	35.66	6373.68	35.80	
	200:1	4665.29	34.72	4784.37	34.87	
BQTerrace	50:1	22395.07	36.08	22800.92	36.12	-3.04%
	100:1	11197.67	35.16	11382.07	35.22	
	150:1	7465.15	34.56	7546.39	34.64	
	200:1	5598.95	34.10	5660.58	34.19	

표 7. Base layer (BL)에서의 BD-rate

Seq.	압축률	Anchor		Acc. Rate model		BD-rate
		Kbps	Ypsnr	kbits	Ypsnr	
Kimono	50:1	11944.10	41.85	12460.17	41.90	5.44%
	100:1	5972.09	39.85	6529.85	39.93	
	150:1	3980.20	38.31	4095.12	38.23	
	200:1	2984.69	37.14	3109.00	37.08	
ParkScene	50:1	11943.17	38.92	12479.52	38.95	4.36%
	100:1	5971.53	36.58	6308.69	36.57	
	150:1	3979.72	35.12	4127.00	35.18	
	200:1	2984.61	34.16	3108.22	34.21	
Cactus	50:1	24882.46	38.16	25417.60	38.17	0.91%
	100:1	12441.49	36.74	12804.59	36.76	
	150:1	8294.07	35.62	8473.78	35.70	
	200:1	6220.83	34.73	6365.59	34.86	
BasketballDrive	50:1	24883.08	38.90	25455.14	38.97	-0.15%
	100:1	12441.52	37.29	12780.98	37.36	
	150:1	8294.05	36.15	8565.21	36.25	
	200:1	6220.09	35.23	6430.67	35.37	
BQTerrace	50:1	29859.90	36.45	31149.53	36.49	-0.50%
	100:1	14929.88	35.40	15755.18	35.48	
	150:1	9953.57	34.78	10134.52	34.85	
	200:1	7465.19	34.34	7603.97	34.41	

Base layer(L0)에서는 기존 HM과 누적 rate을 이용한 rate model의 코딩 효율에 큰 차이가 없었지만, enhancement layer(L1)에서는 코딩 효율이 높아지는 것을 확인하였다. 반면, Kimono와 ParkScene에서는 L2에서 코딩 효율이 떨어지기 시작해 L3에서는 코딩 효율 저하가 증가했다. 이는 누적된 rate을 기반으로 rate model을 만들었기 때문에 rate adaptation과 직접 관련 있는 rate과 RDO에서 계산하는 rate에 차이가 있기 때문이다.

본 논문은 매 picture마다 target bit이 변하는 상황을 가정 하였으므로 target bit과 real bit의 일치 정도가 중요하다. 따라서 정교한 rate control을 위해 기존의 단일 rate model을 그대로 사용하였다. 하지만, 5장에서 다룬 distortion model에서는 hyperbolic 관계를 이용하기 위해 누적 rate model을 적용했다.

2.2 IEEE 802.11ad 프로토콜 설명

기존 Wi-Fi 프로토콜이 사용하던 2.4/5 GHz의 wireless band가 포화상태이고 수 giga-bit data 전송을 하기에 부적합하여 새로운 프로토콜이 필요했다. 802.11ad 프로토콜[8]은 60 GHz band를 사용하는데, 이는 range가 짧아 간섭이 적고 수 giga-bit data 전송에 적합하다.

다음은 802.11ad 프로토콜에서 PHY mode에 따라 달라지는 packet overview를 나타낸 것이다.

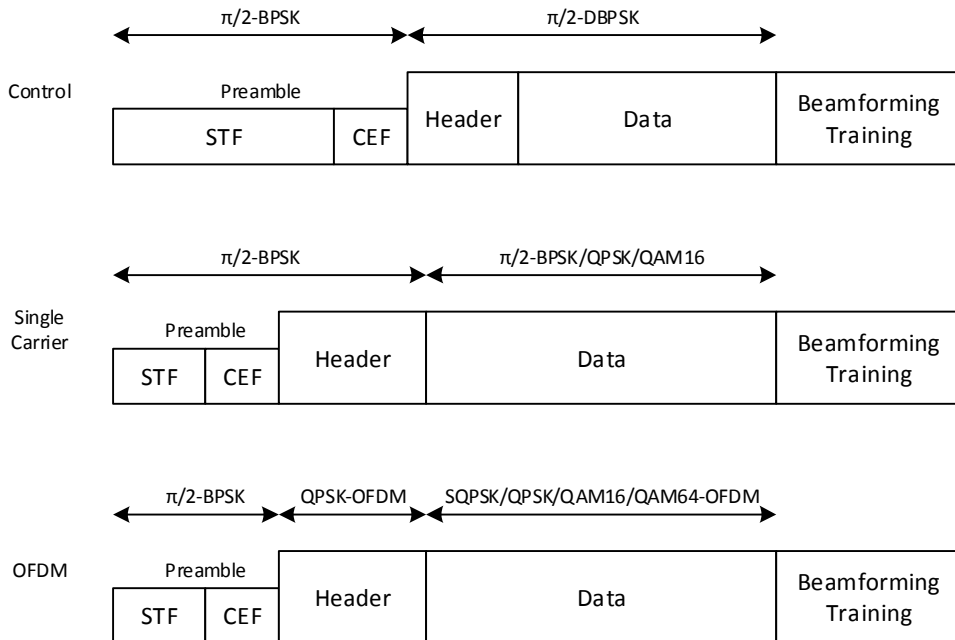


그림 5. PHY modes (packet overview)

2.2.1 Preamble

Preamble의 short training field(STF)는 timing estimation과 automatic gain adjustment를 위한 용도이며, channel estimation field(CEF)는 channel impulse response를 추정하기 위해 사용한다.

STF는 짧은 symbol을 10번 보내 signal detect 및 diversity selection을 수행하고, frequency offset값을 추정하여 timing을 동기화하는 역할을 한다. CEF는 긴 symbol을 2번 보내 채널과 frequency offset값을 추정한다.

2.2.2 Header / Data

그림 5에서 header는 modulation과 channel coding방식에 대한 파라미터를 포함하며, data는 비디오 영상이 담긴 payload 역할을 한다. 표 8은 802.11ad에서 사용하는 modulation과 coding schemes이다.

표 8. Modulation and coding schemes(MCS)

	MCS	Modulation	Repetition	Code rate	Data rate[Mbps]
Control (CPHY)	0	DBPSK	1	1/2	27.5
Single Carrier (SCPHY)	1	$\pi/2$ -BPSK	2	1/2	385
	2	$\pi/2$ -BPSK	1	1/2	770
	3	$\pi/2$ -BPSK	1	5/8	962.5
	4	$\pi/2$ -BPSK	1	3/4	1155
	5	$\pi/2$ -BPSK	1	13/16	1251.25
	6	$\pi/2$ -BPSK	1	1/2	1540
	7	$\pi/2$ -BPSK	1	5/8	1925
	8	$\pi/2$ -BPSK	1	3/4	2310
	9	$\pi/2$ -BPSK	1	13/16	2502.5
	10	$\pi/2$ -16QAM	1	1/2	3080
	11	$\pi/2$ -16QAM	1	5/8	3850
	12	$\pi/2$ -16QAM	1	3/4	4620
Orthogonal Frequency Division Multiplex (OFDMPHY)	13	SQPSK	1	1/2	693.00
	14	SQPSK	1	5/8	866.25
	15	QPSK	1	1/2	1386.00
	16	QPSK	1	5/8	1732.50
	17	QPSK	1	3/4	2079.00
	18	16-QAM	1	1/2	2772.00
	19	16-QAM	1	5/8	3465.00
	20	16-QAM	1	3/4	4158.00
	21	16-QAM	1	13/16	4504.50
	22	64-QAM	1	5/8	5197.50
	23	64-QAM	1	3/4	6237.00
	24	64-QAM	1	13/16	6756.75
Low-Power Single Carrier (LPSCPHY)	25	$\pi/2$ -BPSK	1	13/28	626
	26	$\pi/2$ -BPSK	1	13/21	834
	27	$\pi/2$ -BPSK	1	52/63	1112
	28	$\pi/2$ -QPSK	1	13/28	1251
	29	$\pi/2$ -QPSK	1	13/21	1668
	30	$\pi/2$ -QPSK	1	52/63	2224
	31	$\pi/2$ -QPSK	1	13/14	2503

표 8의 modulation은 digital modulation을 의미하고 수 bit의 symbol을 analog signal pattern으로 변환시키는 역할을 한다. Modulation은 pattern으로 이용할 신호의 특성에 따라 amplitude modulation(AM), frequency modulation(FM), phase modulation(PM)

의 3가지로 나누어진다. 802.11ad에서 적용한 modulation은 PM을 기반으로 AM을 혼합한 방식이다. 먼저, 가장 간단한 Binary phase-shift keying(BPSK) 방식은 carrier signal을 1개 사용한다. Phase로 나타낼 수 있는 패턴은 위상 π 차이로 구분하여 digital 0과 1(symbol bit-length 1)로 나타낸다.

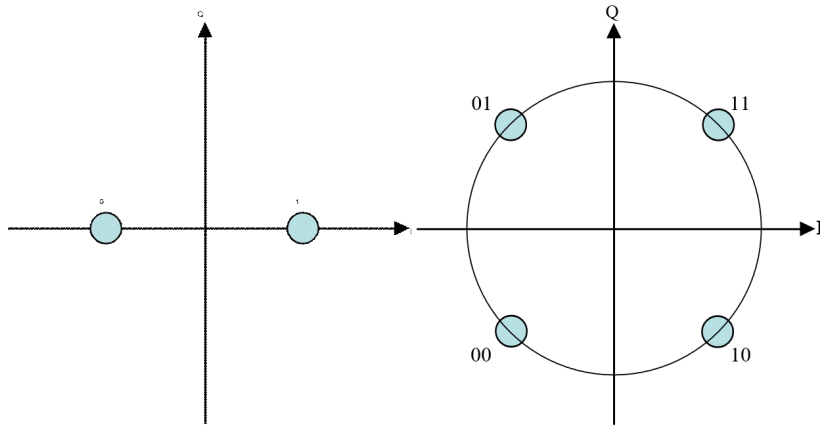


그림 6. Constellation diagram BPSK/QPSK

Phase로 표현할 수 있는 pattern이 4개이면 quadrature phase-shift keying(QPSK)이며, symbol bit length는 2가 된다. 이 때 amplitude로 나타낼 수 있는 pattern을 추가할 수 있다.

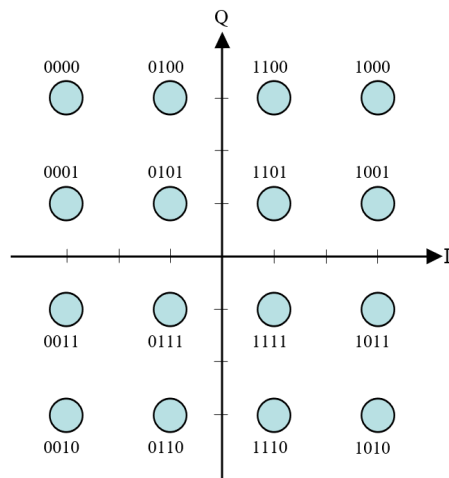


그림 7. Rectangular 16-QAM constellation diagram

Modulation이 나타낼 수 있는 pattern의 개수가 증가할수록 한번에 전송하는 symbol bit length가 증가하므로, data bitrate도 증가하게 된다. 하지만 noise나 shadowing으로 인해 signal에서 pattern을 분리할 수 없는 확률도 늘어나기 때문에, 통신의 신뢰성을 위해 채널 용량에 따라 modulation 적용 방식을 제한해야 한다.

2.2.3 Low Density Parity Check(LDPC)

Low density parity check(LDPC)는 error detection과 correction을 위해 parity check 방식을 이용하며, protection 수준에 따라 coding rate(r)이 달라진다. Coding rate(r)은 codeword의 길이(c) 중 parity bit을 제외한 message bit(m)의 비율을 나타낸 값이다.

$$r = \frac{m}{c} \quad (31)$$

Raw Bit Rate은 channel coding 후 parity bit이 포함된 bitrate을 나타낸 것으로, data bitrate을 계산하기 위해서는 coding rate을 곱하면 된다.

802.11ad 프로토콜의 maximum bandwidth는 2.16 GHz로 8K 영상을 10:1의 압축률로 압축할 때, 모든 채널 용량을 사용한다고 가정한다. 하지만 이는 매우 큰 값으로, 본 논문에서는 실제로 실험에 사용할 1080p 영상은 모든 채널 대역폭을 사용하지는 않는다고 전제했다.

제 3 장 SNR-scalable coding 알고리즘 설명

본 논문에서 사용한 SNR-scalable coding은 H/W 구현을 위해 CTU 단위로 layer를 구분하고, 상위 layer에서는 base layer의 구조를 이용해 CU partition search의 복잡도를 제거했다. In-loop filter 기술은 CTU마다 적용할 수 없기 때문에 최상위 layer에서만 적용하도록 제한하였다.

Base layer에는 in-loop filter 기술을 제외한 기존 HEVC 알고리즘을 그대로 적용했으며, enhancement layer에는 motion estimation 시 하위 layer의 integer motion vector 값을 받은 후 fractional motion estimation만 수행한다. 또한, base layer에서 intra-frame prediction mode가 사용되었다면 enhancement layer에는 inter-layer prediction만 적용하도록 하였다.

