

무선 비디오 스트리밍을 위한 네트워크 적응형 전송오류제어 선택

문성태^o, 김종원

광주과학기술원 정보통신공학과 네트워크미디어 연구실

{stmoon, jwkim}@gist.ac.kr

Network-adaptive Selective Transport Error Control (NASTE) for Video Streaming over WLAN

SungTae Moon^o, JongWon Kim

Networked Media Lab., Dept. of Info. & Com., GIST (Gwangju Institute of Science and Technology)

요 약

무선 네트워크 환경에서의 미디어 전송은 채널의 페이딩 현상 및 채널 잡음 등으로 인한 다량의 패킷손실과 전송지연의 변동으로 급격한 화질 열화를 발생시킨다. 이러한 채널 변동이 심한 무선 네트워크 환경에서 성공적으로 미디어를 전송하기 위해서 FEC (Forward Error Correction), ARQ (Automatic Repeat reQuest) 및 인터리빙을 사용하여 오류를 복구한다. 하지만 각각의 오류제어 모드는 채널 상황 및 오류 패턴에 따라 효율성이 다르기 때문에, 채널 상황에 적응적으로 변경할 수 있는 기법이 필요하다. 본 논문에서는 무선채널 변동을 송·수신측의 피드백 정보를 통해 측정하고 이에 따라 유동적으로 가장 적합한 오류제어 모드를 선택하여 패킷손실을 최소화하는 네트워크 적응형 전송오류제어 기법인 NASTE (Network-Adaptive Selective Transport Error control)를 제안한다. 제안된 기법의 성능을 검증하기 위해 IEEE 802.11g 무선랜 환경에서 NASTE 를 구현해서, MPEG-2 미디어 전송할 때 발생하는 패킷오류를 복구함으로써 전체적인 전송률 개선을 검증한다.

1. 서론

최근 급격한 무선 네트워크 기술의 발달과 다양한 멀티미디어 서비스에 대한 사용자들의 요구들로 인해 무선 네트워크에서의 멀티미디어 전송에 대한 연구와 개발이 활발히 진행되고 있다. 특히 미디어 스트리밍은 네트워크의 변동을 극복하면서 각각의 패킷을 일정한 지연 안에 전송할 수 있어야 한다. 하지만 무선 네트워크에서의 미디어 스트리밍은 채널 특성 및 이동 단말의 성능에 의해 패킷 전송지연, 패킷손실 및 대역폭의 변동과 같은 문제점들을 지니고 있다. 그 중에서도 무선 채널의 페이딩 현상 및 채널 잡음 등은 다량의 채널 오류를 유발한다. 이를 극복하기 위해서 일반적인 무선 네트워크에서는 링크 계층에서 ARQ (Automatic Repeat reQuest) 혹은 FEC (Forward Error Correction)와 같은 전송오류제어를 수행한다. 하지만 모든 오류를 극복할 수 없으며 이는 상위 계층에서 패킷손실 및 전송지연의 변동을 유발하기 때문에 상위 계층에서의 오류제어 기법이 필요하다. 따라서 채널 변동이 심한 무선 네트워크 환경에서 성공적으로 데이터를 전송하기 위해서 전송 계층에서 오류제어 기법인 패킷 단위 FEC 혹은 ARQ 를 추가적으로 사용한다. 한편, 지속적인 오류가 발생하는 경우에는 인터리빙을 사용하여 FEC 성능을 향상시킨다.

패킷 단위 FEC 는 보내고자 하는 패킷 수(k)에 잉여패킷 수 ($m-k$)를 추가하여 전체 n 개의 패킷을 전송한다. 이는 ARQ 와 달리 어떤 피드백도 필요없기 때문에 지연으로 발생하는 문제나 피드백 implosion 문제점 등을 해결할 수 있다. 하지만, 만약 $n-k$ 이상의 패킷손실이 일어날 경우 모든 손실된 패킷은 복원할 수 없다는 문제점 (all-or-none error recovery)을 가지고 있다. 따라서 오류 발생률이 높은 경우 FEC 사용은 비효율적이다. ARQ 는 손실된 패킷을 송신측에서 재전송하는 방식이다. 이 방법은 FEC 와 달리 하나의 손실된 패킷이라도 제대로 전송받게 되면, 그 수만큼 복원 (incremental recovery)이

