실시간 멀티미디어 스트리밍을 위한 계층 통합 패킷 스케줄링 기법

정회원 홍 성 우*, 원 유 집**

Cross-layer Design of Packet Scheduling for Real-Time Multimedia Streaming

Sung-woo Hong*, You-jip Won** *Regular Members*

요 약

멀티미디어의 프레임들은 QoS에 각각 다른 정도의 영향을 미치기 때문에, 패킷 손실을 줄이는 것이 항상 QoS를 향상시키는 것은 아니다. 사용자가 느끼는 QoS를 극대화하기 위해 본 논문은 중요성 기반 패킷 스케줄링을 제안한다. 중요성 기반 패킷 스케줄링 기법은 패킷 스케줄링의 두 가지 근본적인 문제를 해결한다. 하나는 "무엇을 보내는 가" 이고 다른 하나는 "언제 보내는 가" 이다. 이용 가능한 대역폭이 계속 변화할 때, 대역폭에 적응하기 위해 모든 패킷을 보낼 수 없을 경우에는 일부의 패킷을 선택해서 보낼 수밖에 없다. 이를 위해 패킷 Significance를 제안하는데, 패킷 Significance는 효과적으로 프레임 상관관계를 파악하고 프레임의 중요도를 정량화한다. 그리디(greedy) 접근 기법이 패킷 선택 문제에 사용되었고, 패킷 Significance가 전송 스케줄링에 고려되었다. 널리 공개된 MPEG-4 비디오 클립이 실험에 사용되었으며, 시뮬레이션 소프트웨어에 디코딩 엔진이 삽입되었고, 성능 평가를 위해 PSNR을 측정하였다. 크기 기반 패킷 스케줄링 기법 및 비트 레이트 기반 최선(bit-rate based best-effort) 스케줄링 기법과 성능을 비교 분석하였다. 제안하는 기법이 더 중요한 패킷에 차별화 된 보호 기능을 성공적으로 부여하며, QoS를 크게 향상시키는 것을 확인할 수 있었다.

Key Words : Multimedia, Scalable Encoding, Packet Significance, Traffic Smoothing, Packet Scheduling

ABSTRACT

Improving packet loss does not necessarily coincide with the improvement in user perceivable QoS because each frame carries different degree of importance. We propose Significance-aware packet scheduling (SAPS) to maximize user perceivable QoS. SAPS carries out two fundamental issues of packet scheduling: "What to transmit" and "When to transmit?" To adapt to the available bandwidth, it is necessarily to transmit the subset of the data packets if the entire set of packets can not be transmitted. "Packet Significance" quantifies the importance of the frame by elaborately incorporating frames' dependency. Greedy approach is used in selecting packets and transmission schedule is determined based on the Packet Significance. The proposed scheme is tested using publicly available MPEG-4 video clips. Decoding engine is embedded in the simulation software and user perceivable QoS is exposeed in termstermiSNR. Throughout the simulation based experiment, the performance of the proposed scheme is compared

※ 본 연구는 지식경제부 및 정보통신산업진흥원의 대학 IT연구센터 지원사업의 연구결과와(NIPA-2009-(C1090-0902-0033), 2009년도 정부(교육과학기술부)의 재원으로 한국과학재단의 지원을 받아 수행된 연구임(R0A-2009-0083128).

* 한양대학교 전자통신컴퓨터학과 분산 멀티미디어 연구실(toggiya@ece.hanyang.ac.kr)

** 한양대학교 전자통신컴퓨터학과 분산 멀티미디어 연구실(yjwon@ece.hanyang.ac.kr)

논문번호 : KICS2009-09-416, 접수일자 : 2009년 9월 17일, 최종논문접수일자 : 2009년 10월 26일

two other schemes: Size-based packet scheduling and Bit-rate based best effort packet scheduling. SAPS successfully incorporates the semantics of a packet and improves user perceivable QoS significantly. It successfully provides unequal protection to more important packets.