3.1 블록(CTU) 단위 동작

Encoder에서 layer는 CTU 단위로 구분되며, 다음 그림은 한CTU에서의 encoding 순서를 나타낸 것이다.

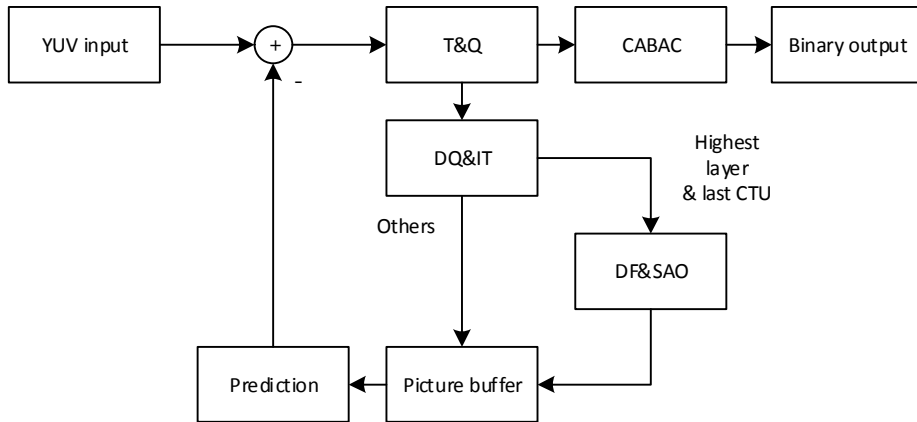


그림 8. 블록 단위로 동작하는 SNR-scalable encoder 구조

기존 HM 구조에서는 in-loop filter에 해당하는 DF와 SAO 단계를 picture buffer 전에 수행하지만, 제안한 구조에서는 최상위 layer에서만 수행한다.

3.2 Prediction 단계에 적용된 fast 알고리즘

Prediction의 경우, layer에 따라 적용하는 mode가 구분되며 다음은 이를 나타낸 그림이다.

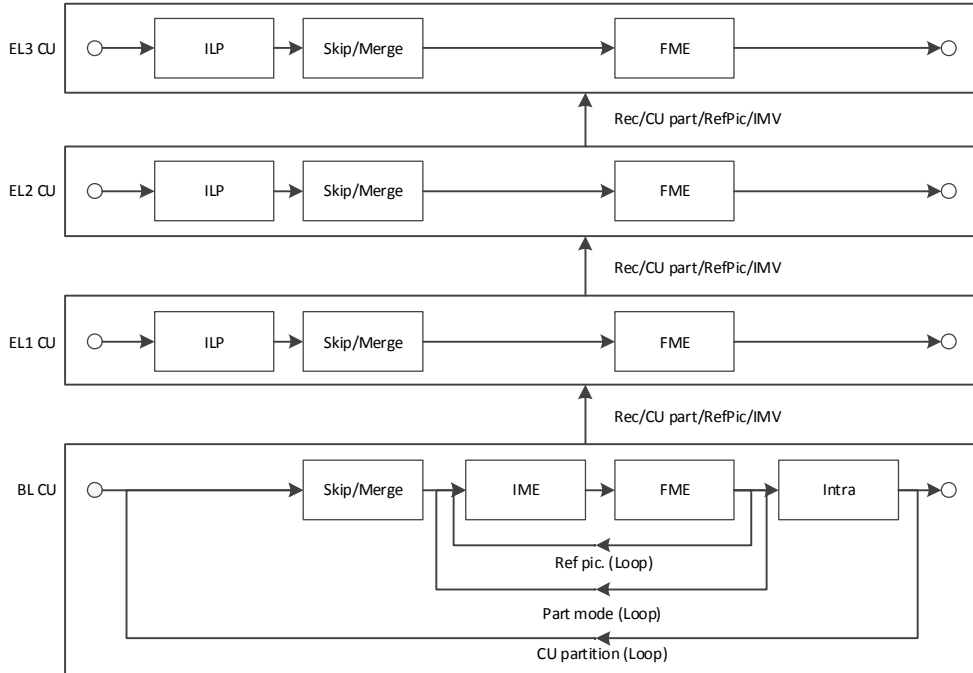


그림 9. Fast 알고리즘이 적용된 prediction 단계

Base layer(BL)에서 결정된 CU/PU 정보는 상위 layer에서 공유되므로 partition mode와 CU partition 정보 등은 BL CU에서만 계산한다. 마찬가지로 reference picture, motion vector 정보 또한 BL에서 계산해 상위 layer와 공유한다. Motion vector의 경우, integer-pel 단위 정보는 BL에서 계산한 값을 사용하며 enhancement layer(EL)에서는 fractional motion estimation만 수행한다. Inter-layer prediction은 CU 단위로 수행하므로 skip/merge 단계 전에 수행한다. 반대로 intra prediction은 base layer에서만 수행하고 enhancement layer에서는 이 과정을 inter-layer prediction으로 대체한다.

3.3 Layer selector를 위한 bitstream에서의 layer

구분

Layer는 CTU 단위로 encoding되기 때문에 bit 출력 또한 CTU 단위로 수행된다. 하지만 이 경우에는 layer 개수 조절 시 CTU마다 layer를 구분하는 scheme이나 bit address를 저장하는 단계가 필요하므로 bitstream에서는 slice 단위로 layer를 구분한다.

그림 10의 왼쪽 그림에서는 CTU 단위로 layer를 인코딩하며, 오른쪽 그림은 순서를 바꾸어 slice 단위로 layer를 인코딩하는 것을 나타내었다.

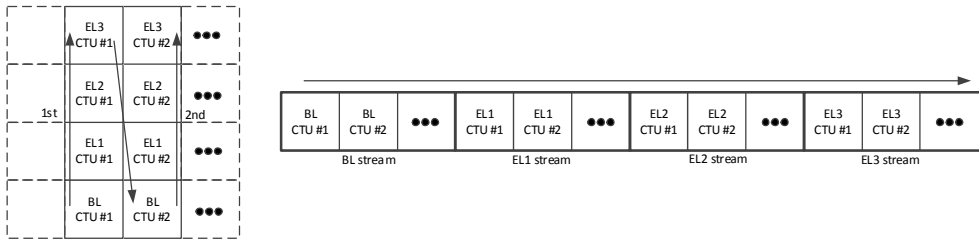


그림 10. Bitstream에서의 layer 구분 (slice 단위)

다음은 채널 용량을 입력 받아 데이터 전송율을 해당 채널 용량 이하로 줄이는 selector의 위치를 나타낸 것이다.

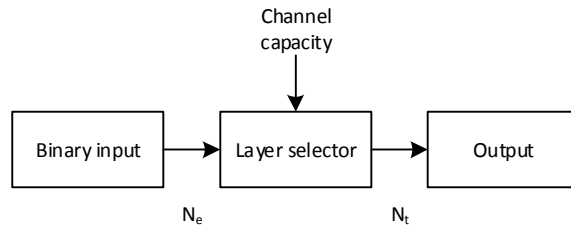


그림 11. Layer 개수 selector

그림 8의 binary output은 encoding 시작 시 입력 받은 채널 용량을 기준으로 선택한 데이터 전송율을 나타낸다. Encoding delay로 인해 채널 용량이 변하는 경우, 상위 layer를 제거해 데이터 전송율을 현재 채널 용량 이하로 맞춰야 한다. 따라서 encoder에서 생성한 layer 개수 N_e 와 전송하는 layer 개수 N_l 는 다음과 같은 관계를 가진다.

$$N_e \geq N_l \quad (32)$$

Layer 개수를 선택하기 전에 전송 채널 용량을 추정하는 단계는 4장에서 채널 용량 모델을 통해 설명한다.

3.4 Decoding 성능을 높이기 위한 error concealment 알고리즘

채널 용량에 맞게 전송 layer 개수를 선택하여도 실제 무선 채널의 신호잡음이나 음영으로 인해 error가 발생할 수 있다. 채널 코딩 단계에 low density parity check(LDPC)과 같은 error correction code가 삽입되어 있어 일정 bit error rate ($<1.0E-6$)은 correction 하지만, correction에 실패하면 bit error는 data bit을 오염시킨다. 본 논문의 블록단위 SNR-scalable coding의 decoder에는 Choi[12]의 알고리즘을 기반으로 개선한 error concealment 알고리즘이 적용되어 있다.

기존의 error concealment 알고리즘은 현재 block이 error로 인해 손상된 경우, 이전 frame의 동일 위치에 있는 block(collocated block)의 motion vector를 이용해 prediction을 수행한다. 본 논문에 적용된 error concealment 알고리즘은 collocated block의 motion vector를

refine해 사용한다. Motion vector refinement 단계에서는 quarter-pel 단위로 21x21 search range를 full search해 boundary sum of absolute difference(SAD)가 가장 낮은 motion vector를 선택한다. 이때, boundary SAD는 outer block matching algorithm(OBMA)을 사용했다.

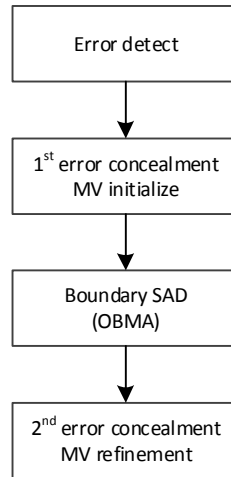


그림 12. Error concealment 알고리즘

Error concealment는 두 stage로 구성되어 있다. 2nd stage에서는 motion vector refinement를 수행하며, 1st stage에서는 이 때 필요한 reconstructed block을 생성한다. Error concealment 수행 시 layer 개수와 layer별 rate 할당을 최적화하기 위해 decoding distortion을 cost로 사용한다.

다음은 1080p 영상을 100:1로 압축한 후, POC 2~4에서 한 slice씩 error concealment를 수행했을 때 해당 slice의 PSNR 평균과 정상 decoding PSNR 평균의 차이를 나타낸 것이다.

표 9. Error concealment 알고리즘의 평균 성능

Sequence	Normal base layer[dB]	Error concealment[dB]	Delta
Kimono	40.44	29.78	-10.66
ParkScene	39.12	34.24	-4.88
Cactus	36.20	32.27	-3.93
BasketballDrive	38.73	26.71	-12.02
BQTerrace	35.00	31.86	-3.14

표 9에 따르면 sequence마다 error concealment의 성능 차이가 크므로 이를 R-D model에 바로 적용하지 않고, error가 발생한 slice의 distortion 값을 error concealment를 적용한 distortion으로 바꾸어 적용한다.

$$D_{dec.est} = A_{normal} \cdot D_{dec} + A_{EC} \cdot D_{dec.EC}, A_{normal} + A_{EC} = 1 \quad (33)$$

제 4 장 Wireless 채널 용량 모델

4장은 Wireless 채널 용량의 estimation과 관련된 내용을 다룬다. 4장의 구성은 다음과 같다. 4.1에서는 Fading 채널 모델을 이용한 feedback 채널 정보의 추정에 관한 내용을 다룬다. 4.2는 4.1에서 논한 feedback 채널 정보 추정으로 얻은 slice 단위 정보를 이용하여 picture 단위 채널 용량을 추정하는 내용을 서술한다. 4.3에서는 k-GOP 단위 채널 용량을 추정하는 방법을 서술한다. 이때, 각 채널 용량 추정은 전 장에서 추정된 단위의 채널 용량 정보를 이용한다. 4.4에서는 SNR과 packet error rate의 관계에 대해 서술한다.

4.1 Fading 채널 모델을 이용한 feedback 채널 정보의 추정

Fading 채널은 채널 용량이 단시간에 감소하는 채널을 의미하며, line-of-sight(LOS)의 비중에 따라 Rayleigh fading 채널과 Ricean fading 채널로 나뉜다. Rayleigh fading 채널은 일반적으로 빌딩 숲 사이에서 전파를 수신하는 상황을 모델링하는데 사용하고, Ricean fading 채널은 실내에서 전파를 수신하는 상황의 모델링에 이용한다. 이때, Rayleigh fading 채널은 LOS값으로 0을 이용하며, Ricean fading 채널은 일반 채널보다 큰 LOS값을 사용한다. 본 논문은 실내 상황을 가정하였기 때문에 Ricean fading 채널 모델을 이용하였고, 이와 관련된 실험은 6.1.1에서 서술한다.

Ricean fading 채널의 모델링은 R. J. Punnoose[7]의 방법을 사용했다. 본 논문에서는 Gans 모델을 이용해 Doppler filter를 만들고, 모든 frequency 성분에 Gaussian random number를 곱한 후, inverse Fourier transform을 거쳐 임의성을 가진 $SNR(t)$ 를 얻었다.

$$SNR(t) = f(\overline{SNR}, K(t), f_D(t)) \quad (34)$$

식(34)의 각 파라미터인 \overline{SNR} 과 $K(t)$, $f_D(t)$ 에 대한 설명은 다음과 같다. \overline{SNR} 은 시간에 따라 변하는 $SNR(t)$ 의 평균을 의미하며, 자세한 설명은 본 절 뒷부분에서 다루도록 한다. $K(t)$ 는 Ricean factor를 의미하며 실제 방에서 측정한 결과를 정리한 논문[13]을 참고하였다.

