

# 저장매체 스트리밍에서 substream에 기초한 비순차 패킷 스케줄링

정희원 최수정\*, 안희준\*\*, 강상혁\*\*\*

## Substream-based out-of-sequence packet scheduling for streaming stored media

Sujeong Choi\*, Heejune Ahn\*\*, Sang Hyuk Kang\*\*\* *Regular Members*

### 요 약

이 논문에서는 스트리밍 미디어에 대해 속도 대비 왜곡을 최적화하는 패킷 스케줄링 알고리즘을 제안한다. 수신단은 패킷을 받으면 ACK를 보내는데, 주기적으로 관측한 채널 throughput을 함께 보낸다. 비디오 패킷의 중요도는 움직임 벡터, 텍스처 필드, 시간계층화를 고려하여 GOP 내에서의 위치에 의해 결정되며, 이것을 이용하여 서브스트림을 정의한다. 서버는 수신단으로부터 온 응답 정보와 비디오의 통계적 특징을 사용하여 임의의 시점에서 전송하고자 하는 최적의 서브스트림을 결정한다. 이를 위해 스트리밍 시스템을 큐잉 시스템으로 모델링하고 각 서브스트림에 있는 프레임의 복호화 실패율을 체감대역폭을 이용하여 계산하고 그 시점에 전송할 최적의 서브스트림을 선택한다. 수신단이 주기적으로 기록하고 전송한 채널정보에 대해, 송신단은 최적의 서브스트림을 새로이 계산하는데, 그때까지 서브스트림의 데이터는 Earliest Deadline First(EDF)방식으로 전송된다. 이 때, 전송 순서는 원래의 재생 순서와 다르게 된다. 그리고 실제 비디오 데이터를 사용한 실험에서, 제안한 스케줄링이 기존의 순차적인 전송보다 성능이 좋다는 것을 보여준다.

Key Words : media streaming, rate-distortion, effective bandwidth, scalability

### ABSTRACT

We propose a packet scheduling algorithms for streaming media. We assume that the receiver periodically reports back the channel throughput. From the original video data, the importance level of a video packet is determined by its relative position within its group of pictures, taking into account the motion-texture discrimination and temporal scalability. Thus, we generate a number of nested substreams. Using feedback information from the receiver and statistical characteristics of the video, we model the streaming system as a queueing system, compute the run-time decoding failure probability of a frame in each substream based on effective bandwidth approach, and determine the optimum substream to be sent at that moment in time. Since the optimum substream is updated periodically, the resulting sending order is different from the original playback order. From experiments with real video data, we show that our proposed scheduling scheme outperforms the conventional sequential sending scheme.

\* 서울시립대학교 전자전기컴퓨터공학부(cris@uos.ac.kr),

\*\* 서울산업대학교 제어계측공학과(hcejune@snut.ac.kr)

\*\*\* 서울시립대학교 전자전기컴퓨터공학부(shkang@uos.ac.kr)

논문번호040110-0308, 접수일자2004년 3월 8일

## I. 서론

인터넷에서 저장 미디어 스트리밍의 응용은 날로 그 인기를 더해가고 있다. 저장 미디어 스트리밍 시스템에서 서버는 부호화된 미디어 데이터를 저장하였다가 클라이언트의 요구에 따라 데이터를 전송하며, 재생하기 전에는 초기지연을 갖고 시작한다. 이상적인 경우, 일단 클라이언트가 재생을 시작하면 끝날 때까지 화면의 끊김이 없이 계속되어야 한다. 미디어의 부호화 및 저장 시에는 채널 상태에 관한 정보가 없으므로, 차후 전송시에 채널의 상태에 따라 화질이 급격하게 나빠지는 일이 없도록 탄력적으로 부호화가 되어야 하고[1], 쌍방향 어플리케이션과 비교했을 때 저장 미디어 스트리밍은 5-20 초 정도의 비교적 큰 초기버퍼를 갖는[2] 특징들이 있는데, 이러한 특징을 이용하여 속도 대비 왜곡(rate-distortion) 최적화 방법과 데이터 패킷 스케줄링을 개발할 수 있다.

부호화된 미디어 데이터는 다른 중요도를 가진 패킷들로 구성되어 있다. 비디오 데이터의 패킷은 I, P 혹은 B 프레임, 그리고 움직임 벡터이나 텍스처 필드에 따라 여러 종류로 나누어진다. 또한, 비디오 패킷들 사이에는 관련성이 있으며, 손실되거나 지연된 데이터 패킷은 비디오 화질에 영향을 끼치게 된다. 그러므로, 속도 대비 왜곡을 최적화하여 여러 가지유형의 데이터 패킷에 UEP(unequal error protection)을 적용하는 것이 바람직하다.

그동안 미디어 전송에서 UEP에 대한 많은 연구가 있었는데, 그 한가지 방법은 패킷의 중요도에 따라 각각 다른 FEC 코드를 적용하는 것이다 [3],[4],[5],[6]. 그러나, 저장 미디어 스트리밍 어플리케이션에서는, 전이중(full-duplex) 채널 및 수신단의 버퍼링을 가정하여 응용 ARQ을 사용하거나, FEC과 결합된 복합ARQ를 사용하는[7] 경향이 있다.

미디어 패킷 스케줄링에 속도 대비 왜곡 최적화를 이용한 연구는 [8]와 그 참고문헌에서 찾아 볼 수 있다. [8]에서, 스케줄링 윈도우에 있는 패킷들에 반복적으로 수행하여 최적의 스케줄링 방법을 찾아 내는데 마코프 결정 과정을 사용하였다. 그러나, 계산상 복잡하기 때문에 실시간으로 구현할 때는 그 성능을 보장하지 못할 수도 있다. 그래서, [9]에서는 복잡한 계산을 줄이기 위해 속도 대비 왜곡 최적화에 필요한 경험적인(heuristic) 식이 제시되었다.

이 논문에서는, 손실이 발생하는 네트워크에서 속도 대비 왜곡 최적화 방법을 바탕으로, 역채널을 이용하는 것이 가능하고 비디오 클립 원래의 재생 시간은 불변이라고 가정하고, 저장 미디어 스트리밍의 패킷 스케줄링을 다루도록 한다. 우선, 가장 중요한 패킷에 그 다음으로 중요한 패킷을 삽입하는 방법으로  $H$  개의 서브스트림을 생성한다. 그리고 나서, 전체 비디오 데이터를 우선순위에 따라 나누기 위해 최적의 중요도  $\theta$ 를 결정한다. 높은 우선순위에 있는 것은 중요도  $0, \dots, \theta$ 를 갖는 서브스트림, 낮은 우선순위의 서브스트림은 중요도  $\theta+1, \dots, H-1$ 를 갖게 된다. 여기서 제안된 스케줄링 알고리즘은 높은 순위의 서브스트림부터, 같은 우선순위인 경우에는 EDF(earliest-deadline-first) 방식으로, 같은 우선순위와 같은 데드라인을 갖는 패킷은 중요도에 따라 전송하고, 데드라인을 놓친 패킷은 전송하지 않는다.

논문의 전반부에서는 I 와 P 프레임의 크기가 고정되었다고 가정하고, 에러가 없는 제한 대역폭 채널에서 왜곡을 분석한다. 그 다음에, 우리는 VBR로 부호화한 비디오와 혼잡이나 채널 에러로 인한 채널 대역폭 변화를 고려한다. 패킷 스케줄링 알고리즘을 위해, SRD(short range dependencies)와 LRD(long range dependencies), 두 가지 특징을 모두 가진 비디오 스트림의 다중화 트래픽을 모델링한[11] 체감 대역폭[10]을 사용한다. 큐잉 모델을 스트리밍 시스템에 적용하기 위해 수신 버퍼의 상태, 시변 채널의 throughput 통계치, GOP의 크기에 대한 autocovariance 등을 고려한 체감대역폭을 이용하여 프레임 복호화 실패율을 계산하였다. 손실이 심한 채널에서 화질의 저하가 완만하게 일어나도록 하기 위해, MPEG-4 데이터 분할과 시간 계층화에 기초한 MTD(motion-texture discrimination)에 우리가 제안한 알고리즘을 적용하였다. 실제 비디오 데이터의 유무선 채널환경에서의 실험을 통해 PSNR 관점에서 그 성능을 평가하였다.

이 논문은 다음과 같이 구성되어 있다. 2장에서는, 속도 대비 왜곡의 최적화 문제를 제한대역폭을 가진, 에러가 없는 채널에 대해 분석하였다. 3장에서는, 체감대역폭의 개념을 기초로 하여 패킷 스케줄링을 제안하기 위해 비디오 비트 속도와 채널대역폭 변화를 고려하였다. 4장에서는, 제안한 방법을 실제 유무선 시나리오에서 평가하였고, 마지막 5장에서는 결론을 내리도록 한다.

## II. 속도 대비 왜곡 분석

이 장에서는 고정된 채널 대역폭에서 스트리밍 왜곡을 분석한다. 그리고, 비디오는 계층화나 MTD 없이 부호화되었음을 가정한다.