가능하다[1]. 따라서 오류 발생률이 높은 경우 ARQ 는 효율적으로 동작한다. 하지만 이 방법은 예측 불가능한 지연을 야기할 수 있으며 엄격한 실시간 제약을 갖는 멀티미디어 응용 서비스에는 적절하지 않다. 한편, 패킷 인터리빙 (interleaving) 방식은 연속적인 패킷손실 (burst loss)을 분산시키는 효과를 가진다. 하지만 손실된 패킷 자체를 복구할 수는 없으며 송신측에서 인터리빙을 수행하고 수신측에서 재구성하는 과정에서 상당한 지연이 발생하는 문제점을 갖고 있다. 기존의 방법들은 FEC 혹은 ARQ 를 적절히 조합하여 사용함으로써 채널 오류 발생을 최소화 하고자 하였다[5]. 하지만 각 오류제어 모드는 채널 상황과 오류 패턴에 따라 성능이 다르게 나타나기 때문에 유동적이고 네트워크에 적응적으로 변화 시켜주는 기법이 필요하다[2].

본 논문에서는 패킷손실 복구를 위하여 무선 채널 변동에 따라 패킷단위 FEC, interleaved FEC 및 delay-constrained ARQ 를 선택적으로 조합한 오류제어 모드를 이용하여 패킷손실을 최소화하고 효율적으로 채널을 사용하는 네트워크 적응형 오류 제어 기법(NASTE)을 제안한다. 보다 정확하게 채널상태를 파악하기 위해서 기존 수신측의 피드백 방식과 더불어 송신측 하위 계층을 모니터링하는 cross-layer 적인 방식을 병행하도록 한다. 제안된 기법은 IEEE 802.11g 무선랜 환경에서 MPEG-2 미디어 전송을 통해 전송률을 검증하기 위해 VideoLAN[3] 오픈 소스에 결합하여 제안한 오류제어 모듈을 추가하였으며, 일반적인 전송오류제어 방법과 제안한 기법 성능을 비교하여 검증한다.

본 논문의 구성은 다음과 같다. 2 절에서는 관련 연구들에 대한 소개하고, 3 절에서는 제안한 NASTE 기법을 제시한다. 4 절에서는 실험 결과를 통해 제안된 기법의 성능을 검증하며, 마지막으로 5 절에서 본 논문을 마무리한다.

2. 관련 연구

FEC, ARQ 및 인터리빙을 혼합하여 사용하는 전송 오류제어방법은 그 동안 많은 연구들이 수행되어왔다 [4-6]. [5]에서는 ARQ와 FEC를 결합하여 사용하는 혼합형 ARQ (hybrid ARQ) 방식을 사용하여 수신측의 요청이 있는 경우에 한해서 송신측이 인여패킷을 전송함으로써 보다 효율적으로 오류를 복구하는 방법을 제시하였다. 특히 [6]에서는 ARQ와 FEC를 채널 상황에 따라 혼합형 ARQ 방식과 ARQ 및 FEC를 채널 상황에 따라 모드를 선택하여 사용한다. 이 방법은 본 논문에서 제안된 NASTE처럼 재생버퍼와 오류 발생률을 고려하는 점에서 비슷하나 인터리빙을 사용하지 않기 때문에 연속적인 오류에 대한 해결 방법이 부족하다. 뿐만 아니라 오류 패턴을 반영하지 않아서 적합하지 않은 모드로 작동할 수 있다. 한편, [4]에서는 무선 채널상태나 전송지연을 고려하여 조건부 재전송 방식과 FEC기반에 low-delay 인터리빙 방식을 병행하여 데이터를 전송하는 기법에 대해서 제안한다. 이 방법은 다른 방법과 달리 인터리빙을 사용하여 연속적인 오류를 고려하지만, 채널상태에 따른 변환 기법에 대해서는 고려하지 않기 때문에 불필요한 인터리빙 사용으로 인해 지연이 증가한다.