$f_D(t)$ 는 Doppler frequency shift를 의미하며 다음 식을 이용하여 계산한다. v 는 수신단이 이동하는 속도를 의미하며, c 는 광속이다. 본 논문에서 가정한 802.11ad는 carrier frequency f_c 로 60GHz를 이용한다.

$$f_D = \frac{v}{c} f_c \quad (35)$$

앞서 언급한 \overline{SNR} 의 계산식은 송수신단 사이의 거리에 따른 power의 변화를 정의하는 two-ray path loss 모델식 식(36)으로부터 유도한다.

$$P_r = P_t \frac{G h_t^2 h_r^2}{d^4} \quad (36)$$

식(36)에서 P_r , P_t 는 각각 수신단과 송신단의 power를 나타내며, G 는 antenna gain, h_t , h_r 은 각각 수신단과 송신단의 antenna height를 나타낸다. 본 논문에서는 송수신단 사이 거리가 2.5m에서 9.303dB를 SNR값으로 갖는 Wi-Fi를 가정한다. 식(36)을 데시벨 단위로 변환한 후 정리하여 path loss를 계산하면 식(37)과 같다. 이 식을 수신단의 power로 정리하면 식(38)과 같이 나타낼 수 있다.

$$PL = P_{t,dB} - P_{r,dB} = 40 \log_{10}(d) - 10 \log_{10}(G h_t^2 h_r^2) \quad (37)$$

$$P_{r,dB} = P_{t,dB} + 10 \log_{10}(G h_t^2 h_r^2) - 40 \log_{10}(d) \quad (38)$$

식(38)에서 $P_{t,dB} + 10 \log_{10}(G h_t^2 h_r^2)$ 는 송신단의 power와 antenna gain, 그리고 송수신단의 antenna height가 일정하다 가정하고 고정 상수 값을 이용한다. 앞서 송수신단 간격이 2.5m에서 수신단의 SNR값이

9.303dB라 가정하였으므로 이 값을 대입하면 해당 고정 상수 부분은 25.2206dB가 된다. 따라서 식(38)은 다음과 같이 나타낼 수 있다.

$$P_{r,dB} = 22.2206 - 40\log_{10}(d) \quad (39)$$

본 논문에서는 MATLAB을 이용해 채널 용량의 simulation을 하였고, 0.1초 단위로 SNR(t)를 생성한 뒤 연결하였다. 이 때, packet 단위 채널 용량은 simulation을 통해 얻은 point 10개의 평균 값을 이용하였다. 실험에서 picture 1개는 k_{slice} 개의 slice로 구성했으며, slice 1개는 k_{packet} 개의 packet으로 구성하였다. Picture는 1초에 60장씩 전송하는 상황을 가정하였기 때문에 sample time은 1/24000 sec를 이용하였다.

다음은 위 방식에 따라 MATLAB을 이용하여 생성한 Ricean fading 채널 패턴의 예시를 나타낸 것이다.

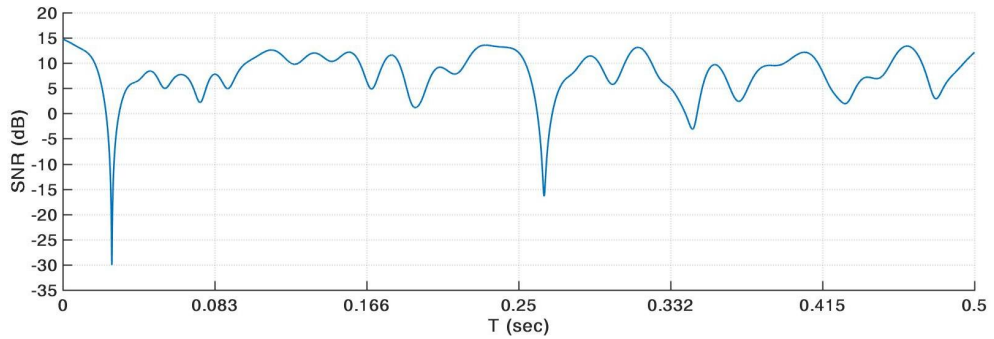


그림 13. Ricean fading channel 모델의 예시

$$(\overline{SNR} = 10dB, K=2, f_D = 50Hz)$$

다음 그림은 현재 가정하고 있는 SNR-scalable encoder에서 채널로부터 feedback 받은 SNR을 이용해 layer의 수와 layer별 bit을 할당하는 feedback loop을 나타낸 것이다. 검정색 음영은 encoding delay를

나타낸 것이며, 나머지 회색 음영은 transmission delay를 나타낸 것이다. 해당 delay로 인해 rate allocation block은 전송 시 채널 용량을 정확히 알 수 없으므로, 실제 채널 용량을 feedback 받아 추정 error를 최소화하는 방향으로 rate allocation을 수행한다.

Video encoding 이전에는 이전 k-GOP 단위의 전송 layer 수 평균인 \bar{K} 를 이용하여 추정한 채널의 상태를 바탕으로 coding layer 수 N을 결정한다.

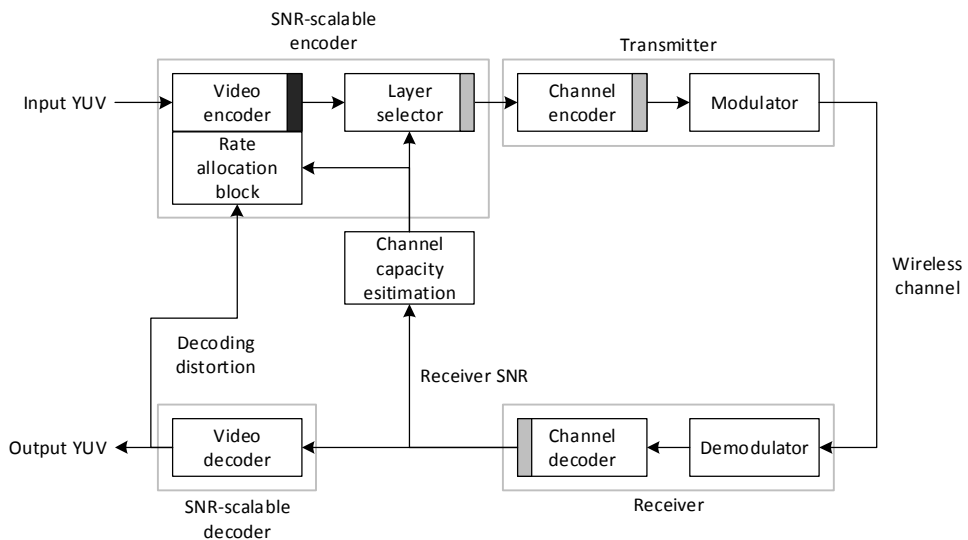


그림 14. 가정한 SNR-scalable coding과 wireless modem의 관계

Slice 정보를 이용한 채널 용량 추정 시에는 regression을 이용하였다. 실험은 2차 다항식으로 근사해 진행하였고, 이 때 5개 slice 정보를 이용하여 1 frame(10 slice) 이후의 채널 용량을 추정하였다.

채널로부터 받은 정보에는 채널 용량을 추정하는데 필요한 SNR값과, 채널 용량 중 실제 비디오 영상에 해당하는 bit/s를 보기 위한 coding rate값이 들어있다. 다음은 Shannon capacity를 나타낸 수식으

로, 채널 용량 추정 시 해당 식과 SNR을 이용해 채널 용량을 추정한다. 이 때, C 는 채널 용량[bit/s], B 는 bandwidth[Hz], SNR은 signal의 power와 noise의 비를 나타낸다.

$$C = B \log_2(1 + SNR) \quad (40)$$

시간에 따른 채널 용량 변화를 관찰하기 위해서는 짧은 샘플링 시간의 평균 SNR을 사용한다. 채널로부터 SNR 값을 feedback 받는 단위는 packet이지만, 본 논문에서는 slice 단위로 채널 용량 정보를 추정하였다. 이 때, k_{packet} 개의 packet이 1개의 slice를 구성한다고 가정했다.

802.11 프로토콜에 따르면 maximum transmission unit 크기는 2304byte로, 이는 1080p 영상을 frame당 slice 10개로 나눈 뒤 100:1로 압축한 경우의 slice 1개당 평균 크기(3110.4byte)보다 작은 수치이다.

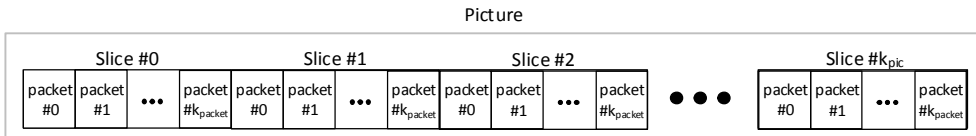


그림 15. Picture내 slice와 packet의 관계

채널 용량의 변화를 추정하기 위해, 본 논문에서는 수신단의 신호 세기(SNR)를 매 packet마다 feedback 받는다고 가정했다. 다만, MCS index는 PHY layer에 존재하지만 본 논문에서는 이 값을 picture 단위로 변경한다.

SNR feedback을 받을 때에는 delay가 발생하는데, 대표적인 것이 encoding delay(slice 10개)와 transmission delay(slice 1개)이다. 해

당 delay로 채널 용량 추정 시 delay만큼의 오차가 발생하고, 이로 인해 채널 용량추정치와 실제 전송 시의 채널 용량이 달라질 수 있다. 이를 해결하기 위해 encoder에서는 feedback 받은 채널 통계로 채널 모델을 세워 전송 시의 채널 용량을 추정한다.

4.2 Slice/picture 정보를 이용한 picture/2-GOP

단위 채널 용량 추정

Picture 단위 채널 용량은 4.1에서 regression으로 얻은 slice 단위 채널 용량 추정 결과를 이용하여 추정한다. 본 논문에서는 picture 1개가 slice 10개로 구성되었다고 가정 하였으므로, regression으로 도출한 slice 단위 채널 용량 10개를 평균 내어 picture 단위 채널 용량으로 추정하였다.

k-GOP 단위 채널 용량 또한 이와 마찬가지로 picture 단위로 추정한 채널 용량의 평균 값을 이용하였다. 이 때, 2-GOP 단위로 채널의 특성을 나타내는 LCR, AFD, AIFD 등의 채널 특성 파라미터를 계산하였다. 이 값은 이후 coding layer 수 N 결정에 이용한다.

본 논문의 시나리오에 해당하는 picture/2-GOP 단위 채널 용량 추정 결과는 6.1.3에서 자세히 다룬다.

4.3 SNR과 packet error rate의 관계

본 논문에 적용된 error concealment 알고리즘은 slice 단위로 동작하므로 packet error rate(PER)과 관련이 있다. SNR과 PER의 관계는 modulation 방법에 따라 변한다. 따라서, 실제 전송 시의 채널 용량을 구하고 이때 사용한 modulation 방법을 MCS table에서 얻는 과정이 필요하다. 표 10은 modulation과 bit error probability 근사 관계를 나타낸 것이다. 표에서 γ_b 는 SNR을, P_b 는 bit error probability를 의미한다.

표 10. Modulation과 bit error probability의 근사 관계

Modulation	$P_b(\gamma_b)$
BFSK	$P_b = Q(\sqrt{\gamma_b})$
BPSK	$P_b = Q(\sqrt{2\gamma_b})$
QPSK, 4QAM	$P_b \approx Q(\sqrt{2\gamma_b})$
MPAM	$P_b \approx \frac{2(M-1)}{M \log_2 M} Q\left(\sqrt{\frac{6\bar{\gamma}_b \log_2 M}{(M^2-1)}}\right)$
MPSK	$P_b \approx \frac{2}{\log_2 M} Q(\sqrt{2\gamma_b \log_2 M \sin(\pi/M)})$
Rectangular MQAM	$P_b \approx \frac{4(\sqrt{M}-1)}{\sqrt{M} \log_2 M} Q\left(\sqrt{\frac{3\bar{\gamma}_b \log_2 M}{(M-1)}}\right)$
Nonrectangular MQAM	$P_b \approx \frac{4}{\sqrt{M} \log_2 M} Q\left(\sqrt{\frac{3\bar{\gamma}_b \log_2 M}{(M-1)}}\right)$

예를 들어, DPSK의 경우 bit error probability는 다음과 같다.

$$P_b = \frac{1}{2} e^{-\gamma_b} \quad (41)$$

1080p 영상을 100:1로 압축한 경우, slice 1개는 24880bit이다. 따라서 packet error rate은 식(42)를 이용해 계산한다.

$$PER = 1 - (1 - BER)^{packet_bit} \quad (42)$$

제 5 장 Layer 수 결정과 layer별 bit allocation

5장은 layer 수와 layer별 bit allocation 과정을 다룬다. 5장의 구성은 다음과 같다. 5.1에서는 GOP 2개 단위 코딩 layer 수 결정에 대한 내용을 서술한다. 5.2는 picture 단위 layer별 bit 할당을 다루고, 5.3은 slice 단위 layer별 bit 할당에 대해 서술한다. 마지막으로, 5.4에서는 MCS index를 결정하는 방법에 대해 서술한다.

5.1 k-GOP 단위 평균 layer bit 할당과 코딩 layer 수 결정

4장에서는 채널 용량 추정 방식에 대해 다루었고, 5.1에서는 이를 기반으로 layer를 나누고 rate를 분할 및 할당 과정에 대해 서술한다.

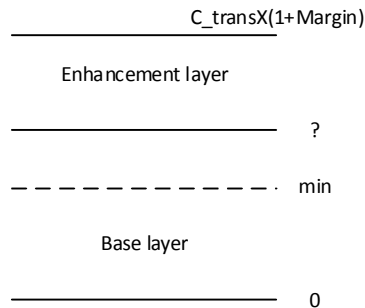


그림 16. Layer 결정 전 rate allocation 문제

채널 용량 추정이 거의 정확하다면 HM과 동일하게 single layer로 encoding을 수행하면 된다. 하지만 채널 용량 추정이 항상 맞지는 않으므로, 채널 용량이 변하는 상황에서도 decoding이 가능하도록 layer를 나누어야 한다.