먼저, 프레임 전송속도  $f$  frame/sec를 갖는 비디오 프레임 시퀀스  $F_n(n=0,1,2,\dots)$ 을 가정하자. 그리고 수신단은  $t=0$ 초에  $F_0$ 을 재생하기 시작했다고 가정하자. 그러면  $F_n$ 의 데드라인은  $t=n/f$  초가 된다. 비디오 시퀀스는 여러 개의 GOP로 구성되며, 하나의 GOP는 1개의 I 프레임과  $N-1$ 개의 P 프레임으로 구성되어 있다고 가정하자.  $k=(n \bmod N)$ 라고 하면,  $k=0$  일 때  $F_n$ 은 I 프레임으로,  $k \neq 0$ 일 때 Pk 프레임으로 얻게 된다. GOP에서,  $j$  번째 P프레임을 Pj ( $j=1,2,\dots,N-1$ )로 표기한다. GOP에 있는 데이터의 중요도를  $H$  단계로 나누는데, 가장 중요한 데이터에  $h=0$ , 가장 덜 중요한 데이터에  $h=H-1$ 을 할당한다. 따라서, 이 장에서는  $h=0$ 를 I프레임에,  $h=j$ 를 Pj ( $j=1,2,\dots,N-1$ )에 할당을 하였다.

$S^{(I)}$ 와  $S^{(P)}$ 를 I, P 프레임의 고정된 크기로 하겠다. 그리고 프레임 전송 속도  $f$ 를  $N$  frame/sec로 가정한다. 즉, GOP 하나의 지속 시간이 1초이다. 그러면, 평균 비디오 속도는  $S^{(I)} + (N-1)S^{(P)}$  bits/sec이다.  $C$  bits/sec의 고정대역폭을 갖고 에러가 없는 채널을 생각해 보자. 여기에서  $C$ 는

$$\frac{S^{(I)} + (N-1)S^{(P)}}{2} < C < S^{(I)} + (N-1)S^{(P)} \quad (1)$$

을 만족한다고 가정하자. 에러가 없는 채널을 가정한 것은 송신단이 패킷을 재전송할 필요가 없고 결국, 채널 throughput이 채널 대역폭  $C$ 와 같다는 뜻이다.

프레임  $F_n$ 에 대하여 그 GOP의  $\{F_{n \cdot (n \bmod N)}, F_{n \cdot (n \bmod N) + 1}, \dots, F_{n-1}\}$ 가 성공적으로 복호화되었고,  $F_n$ 의 데이터가 데드라인 이내에 수신단에 도착했을 때만,  $F_n$ 이

$t=n/f$  초에 복호화에 성공했다고 하겠다.

I 프레임과  $P_1, P_2, \dots, P_{N-1}$  프레임의 복호화 성공율을  $g_0, g_1, g_2, \dots, g_{N-1}$ 이라 하자. 이때 스케줄링 알고리즘의 목적은 왜곡을 최소화시키는  $g$ 의 최적값을 결정하는 것이다. 또한, 복호화에 성공한 Pj프레임들의 개수는 이전 프레임 Pj-1이 복호화에 성공한 개수보다 크면 안된다. 따라서,

$$1 \geq g_0 \geq g_1 \geq \dots \geq g_{N-1} \geq 0 \quad (2)$$

가 된다.

이 장에서, 데드라인을 놓친 프레임에 대해서는 그 GOP 나머지를 버리는 방법과 RTT(round trip time)=0을 가정하고 있다. 그러므로, 제때 도착하지 못한 패킷은 더 이상 쓸모가 없게 되어, goodput과 throughput이 같게 된다(goodput은 중복된 패킷이나 쓸모없는 패킷을 제외하고 수신측에 전달된 트래픽 전송속도로 정의된다). 이때, throughput은 채널대역폭과 같으므로,

$$S^{(I)}g_0 + S^{(P)} \sum_{j=1}^{N-1} g_j = C \quad (3)$$

이 성립한다.

이제 복호화에 성공한 프레임을 복사하여 실패한 프레임에 대한 에러를 숨기는 방법을 고려한다. 예를 들어, P4 프레임이 손실되었다고 하자. 만약 그 GOP의 I, P1, P2, P3은 성공적으로 복호화되었다면, 복호기는 P4의 재생 시간에 P3을 재생하고 왜곡은 작아진다. 또한, P4의 복호화 실패가 그 GOP의 P1의 손실로 인한 것이라면, 복호기는 P4의 재생 시간에 I 프레임을 재생하고 왜곡은 훨씬 더 크게 된다. 결과적으로, 복호화에 실패한 프레임의 왜곡은 그 프레임과 실패의 원인이 되는 프레임 간의 시간차와 관련된 함수라 할 수 있다.

$\delta_j(j=1,2,\dots,N)$ 를  $F_{n-j+1}$ 의 손실로 인해  $F_n$ 이 복호화에 실패함으로써 발생한 왜곡이라고 하자. 즉, 수신단이  $F_n$ 의 재생시간에  $F_{n-j}$ 를 재생함으로써 발생하는 왜곡을 말한다. 예를 들어, P2가 성공적으로 복호화되고 그 다음에 오는 P3가 손실되었다면, P3의 왜곡은  $\delta_1$ 이 되고 P4의 왜곡은  $\delta_2$ 가 된다. 특히,  $\delta_j$ 을

$$\delta_j = \text{MSE}(F_n, F_{n+j}), \quad j=1,2,\dots,N \quad (4)$$

로 정의하는데, 여기에서  $\text{MSE}(F_k, F_{k+j})$ 는

$F_k$ 와  $F_{k+j}$  사이의 평균제곱오차이며, 다음을 만족한다.

$$0 < \delta_1 < \delta_2 < \dots < \delta_N \quad (5)$$

송신단에서 I 프레임이 버리는 경우는 이전 GOP의 모든 프레임이 성공적으로 전송되었을 때만 가능하다고 가정하면 I 프레임의 손실로 인한 왜곡은  $\delta_1$ 이다.

Pj 프레임의 복호화 실패율  $e_j$ 는

$$e_j = e_0 + (e_1 - e_0) + (e_2 - e_1) + \dots + (e_j - e_{j-1}) \quad (6)$$

로 표현되며, 첫 번째 항은 I 프레임의 복호화 실패를 의미하고 두 번째 항은 I 프레임이 성공적으로 복호화 되었다는 조건 하에 P1 프레임이 복호화에 실패한 경우를, 세 번째 항은 P1 프레임이 성공적으로 복호화되고 P2 프레임이 실패한 경우를 나타내고 있다. 만약 Pj 프레임이 I 프레임의 손실 때문에 복호화가 실패하였다면, 이전 GOP의 P<sub>N-1</sub> 프레임을 재생하기 때문에 Pj 프레임의 왜곡은  $\delta_{j+1}$ 가 된다. 비슷한 방법으로, Pj 프레임이 실패한 것이 그 GOP의 P1 프레임의 실패 때문이라면 Pj 프레임의 왜곡은  $\delta_j$ 가 된다. 간단히 정리해 보면, I프레임의 왜곡은  $e_0\delta_1$ 가 되고 Pj 프레임의 왜곡은

$$e_0\delta_{j+1} + (e_1 - e_0)\delta_j + (e_2 - e_1)\delta_{j-1} + \dots + (e_j - e_{j-1})\delta_1 \quad (7)$$

가 된다. 그러므로, GOP의 왜곡은 아래와 같이 나타난다.

$$\begin{aligned} D &= e_0\delta_1 + e_0\delta_2 + (e_1 - e_0)\delta_1 + e_0\delta_3 + (e_1 - e_0)\delta_2 + (e_2 - e_1)\delta_1 + \dots + e_0\delta_N + (e_1 - e_0)\delta_{N-1} + \dots + (e_{N-1} - e_{N-2})\delta_1 \\ &= \sum_{j=0}^{N-1} e_j\delta_{N-j} \\ &= \sum_{j=1}^N \delta_j - \sum_{j=0}^{N-1} g_j\delta_{N-j} \end{aligned} \quad (8)$$

이제 남은 과정은, (2)와 (3)을 이용하여 D를 최소

로 만드는  $g_j$  값들을 찾는 것이다. 최적화된  $g_j$ 은  $C, S^{(I)}, S^{(P)}, N, \delta_j$ 의 함수인데, 부록 A를 참조하면, 최적의 조건식은 아래의 (a), (b) 중 하나를 만족한다.

$$\begin{aligned} (a) \quad g_0 &= g_1 = \dots = g_L = 1. \\ g_{L+1} &= \frac{C - S^{(I)} - LS^{(P)}}{L}, \quad (9) \\ g_{L+2} &= \dots = g_{N-1} = 0 \end{aligned}$$

여기에서

$$L = \lfloor \frac{C - S^{(I)}}{S^{(P)}} \rfloor \quad (10)$$

이고,

$$D = (1 - \frac{C - S^{(I)} - LS^{(P)}}{L})\delta_{N-L-1} + \sum_{j=1}^{N-L-2} \delta_j \quad (11)$$

이다.

(b) 아래 (N-L-1) 가지 중 하나이다.

$$\begin{aligned} g_0 &= g_1 = \dots = g_M = \frac{C}{S^{(I)} + MS^{(P)}}, \\ g_{M+1} &= \dots = g_{N-1} = 0, \quad M = L+1, L+2, \dots, N-1 \end{aligned} \quad (12)$$

여기에서  $\lfloor x \rfloor$ 는 x보다 작거나 같은 정수 중 가장 큰 값이다. 따라서,

$$D = \sum_{j=1}^N \delta_j - \frac{C}{S^{(I)} + MS^{(P)}} \sum_{j=0}^M \delta_{N-j} \quad (13)$$

이다.