뿐만 아니라 대부분 채널 상황을 측정하기 위해 주로 수신측의 피드백 정보에만 의존하기 때문에 전송되는 데 걸리는 지연과 피드백 손실에 대한 문제점을 가지고 있다. 이 경우 실시간으로 채널 상황을 측정하기 어렵다는 단점이 있다. 더욱이, 주로 시뮬레이션 기반으로 수행되었기 때문에 실제 환경에는 적용하기에 고려해야 할 사항들이 많다.

3. 네트워크 적응형 전송오류제어 선택 기법 (NASTE)

본 논문에서 제안된 NASTE는 패킷단위 FEC, delay-constrained ARQ 및 인터리빙을 채널 상황에 따라 적절하게 선택 및 조합하여 사용한다. 따라서 현재 어플리케이션과 네트워크 상태를 정확하게 파악하고, 이를 기반으로 적합한 오류제어 모드를 선택한다. 제안된 NASTE에서는 오류제어 모드를 선택하기 위해 사용가능성을 측정하고 패킷오류 발생률 및 패턴을 고려한다. 모드 사용가능성을 측정하기 위해서 재생버퍼 (playback buffer)가 견딜 수 있는 시간을 각 모드의 임계값과 비교한다. 한편 패킷오류 발생률과 패턴은 송수신측을 모니터링하여 구한다.

그림 1은 제안된 기법에 대한 전체적인 구조도를 나타낸다. 이 오류제어 기법은 network adaptive manager 모듈에 의해서 효과적으로 제어된다. 이 모듈은 송수신측에서 전달해주는 전송 정보를 수집하고 이를 바탕으로 NASTE 인코딩 부분에서 현재 어플리케이션과 네트워크 상황에 적합한 오류제어 방법을 선택하도록 도와준다. 일단 수신측에 도착한 모든 데이터들은 버퍼에 저장되는데 이는 전송과 재생버퍼가 결합된 방식으로 분리된 버퍼 사용으로 인한 지연을 최대한 줄였다.

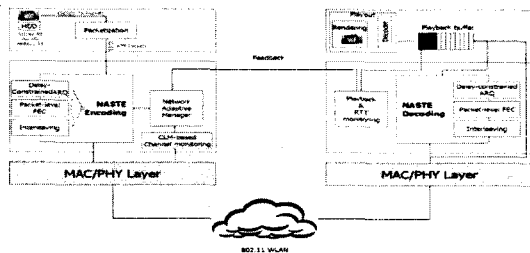


그림 1. NASTE의 기본 구조.

이때 NASTE의 성능 평가를 위해 하위 계층에서 지원하는 오류제어 방법은 최소화 한다. 이 방법은 1 홉에서의 단일 node 경우 발생하는 오류를 제어하는데 목적을 두고 있기 때문에 다른 상황에서 발생하는 오류는 고려하지 않는다. 한편 가용 대역폭은 NASTE를 사용하기에 충분하다고 가정한다. 마지막으로 제안된 기법은 수신측의 재생버퍼상태가 최대한 인터리빙 깊이 (interleaving depth)를 10 개까지 수용하기 위해 90 frame (3 초)를 버퍼링 한다.

3.1. 모드 별 임계값 과 오류 패턴 측정

제안된 NASTE 기법은 상황에 따라 효과적인 모드를 선택하기 위해서 송·수신측의 피드백을 통하여 모드 별 임계값 (msec)과 오류 패턴을 측정한다. 먼저 모드 별 임계값과 비교하기 위해 수신측 재생버퍼 레벨을 측정한다. 재생버퍼 레벨은 버퍼에 저장된 프레임 단위의 데이터 양 (N_{buffer})을 의미하는데, 이는 수식 (1)과 같이 시간 도메인 ($T_{playback}$)으로 변경해서 근사화할 수 있다.

$$T_{playback} = \frac{N_{buffer}}{framerate} \times 1000. \quad (1)$$

$T_{playback}$ 과 비교하여 사용 가능한 모드를 알기 위해 수신측에서 요구하는 ARQ와 인터리빙의 임계값을 측정한다. ARQ 임계값 (T_{ARQ})은 수식 (2)와 같이 표현된다.