본 논문의 블록 단위 SNR-scalable coding은 실제 채널 용량에 근접하게 rate를 할당하는 것도 중요하지만, inter-layer dependency로 인한 decoding 화질도 고려해 rate를 할당해야 한다. 2.1.2에서 제안한 누적 rate model은 layer별로 별개의 R-D curve를 가진다고 가정하는 것이다. 마찬가지로 rate allocation 단계에서도 layer별 별개의 R-D model을 세워 decoding 화질을 추정할 필요가 있다.

우선, 추정한 채널 용량(c_{trans})과 영상 데이터의 bitrate(r_{total})를 구분하는 작업을 수행한다. 이는 MCS table에 있는 coding rate(r_{code})를 이용해 계산이 가능하다.

$$r_{total} = c_{trans} \cdot r_{code} \quad (43)$$

Coding rate은 전송 throughput에 따라 결정되는데, 본 논문에서는 채널 용량을 이용해 data bitrate를 역산하는 방법을 사용했다.

Decoding MSE 값의 추정은 현재 rate model에서 사용하고 있는 R-D 관계(hyperbolic)를 이용했으며, C, K는 실제 코딩 결과를 이용하여 picture 단위로 update한다.

$$D(R) = C \cdot R^K \quad (44)$$

Layer마다 사용한 rate이 변하면 distortion도 달라지므로, layer마다 R-D model을 다르게 두어 layer별 distortion을 추정한다. 이 때,

single layer의 rate model과는 달리 하위 layer의 누적 rate 개념이 필요하다. 식(45)는 하위 layer의 누적 rate를 나타낸 것으로, R 은 layer 별 rate, r 은 하위 layer의 누적 rate를 의미한다.

$$r_i = \sum_{k=0}^i R_k \quad (45)$$

$$\arg \min_{N, r_i} \{D_{dec}\} \quad (46)$$

이 때, decoder distortion은 coding layer 수 결정 후 계산한다. Coding layer 수는 이전 2-GOP 단위에서의 transmission layer 수의 평균(\bar{K})을 이용해 결정한다.

$$N \geq \bar{K} \quad (47)$$

예를 들어, \bar{K} 가 2.3인 경우에는 N 값이 3 또는 4가 될 수 있다. 본 논문에서는 coding layer 수 N 결정 시 채널의 상태를 2-GOP 단위로 추정된 결과를 이용한다. 이 때, 2-GOP 내에서의 fading depth와 fading 횟수 변화를 반영하였다.

5.2 Picture 단위 layer별 bit 할당

Picture 단위 rate allocation에서는 GOP 2개 단위 rate allocation 단계에서 다루지 않았던 distortion model parameter update 과정이 추가된다. 이는 picture를 encoding하는 동안 새로운 SNR과 decoder distortion을 feedback 받을 수 있기 때문이다. 이 때 encoder에서는 확인할 수 없는 error concealment 성능을 보기

위하여 decoder distortion을 feedback 받는다.

R-D 관계는 hyperbolic하다고 가정했으며 layer별로 따로 C, K를 관리한다고 전제하였다. 또한, rate model에서 사용하는 파라미터 adaptation 수식을 사용하여 C, K를 매 picture마다 update하도록 하였다.

$$C_{new} = C_{old} + \delta_C \cdot (\ln D_{real} - \ln D_{est}) \cdot C_{old} \quad (48)$$

$$K_{new} = K_{old} + \delta_K \cdot (\ln D_{real} - \ln D_{est}) \cdot \ln bpp_{real} \quad (49)$$

식(48)과 식(49)에서 D_{real} 은 실제 decoding distortion이며, D_{est} 은 식(50)을 이용해 추정된 decoding distortion 값이다. 실제로 error concealment를 적용한 distortion은 알 수 없으므로 5.1과 마찬가지로 error concealment를 적용한 distortion을 추정해서 사용한다.

$$D_{dec.est} = (1 - P_b) \cdot D_{dec} + P_b \cdot D_{dec.EC} \quad (50)$$

본 논문에서는 layer별 target bit을 계산하기 위해 추정된 source rate의 maximum(추정된 source rate과 동일한 값)과 minimum(rate control에 이용하는 최소값) 사이를 구간으로 나누어 decoder distortion을 계산했다. 다음은 N개 layer에 대한 decoder distortion을 나타낸 것이다.

$$D_{dec} = D_N(r_N) \quad (51)$$

$$D_i(r_i) = C_i \cdot bpp(r_i)^{K_i} \quad (52)$$

Rate는 layer 수와는 달리 실수 값을 가지므로 모든 값에 대한 실

힘을 수행할 수 없다. 따라서, 추정된 source rate의 maximum과 minimum 사이를 200개의 구간으로 sampling하여 distortion을 추정한다.

Enhancement layer의 rate이 base layer의 rate과 차이가 큰 경우에는 QP가 커지는 경향이 있어, enhancement layer의 rate이 base layer의 rate의 10% 미만인 경우에는 enhancement layer를 사용하지 않도록 제한했다.

하지만 해당 R-D model에서는 error concealment 알고리즘이 적용되었을 때의 decoding distortion을 확인할 수 없으므로, error concealment 알고리즘을 적용함으로써 인해 화질이 저하되어 distortion이 증가하는 요소를 삽입해야 한다. 식(50)은 이를 반영해 3.4의 error concealment bpp($D_{dec.EC}$)와 4.3의 SNR(γ_b)-PER 관계로 수정한 R-D model의 distortion이다.

본 논문에서는 adaptive power/modulation의 목표 BER인 10^{-6} 을 고정된 BER로 사용 [5]하고 각 bit마다 error가 발생할 확률이 동일하다고 가정하면 PER은 다음과 같이 계산된다. PER은 1080p 영상 100:1 압축률을 기준으로 packet bit이 12441bit 일 때, PER 값은 0.0124이다.

$$PER = 1 - (1 - BER)^{packet_bit} \quad (53)$$

다음은 layer가 4개일 때 decoding distortion이 최소가 되는 rate을 찾는 pseudo code이다. 모든 rate에 대해 cost를 구하는 것은 경우의 수가 너무 많으므로, 앞서 언급한대로 추정된 채널 용량과 동일한 값과, 0 사이 구간을 200개로 sampling하여 rate을 나타냈다.

```

COST = D(r[3], 3)
FOR i=0; i<200; i++
  FOR j=0; j<200; j++
    FOR k=0; k<200; k++
      IF i<j & j<k
        r[0] = RateConv(i)
        r[1] = RateConv(j)
        r[2] = RateConv(k)
        FOR n=0; n<3; n++
          COST = COST + D(r[n], n)
        IF COST < MIN
          MIN = COST
        FOR n=1; n<4; n++
          R[n] = r[n] - r[n-1]

```

5.3 Picture 단위 전송 layer 수 결정

Picture 단위 전송 layer 수 K 결정 과정에 대해 설명하기에 앞서, modulation coding scheme(MCS) 사용의 예시에 대한 설명을 한다. MCS는 802.11 표준에서 공간 스트림, 변조 형태, 부호화율 등을 조합하는 구성으로, 원하는 전송 속도를 결정하기 위해 사용한다. 본 논문에서는 8K 444 10bit @60Hz 영상을 전송하기 위해 802.11ad 프로토콜을 사용하는 환경을 가정한 후 MCS를 사용했다. 표 11은 본 논문에서 사용한 MCS 구성 표의 일부분을 나타낸 것이다.

표 11. 사용한 MCS table

MCS	Code rate	Channel data rate [Mbps]	Source data rate [Mbps]
13	1/2	17.33	8.66
14	5/8	21.66	13.53
15	1/2	34.65	17.32
16	5/8	43.31	27.06
17	3/4	51.98	38.98
18	1/2	69.30	34.65
19	5/8	86.63	54.14
20	3/4	103.95	77.96
21	13/16	112.61	91.49
22	5/8	129.94	81.21
23	3/4	155.93	116.94
24	13/16	168.92	137.24

Wireless channel 용량의 1/40만 사용하는 것으로 가정 하였으므로 표 11을 수정하면 다음과 같다.

표 12. 사용한 MCS table (picture 단위)

MCS	Code rate	Channel data rate [kbps]	Data rate [kb per picture]	Source data rate [kb per picture]
13	1/2	433.25	7.22	3.61
14	5/8	541.5	9.02	5.64
15	1/2	866.25	14.43	7.22
16	5/8	1082.75	18.04	11.28
17	3/4	1299.5	21.65	16.24
18	1/2	1732.5	28.87	14.44
19	5/8	2165.75	36.09	22.56
20	3/4	2598.75	43.31	32.48
21	13/16	2815.25	46.92	38.12
22	5/8	3248.5	54.14	33.84
23	3/4	3898.25	64.97	48.73
24	13/16	4223	70.38	57.18

식(54)는 data rate과 source coding에 이용할 target bitrate 간의 관계를 나타낸 것이다.

$$m = r \times c \quad (54)$$

r은 coding rate으로, MCS table의 code rate에 해당한다. m은 실제 message의 bit length로 source coding에 이용할 target bitrate, c

는 channel coding을 한 codeword의 bit length로 MCS table의 data rate에 해당한다. MCS에서 다루는 c 값(data rate)은 실제 전송 bitrate인 throughput을 의미하고 채널 용량은 이론적인 maximum bitrate을 의미하므로 차이가 있으나, 본 논문에서는 간단하게 throughput과 채널 용량의 구분을 없애고 채널 용량만큼 데이터가 전송된다고 가정한다.

전송 layer 수 K 는 인코딩 결과 실제 사용된 layer별 bitrate을 예측한 채널 용량으로 나눈 값을 이용하여 결정한다. 이 때 각 layer별로 계산된 값을 본 논문에서 이용한 MCS code rate의 최대값인 13/16과 비교하여, 이 값보다 작은 값을 가지는 layer까지 전송할 수 있는 것으로 가정하였다.

5.4 Slice 단위 layer별 bit 할당

기존 HM과 SHVC에서는 slice 단위의 rate control을 수행하지 않는다. 하지만 본 논문에서는 slice 단위 채널 용량을 추정하고 있으므로 slice 단위 rate control의 사용을 고려해 볼 수 있다.

[14]에서는 저지연 환경에서 영상 전송 시 all intra에서 frame보다 작은 단위인 slice 단위의 rate control을 수행하는 과정을 설명한다. Slice 단위 rate control 시에는 slice 단위로 bit 할당 후, 해당 slice에 대한 complexity를 계산한다. 이후, complexity와 target bit를 바탕으로 λ 와 QP를 차례로 결정한다. 이 때 complexity per pixel(cpp)로 이 전 slice와 현재 slice의 complexity를 비교하여 weight를 조절한다.

식(55), (56), (57)은 slice 단위 target rate를 계산하는 과정으로, 입력 받은 target bps로 picture와 slice 단위 평균 bit를 도출한다. 식

(57)은 이를 이용해 slice의 target bit를 계산하는 과정을 보여준다.

$$Bit_{avg}^{pic} = \frac{target_bps}{fps} \quad (55)$$

$$Bit_{avg}^{slice} = Bit_{avg}^{pic} \cdot \frac{area_of_target_latency}{area_of_frame} \quad (56)$$

$$Bit_{target}^{slice}(i) = Bit_{avg}^{slice} - (B_f - Bit_{avg}^{slice}) \quad (57)$$

식(58), (59), (60)은 R-λ model에서 λ를 계산하는 과정이고, 식(61)는 이를 이용해 QP를 계산하는 과정이다.

$$cpp_{pred-err}^{slice} = C_{pred-err}^{slice}(i) / pixels_{slice}(i) \quad (58)$$

$$bpp_{slice} = Bit_{target}^{slice}(i) / pixels_{slice}(i) \quad (59)$$

$$\lambda_{slice}(i) = \frac{\alpha(i)}{256} \cdot \left(\frac{cpp_{pred-err}^{slice \cdot 1.2517}}{\gamma(i) \cdot bpp_{slice}} \right)^{\beta(i)} \quad (60)$$

$$QP_{slice}(i) = \lfloor 4.2005 \cdot \ln(\lambda_{slice}(i)) + 14.2122 \rfloor \quad (61)$$

Slice 단위 rate control 수행 시에는 파라미터의 update 또한 slice 단위로 수행한다. 이 때, 각 slice마다 R-λ model의 특성이 상이할 수 있으므로 각기 다른 R-λ model을 가지도록 한다.

$$\ln cpp^{1.2517} = \ln(C_{intra} / pixels)^{1.2517} \quad (62)$$

$$\Delta(\ln \lambda) = \ln(\lambda_{actual}) - \ln(\lambda_{target}) = \beta \cdot (\ln Bit_{actual}^{slice} - \ln Bit_{target}^{slice}) \quad (63)$$

$$\Delta(\ln \lambda)^* = clip3(-0.125, 0.125, 0.25 \cdot \Delta(\ln \lambda)) \quad (64)$$

$$\alpha_{new} = \alpha_{old} \cdot e^{\Delta(\ln \lambda)^*} \quad (65)$$

$$\beta_{new}(i) = \beta_{old}(i) + \left(\frac{\Delta(\ln \lambda)^*}{\ln cpp^{1.2517}} \right) \quad (66)$$

$$\gamma_{new} = \gamma_{old} \cdot e^{-\frac{\Delta(\ln \lambda)^*}{\beta}} \quad (67)$$

하지만 본 논문에서 사용하는 inter prediction에서의 R- λ model에는 complexity 개념이 없으므로 해당 수식을 바로 적용할 수 없다. 만약 inter prediction에도 cpp의 개념을 도입한다면 [14]와 같이 slice 단위 rate control를 시도할 수 있을 것으로 예상된다.