이 때, (9)와 (12)의 의미를 살펴해보도록 하자. (9)는 I 프레임과 P<sub>1</sub>부터 P<sub>L</sub> 프레임까지는 모두 확률 1로 전송하며 P<sub>L+1</sub>은  $(\frac{C - S^{(I)} - LS^{(P)}}{L})$ 의 확률로 전송하고, 나머지 P<sub>L+2</sub> 프레임부터 P<sub>N-1</sub> 프레임은 버린다. 여기에서 L을 protection scope라 한다. (12)에서는 I 프레임과 P<sub>1</sub> 프레임부터 P<sub>M</sub> 프레임까

지는  $\frac{C}{S^{(D)} + MS^{(P)}}$ 의 확률로 전송하며,  $P_{M+1}$  프레임부터  $P_{N-1}$  프레임까지는 버리도록 하는데, 여기에서는  $M$ 이 protection scope이다. 이때,  $M > L$ 임을 주의한다.

$C, S^{(D)}, S^{(P)}, N, \delta_j$ 가 주어져 있을 때, 왜곡을 최소로 하는 최적의 조건을 아래와 같이 선택한다. (11)과 (13)을 결합하여,  $D_k (k = L, L+1, \dots, N-1)$ 을 protection scope가  $k$ 인 GOP의 왜곡이라고 하자. 즉,

$$D_k = \begin{cases} \sum_{j=1}^{N-k-2} \delta_j + (1 - C + S^{(D)} + kS^{(P)})\delta_{N-k-1}, & k=L, \\ \sum_{j=1}^N \delta_j - \frac{C}{S^{(D)} + kS^{(P)}} \sum_{j=1}^k \delta_{N-j}, & k=L+1, L+2, \dots, N-1. \end{cases} \quad (14)$$

이제 왜곡을 최소화시키는 최적값  $k_{opt}$ 을 다음과 같이 계산하도록 한다.

$$k_{opt} = \arg \min_k D_k, \quad k = L, L+1, \dots, N-1 \quad (15)$$

만약  $k_{opt} = L$ 라면 최적의 조건은 (9)와 같고, 그렇지 않은 경우에는

$$\begin{aligned} g_0 = g_1 = \dots = g_{k_{opt}} &= \frac{C}{S^{(D)} + k_{opt}S^{(P)}}, \\ g_{k_{opt}+1} = \dots = g_{N-1} &= 0 \end{aligned} \quad (16)$$

가 된다. 이제부터는 (15)의  $k_{opt}$ 을 최적 중요도라고 하겠다. 이제는  $k_{opt}$ 을 기준으로 프레임을 두 부류로 나누는데, 높은 우선순위에는 I 프레임부터  $P_{k_{opt}}$ 까지를, 낮은 우선순위에는 그 나머지인  $P_{k_{opt}+1}, \dots, P_{N-1}$ 로 한다. 높은 순위의 모든 패킷은 낮은 순위의 패킷보다 먼저 전송이 되도록 설정하고, 우선순위가 같을 때는 EDF에 따라 전송이 된다. I와 P프레임의 크기를 고정된 값으로 가정하고 에러가 없는 제한대역폭 채널을 가정하였을 때, 높은 우선순위의 프레임은 어느 정도는 확실한 복호화 성공율을 보장받는다. 특히, (9) 조건에서는 I 프레임부터  $P_{k_{opt}} (k_{opt} = L)$ 까지의 높은 우선순위의 패킷에 복호화 성공률 1을, 나머지 낮은 우

선순위의 패킷에 낮은 성공률을 각각 할당하였다. 또한 (16) 조건에서는, 높은 우선순위 범위를 I 프레임부터  $P_{k_{opt}} (k_{opt} > L)$ 까지 확장시켰으며 감소된 복호화 성공률

$$\frac{C}{S^{(D)} + k_{opt}S^{(P)}} \text{을 가지고 있다.}$$

여러 가지 비디오 데이터틀 가지고 실험을 해 본 결과,  $k_{opt}$ 는 I 프레임과 P 프레임의 크기의 비율에 대하여 달라지지만, 대체적으로  $k_{opt} \approx L$ 의 결과를 얻었다.

### III. 체감대역폭을 기초로 한 패킷 스케줄링

앞장에서, throughput이 채널 대역폭과 같고 RTT가 0인 에러가 없는 채널과 고정된 크기의 I와 P 프레임을 가정했을 때 속도 대비 왜곡 최적화 문제를 분석하였다. 그러나, 실제 상황에서는 비디오 속도의 변화와 시간에 따라 변하는 수신단 버퍼의 큐 길이, 혼잡이나 비트에러에 따라 변하는 채널 대역폭까지 고려해야 한다. 이런 경우 채널 대역폭, throughput, goodput은 같을 수 없다. 이번 장에서는 실시간으로 구현가능한 스케줄링 알고리즘을 개발하기 위해 위의 실질적인 조건들도 고려한다. 앞장에서 나온 결과들을 바탕으로 하여, 최소의 왜곡을 위한 최적의 조건으로 대체로 식 (9)를 사용한다.

#### 1. 계층화와 중요도

중요도 문제는 계층화와 밀접한 관련이 있다. 계층적 부호화방법은 채널 변동을 점차적으로 감소시킨다[13],[14]. 계층화를 이용한 기법은 SNR, 공간적, 시간적 기법들이 있는데[1], 이 논문에서는, MTD(Motion Texture Discrimination)와 시간 계층화를 사용하도록 한다. MTD는 SNR 계층화의 한 형태로 볼 수 있다. 계층화를 활용한 방법은 전체적인 데이터 크기를 증가시키므로, 속도 대비 왜곡이 개선되면 데이터 크기는 필수적으로 증가하게 된다.

프레임에 있는 데이터는 MTD를 하지 않을 때는 같은 중요도를 가지며, MTD의 경우에 P프레임의 데이터는 움직임 벡터인지 텍스처 필드인지에 따라 두 단계의 중요도로 나누어진다. 그림 1에서는 의존관계를 화살표로 나타내어 계층적 부호화의 구조를 표현하였다. 각 계층화 방법에서는, 더 높은 중요도를 가진 것은 기저층에 배치하고, 같은 계층에서는

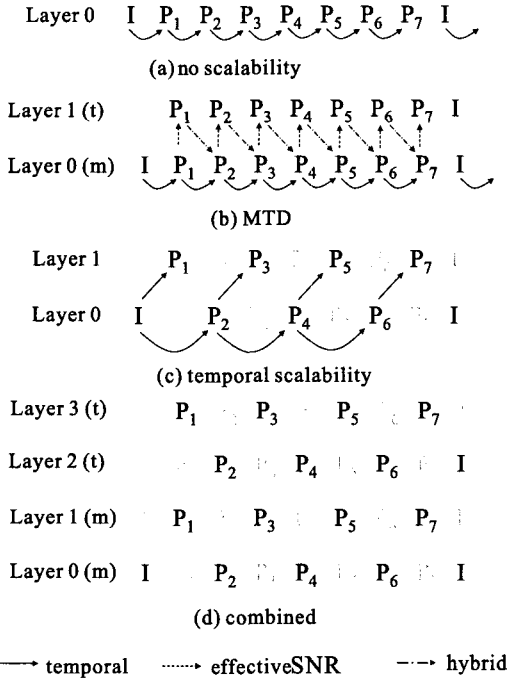
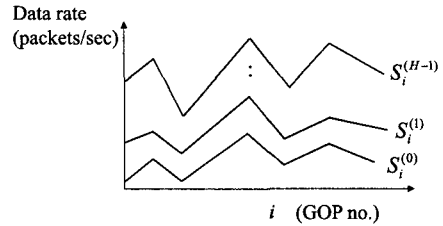
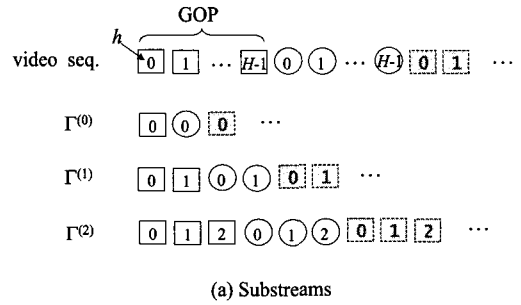


그림 1. 계층화와 중요도

프레임 번호가 작을수록, 같은 프레임에서는 먼저 재생해야 할 데이터일수록 높은 중요도를 갖고 있다.

그림 1의 (a)는 순차적으로 패킷화된 프레임들의 의존관계를 나타낸 그래프이다. 이 경우에 하나의 GOP에 있는  $N$  개의 프레임은 총  $H = N$ 개의 중요도를 갖게 된다.

움직임 보상을 하면 한 프레임의 부호화된 데이터는 움직임 벡터와 텍스처 필드로 나누어지게 된다. 여기에서 움직임 벡터가 텍스처 필드보다 데이터의 크기가 작으면서도 중요한 정보를 더 많이 포함하고 있음을 유의하자. 실험결과를 예로 들면, 토크쇼에서는 움직임 벡터가 텍스처보다 27%까지 작으며, 예고편 비디오인 경우에는 15%까지 작게 나타났다. MPEG-4 는 데이터 분할을 통해 MTD를 지원하고 있는데, 전체적인 데이터 크기는 거의 증가시키지 않고 있음을 주목하자. 예를 들어, MPEG-4 데이터 분할을 했을때, 전체 데이터 크기는 여러가지 양자화 방법을 거쳐도 단지 0.2%내지 0.6% 정도만 증가하였다. 그림 1의(b)에서는 MTD의 계층구조를 나타내었는데 I 프레임과 P 프레임의 움직임 벡터는 기저층에, P 프레임의 텍스처필드는 향상층에 놓았다. 이때, 소문자 'm'과 't'는 움직임(motion)과 텍스처(texture)를 각각 나타내고 있다.