$$T_{ARQ} = RTT + T_{process,ARQ}. \quad (2)$$

여기서 $T_{process,ARQ}$ 는 ARQ 사용시 송·수신측의 대략적인 처리시간을 의미한다. 한편 인터리빙을 사용하기 시작할 때 버퍼는 비어있는 상태라고 가정하고, worst case에서 인터리빙 임계값 (T_{inter})은 수식 (3)과 같이 표현된다.

$$T_{inter} = d_{min} \cdot n \cdot L \cdot \left(\frac{1}{r_{source}} + \frac{1}{r_{sending}} \right) + RTT + T_{process,inter}. \quad (3)$$

여기서 d_{min} 은 인터리빙에서 필요한 최소한의 깊이 (depth)를 의미한다. 본 논문에서는 d_{min} 을 2로 설정하였다. n 은 FEC 블록 크기를 나타내고 L 은 패킷 길이를 의미하고 이때 L 은 1500byte로 설정한다. 한편 r_{source} 는 미디어 전송률을 의미하고 $r_{sending}$ 은 전송 계층의 전송률을 의미한다. 마지막으로 $T_{process,inter}$ 은 인터리빙 사용시 수신측의 대략적인 처리 시간을 나타낸다.

오류 패턴은 오류 발생률과 오류 지속성을 측정하여 결정된다. 오류 발생률은 수신측의 피드백을 통하여 일정 기간동안 발생한 오류양으로 계산한다. 하지만 위에서 언급한 것처럼 피드백을 통하여 네트워크 상황을 모니터링하기에는 네트워크 상황의 악화로 인한 손실가능성이 크다. 특히 실시간으로 네트워크 상황을 파악해야 하는 경우 피드백 손실은 전체 성능에 큰 영향을 준다. 따라서 본 논문에서는 오류 발생률을 구하기 위해 송신측 MAC 계층에서 일정 기간동안 전송된 패킷과 수신된 ACK를 모니터링한다 [9]. 오류 지속성은 EWMA (Exponential Weighted Moving Average)를 이용하여 측정한다. 본 논문에서는 100msec 마다 오류 발생률을 샘플링하고 α 값을 0.1로 설정하였다.

한편, 수신 신호강도를 나타내는 RSSI (Receive Signal Strength Indicator)는 오류 발생의 근본 원인으로 볼 수 없지만 오류가 자주 발생하는 시점을 나타낼 수 있다 [7]. 특히 실험을 통해 실내공간의 IEEE 802.11g 무선랜 환경에서 54Mbit/s로 전송 시

신호강도에 따른 오류 발생률을 측정한 결과 RSSI 값이 215 이하인 경우 오류가 자주 발생함을 알 수 있었다 (그림 2). 본 논문에서는 오류 발생이 증가되는 시점을 나타내기 위해 RSSI 를 사용한다.

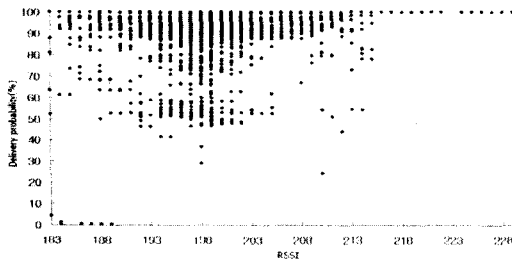


그림 2. RSSI 에 따른 전송률 분산도.

3.2. 네트워크 적응형 모드 변경

무선 네트워크 환경은 채널 변경에 따라 패킷손실의 차가 매우 크다. 특히 사용자의 이동성이 큰 환경에서는 오류 발생률 및 발생 패턴이 시간에 따라 급변한다. 이런 환경에서 미디어를 전송하기 위해서는 네트워크 상황에 맞추어 적절한 오류제어 모드를 선택해야 한다. 본 논문에서는 오류제어 모드를 표 1 과 같이 오류 패턴에 따라 4 가지로 분류한다.