5.5 MCS index 결정

MCS index는 현재 본 논문에는 적용되지 않았지만, packet 단위로 변경이 가능하다. 이에 대한 설명은 다음과 같다.

우선, 이전 α 번째 packet의 채널 용량을 이용하여 이후 β (α + encoding delay) 번째 packet의 채널 용량을 추정한다. 이후, 표 11에서 가장 작은 code rate 값인 1/2을 초기값으로 이용하여 인코딩 이전 source rate R_s 를 구하고 인코딩을 수행한다. Encoding delay 이후, 인코딩에 사용된 β 번째 packet의 source rate R_s^β 를 구한다. 마지막으로 앞서 계산한 R_s^β 와 추정 값인 R_{ch}^β 의 비를 이용하여 MCS index를 결정한다.

하지만 채널 용량의 추정 값으로 MCS index를 도출한 것과, 실제 채널 용량 값으로 MCS index를 계산한 결과는 다를 수 있다. 해당 오차는 채널 용량의 추정이 잘못되어 발생한 것이다. 따라서, 이를

feedback하여 다음 채널 용량 추정 값을 조정해 MCS index 추정 오차를 줄일 수 있다.

본 논문에서는 위와 같이 packet 단위로 MCS index를 결정하지는 않는다. 다만, picture 단위로 추정된 채널 용량과 인코딩 후의 source rate의 비를 MCS code rate으로 간주하여 전송 layer 수 결정에 활용한다.

제 6 장 Fading channel 환경의 실험 결과

6.1 채널 용량 추정 관련 실험

6.1.1 Fading channel 시나리오 구성

Fading channel 시나리오는 4.1에서 언급한 내용을 바탕으로 한다. 해당 시나리오는 2초 분량으로, 송신단과 수신단이 2m 떨어진 상황에서 시작한다. 수신단은 송신단으로부터 일정 속도로 멀어지며, 이는 f_D 값을 변화를 초래한다. 본 논문에서는 1초에서 1.5초 사이에는 장애물이 출현한다고 가정했다.

평균 SNR은 송수신단 사이 거리가 2.5m일 때 SNR값이 9.303dB라 가정한 식(39)를 이용하였다. [13]에서는 바닥으로부터 Wi-Fi 안테나의 높이가 20cm, 60cm, 100cm일 때를 기준으로 K 값을 측정하여 근사식을 만들었다. 본 논문에서는 바닥에서부터의 안테나의 높이로 60cm를 시나리오에 이용하였다. 이를 식으로 나타내면 다음과 같고 x

는 미터 단위이다.

$$K(x) = -0.05x^3 + 0.72x^2 - 2.88x + 5.52 \quad (68)$$

본 논문에서는 K 값이 식(68)에 따라 변하다가 1초에서 1.5초 사이에 장애물로 인해 0.1로 감소하는 상황을 가정하였다.

앞서 송수신단 사이의 상대 속도에 의해 f_D 가 변한다고 서술하였다. 본 논문에서는 송신단이 정지한 상황에서 수신단이 0.5m/s, 1m/s, 2m/s, 4m/s의 속도로 이동하는 시나리오를 가정했으며, 이 때 f_D 의 값은 각각 100Hz, 200Hz, 400Hz, 800Hz이다. 다음 그림은 각 Doppler shift frequency에서 시나리오로 생성한 채널 상태를 나타낸 그래프이다.



그림 17. $f_D = 100\text{Hz}$ 일 때의 채널 상태



그림 18. $f_D = 200\text{Hz}$ 일 때의 채널 상태

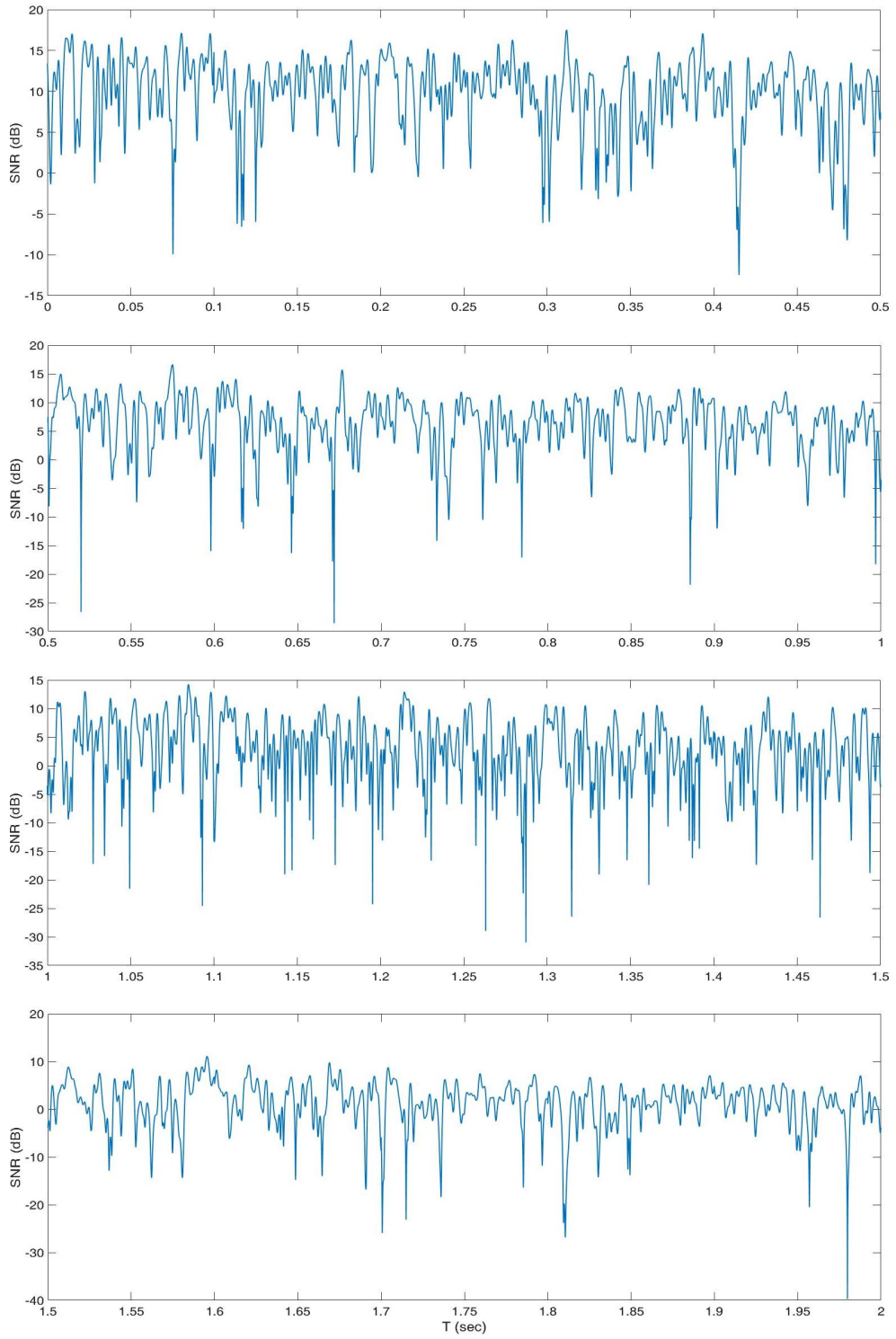


그림 19. $f_D = 400\text{Hz}$ 일 때의 채널 상태

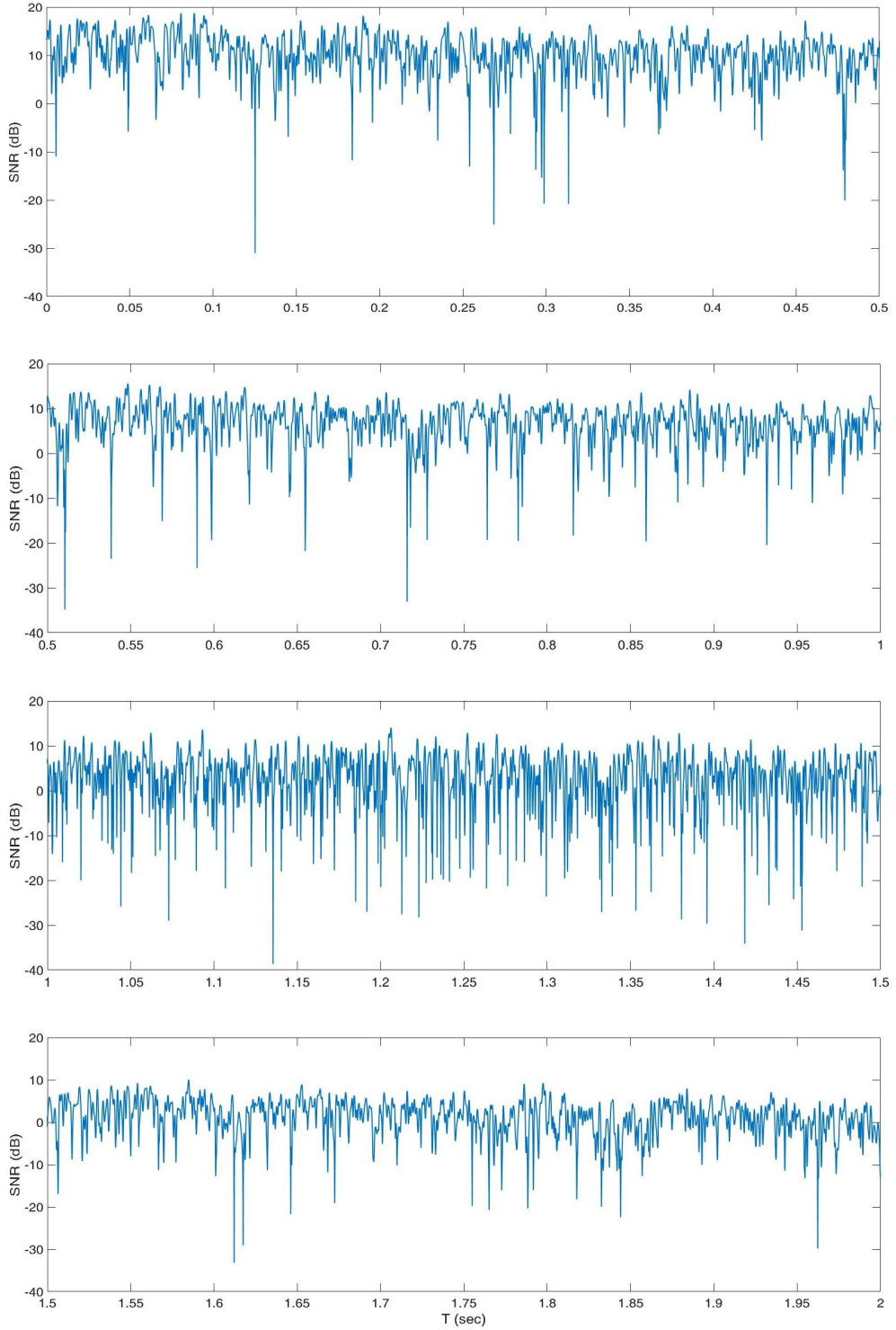


그림 20. $f_D = 800\text{Hz}$ 일 때의 채널 상태

6.1.2 채널 특성 파라미터 측정

6.1.1에서 구성한 fading channel의 특성을 파악하기 위해 각 상황에서 채널 특성 파라미터를 측정하였다. 파라미터는 LCR, AFD, AIFD를 측정하였으며 추가적인 채널 특성 파악을 위해 fading 주기 당 fade된 횟수를 측정하였다. 이 때, fading 주기는 AFD와 AIFD의 합으로 계산하였다.

Threshold는 전체 SNR 값의 평균을 기준으로 5dB씩 감소시켰다. 실험 결과에 의하면 LCR은 f_D 에 비례하여 증가하였고, AFD와 AIFD는 감소하였다. Fading 주기 당 fade가 발생한 횟수는 threshold가 높을 때는 f_D 에 따라 큰 차이가 없었지만, threshold가 낮을 때는 그 값이 커지는 것을 확인하였다.

이러한 채널 특성 파라미터들은 GOP 단위로 측정하여 5.1에서 언급한 coding layer 수 N의 결정에 사용할 수 있다.