(b) Data rates of substreams

그림 2. 서브스트림

우리는 텍스처 필드보다는 움직임 벡터에 더 높은 중요도를 할당하였으며, 특히, I 프레임과 P<sub>1</sub>에서 P<sub>N-1</sub> 프레임까지의 움직임벡터에 중요도  $h$ 를 0부터  $N-1$ 까지 할당하였다. 그리고, P<sub>1</sub>에서 P<sub>N-1</sub> 프레임까지의 텍스처 필드에 중요도  $N$ 에서  $2N-2$ 를 할당하였다. 따라서, 총 중요도  $H$ 는  $2N-1$ 이 된다.

시간 계층화는 시각적 관점에서나 PSNR 같은 객관적인 수치상으로도 그 성능에 있어서 개선을 가져왔다. 프레임 속도에 있어서 완만한 저하를 이끌어 낸 것으로는 [13]에서의 방식이 가장 좋은 결과를 얻어냈다. 그러나, 채널대역폭이 제한된 실제 상황에서는, 패킷스케줄링으로 인해 왜곡이 개선되면 계층적 부호화의 속도가 증가하는 것은 당연하다. 이 논문에서 우리는, P 프레임과 그것을 참조한 프레임들 사이에 시간차를 최소화함으로써, 결국 시간 계층화로 인한 비트 속도 증가를 최소화하기 위하여 그림 1(c)에 있는 두 계층에 제한한 스케줄링 알고리즘을 적용하였다. 이 경우, 총 중요도  $H$ 는 그림 1(a)의 경우와 같은  $N$ 이 되고, I 프레임과 짝수번 P 프레임이 기저층을 이루고, 홀수번 P프레임이 향상층을 형성한다. 가장 높은 중요도( $h$ ) 0은 I 프레임에, 다음 중요도 1은 P<sub>2</sub>에, 2는 P<sub>4</sub>에 할당되는 방식으로 각각 짝수번 P프레임에 중요도

가 부여된다. 그리고, 홀수번 P 프레임의 경우,  $h = \frac{N}{2}$  은 P<sub>1</sub>에,  $h = \frac{N}{2} + 1$  은 P<sub>3</sub>에 대응시키는 방법으로 중요도가 매겨진다.

시간, 공간 계층화를 결합한 방식은 [14]에서 논의되었는데, 그림 1(d)에서는 MTD와 2계층 시간 계층화의 결합을 고려하여 나타내었다. 이 경우에, 중요도는 총  $2N-1$  단계인데, 중요도 0부터  $N-1$ 까지는 I 프레임과 그림 1(b)의 움직임에, 중요도  $N$ 부터  $2N-2$ 까지는 텍스처 필드에 부여되었다. 또한, 움직임 벡터와 텍스처에서는, 짝수번 P 프레임이 홀수번 P 프레임보다 더 높은 중요도를 가지고 있다.

2. 비디오 데이터의 체감 대역폭

GOP의 데이터는 중요도에 따라 H 단계로 나누어지며, 전송할 때에는 고정된 크기의 패킷으로 나누어진다. 그림 2(a)에서는 H 단계의 중요도를 이용하여, 중요한 데이터에 덜 중요한 데이터를 끼워 넣는 방법으로 H 개의 서브스트림을 생성하였다. 넓은 의미로 안정되어 있다는 가정 하에, i 번째 GOP에서 중요도가 h인 데이터의 크기를 패킷단위로 나타낸 수열을 랜덤프로세스  $\{X_i^{(h)}; i=0, 1, \dots\}$ 라 정의하자. 또한  $S_i^{(\theta)}$  ( $\theta=0, 1, \dots, H-1$ )을 그림 2(b)에서처럼, 아래와 같이 정의한다.

$$S_i^{(\theta)} = \sum_{h=0}^{\theta} X_i^{(h)} \tag{17}$$

그리고,  $S_i^{(\theta)}$ 의 autocovariance는 아래와 같이 정의된다.

$$v^{(\theta)}[k] = E[(S_i^{(\theta)} - E[S^{(\theta)}])(S_{i+k}^{(\theta)} - E[S^{(\theta)}])], k=0, 1, \dots. \tag{18}$$

$\theta$ 에 대해서는, 높은 우선순위는 중요도 0부터  $\theta$ 까지, 낮은 우선순위는 중요도  $\theta+1$ 부터  $H-1$ 까지로 나눈다. 비디오 시퀀스 전체에 대해, 각 GOP에서 높은 우선순위의 데이터로 구성된 데이터 스트림을  $\Gamma^{(\theta)}$ 라고 하자. GOP 지속시간이 1초라는 가정 하에,  $\Gamma^{(\theta)}$ 에서 i 번째 GOP의 데이터의 크기

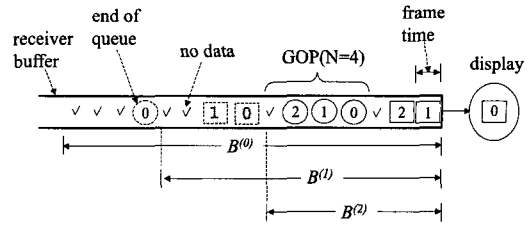


그림 3. 수신단 버퍼

는  $S_i^{(\theta)}$ 이고  $\Gamma^{(\theta)}$ 의 평균 데이터 속도는  $E[S^{(\theta)}]$  packets/sec 가 된다.

앞장에서, I와 P프레임은 고정된 크기를 가지며 채널에 에러가 없는 경우를 가정했을 때, (9)에 의해 높은 순위의 스트림은  $\Gamma^{(k_{opt})}$ 의 프레임 복호화 실패율 0을 보장한다. 그러나, 이번 장에서는 비디오 전송 속도 변화를 고려하여 확률과정에 의한 비디오 전송 속도를 모델링한다. 이 경우에, 확률적으로 성능을 보장하기 위해 높은 우선순위를 차지하는 부분의 복호화 실패율이 0이 아닌 경우를 다룰 필요가 있다.

서브스트림  $\Gamma^{(\theta)}$ , 채널대역폭과 같은 출력 서비스 속도  $C$  packets/sec, 버퍼 크기  $B^{(\theta)}$ 을 갖고 있는 큐잉 모델을 제안한다. 큐잉 모델을 효과적으로 만들기 위해서는, 큐잉모델의 버퍼 크기  $B^{(\theta)}$ 을 수신단 버퍼 상태를 고려하여 신중하게 결정할 필요가 있다. 특히,  $B^{(\theta)}$ 는  $\Gamma^{(\theta)}$ 의 평균 데이터 속도와 미리 저장된 비디오 지속시간에 의해 수신단 버퍼에서 결정이 된다. 그림 3과 같이  $B^{(\theta)}$ 을 중요도 0에서  $\theta$ 까지의 데이터를 가진 GOP의 개수와 연관지어서 접근한다. 또한, 서버는 모든 패킷의 ACK를 받는 것으로 가정한다. 수신단이 k번째 GOP의 한 프레임을 복호화하고 있고, 중요도  $\theta$ 까지의 패킷에 대한 ACK를 모두 받은 것 중 가장 큰 GOP 번호가  $m_{\theta}$  ( $m_{\theta} \geq k$ ) 라고 가정하자. 평균 데이터 속도가  $E[S^{(\theta)}]$ 이고  $(m_{\theta} - k)$ 초 동안의 버퍼크기를 패킷 단위로 나타내면,  $B^{(\theta)}$ 는 아래와 같다.

$$B^{(\theta)} = (m_{\theta} - k)E[S^{(\theta)}][\text{packets}] \tag{19}$$

$B^{(\theta)}$ 을 시간 개념으로 표현하기 위해, 수신단 버퍼

에 도착하지 않은 데이터 프레임은  $\vee$ 표시로 나타냈는데, 그 프레임이 결국 테드라인 내에 도착하지 못하면 수신단은 마지막으로 복호화에 성공한 프레임을 재생한다. 위 식에서  $E[S^{(\theta)}]$ 은 독립적인 상수인 반면,  $m_\theta$ 와  $k$ 는 시간에 따라 변하는 변수들이다.