표 1. 오류제어 모드.

| 오류제어 모드 | 오류 패턴 | T_{playback} |
|-----------------------|------------|-----------------------|
| FEC | 지속적인 오류 | - |
| ARQ | 갑작스런 오류 | T_{ARQ} 이상 |
| FEC & ARQ | 많고 연속적인 오류 | T_{inter} 이하 |
| Interleaved FEC & ARQ | 많고 연속적인 오류 | T_{inter} 이상 |

지속적인 오류는 오류 발생률이 적고 오류가 일정 기간동안 지속적으로 나타나는 경우로 이 경우 FEC 모드를 사용한다. 갑작스런 오류는 오류 발생률이 적고 가끔씩 발생하는 오류를 나타낸다. 이 경우 FEC 모드를 사용하면 불필요한 잉여패킷 사용으로 비효율적이기 때문에 ARQ 모드를 사용한다. 한편 오류 발생률이 많은 경우 연속적인 오류가 나타나는데 이 경우 interleavedFEC&ARQ 모드를 사용하여 오류를 분산시킨다. 하지만, T_{inter} 임계값을 충족시키지 못하는 경우 FEC&ARQ 모드로 변경된다.

채널 상황에 따라 오류제어 모드 선택 절차는 그림 3 과 같다. 먼저 전체적인 채널상태를 파악하기 위해 RSSI 값을 측정한다. 이때 RSSI 값이 임계값 이상인 경우 채널 상황이 좋다고 판단하고 오류 지속성에 따라 ARQ 혹은 FEC 모드를 사용한다. 하지만 T_{ARQ} 가 충족되지 못하면, ARQ 모드는 사용할 수 없어 FEC 모드를 사용한다. 한편 RSSI 값이 임계값 이하로 떨어져 채널상태가 불안정하다고 판단되면 복합적인 방법으로 오류를 복구한다. 특히 연속적인 오류가 발생하고 T_{inter} 를 충족하는 경우 interleavedFEC&ARQ 모드를 사용한다. 하지만 T_{inter} 를 충족시키지 못하는 경우 이 모드를 사용하면 오히려 지연으로 데이터가 재시간에 도착하지 못하기 때문에 FEC&ARQ 모드를 사용한다.

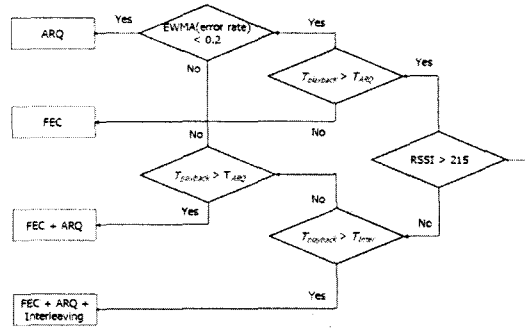


그림 3. 채널 상황에 따른 오류제어 방법 선택 절차.

제한된 NASTE 기법은 적응적으로 FEC 의 잉여패킷의 양을 조절하여 패킷손실률을 최소화함과 동시에 대역폭을 효율적으로 사용한다. 또한 연속적인 오류가 발생하는 경우 interleaved FEC 가 사용되기 때문에 FEC 의 잉여패킷의 양이 변경되면 FEC 블럭 크기(n)가 변경되어 인터리빙 블럭 크기(d)도 변경된다. FEC 의 잉여패킷의 양을 조절하는 방식은 오류발생률을 바탕으로 수식 (4)를 이용하여 FEC 의 효율성 (throughput)[8]을 계산하고, 그 결과가 가장 큰 경우의 잉여패킷 양(n-k)를 선택하여 사용한다. 이때 패킷손실은 독립적으로 발생한다고 가정한다.

$$\begin{aligned}
 T_{\text{FEC}} &= \frac{k}{n} \text{Prob}(X \leq n-k) \\
 &= \frac{k}{n} \sum_{j=0}^{n-k} \binom{n}{j} P_e^j (1-P_e)^{(n-j)} \\
 &= \frac{k}{n} \sum_{j=0}^{n-k} P_e^j (1-P_e)^{(n-j)} \frac{n!}{j!(n-j)!}
 \end{aligned} \tag{4}$$

여기서 X 는 n 개의 패킷으로 이루어지는 FEC 블럭에서 손실된 패킷의 개수를 나타내는 랜덤 변수이고, P_e 는 오류 발생률을 나타낸다. 또한 인터리빙은 FEC 와 같이 사용하기 때문에 FEC 의 n 값이 결정되면 T_{playback} 과 비교하여 수식 (5)와 같이 깊이(d)를 계산한다.