표 13. $f_D = 100\text{Hz}$ 일 때의 채널 특성 파라미터

Thres [dB]	LCR		AFD		AIFD [sec]	cycle (AFD+AIFD) [sec]	#fade /cycle
	[cross /sec]	[cross /pic]	[sec]	[pic]			
5	32	0.53	0.0141	0.85	0.0148	0.0289	0.93
0	28	0.47	0.0069	0.41	0.0259	0.0328	0.92
-5	18.5	0.31	0.0034	0.2	0.0409	0.0443	0.82
-10	11.5	0.19	0.0021	0.13	0.0704	0.0725	0.83
-15	8.5	0.14	0.0013	0.08	0.0584	0.0597	0.51
-20	5.5	0.09	0.0007	0.04	0.0688	0.0695	0.38
-25	3	0.05	0.0004	0.02	0.1385	0.1389	0.42
-30	2.5	0.04	0.0002	0.01	0.1734	0.1736	0.43

표 14. $f_D = 200\text{Hz}$ 일 때의 채널 특성 파라미터

Thres [dB]	LCR		AFD		AIFD [sec]	cycle (AFD+AIFD) [sec]	#fade /cycle
	[cross /sec]	[cross /pic]	[sec]	[pic]			
5	67	1.12	0.0065	0.39	0.0073	0.014	0.93
0	60.5	1.01	0.003	0.18	0.0122	0.015	0.92
-5	36	0.6	0.0017	0.1	0.0225	0.024	0.87
-10	21	0.35	0.0009	0.05	0.0352	0.036	0.76
-15	10.5	0.18	0.0005	0.03	0.0644	0.065	0.68
-20	6.5	0.11	0.0003	0.02	0.0902	0.090	0.59
-25	3.5	0.06	0.0001	0.01	0.1171	0.117	0.41
-30	1.5	0.03	0	0	0.3124	0.312	0.47

표 15. $f_D = 400\text{Hz}$ 일 때의 채널 특성 파라미터

Thres [dB]	LCR		AFD		AIFD [sec]	cycle (AFD+AIFD) [sec]	#fade /cycle
	[cross /sec]	[cross /pic]	[sec]	[pic]			
5	127.5	2.13	0.0038	0.23	0.004	0.0078	0.99
0	117.5	1.96	0.0016	0.1	0.0069	0.0085	1.00
-5	70.5	1.18	0.0008	0.05	0.0128	0.0136	0.96
-10	33	0.55	0.0005	0.03	0.0236	0.0241	0.79
-15	18.5	0.31	0.0003	0.02	0.0403	0.0406	0.75
-20	9	0.15	0.0002	0.01	0.0857	0.0859	0.77
-25	4.5	0.08	0.0001	0.01	0.1824	0.1825	0.82
-30	1	0.02	0.0001	0.01	0.6926	0.6927	0.69

표 16. $f_D = 800\text{Hz}$ 일 때의 채널 특성 파라미터

Thres [dB]	LCR		AFD		AIFD [sec]	cycle (AFD+AIFD) [sec]	#fade /cycle
	[cross /sec]	[cross /pic]	[sec]	[pic]			
5	270	4.5	2.E-03	1.E-01	0.002	0.0037	0.99
0	231.5	3.86	8.E-04	5.E-02	0.0035	0.0043	0.99
-5	142	2.37	4.E-04	2.E-02	0.0066	0.0070	1.00
-10	80.5	1.34	2.E-04	1.E-02	0.0122	0.0124	1.00
-15	44	0.73	1.E-04	7.E-03	0.021	0.0211	0.93
-20	22.5	0.38	6.E-05	4.E-03	0.0417	0.0418	0.94
-25	10.5	0.18	5.E-05	3.E-03	0.0918	0.0919	0.96
-30	3.5	0.06	4.E-05	3.E-03	0.2478	0.2478	0.87

6.1.3 Regression을 이용한 채널 용량 추정

다음 그림은 slice, picture, 2-GOP 단위로 채널 용량을 추정한 결과를 나타낸 그림이다. Encoding delay가 1 frame이므로, slice 10개 이후의 채널 용량을 추정해야 한다. 본 논문에서는 이전 5개 slice 정보를 이용하여 1 frame 뒤의 slice 정보를 2차 함수를 이용한 regression으로 추정하였다. 이 때, slice 1개의 채널 용량은 시뮬레이션 결과로 얻은 40 point를 평균 내어 도출하였다.

Picture 단위와 2-GOP 단위의 채널 용량 추정은 regression을 이용한 slice 단위 채널 용량 추정 결과를 각각 10개, 80개씩 평균 내어 얻었다. 각 그림에서 Est로 명시된 것은 1 frame encoding delay를 고려하여 얻은 채널 용량 추정 결과이고, 그렇지 않은 것은 encoding delay를 고려하지 않고 각 채널 용량을 평균 낸 것이다.

다음 그림은 Doppler frequency shift가 200Hz일 때의 slice, picture, 2-GOP 단위 채널 용량 추정 결과를 나타낸 것이다. 실험 결과로 나온 error rate은 slice 단위에서는 86%, picture 단위 30.3%, 2-GOP 단위로는 4.66%이다. Error rate이 높게 나온 것은 encoding delay가 10개 slice이므로 10 point 이전에 정보를 추정해야 하기 때문이다.

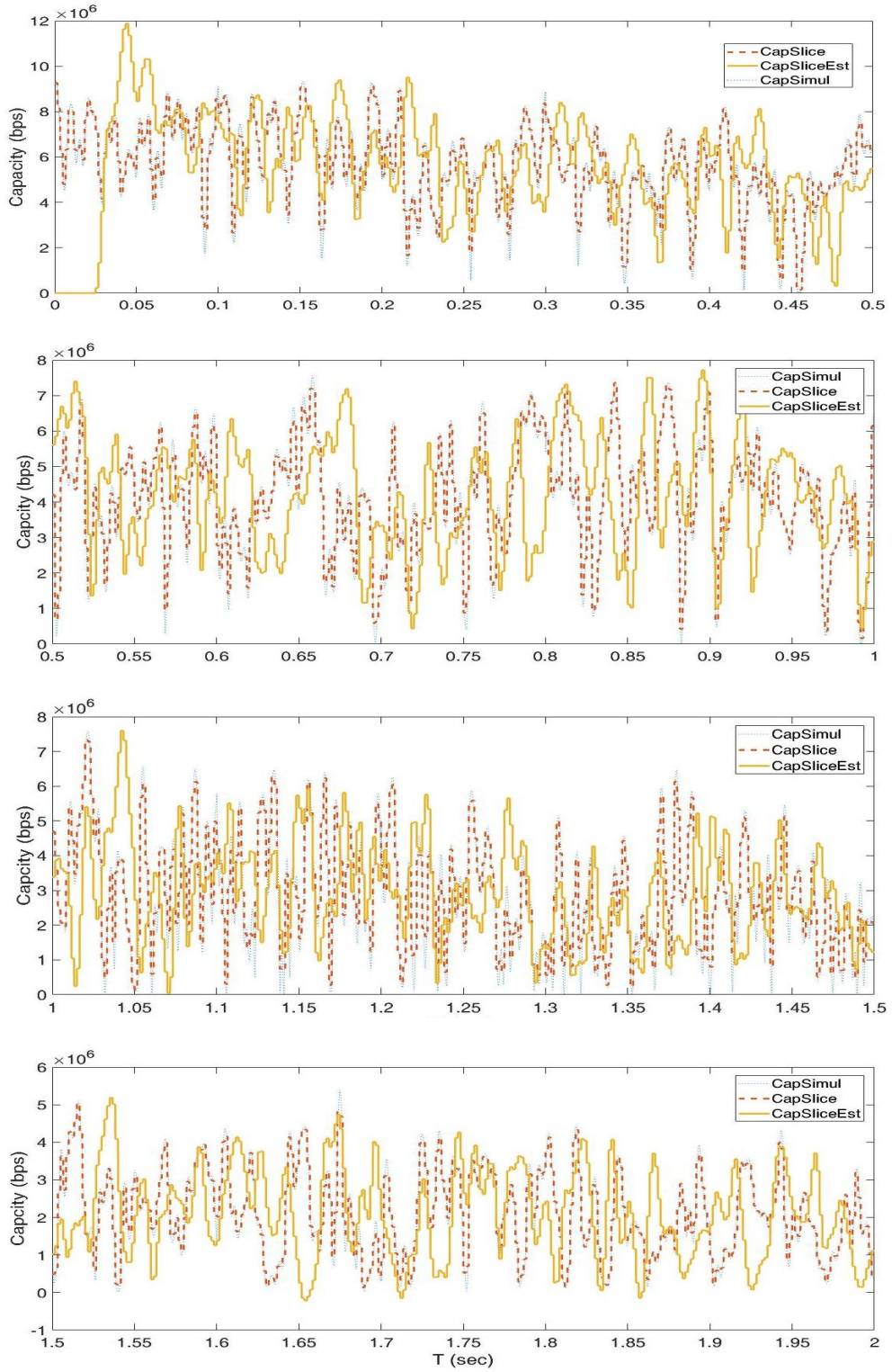


그림 21. Slice 단위 채널 용량 추정 ($f_D = 200\text{Hz}$)

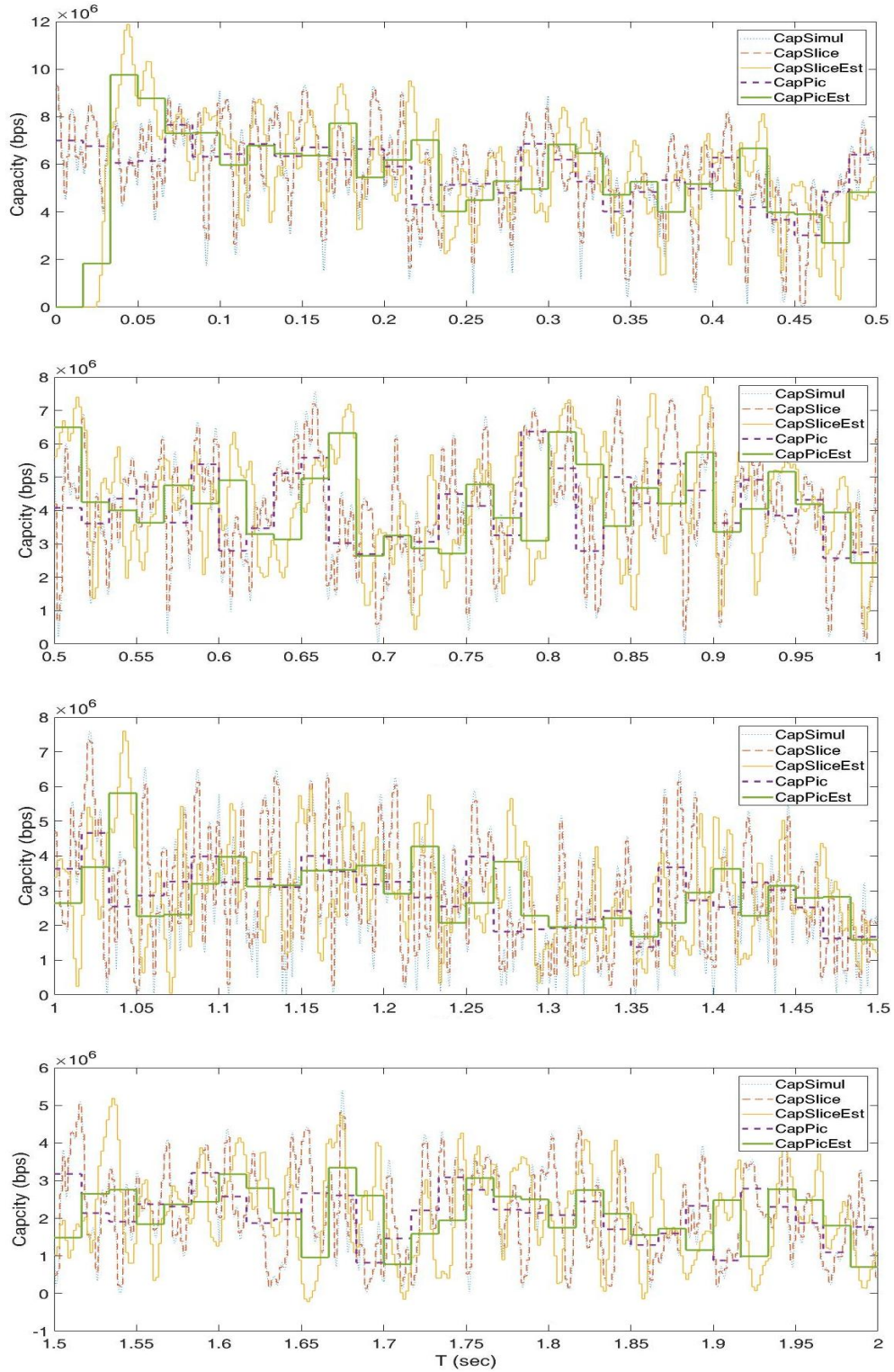


그림 22. Picture 단위 채널 용량 추정 ($f_D=200\text{Hz}$)