평균 데이터 속도  $E[S^{(\theta)}]$ 이 채널대역폭  $C$ 보다 작도록  $\Gamma^{(\theta)}$ 가 주어졌을 때, 입력 스트림  $\Gamma^{(\theta)}$ 는 비디오 속도의 변화 때문에 패킷 손실이 발생한다. 이제  $W_{eff}^{(\theta)}$ 를  $\Gamma^{(\theta)}$ 의 체감대역폭이라고 하자. 크기가  $K$ 인 유한 버퍼와, 상수인 서비스 속도를 가진 큐잉 시스템에서의 입력 스트림이  $\Gamma^{(\theta)}$ 라 하면,  $\Gamma^{(\theta)}$ 의 데이터 속도 프로세스  $\{S_i^{(\theta)}; i=0, 1, \dots\}$ 의 체감대역폭은  $\varepsilon$  보다 작은 손실율을 보장하기 위한 큐의 최소 서비스 속도로 정의된다. [10]을 참고하면, 체감대역폭은 아래와 같다.

$$W_{eff}^{(\theta)} = E[S^{(\theta)}] + \frac{-\log(\varepsilon)}{2K} \sum_{k=0}^{\infty} v^{(\theta)}[k] \quad (20)$$

여기서 비디오 시퀀스가 LRD 성질을 갖지 않으므로 자기공분산  $v^{(h)}[k]$ 의 유한함을 가정한다.  $\Gamma^{(\theta)}$ 의 체감대역폭은 평균 대역폭  $E[S^{(\theta)}]$ 와 비교하였을 때, 입력 프로세스의 특징, 큐잉 시스템의 버퍼 크기, 손실율에 관한 함수만큼 크다. 직관적으로 요구되는 손실율이 작아지거나 입력 프로세스의 변화가 심해지거나, 큐잉 시스템의 버퍼 크기가 감소하면 그 차는 점점 커짐을 알 수 있다.

위에서 언급한 체감대역폭의 개념을 적용하면,  $W_{eff}^{(\theta)}$ 을 채널 throughput  $C$ 로 바꾸고,  $K$ 를  $B^{(\theta)}$ 로 바꾸어서 (20)를  $\varepsilon$ 에 관한 식으로 정리하면 다음 식을 얻게 된다.

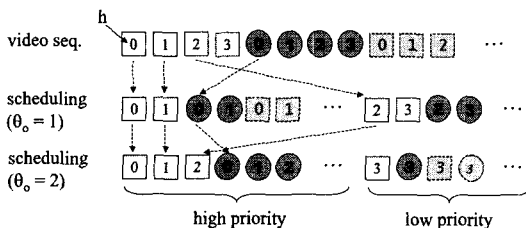


그림 4. 스케줄링

$$\varepsilon_\beta^{(\theta)} = e^\beta \quad (21)$$

이고 여기서

$$\beta = \frac{-2B^{(\theta)}(C - E[S^{(\theta)}])}{\sum_{k=0}^{\infty} v^{(\theta)}[k]} \quad (22)$$

이다.

$\Gamma^{(\theta)}$ 에 대해, 위의 등식은 패킷 손실율  $\varepsilon_\beta^{(\theta)}$ 이  $E[S^{(\theta)}]$ 와,  $B^{(\theta)}$ , autocovariance로 표현된 비디오 변화에 관한 함수임을 의미한다. 즉,  $B^{(\theta)}$ 가 작아지거나 비디오의 변화가 심해질 때, 또한  $E[S^{(\theta)}]$ 이 커지면  $\varepsilon_\beta^{(\theta)}$ 이 커진다. 또한,  $\varepsilon_\beta^{(\theta)}$ 은  $[0,1]$ 안의 값이므로  $E[S^{(\theta)}] < C$  이어야 한다.

$\Gamma^{(\theta)}$ 에서, GOP의 어떤 움직임 패킷이 손실되면 그 GOP가 복호화에 실패했다고 정의하도록 한다.  $\varepsilon_{GOP}^{(\theta)}$ 을  $\Gamma^{(\theta)}$ 에 대한 GOP 복호화 실패율이라고 하자. 부록 B를 참조하여, 확률  $\varepsilon_{GOP}^{(\theta)}$ 는 다음과 같이 근사 시킨다.

$$\varepsilon_{GOP}^{(\theta)} = \begin{cases} 1 - (1 - \varepsilon_\beta^{(\theta)})^{E[S^{(\theta)}]}, & \theta \leq N-1 \\ 1 - (1 - \varepsilon_\beta^{(\theta)})^{E[S^{(N-1)}]}, & \theta > N-1 \end{cases} \quad (23)$$

큐잉 이론에 의해서,  $\varepsilon_{GOP}^{(\theta)}$ 가  $\theta$ 에 관한 단조증가 함수임을 주목하자. 위의 GOP 복호화 실패율은 다음 장에 나오는 스케줄링 알고리즘에 있는 최적값  $\theta$ 을 결정짓기 위해 필요한 특정값과 같다.

### 3. 패킷 스케줄링

GOP 복호화 실패율이  $\gamma$  보다 작아야한다고 가정하자.  $C$ 가 주어질 때, 서버는 다음을 만족하는 경계값  $\theta_0$ 을 결정한다.

$$\theta_0 = \arg \max_\theta \{ \varepsilon_{GOP}^{(\theta)} < \gamma \text{ and } E[S^{(\theta)}] < C \} \quad (24)$$

그러므로 최적의 중요도는  $\Gamma^{(\theta)}$ 의 GOP 복호화 실패율이  $\gamma$ 보다 작고, 서브스트림의 평균 데이터 속도가 채널대역폭  $C$ 보다 작도록 하는 것들 중 가장



큰  $\theta$  값이 된다. 앞서 언급했듯이, 이 두 번째 조건은  $\varepsilon_p^{(0)}$ 가  $[0,1]$ 에 포함되기 하기 위한 필요 조건이다.  $\theta_0$ 는 정수이므로, 서버는 수신단 버퍼의 과부족을 조절하고 채널대역폭을 최대한 활용하기 위해  $\theta_0$ 와  $\theta_0+1$ 을 선택적으로 사용해야 한다. 수신단 버퍼가 가득 차면  $\theta_0+1$ 를 선택하고, 버퍼가 비어있으면  $\theta_0$ 을 선택한다. 다음의 예를 생각해 보도록 하자. 채널 throughput이 100Kbps 이고, 평균 비디오 데이터 속도가 100Kbps, 15프레임이 1 GOP이고, 프레임 속도가 15 frame/sec 임을 가정 하자. 높은 순위의 데이터는 단지 1 프레임뿐이고 서버는 1 프레임을 모두 먼저 보내도록 한다. 이때 평균 데이터 속도는 10Kbps, 총 비디오 속도는 100Kbps로 가정한다. 복호기는 1초 동안 15개의 1 프레임을 재생하지 않고 단지 하나의 1 프레임만을 재생한다. P 프레임의 재생시간에도 송신단은 1 프레임을 여전히 보내지만, 수신단은 같은 1 프레임을 재생하고 있으므로 다음 1 프레임의 재생시간 때까지 수신단의 버퍼는 고갈되지 않는다.

$\theta_0$ 을 기준으로 중요도 0부터  $\theta_0$ 까지는 높은 우선 순위,  $\theta_0+1$ 부터  $H-1$ 까지는 낮은 우선순위를 갖는다. 높은 순위를 가진 패킷은 낮은 것보다 우선적으로 전송되도록 스케줄링 되고, 같은 우선순위를 가진 패킷들은 EDF로 전송된다. 중요도 0에서 3까지의 GOP의 예를 그림 4에 나타냈다. 중요도 0은 1 프레임에, 1은 P<sub>1</sub> 프레임에 대응시켰다고 가정한다. 시간  $t_1$ 에서,  $\theta_0$ 는 1이고 중요도 0과 1을 가진 모든 프레임들은 중요도 2나 3을 가진 프레임보다 먼저 전송되도록 스케줄링 되어 있다.  $\theta_0$ 이 2로 업데이트 되는 시점  $t_1+T$  까지, 서버가 제일 앞의 세 프레임, 즉, 중요도 0과 1을 가진 사각형 모양의 두 프레임과 중요도 0을 가진 원 모양의 한 프레임을 보냈다고 가정하자.  $\theta_0$ 이 2로 업데이트 되면 그림 4의 제일 아래 그림처럼, 중요도 0,1,2를 가진 프레임들이 높은 우선순위를 차지하게 되고 3을 갖는 프레임이 낮은 우선순위를 차지하게 된다. 이제 서버는 새로운 스케줄에 따라 사각형 모양의 세 번째인 중요도 2의 프레임을 전송한다. 원래의 비디오 시퀀스와 비교를 해 보면, 이 예에서의 전송 순서는 원래의 재생순서인  $h=0,1,2,3,0,\dots$  대신에  $h=0,1,0,2,\dots$ 가 된다.

#### 4. 미디어 스트리밍 프로토콜

우리는 제안한 스케줄링 방법이 프로토콜 상에서는 응용 계층과 전송 계층 사이에 있다고 가정하였다. 각각의 수신된 패킷에 대해서는 수신단이 ACK을 보낸다. 그러면 송신단은 유용한 패킷들을 acknowledged packet, outstanding packet, non-sent packet으로 나눠서 패킷 상태를 업데이트한다. 여기에서 유용하다는 뜻은 GOP에서 이전 프레임의 전송실패로 인하여 버려지는 패킷과 테드라인을 놓쳐 버린 패킷들을 제외한, 오로지 전송대상으로 고려되는 패킷들을 의미한다. 우리는 비디오의 원래 재생 시간은 불변이라고 가정하고 있기 때문에, 서버는 현재 시간과 주어진 프레임 속도로부터 각 프레임의 테드라인을 계산할 수 있다.