Ⅰ. 서 론

1.1 동기

무선 네트워크 기술의 발달은 사용가능한 네트워크 대역폭의 급격한 증가를 가져왔다. IEEE 802.11(a, b, g) 서비스는 움직이지 않는 노드에게 10 MHz 이상의 대역폭을 무선으로 제공할 수 있다. IEEE 802.16 기술은 1 MHz 이상의 대역폭을 움직이는 노드에 제공할 수 있으며, 3.5 세대 (예를 들면 HSDPA) 또는 4세대 통신 기술은 그보다 더 낮은 대역폭을 더 빨리 움직이는 노드에게 제공가능하다. 이와 같이 네트워크 기술의 발달은 대역폭의 증가를 가져왔을 뿐 아니라, 상황에 맞게 사용할 수 있는 선택권의 증가 또한 가져왔다. 최근에 등장하는 모바일 장치들은 이와 같이 상황에 따라 사용가능한 다양한 서비스와 기술들을 선택해서 사용할 수 있게 되었다. 예를 들면, [1]에 등장하는 모바일 장치는 IEEE 802.16 기반의 기술과 3G CDMA 기술을 상황에 맞게 선택해서 사용가능하다. 이와 같이 무선 컴퓨터 네트워크 기술의 급격한 발달은 멀티미디어 서비스를 유비쿼터스한 방식으로 서비스를 가능하게 하였다. 그러나 현재까지 무선 네트워크상에서 멀티미디어 서비스의 비스는는 만족할 만한 상황은 아니다. 이는 무선 네트워크의 특성과 멀티미디어 응용 프로그램의 시스템 요구사항 등에 기인한다. 실시간 멀티미디어 서비스는 문자 기반의 (textbased) 최선의 서비스(best effort)와는 별개의 특징을 가진다. 실시간 멀티미디어 서비스는 지속적으로 이용 가능한 대역폭을 요구하며 보내진 데이터가 정해진 시간 (deadline) 안에 보내어 져야 하며, 만약 정해진 시간 안에 데이터가 도달하지 않을 경우, 보내어진 데이터는 버려지게 된다. 따라서 멀티미디어 서비스는 일정 크기 이상의 대역폭을 통계적으로 (stochastically) 또는 결정적으로 (deterministically) 예약해 놓을 것을 요구한다. 그러나 무선 네트워크 특성상 여러 가지 이유 (해당 노드의 이동, 채널 간섭, 노이즈 등) 으로 인해 일정 크기 이상의 이용 가능한 대역폭이 지속적으로 보장되기는 힘들다. 또한 병목 (bottleneck node) 등의 패킷 지연 및 bursty한 트래픽으로 인한 버퍼 오버플로 (buffer overflow) 및 패킷 손실 등의 현상은 더욱 더 무선 네트워크상에서 멀티미디어 서비스를 어렵게 한다. 무선 네트워크상에서 멀티미디어 서비스의 Qos를 극대화하기 위해서는 이용 가능한 서비스의 적절한 선택 그리고 멀티미디어 어플리케이션의 특성 놓을요구에 맞는 최적화 기능을 하나의 프레임 워크 내에서 실행하여야 한다.

현재의 압축 기술은 해당 비디오에서 중복성 (redundancy)을 제외해서 만들어진다. 중복성은 시간적 (temporal), 공간적 (temporal), 그리고 신호대 노이즈 비 (SNR) 축을 따라서 존재한다. 압축된 비디오 정보는 프레임 (frame) 에 저장되어 전송되어 진다. 각각의 프레임은 전송하고자 하는 정보에 따라 다양한 크기로 나타나게 된다. I 프레임의 경우 프레임에 비디오 정보가 전부 압축되어 있어 크기도 가장 크게 나타나는 편이며, 다른 프레임의 도움 없이 디코딩 (decoding) 가능하다. P 프레임의 경우 이전 장면과 현재 장면의 차이를 나타내는 정보만을 가지고 있어 대개 I 프레임 정보보다는 작은 크기를 가진다. B 프레임의 경우 이전 I 프레임 (또는 P 프레임)그리고 이후 I 프레임 (또는 P 프레임) 양쪽에서 나타나는 프레임 정보의 차이를 저장하며 일반적으로 가장 작은 크기를 가진다. 이와 같이 각각의 비디오 정보를 프레임에 담아 저장하며, 각각의 프레임들은 다양한 크기를 가지게 된다. 이러한 프레임 크기들의 다양한 크기는 전송 시에 트래픽의 burstiness로 나타나게 된다. bursty한 트래픽은 노드의 버퍼 오버플로를 발생시키고, 이는 패킷 손실로 이어지고, 나아가 해당 어플리케이션의 QoS를 저하시키게 되는 요인으로 작동하게 된다. 트래픽의 burtiness 를 줄이기 위해 많은 연구가 행해졌으며, 일련의 이러한 기법을 트래픽 스무딩 (smoothing) 또는 트래픽 쉐이핑 (shaping) 이라고 한다.