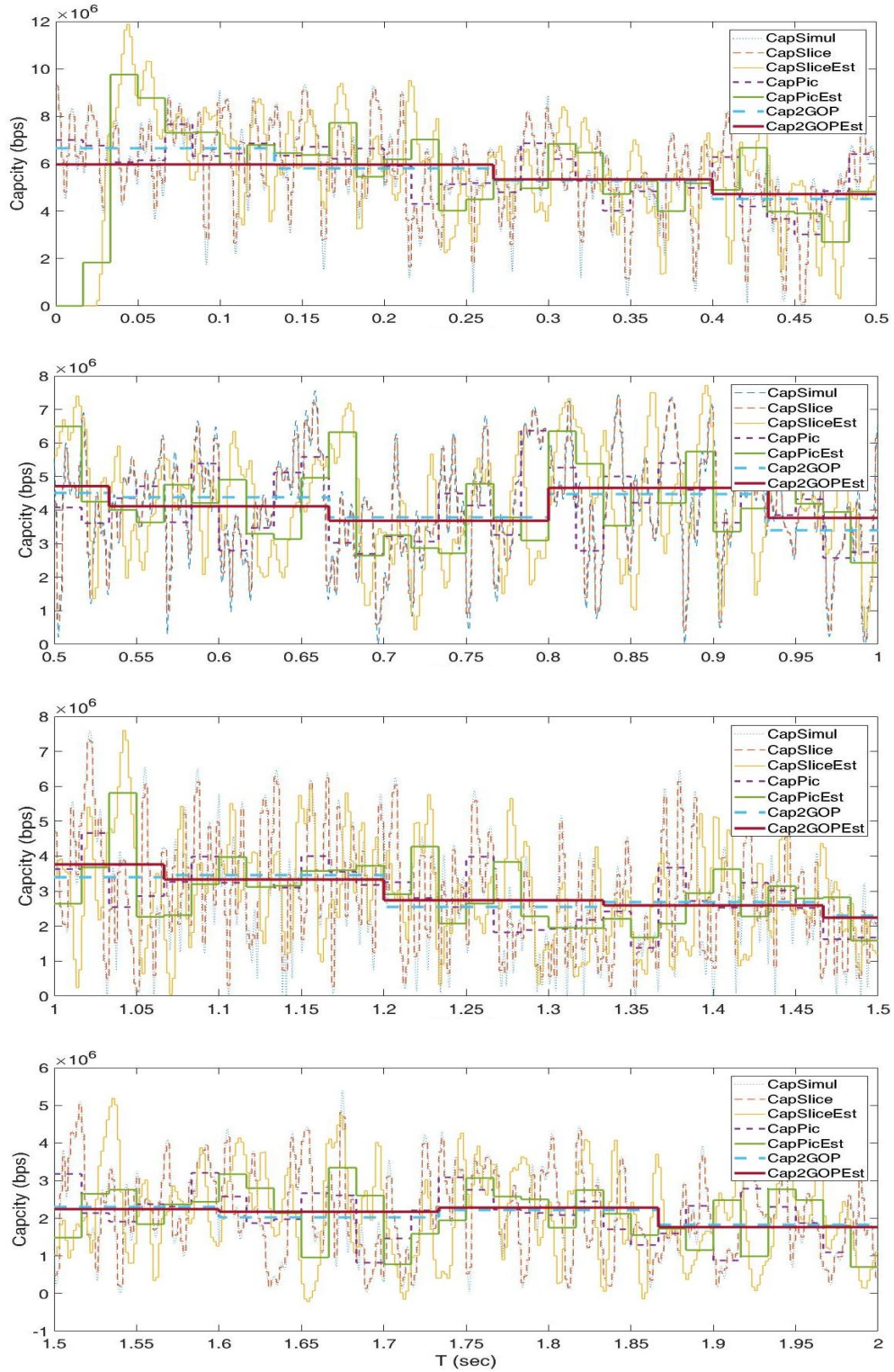


그림 23.2-GOP 단위 채널 용량 추정 ($f_D=200\text{Hz}$)

6.2 Layer 수 결정 및 layer별 bit allocation 관련 실험

6.2.1 Coding layer 수와 전송 layer 수의 비교

다음은 본 논문에서 구현한 SNR-scalable coding을 이용하여 coding layer 수와 전송 layer 수의 변화를 측정한 결과이다. 실험은 Class B 5개 시퀀스 (BasketballDrive, BQTerrace, Cactus, Kimono, ParkScene), 100:1 압축률, 120장, 60 fps에 대하여 진행하였다. 또한, in-loop filter와 SAO는 수행하지 않았다.

표에서 default는 coding layer 수를 4개로 고정하고 각 layer별로 동일하게 bit를 할당한 것이며, adaptive는 본 논문의 방식을 사용한 것이다. Adaptive에서는 첫 번째 2-GOP의 N 값으로 4를 할당하였고, 이후에는 이전 2-GOP의 K 값 평균과 채널 상태 추정 결과를 고려하여 N을 결정하였다. 표 17부터 표 21까지는 2-GOP 단위로 N과 K의 평균을 나타낸 것으로, default의 N 값은 4로 고정된 반면 adaptive의 N 값은 2-GOP 단위로 변하는 것을 확인할 수 있다.

표 17. 2-GOP 단위 N과 K 평균 값의 비교 (BasketballDrive, 100:1)

2-GOP idx	Default		Adaptive	
	N	K 평균	N	K 평균
0	4	2	4	3
1	4	1.75	4	1.875
2	4	1.75	3	1.5
3	4	1.75	3	1.375
4	4	1.875	1	1
5	4	2	1	1
6	4	2.125	1	0.75
7	4	2	2	1.25
8	4	2.125	3	1.75
9	4	2.375	2	1.75
10	4	2.25	3	2.5
11	4	2.125	2	1.5
12	4	2.25	1	1
13	4	2.25	1	0.75
14	4	2.375	1	0.875

표 18. 2-GOP 단위 N과 K 평균 값의 비교 (BQTerrace, 100:1)

2-GOP idx	Default		Adaptive	
	N	K 평균	N	K 평균
0	4	1.875	4	3.125
1	4	0.75	4	2
2	4	0.875	4	1.625
3	4	0.5	3	1.625
4	4	0.625	1	0.875
5	4	0.625	1	0.5
6	4	0.875	1	0.625
7	4	0.5	2	1.375
8	4	0.5	3	1.75
9	4	0.625	2	1.125
10	4	0.5	3	2.125
11	4	0.625	2	1.375
12	4	0.75	1	0.625
13	4	0.875	1	0.75
14	4	0.75	1	0.75

표 19. 2-GOP 단위 N과 K 평균 값의 비교 (Cactus, 100:1)

2-GOP idx	Default		Adaptive	
	N	K 평균	N	K 평균
0	4	1.875	4	2.875
1	4	0.75	4	1.125
2	4	0.875	3	0.5
3	4	0.5	2	0.875
4	4	0.625	1	0.625
5	4	0.625	1	0.5
6	4	0.875	1	0.375
7	4	0.5	2	1
8	4	0.5	3	0.625
9	4	0.625	1	0.5
10	4	0.5	2	1
11	4	0.625	1	0.5
12	4	0.75	1	0.625
13	4	0.875	1	0.75
14	4	0.75	1	0.75

표 20. 2-GOP 단위 N과 K 평균 값의 비교 (Kimono, 100:1)

2-GOP idx	Default		Adaptive	
	N	K 평균	N	K 평균
0	4	1.625	4	2.375
1	4	1	4	1
2	4	1.25	3	1.625
3	4	1	3	0.875
4	4	1	1	0.875
5	4	1	1	0.875
6	4	1.125	1	0.625
7	4	0.875	2	1.125
8	4	1.5	3	1.375
9	4	1	2	1.125
10	4	1.125	3	1.75
11	4	1.375	1	1
12	4	1.25	1	0.875
13	4	1.375	1	0.875
14	4	1.625	1	0.75

표 21. 2-GOP 단위 N과 K 평균 값의 비교 (ParkScene, 100:1)

2-GOP idx	Default		Adaptive	
	N	K 평균	N	K 평균
0	4	1.625	4	2.75
1	4	1	4	0.875
2	4	1.25	2	1.25
3	4	1	3	1.125
4	4	1	1	0.625
5	4	1	1	0.5
6	4	1.125	1	0.625
7	4	0.875	2	1.25
8	4	1.5	3	1.5
9	4	1	2	1
10	4	1.125	3	1.75
11	4	1.375	1	1
12	4	1.25	1	1
13	4	1.375	1	0.75
14	4	1.625	1	0.625

6.2.2 Allocation에 따른 성능 평가

본 절은 allocation에 따른 성능 평가를 다룬다. 앞서 언급한 실험 결과를 POC에 따른 수신단 Y-PSNR로 나타내었으며, 이는 다음 그림과 같다. 전송 layer 개수가 0인 경우에는 수신단에서 error concealment가 수행된다. 이 때, Y-PSNR 값은 표 9의 시퀀스 별 평균 error concealment 결과 값을 이용하였다. 다만, 이 때의 error concealment 결과는 frame 내 slice 들 중 1개의 slice에 대해서만 error concealment를 적용한 결과의 평균 값이므로, 전송 layer 개수가 0일 때와 같이 frame 전체에 error concealment가 적용되어야 하는 경우와는 차이가 있다. 해당 부분에 대해서는 추가 실험을 진행할 필요성이 있다.

표 22는 시퀀스 별 성능 평가 결과를 정리한 것으로, default와

adaptive의 Y-PSNR의 평균과 표준 편차를 나타낸 것이다. 표에서 보는 것과 같이 본 논문에서 사용한 adaptive는 default에 비해 평균 Y-PSNR이 0.522~0.984 dB 개선되었다. 그림 24부터 그림 28은 POC에 따른 수신단에서의 Y-PSNR을 그래프로 나타낸 것으로, adaptive가 default에 비해 Y-PSNR이 높은 것을 확인할 수 있다.

표 22. 수신단 기준 Default와 adaptive의 Y-PSNR 비교

(Class B, 100:1)

Sequence	Average [dB]	StdDev [dB]
BasketballDrive	0.762	1.239
BQTerrace	0.984	3.091
Cactus	0.784	1.029
Kimono	0.522	2.578
ParkScene	0.788	1.185

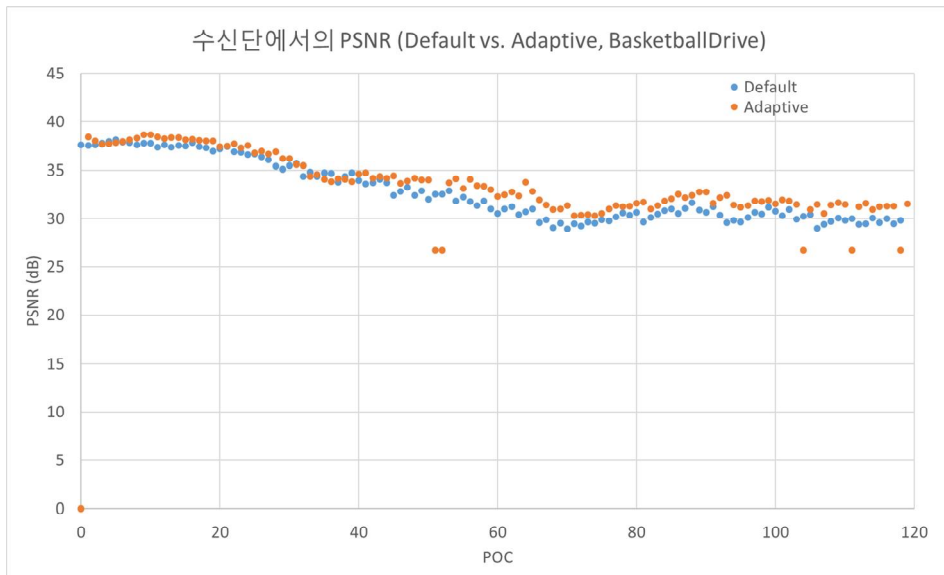


그림 24. 수신단에서의 Y-PSNR (BasketballDrive, 100:1)

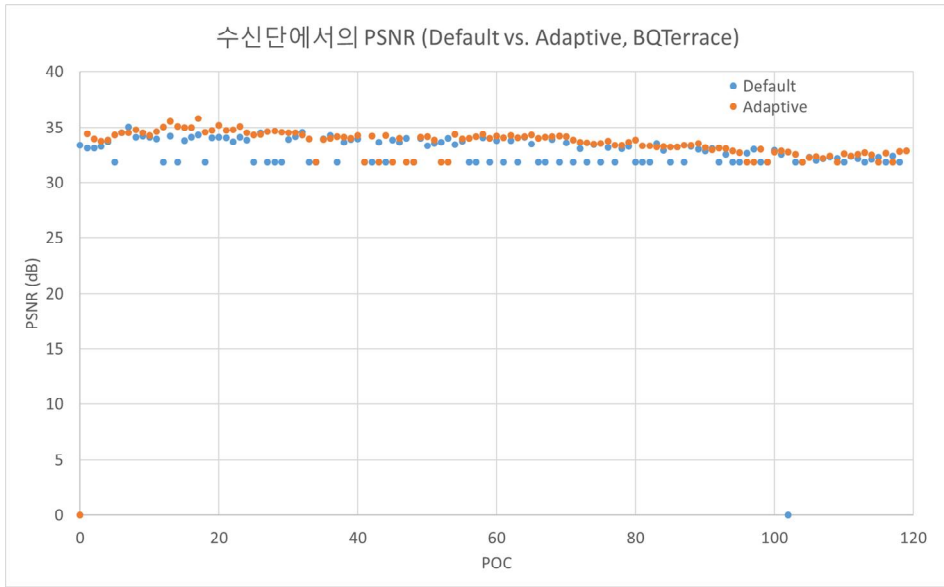


그림 25. 수신단에서의 Y-PSNR (BQTerrace, 100:1)