우리가 제안한 방법을 구현함에 있어서, 서버는 순방향과 역방향의 전송시간의 합인 왕복시간을 측정하기 위해 각 패킷의 끝단에 시간 스탬프를 넣는다. 송신단은  $RTT_{max}$  로 표기하는 RTT 최고값을 측정한다. 순방향과 역방향 링크 사이에서의 가능한 비대칭적 트래픽 부하를 고려할 때, 송신단에서 수신단으로의 순방향 전송시간은 상한값  $RTT_{max}$  를 가정하고 있다. 그러므로, 송신단은 패킷을 전송했을 때 테드라인 이전에 패킷이 수신단에 도착했는지를 예측하기 위해  $RTT_{max}$  를 사용한다. 일단 패킷이 전송되면 그 패킷은 outstanding packet이 되고, 타이머는  $RTT_{max}$  초 후에 멈추도록 맞춰진다. 만약 타이머가 멈추기 전에 ACK가 도착하면 acknowledged packet이 된다. 만약 ACK 없이 타이머가 멈추면, 패킷은 재전송해야 하는 non-sent packet이 된다. 이때, 송신단은 재전송해야 할 패킷인지 처음 전송하는 패킷인지 구별하지 못한다.

한편, 시간적으로 변화하는 채널 대역폭을 고려하기 위하여 서버는 주기적으로 클라이언트가 측정하여 보고하는 채널의 대역폭의 평균값을 계산한다. 주기를  $T$ 라 할 때  $Y_m$ 을 시간  $t=mT$ 일 때 클라이언트가 측정, 보고한 순간 대역폭이라고 하자. 그러면 평균 채널 대역폭  $C$ 는 Exponentially weighted moving average(EWMA) 방법으로 다음과 같이 구한다.

$$C \leftarrow \alpha C + (1-\alpha) Y_m \quad (30)$$

#### IV. 성능 평가

이 논문에서 제안한 최적화 스케줄링 방법을 유무선 채널에서 실제 비디오를 전송하여 평가하였다. 토크쇼 "Larry King Live"와 영화 "Preacher's Wife"의 예고편을 사용하였다. 둘 다 QCIF 포맷이며 15 frames/sec 로 부호화되었다. 토크쇼는 1689 프레임이고 예고편 비디오는 1981 프레임으로 구성되어 있다. 비디오 데이터를 부호화, 복호화 하기 위해 Microsoft MPEG-4 Visual Reference Software version 2 FDMA 1-2.3-001213을 사용하였다. 또한, MTD와 시간 계층화와 관련된 여러 복구와 어려움점 특징을 추가하였으며, 제안한 패킷 스케줄링 알고리즘은 UDP를 사용하는 전송 계층 위에 구현되었다. 도착하는 패킷 1000Kbyte에 대해, 수신단은 시간 스탬프와 측정된 throughput를 포함하는 ACK 3 byte를 전송한다.

이제 제안된 스케줄링 방법을 체감대역폭 스케줄링(EBS)라 표기하고, 순차적 전송(SS)와 비교하였다. SS와 EBS 둘 다, 송신단이 데드라인을 놓친 프레임을 발견하면 현재 GOP의 나머지 데이터는 버린다. 그리고, 수신단의 버퍼 크기는 충분히 크다고 가정하였다. pre-roll 버퍼는 10초이다. 즉, 수신 버퍼에 비디오 10초 분량이 채워지면 재생을 시작하고, 제안한 방법에서는, 높은 순위에 요구되는 복호화 실패율  $\gamma$  는  $10^{-4}$ 를, EWMA 인수  $\alpha$  는 0.9를 사용하였다.

먼저, KAIST 와 U.C. Berkeley 간의 유선 인터넷 환경에서 토크쇼 비디오를 전송하였더니, 평균 왕복시간은 250~300 msec로 측정되었다. 실험에서는, 시간지연을 증가시키기 위해 반복적으로 비디오를 전송하였다. 또한, 평균 비디오 속도가 125.5Kbps인 토크쇼 비디오를 사용하였다. 유선 인터넷의 실제 채널대역폭은 비디오 속도보다 훨씬 크게 측정되기 때문에, 수신단에서 ON/OFF 타입으로 인위적인 에러를 적용하였다. 채널 상태는 좋거나 나쁜 경우로 나누었는데, 상태가 나쁜 때 수신된 패킷은 ACK 없이 손실되거나 훼손되고 버려진 것으로 가정한다. 각 상태의 지속시간은 지수 분포를 따르며, 채널 상태가 나쁜 구간은 평균적으로 50msec동안 지속되고, 1~15%정도를 차지한다. 그림 5에서, 채널 throughput의 함수로 본 왜곡 곡선을 다양한 방법에 대해 나타내었다. 기호를 살펴보면, TS는 시간 계층

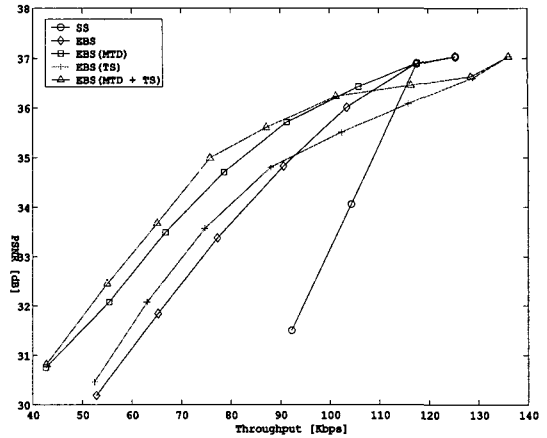


그림 5. 속도 대비 왜곡 곡선

화를 표현하고 있다. SS에 해당하는 원으로 된 곡선은 throughput가 감소하면 비디오 화질이 급격히 떨어지고 있음을 보여주고 있다. 다이아몬드로 된 커브가 이 논문에서 제안한 EBS 이며, 중요도는 GOP 안에서의 프레임 위치를 기초로 하였고 비디오 속도의 73%인 throughput 92.1Kbps에서 SS보다 4dB이상의 향상을 가져왔다. 다음으로 우리는 MTD를 적용하여 사각형 모양의 곡선을 그렸는데 throughput 92.1Kbps에서 1.3dB의 향상을 보여주고 있다. 이 두 가지 방법은 모든 채널 throughput에서 성능이 우수하게 나왔다. 삼각형과 십자 모양의 곡선은 2계층 시간 계층화를 사용하였는데 10%까지 부호화된 데이터의 양을 증가시킨 것으로 나왔다. 그러므로, 125.5Kbps의 채널 throughput으로, 모든 다른 방법들과 비교하였을 때 약 0.5dB의 감소를 보이고 있다. 95Kbps미만의 채널 대역폭에서는 TS와 결합된 MTD는 다른 모든 방법들보다 성능이 우수하며, MTD 만의 스케줄링은 TS 만의 스케줄링보다 우수하다. 요약해 보면, 125.5Kbps에서 제 외한 모든 채널 throughput에서 하나 이상의 EBS 방법이 SS보다 우수하다는 것을 알 수 있다. 그림 6에서는, 시간에 관한 함수로서의 다양한 통계값을 보여주면서 SS와 EBS를 비교하는데, throughput은 비디오 속도의 53%, 즉, 67Kbps이다. EBS에서는, 하나의 GOP에 16프레임 MTD를 사용하였고, 따라서 총 31단계의 중요도를 갖게 된다. 10초 동안의 초기 버퍼링 후에, 수신단은 시간 0에서 재생을 시작한다. 그림 6(a)를 보면 50초와 110초 부근에서 비디오 속도의 변화가 심각하며, 그림 6(b)에서는 110초에서 채널의 심각한 손실이 관찰된다. SS

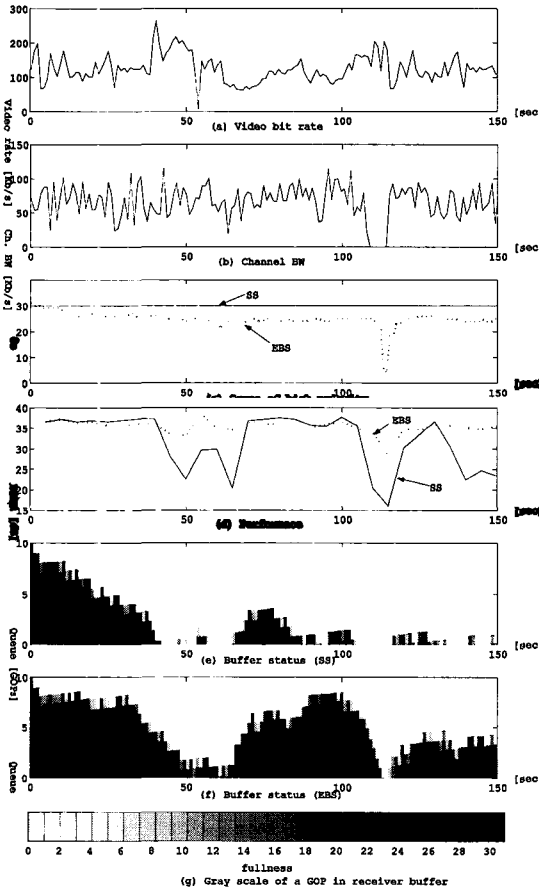


그림 6. 토크쇼 비디오의 실험결과

방법에서, 서버는 같은 우선순위인 모든 중요도의 데이터를 전송하려고 한다. 즉,  $\theta_0$ 은 그림 6(c)에서 보이는 것처럼 30으로 일정하다. 그러나, EBS를 사용하면,  $\theta_0$ 은 그림 6(c)에서처럼 채널 throughput 변화와 수신 버퍼의 큐 길이에 따라 적절하게 결정된다. 그림 6(d)에서는, PSNR 관점으로 두 방법을 비교하였는데, PSNR은 MPEG-4 부호화 전에 계산된 값이며 5초마다 기록되도록 하였다. SS 방법은 속도가 지속적으로 높거나 채널 애러가 심할 때 급격한 감소를 보여주고 있다. 어떤 부분에서는, 중요한 데이터를 먼저 보내 대역폭을 확보하고자 하는 스케줄링 방법보다 SS가 더 좋은 PSNR을 보여준다. 그러나, 전반적으로는, 제안한 방법이 심하게 왜곡된 부분을 완회시키고 PSNR에서 완만한 변화를 이끌어냄으로써 SS보다 성능이 우수하다.