본 논문에서는 이러한 트래픽 스무딩을 통해 실시간 멀티미디어 서비스의 QoS를 극대화 하고자 한다. 멀티미디어 QoS를 극대화시키기 위해 스트리밍 서버는 몇 가지 올바른 선택을 하여야 한다. 먼저 사용가능한 서비스의 종류 (예를 들면 IEEE 802.11 또는 HSDPA) 를 파악하고, 서비스를 선택한 뒤에는 이러한 서비스를 제대로 이용하기 위해 멀티미디어 특성에 맞게 데이터를 전송해야 한다. 데이터 전송을 위해 우선 사용 가능한 대역폭을 알

아봐야 하며, 대역폭에 맞게 보내어질 데이터가 결정되면 트래픽 스무딩 기법을 통해 burstiness를 최소화 시켜서 전송해야 한다. 대역폭에 맞추어 데이터를 선택하는 것은 변화하는 대역폭에 대해 프레임을 선별해서 보내는 것으로 만약 대역폭이 충분하다면, 이와 같은 과정은 생략될 것이다. 그러나 무선 네트워크의 특성상 이용 가능한 대역폭의 변동 폭이 커서 이러한 대역폭 적응 시스템은 반드시 필요하다. 따라서 어떤 프레임이 중요하고 어떤 프레임이 중요하지 않는가를 정확히 판단해야 한다. 이와 관련해 대역폭 변화에 따라 효과적으로 대처하기 위해 점층적 인코딩 (scalable encoding) 기법이 제안되었다[2]. 점층적 인코딩 기법은 본래의 이미지를 여러 개의 버전으로 나눠서 인코딩한 것으로 부분만의 (partial) 정보만을 가지고도 이미지를 복원할 수 있는 기법이다. 더 많은 정보를 가지게 되면 더 좋은 이미지를 복원할 수 있다. 그러나 점층적 인코딩 기법은 네트워크의 대역폭에 정확히 대응할 수 없다는 단점이 존재하며, 인코딩과 디코딩시의 복잡성 때문에 널리 쓰이지는 않는다. 이용 가능한 대역폭에 맞추어 데이터를 선택하고 나면 다음은 선택된 데이터를 클라이언트에 패킷 손실 없이 보내야 한다. 즉, 트래픽의 burstiness를 줄여서 보내야 한다. 보내고자 하는 데이터를 트래픽 스무딩 기법을 통해 트래픽 burstiness를 최소화 시켜서 패킷 손실을 감소할 수 있다. 현재의 트래픽 스무딩 기법은 보내고자 하는 패킷 또는 프레임의 크기만을 고려하여 데이터를 전송한다. 즉, 모든 패킷 또는 프레임의 중요성은 동일하다는 가정 하에 보내고 있다. 그러나 각각의 프레임이 QoS에 끼치는 영향은 다르며, QoS를 극대화하기 위해서는 각각의 프레임의 중요성을 고려한 트래픽 스무딩 기법이 필요하다. 본 논문의 의의는 다음과 같다. 첫 번째로 본 논문에서는 프레임의 중요성을 정확하게 파악하기 위해 패킷 Significance 개념을 도입하였다[3],[18],[25]. 본 연구는 [3],[18],[25]에서 제안된 수식을 발전시켰을 뿐 아니라 그 기법을 계층적으로 인코딩된 비디오 영상에도 적용시켰고, 제안하는 기법의 성능을 입증하였다. 패킷 Significance 는 패킷 손실에 따른 QoS에 끼치는 PSNR 손실을 수치화 한 것이다. 두 번째로 본 논문에서는 패킷 Significance를 이용하여 트래픽 스무딩 기법을 적용하였다. 본 논문에서 제안하는 기법을 통해, 패킷 손실이 더 크더라도 QoS는 더욱 크게 향상될 수 있다는 것을 나타내었다. 세 번째로 본 논문에서는 패킷 Significance를 이용한 패킷 스케줄링 즉 대역폭 적응 및 트래픽 스무딩 기법이 하나의 프레임 워크 내에서 실현되었다. 기존의 기법은 대역폭 적응 또는 트래픽 스무딩 기법이 각각 다른 문맥 (context) 내에서 시도되었으나 QoS를 극대화하기 위해서 대역폭 적응 및 트래픽 스무딩 기법은 하나의 프레임 워크 내에서 실현될 필요가 있다.