$$d = \left\lceil \frac{D_{\text{playback}}}{n \cdot L \cdot \left(\frac{1}{r_{\text{source}}} + \frac{1}{r_{\text{sending}}} \right) + RTT + T_{\text{process,inter}}} \right\rceil \tag{5}$$

한편 인터리빙을 사용하게 되면 전체적으로 패킷오류가 분산되기 때문에 FEC 의 n 값도 또한 변경해주어야 한다. 연속적인 오류가 발생하게 되면, 오류가 d 만큼 분산되기 때문에 최소한 $\text{PacketLossRate}/d$ 의 오류 발생에 복구 가능하도록 인터리빙 사용이 지속되는 동안은 n 값을 유지한다.

4. 실험결과

본 논문에서는 NASTE 의 성능 측정을 위해 IEEE 802.11g HostAP 기반 무선랜 환경에서 테스트 하였다. 이를 위하여 미디어 스트리밍 서버를 구축하였고, MPEG-2 비디오 콘텐츠를 사용하였다. 한편 NASTE 성능 실험을 위해 다른 계층에서 사용되는 오류제어를 최소화하기 위해 MAC 계층의 재전송

횟수를 1로 설정하였다. 특히 클라이언트에 이동성을 부여함으로써 시간에 따라 채널상태가 변화되도록 하였다. 따라서 경우에 따라서는 채널상태가 좋지 않아 거의 모든 데이터가 전송이 되지 않은 경우도 발생한다. 생성되는 비디오 스트림은 SD 급으로 약 7.5Mbps 정도의 전송률을 가지며 CBR (constant bit rate)이다.

그림 4-7은 각 오류 패턴에 따라 선택된 모드를 통하여 오류를 복구하는 결과를 나타낸다. 그림 4에서는 적은 오류가 났을 때 ARQ 모드를 사용해 복구한다. 하지만, block number가 408인 부분에서 T_{playback} 이 T_{ARQ} 을 충족시키지 못하였기 때문에 오류를 복구 할 수 없었다. 반면에 오류가 지속적으로 나는 경우는 그림 5와 같이 FEC 모드를 사용하여 복구한다. 한편 오류가 많이 나는 경우는 그림 6과 같이 FEC& ARQ 모드를 병행하여 복구한다. 하지만, 이 경우 수신측의 피드백이 늦어져서 잉여패킷이 후에 적용됨을 볼 수 있다. 반면에 오류가 연속적으로 나는 경우 그림 7과 같이 interleaved FEC&ARQ 모드를 사용하여 연속적으로 발생한 오류를 분산시키고 FEC와 ARQ를 통해 복구한다.

NASTE에 대한 성능 평가를 위해 기존 적용적으로 FEC와 ARQ를 혼합하여 사용하는 경우와 오류제어(EC)를 전혀 사용하지 않는 경우를 실험을 통하여 얻은 데이터를 기준으로 계산해보면 표 2와 같다. 전체적인 오류 발생률은 ARQ와 FEC를 사용하는 것과 비교할 때 인터리빙을 사용함으로써 크게 감소하였으며, FEC로 인한 잉여패킷의 부하는 약 18.5% 정도로 나타났다.

5. 결론 및 향후 연구 계획

본 논문에서는 실시간으로 가변하는 무선 채널로 인한 다량의 패킷손실을 최소화하고 변동하는 무선 네트워크 환경에 적용하는 NASTE 기법에 대해서 살펴보았다. 그리고 이를 검증하기 위해 IEEE 802.11g 무선랜 환경에서 미디어를 전송해 보았고 제안된 기법의 성능이 우수함을 확인할 수 있었다. 특히 연속적인 오류가 발생시 적용적으로 인터리빙을 사용하여 오류제어 효과를 증가 시켰다. 하지만 현재 실험 결과는 어플리케이션 및 네트워크 상태가 매우 안 좋을 경우를 대비한 전송속도 조절 부분이 아직 고려되지 않아서 보완이 필요하다.