그림 6(e)와 (f)에서는 시간에 관한 함수로서의 수신단 버퍼 상태를 매초마다 나타냈고, 세로축은

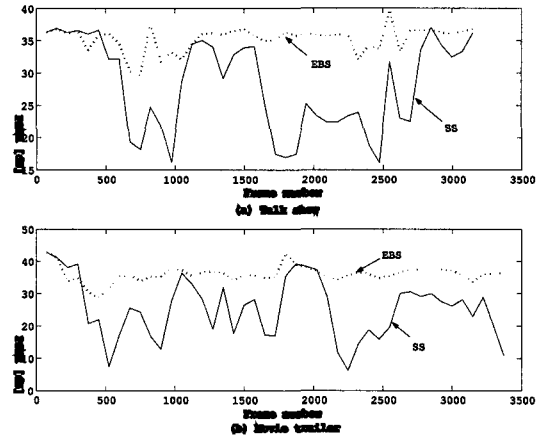


그림 7. 무선 랜에서의 결과

수신단 버퍼에 최소한 I 프레임이라도 갖고 있는 GOP의 개수를 뜻한다. 그리고, GOP에 있는 프레임의 중요도를 흑백의 여러 단계로 나타냈는데, 수신 버퍼에서 중요도 0인 데이터만 갖고 있는 GOP는 흰 색으로, 수신 버퍼에 가장 낮은 중요도의 프레임까지 모두를 가진 GOP는 검은 색으로, 그림 6(g)에 있는 표에 따라 GOP가 보유하고 있는 프레임의 중요도를 표현하였다. 새로운 패킷이 수신단에 도착하면 GOP 번호는 커지고 매  $N/f=16/15$ 초마다 하나씩 감소한다. 여기에서,  $N/f$ 는 이 실험에서 GOP 지속시간을 뜻한다. 처음에 수신단은 버퍼에 10초 정도의 데이터에 해당하는 150개의 프레임, 즉 대략 9개의 GOP를 가지고 있다. SS는 모든 중요도의 프레임을 전송하기 때문에 중요도 단계가 큰, 검은 GOP로 수신단 버퍼를 채우고, 결국 초기 pre-roll 버퍼는 빨리 고갈된다. 한편, EBS는 그림 6(f)에 있는 회색의 GOP처럼 중요도의 최적 범위를 가지고 수신버퍼에서 큐 길이를 조절하게 된다. 우리가 제안한 방법은 50초 부근에서 효과적으로 비디오 변동을 줄이고 110초 부근에서 심각한 채널 변동을 줄이는 것으로 나타났다.

우리는 이제 무선 채널을 고려하도록 한다. 그림 7에서는 IEEE 802.11 무선 랜에서 토크쇼와 예고편 비디오를 전송했을 때의 PSNR 성능을 나타냈다. 2.1Mbps의 백그라운드 트래픽을 생성했을 때, 서버는 SS와, MTD를 사용한 EBS를 위해 두 개의 세션을 열도록 했다. 이 때, 채널 혼잡과 간섭이 두 방법에 동일한 영향을 주도록 하였다. 그림 9에 보는 것처럼, 스케줄링 방법이 SS보다 향상된 부분은 이전의 유선 네트워크 경우에서 보다는 크다. SS와

EBS의 평균 PSNR은 그림 7(a)를 보면 토크쇼에 대해서 27.0dB와 35.8dB이고, 그림 7(b)를 보면 예고편 비디오에 대해서 22.1dB와 36.2dB이다. 그림을 보면 시작할 때 비디오 품질은 초기 pre-roll 버퍼링 때문에 화질이 좋지만, 시간의 경과에 따라 비교를 해보면, SS방법은 수신단 버퍼가 대부분 비어 있기 때문에 화질이 나빠지는 것을 볼 수 있는 반면, EBS는 좋은 비디오 화질이 지속됨을 볼 수 있다.

### V. 결론

우리는 지금까지 채널 변화와 비디오 속도의 변화를 고려하여 속도 대비 왜곡을 최적화하는 패킷 스케줄링 알고리즘을 소개하였다. 이를 위해, 시간 제충화와 MTD를 이용하였고, 전송할 최적의 서브스트림을 결정하는데 큐잉 이론의 체감대역폭 개념을 이용하였다. 유선 인터넷과 무선 랜에서 실제 비디오 시퀀스를 전송할 때, 순차적으로 전송하는 방법보다 제안한 방법이 더 효율적임을 보였다. 앞으로의 연구는 유선과 무선 링크가 결합된 네트워크나 VCR 기능 활용, 멀티캐스트 네트워크에 EBS를 적용하는 것으로 진행이 될 것이다.

### 부록 A

여기에서는, (2)와 (3)을 바탕으로 (13)을 최소화시키는 최적의  $g_j(j=0, 1, \dots, N-1)$ 을 유도한다.  $[g_0 g_1 \dots g_{N-1}]$ 로 생성된  $N$  차원의 공간을 고려해보자. 그러면 (2)의 부등식 조건으로부터  $(N+1)$ 개의 점  $[00\dots0], [10\dots0], [110\dots0], \dots, [11\dots1]$ 로 다면체를 만든다. 각 점은 (2)에 있는 하나의 부등식과  $N$ 개의 등식에 따라 대응된다. 다면체는  $\binom{N+1}{2}$ 개의 모서리를 갖고 있는데,  $N$ 개는 원점에 연결되어 있고 각각 다음과 같은  $N$  개의 조건에 대응된다.

$$\begin{aligned}
 1 > g_0 > g_1 = \dots = g_{N-1} &= 0, \\
 1 > g_0 = g_1 > g_2 = \dots = g_{N-1} &= 0, \\
 &\vdots \\
 1 > g_0 = g_1 = \dots = g_{N-1} > 0 &
 \end{aligned}
 \tag{A.1}$$

그리고, 나머지  $\binom{N+1}{2} - N$  개의 모서리는 다음 조건에 각각 대응된다.

$$\begin{aligned}
 1 = g_0 > g_1 > g_2 = \dots = g_{N-1} &= 0, \\
 1 = g_0 > g_1 = g_2 > g_3 = \dots = g_{N-1} &= 0, \\
 &\vdots \\
 1 = g_0 > g_1 = \dots = g_{N-1} > 0, \\
 &\vdots \\
 1 = g_0 = g_1 > g_2 > g_3 = \dots = g_{N-1} &= 0, \\
 1 = g_0 = g_1 > g_2 = g_3 > g_4 = \dots = g_{N-1} &= 0, \\
 &\vdots \\
 1 = g_0 = g_1 > g_2 = g_3 = \dots = g_{N-1} > 0, \\
 1 = g_0 = \dots = g_{N-2} > g_{N-1} > 0 &
 \end{aligned}
 \tag{A.2}$$

이제 교집합을 만들기 위해 위의 유도된 다면체에 초월평면 (3)에 대입하면, 두 도형이 만나는 교집합은 초월 평면 위에서 다각형을 만든다. 다각형의 꼭지점의 수는  $C, S^{(j)}, S^{(p)}, N$ 에 관한 함수로 표현된다. (13)을 최소화시키는 최적의  $g_j$ ;

$(j=0, 1, \dots, N-1)$ 은

$C, S^{(j)}, S^{(p)}, N, \delta_j$ 에 관한 함수이고, 다각형의 꼭지점 중 하나일 것이다.

먼저, (A.2)를 만족시키는 모서리에 놓여 있는 꼭지점들 중에서 최적의  $g_j$ 에 대한 조건을 찾는다. (A.2)의 모든 조건에 대해

$$g_0 = 1 \tag{A.3}$$

이기 때문에, 주어진 최적화 문제는 다음과 같이 간단히 할 수 있다.

(13)에서 두 번째 항을 최대로 만드는 최적의  $g_j(j=1, 2, \dots, N-1)$ 을 찾는 문제로 접근이 되며, 이때 아래의 표기를 사용한다.

$$\Delta = \delta_N + g_1 \delta_{N-1} + g_2 \delta_{N-2} + \dots + g_{N-1} \delta_1. \tag{A.4}$$

또한, 아래 식을 만족한다.

$$S^{(N)} + S^{(P)}(g_1 + \dots + g_{N-1}) = C \quad (A.5)$$

(A.5)를 (A.4)에 대입하면,

$$\Delta = \delta_{N+1} + \frac{\delta_1(C - S^{(N)})}{S^{(P)}} + \sum_{j=1}^{N-2} g_j(\delta_{N-j} - \delta_1) \quad (A.6)$$

를 최대로 만드는  $g_j$  를 찾으려 한다. 이 등식에서, 첫 번째 항과 두 번째 항은  $g_j$  와 무관하다. 여기에서

$$\delta_{N-1} - \delta_1 > \delta_{N-2} - \delta_1 > \dots > \delta_2 - \delta_1 \quad (A.7)$$

이기 때문에, 가능한 작은  $j$  로 큰  $g_j$  값을 찾으면 (A.6)가 최대값이 되며, (A.2)와 연립하여 얻은 최적의 조건은

$$\begin{aligned} g_0 &= g_1 = \dots = g_L = 1, \\ g_{L+1} &= \frac{C - S^{(N)} - LS^{(P)}}{L}, \\ g_{L+2} &= \dots = g_{N-1} = 0 \end{aligned} \quad (A.8)$$

인데 여기에서  $L$ 은  $\lfloor \frac{C - S^{(N)}}{S^{(P)}} \rfloor$  이다.