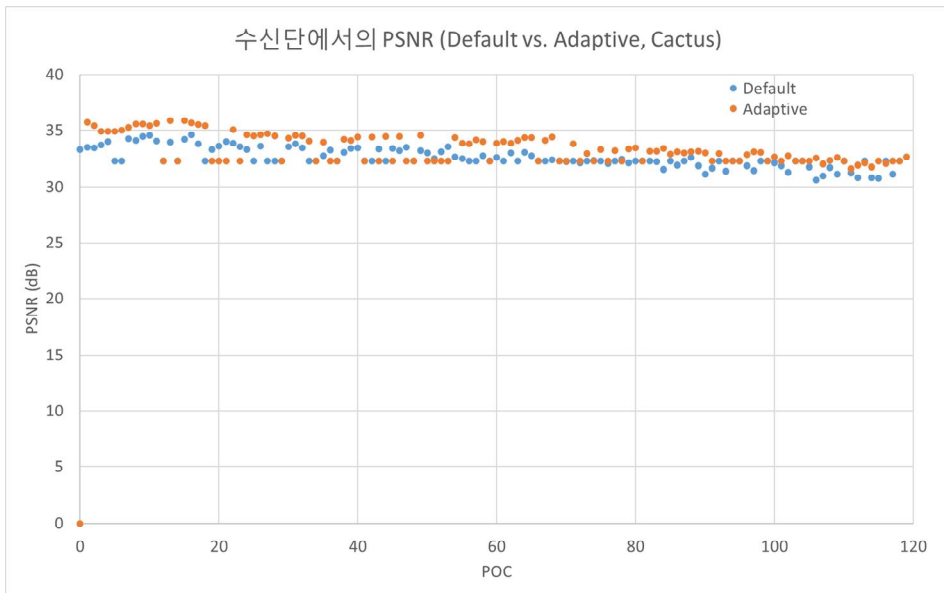


그림 26. 수신단에서의 Y-PSNR (Cactus, 100:1)

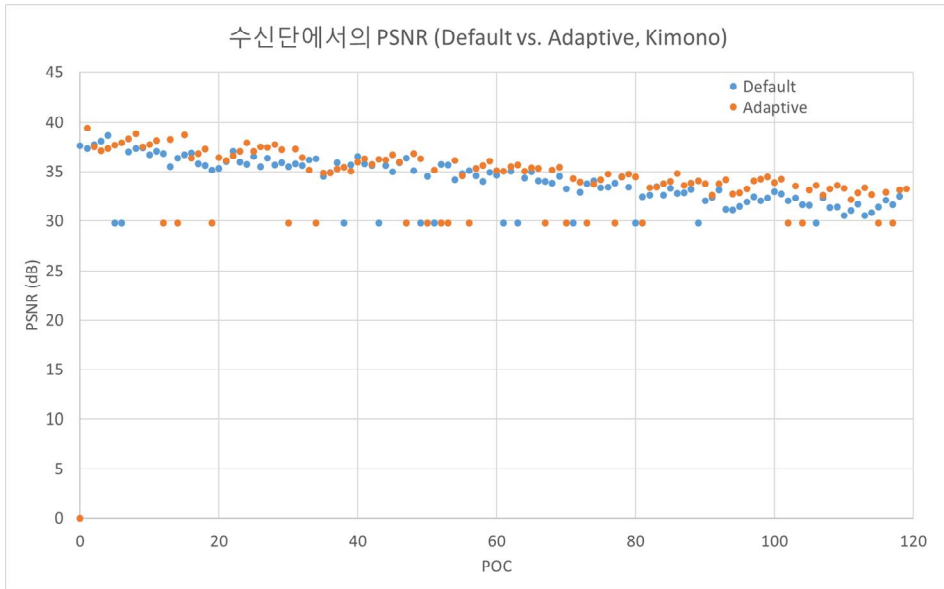


그림 27. 수신단에서의 Y-PSNR (Kimono, 100:1)

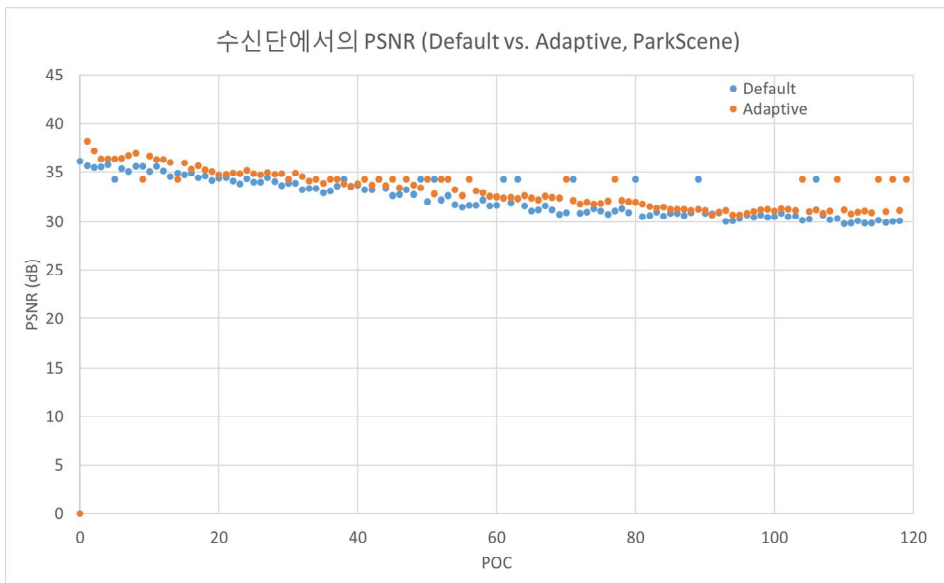


그림 28. 수신단에서의 Y-PSNR (ParkScene, 100:1)

제 7 장 결 론

Fading 채널에서는 채널 용량이 급격하게 변하여 인코딩한 영상을 끊김 없이 전송하는 것이 어렵다. 이를 해결하기 위해 본 논문에서는 SNR-scalable coding 방식을 사용하였다. 기존 HEVC scalable extension의 reference S/W인 SHVC와 달리 layer를 CTU 단위로 구분하고, 상위 layer에서는 base layer의 구조를 이용해 CU partition search의 복잡도를 제거하였다.

Fading 채널 모델은 실내에서 수신자가 이동하는 중에 장애물이 발생하는 상황을 가정하였다. 채널 용량의 추정은 feedback 받은 SNR 정보를 이용하였다. 이 때, encoding delay 이후의 채널 용량을 추정하는데 slice 단위부터 시작하여 picture, 2-GOP 단위의 채널 용량을 차례로 추정하였다. 이 때, 2-GOP 단위로는 몇 가지 채널 특성 파라미터를 이용하여 채널 상태도 추정하였다.

Coding layer 수의 결정에는 앞서 언급한 채널 상태 추정 정보와 이전 2-GOP 단위 전송 layer 수의 평균을 이용하였다. Picture 단위 layer 별 bit 할당 시에는 추정한 채널 용량에 MCS 값을 곱하여 source rate을 계산하였다. 이 값을 최대값으로 하여, 앞서 결정된 coding layer 수로 나눈 결과들 중 decoding distortion을 최소화하는 경우를 선택하였다. 전송 layer 수 K 는 인코딩 결과 실제 사용된 layer 별 bitrate을 예측한 채널 용량으로 나눈 값을 이용하여 결정하였다. 이 때 각 layer별로 계산된 값을 본 논문에서 이용한 MCS code rate의 최대값과 비교하여, 이 값보다 작은 값을 가지는 layer까지 전송할 수 있는 것으로 가정하였다.

실험은 Class B 5개 시퀀스 (BasketballDrive, BQTerrace, Cactus, Kimono, ParkScene), 100:1 압축률, 120장, 60 fps에 대하여 진행하였고, 이 때 in-loop filter와 SAO는 수행하지 않았다.

실험에서 이용한 Ricean fading 채널 시나리오는 2초 분량으로, 송신단과 수신단이 2m 떨어진 상황에서 수신단이 1m/s ($f_D = 200\text{Hz}$)로 멀어지며, 1초에서 1.5초 사이에 Ricean factor 값이 0.1인 장애물이 출현한다고 가정하여 만들었다. 실험에서 평균 SNR 값은 송수신단 사이 거리에 따라 유도한 값을 이용하였으며, Ricean factor는 실내 환경에서 송수신단 사이 거리에 따라 측정된 참고 문헌의 값을 이용하였다. 이 때, 0.1초 간격으로 fading 채널 20개를 생성한 후 이를 연결하여 2초 분량 시나리오를 생성하였다.

실험에서는 coding layer 수를 4개로 고정하고, bit을 layer 별로 동일하게 할당한 것을 anchor로 사용하였다. 실험 결과, 채널 상태 추정 결과에 따라 coding layer 수가 변하는 것을 확인할 수 있었고, 수신단에서 측정한 Y-PSNR은 평균 0.768dB가 개선된 것을 확인하였다.

본 논문에서는 coding layer 수를 결정할 때 fade 횟수와 fading depth를 고려하였다. 이 외에 AFD, AIFD 등을 이용하면 coding layer 수를 더 정확하게 결정할 수 있을 것이다. 또한, 본 논문에서는 rate-distortion 모델을 decoding 성능을 개선하기 위하여 이용하였지만 실험 환경의 SNR에서는 PER이 작아 decoder의 성능 개선 정도가 크지 않았다. 이를 error 면적을 고려한 R-E 모델을 이용하여 개선할 수 있을 것으로 예상된다.

참고 문헌

- [1] G. J. Sullivan, J.-R. Ohm, W.-J. Han and T. Wiegand, "Overview of the high efficiency video coding (HEVC) standard," IEEE Trans. Circuits Syst. Video Technol., vol. 22, no. 12, pp.1649 –1668, 2012
- [2] G. J. Sullivan, J. M. Boyce, and et al, "Standardized Extensions of High Efficiency Video Coding (HEVC)," IEEE Journal of selected topics in Signal Processing, vol. 7, no. 6, 2013
- [3] G. Bianchi, A. T. Campbell, and R. R.-F. Liao, "On Utility-Fair Adaptive Services in Wireless Networks", In Quality of Service, 1998 Sixth International Workshop on IEEE, 1998
- [4] L. P. Kondi, F. Ishtiaq, and A. K. Katsaggelos, "Joint Source-Channel Coding for Motion-Compensated DCT-Based SNR Scalable Video", IEEE Trans. on Image Processing, vol. 11, no. 9, 2002
- [5] A. Goldsmith, Wireless communications. Cambridge university press, 2005.
- [6] J. B. Andersen, J. O. Nielsen, G. F. Pedersen, G. Bauch, and G. Dietl, "Doppler spectrum from moving scatterers in a random environment," IEEE Transactions on Wireless Communications, vol. 8, no. 6, pp. 3270–3277, 2009.
- [7] R. J. Punnoose, P. V. Nikitin, and D. D. Stancil, "Efficient simulation of Ricean fading within a packet simulator," in Vehicular Technology Conference Fall 2000. IEEE VTS Fall VTC2000. 52nd Vehicular Technology Conference (Cat. No.00CH37152), 2000, vol. 2, pp. 764–767

vol.2.

[8] IEEE 802.11ad, Amendment 3: Enhancements for Very High Throughput in the 60 GHz Band. IEEE 802.11 Working Group (2012).

[9] B. Li, H. Li, and et al, " λ -domain Rate Control Algorithm for High Efficiency Video Coding," IEEE Trans. on Image Multimedia, 2014

[10] L. Li, B. Li, and et al, " λ -domain Rate Control Algorithm for HEVC Scalable Extension," IEEE Trans. on Image Multimedia, 2016

[11] B. Li, J. Xu, and et al, "QP refinement according to Lagrange multiplier for High Efficiency Video Coding," IEEE Circuits and Systems, 2013

[12] K. H. Choi and D. H. Kim, "A downhill simplex approach for HEVC error concealment in wireless IP networks," in 2016 IEEE International Conference on Consumer Electronics (ICCE), 2016, pp. 143–146.

[13] S. Wyne, T. Santos, F. Tufvesson, and A. F. Molisch, "Channel Measurements of an Indoor Office Scenario for Wireless Sensor Applications," in IEEE GLOBECOM 2007 – IEEE Global Telecommunications Conference, 2007, pp. 3831–3836.

[14] 정현민, "저지연 비트율 제어 기술 및 상향식 구조의 all intra 인코더". 공학석사학위논문, 2016.

Abstract

Block based SNR–scalable coding for deep fading channels

Seongjin Hong

Department of Electrical and Computer

Engineering

The Graduate School

Seoul National University

This thesis proposes a new algorithm that selects the numbers of coding layers for multiple–layer SNR–scalable coding, allocates the target rate for each coding layer based on estimated channel capacity from the channel state information obtained through a feedback channel and determine the number of transmitted layers for deep fading wireless channels. The multi–layer SNR–scalable coding adopted in this thesis is block–based for easy hardware implementation, which is required to modify the HEVC standard.

Assume that the channel state information in the feedback channel is the average SNR per packet after about a one–picture delay. After obtaining the expected value of the channel capacity for an interval of 2 GOPs, the number of video coding layers is first

selected which minimizes decoding distortion-based cost, then average target bit rate for each layer is determined.

The Ricean channel with three parameters such as the average SNR, the line-of-sight ratio, and the maximum Doppler shift is employed for channel simulation. To test various channel fading some of the parameters are changed for each test case, the level cross rate (LCR), the average fade duration (AFD), and the average inter-fade duration (AIFD) are calculated to estimate the severity of channel fading. The number of coding layers is determined after estimating the number of fading occurrence for each 2-GOP interval and the depth of each fading. Picture-level bit allocation for each coding layer is performed by selecting the case that minimizes the distortion-based cost among the all possible cases.

The number of transmitted erred layer is selected based on the average ratio of the coded bitrate and the estimated channel capacity so that the transmitted data rate must be smaller than the estimated channel capacity.

Keywords : high efficiency video coding (HEVC), SNR-scalable coding, rate control, channel capacity, fading channel

Student Number : 2015-21009