표 2. NASTE의 성능 비교.

| | No EC | FEC + ARQ | NASTE |
|---------------------|-------|-----------|-------|
| Total loss rate (%) | 4.88 | 3.28 | 2.99 |
| Total overhead (%) | 0 | 18.5 | 18.5 |

6. Acknowledgment

본 연구는 한국과학재단 특정기초연구(R01-2006-000-10556-0)지원으로 수행되었음.

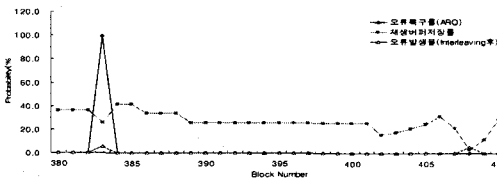


그림 4. ARQ 모드를 통한 오류제어.

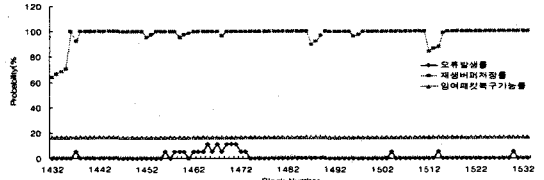


그림 5. FEC 모드를 통한 오류제어.

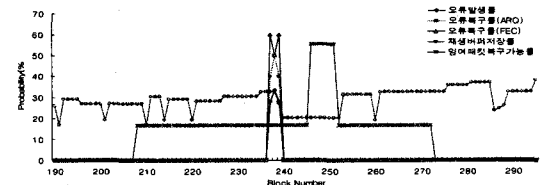


그림 6. FEC & ARQ 모드를 통한 오류제어.

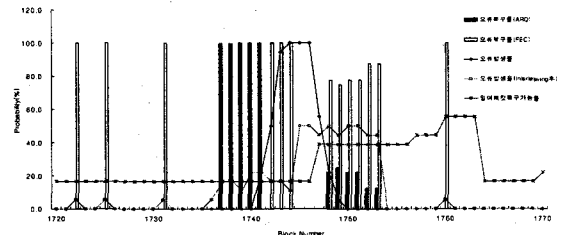


그림 7. Interleaved FEC & ARQ 모드를 통한 오류제어.

7. 참고 문헌

- [1] S. Chan and X. Zheng, "Video loss recovery with FEC and stream replication," in *Proc. IEEE Transaction on Multimedia*, Vol. 8, Issue 2, pp.370-381, Apr. 2006
- [2] J. Kim and J. Shin, "Dynamic network adaptation framework employing layered relative priority index for adaptive video delivery," in *Proc. IEEE Pacific-Rim Conference on Multimedia (PCM 2002)*, Dec. 2002.
- [3] VideoLAN, <http://www.videolan.org>.
- [4] S. Aramvith, C.-W. Lin, S. Roy, and M.-T. Sun, "Wireless video transport using conditional retransmission and low-delay Interleaving," *IEEE Trans. on Circuits and Systems for Video Technology*, Vol. 12, No. 6, June 2002.
- [5] D. G. Sachs, I. Kozintsev, and M. Yeung, "Hybrid ARQ for robust video streaming over wireless LANs," in *Proc. ITCC '01*, Las Vegas, LV, April 2001.
- [6] F. Hartanto and H. R. Srisena, "Hybrid error control mechanism for video transmission in the Wireless IP networks," in *Proc. IEEE LANMAN '99*, Sydney, Australia, Nov. 1999.
- [7] D. Aguayo, J. Bicket, S. Biswas, G. Judd, and R. Morris, "Link-level Measurements from an 802.11b Mesh Network," *SIGCOMM 2004*, Aug 2004
- [8] D. G. Sachs, I. Kozintsev, and M. Yeung, "Hybrid ARQ for robust video streaming over wireless LANs," in *Proc. ITCC '01*, Las Vegas, LV, April 2001.
- [9] S. Park, H. Yoon and J. Kim, "Network-Adaptive HD MPEG-2 Video Streaming with Cross-layered Channel Monitoring in WLAN," in *Proc International Packet Video Workshop (PV2006)*, Hangzhou, China, April 2006