두 번째로, 원점에 연결된 모서리에 놓여 있는 꼭지점에서 최적의  $g_j$ 를 생각해 보자. 이런 경우에, 꼭지점은 다면체의 다음 모서리 들 중 하나에 놓여진다.

$$1 > g_0 = g_1 = \dots = g_M > g_{M+1} = \dots = g_{N-1} = 0 \quad (A.9)$$

결과적으로, 최적의 조건은

$$\begin{aligned} g_0 &= g_1 = \dots = g_M = \frac{C}{S^{(N)} + MS^{(P)}}, \\ g_{M+1} &= g_{M+2} = \dots = g_{N-1} = 0 \end{aligned} \quad (A.10)$$

이고,  $M = L + 1, L + 2, \dots, N - 1$  이다.

간단히 말하면, (2)와 (3)에 연관된 (13)을 최소화시키는 적당한  $g_j$  는 (A.8)이나 (A.10), 둘 중 한 가지로 표현된다.

## 부록 B

여기에서는, 큐잉 모델에서 서브스트림  $\Gamma^{(\Theta)}$ 의 패킷 손실율이  $\varepsilon_p^{(\Theta)}$ 로 주어져 있을 때  $\varepsilon_{GOP}^{(\Theta)}$ 의 근사값을 유도한다. 모든 움직임 패킷들이 데드라인에 맞춰 제대로 수신됐을 때, GOP 복호화가 성공한 것으로 정의하였다. 그림 2를 참고하여, 움직임 데이터가 어떤 계층화에서든 중요도 0부터  $N-1$ 을 차지하고 있음을 알 수 있다.

서버가 패킷 손실율이  $\varepsilon_p^{(\Theta)}$ 인 채널에서  $\Theta \leq N-1$ 인  $\Gamma^{(\Theta)}$ 을 전송한다고 가정하자.  $\Theta \leq N-1$ 일 때,  $\Gamma^{(\Theta)}$ 에 있는 모든 패킷은 움직임 부분을 포함하고 있으므로 임의의 패킷 손실은 GOP 복호화 실패로 연결된다. GOP의 평균 크기는 패킷 단위의  $E[S^{(\Theta)}]$ 이기 때문에, GOP 복호화 실패율은

$$\varepsilon_{GOP}^{(\Theta)} = 1 - (1 - \varepsilon_p^{(\Theta)})^{E[S^{(\Theta)}]} \quad (B.1)$$

가 된다. 그림 2(b)와 (d)에서 MTD를 접목시키면 텍스처 필드는 중요도  $h$ 가  $N-1$ 보다 큰 값을 갖는다. 송신측이 패킷 손실율  $\varepsilon_p^{(\Theta)}$ 을 갖는 채널에서  $\Theta > N-1$ 인 스트림  $\Gamma^{(\Theta)}$ 을 전송하면,  $\Gamma^{(\Theta)}$ 에 있는 패킷 가운데 중요도 0부터  $N-1$ 에 해당하는 패킷만 GOP 복호화 실패에 영향을 끼칠 것이다. GOP의 움직임 부분에 해당하는 데이터의 평균 크기는  $E[S^{(N-1)}]$ 가 되고, GOP 복호화 실패율은

$$\varepsilon_{GOP}^{(\Theta)} = 1 - (1 - \varepsilon_p^{(\Theta)})^{E[S^{(N-1)}]} \quad (B.2)$$

가 된다.

서브스트림  $\Gamma^{(\Theta)}$ 이 주어져 있을 때, 위에서 구한 GOP의 복호화 실패율은 우리가 제안한 스케줄링 알고리즘에서 최적의  $\Theta$ 를 찾기 위한 필요한 조건이다.

## 참고 문헌

- [1] H. Radha, Y. Chen, K. Parthasarathy, and R. Cohen, "Scalable Internet video using MPEG-4", *Signal Processing: Image Communication*, vol. 15, no 1 - 2 pp.95 - 126, Sept. 1999.

[2] G. J. Conklin, et. al., "Video coding for streaming media delivery on the Internet," *IEEE Trans. Circuits and Systems for Video Technology*, vol. 11, no. 3, pp.269 - 281, Mar. 2001.

[3] W. R. Heinzelman, M. Budagavi, and R. Talluri, "Unequal error protection of MPEG-4 compressed video," *IEEE ICIP'99*, vol. 2, pp.530 - 534, Oct. 1999.

[4] A. E. Mohr, E. A. Riskin, and R. E. Ladner, "Unequal loss protection: Graceful degradation of image quality over packet erasure channels through forward error correction," *IEEE Journal of Selected Areas in Communications*, vol. 18, no. 6, pp.819 - 828, June 2000.

[5] J. Cai, Q. Zhang, W. Zhu, and C. W. Chen, "An FEC-based error control scheme for wireless MPEG-4 video transmission," *IEEE WCNC 2000*, vol. 3, pp.1243 - 1247, Sept. 2000.

[6] M. Zhao and A. N. Akansu, "UEP of progressive images in wireless channels," *IEEE VTC 2000*, vol. 2, pp.479 - 483, 2000.

[7] M. Zorzi, "Performance of FEC and ARQ error control in bursty channels under delivery constraints," *IEEE VTC'98*, pp.1390 - 1394, May 1998.

[8] P. A. Chou and Z. Miao, "Rate-distortion optimized streaming of packetized media," submitted to *IEEE Trans. Multimedia*, Feb. 2001.

[9] Z. Miao and A. Ortega, "Expected run -time distortion based scheduling for delivery of scalable media," *Int'l Packetvideo Workshop 2002*, Apr. 2002.

[10] C. Courcoubetis and R. Weber, "Effective bandwidths for stationary sources," *Prob. Eng. Inf. Sci.* vol. 9, pp.285 - 296, 1995.

[11] K. Nagarajan and G. T. Zhou, "Self-similar traffic sources: modeling and real-time resources allocation," *Statistical Sig. Processing 2001*, pp.74 - 76, 2001.

[12] S. Floyd, M. Handley, J. Padhye, and J. Widmer, "Equation-based congestion control for unicast applications," *ACM SIGCOMM 2000*, pp. 3 - 56, Aug. 2000.

[13] S. McCanne, M. Vetterli, and V. Jacobson, "Low-complexity video coding for receiver-driven layered multicast," *IEEE Journal on Selected Areas in Communications*, vol. 15, no. 6, pp.983 - 1001, Aug. 1997.

[14] F. Wu, S. Li, R. Yan, X. Sun, and Y.-Q. Zhang, "Efficient and universal scalable video coding," *IEEE ICIP 2002*, vol. 2, pp.37 - 40, 2000.

최수정 (Sujeong Choi)

정회원

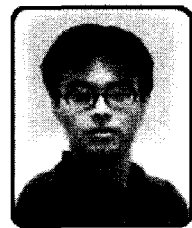


1996년 2월 서울시립대학교 수학과 졸업  
 1999년 8월 서울시립대학교 수학과 석사  
 2004년 8월 현재 서울시립대학교 전자전기컴퓨터공학부 박사과정

<관심분야> 멀티미디어 전송, 디지털 방송

안희준 (Heejune Ahn)

정회원



1993년 2월 한국과학기술원 전기및 전자공학과 졸업  
 1995년 2월 한국과학기술원 전기및 전자공학과 석사  
 2000년 2월 한국과학기술원 전기및 전자공학과 박사  
 1999년 8월~2000년 2월 독일

얼랑엔-뉴롬버그대학 방문 연구원

2000년 2월~2002년 8월 LG전자 정보통신 차세대 단말 연구소 (GSM, UMTS 프로토콜 팀, 선임연구원)

2002년 9월~2003년 12월 Tmax Soft 개발 연구소 (J2EE 팀 팀장, 책임 연구원)

2004년 2월~현재 서울산업대학교 제어계측공학과 (전임강사)

<관심분야> 혼잡제어, 멀티미디어 통신, 임베디드 플랫폼 소프트웨어

강 상 혁 (Sang Hyuk Kang) 정회원



1990년 2월 한국과학기술원 전  
기 및 전자공학과 졸업

1994년 2월 한국과학기술원 전  
기 및 전자공학과 석사

1997년 2월 한국과학기술원 전  
기 및 전자공학과 박사

2001년 1월 ~ 2002년 12월 :

University of California at Berkeley 연구원

1997년 9월 ~ 현재 : 서울시립대학교 전자전기컴퓨터  
공학부 조교수

<관심분야> 초고속 통신망, 라우터 및 스위치, 멀티  
미디어전송, 무선네트워킹