1.2 관련 연구

실시간 통신은 QoS보장 정도에 따라 결정적 그리고 통계적 보장 두 가지로 나눌 수 있다[4]. 결정적 보장 기법에서는 응용 프로그램의 요구사항에 부합하기 위해 대역폭 등의 시스템 자원을 최악의 경우에 맞추어 선점해 놓는데, 이는 트래픽이 bursty한 경우 시스템 자원 응용 차원에서 이는 아주 비효율적이다. 반면에, 시스템 자원을 효율적으로 사용하기 위해 통계적 보장 기법에서는 확률적으로 응용 프로그램이 요구하는 사항에 맞게 보장하려 한다. 예를 들어, 통계적 보장 기법에서 VBR (variable bit-rate) 응용 프로그램에 보장하는 대역폭은 응용 프로그램의 최고치 (peak rate) 대역폭보다는 낮게, 그러나 최저치 또는 평균 대역폭 보다는 높게 할당된다. 따라서 응용 프로그램이 bursty한 경우에 일정 수 이상의 패킷이 손실이 날 경우가 있다. Giordano 는 그의 논문에서 결정적 보장 기법에 비해 통계적 보장 기법의 성능 게인(gain)을 보여주었다[26].

통계적 멀티 플렉싱 기법은 각각의 패킷의 종단 (end-to-end) 지연에 대해 아무런 보장을 하지 않는다. 또한 각각의 패킷 스트림에 관해서 지연의 변화 (variation) 에 대한 보장도 하지 않는다. 이용 가능한 대역폭은 동적으로 변하는데, 이는 TCP 혼잡 제어 기능 또는 세션의 시작/ 끝에 따르는 것에 기인한다. 통계적 보장 기법에서는 자원이 결정적으로 보장되는 것이 아니기 때문에, 한정된 자원 내에 이를 효율적으로 사용하여 멀티미디어 QoS를 향상시키기 위해서 기존에 많은 선행 연구들이 있었다[2],[5]-[10].

이용 가능한 대역폭의 변화에 대처하기 위해 점층적 인코딩 기법(layered encoding scheme) 이 제시되었다[2]. 또한 MPEG-4 에서는 FGS (Fine Granularity Scalability) 인코딩 기법을 채택하여, 대역폭 변화에 대처할 수 있도록 하였다[2]. 그러한 코더들은 각각의 레이어가 요구하는 밴드대역폭과 그 수를 조절함으로써 약간의 화질 저화만 허용하는

내에 이용 가능한 대역폭에 적용될 수 있다. 그러나 MPEG 표준은 네트워크의 대역폭에 제대로 적용하기 위한 복잡성과 그리고 QoS를 극대화하기 위해 어떤 식으로 인터넷에 전송되어야 하는지에 대해서 지정하고 있지 않다. 한정된 자원 내에 그리고 때때로 에러가 발생하는 인터넷에 비디오 데이터를 효과적으로 전송하여 QoS를 향상시키기 위하여 Wu 등은 트래픽 burstiness를 통해 비디오 전송률 (rate)의 변화를 줄여 비디오 재생 시 화질변화 (playback quality variation) 를 줄이는 것을 제안하였다[5]. 이것은 인코딩 시에 필터 (arithmetic averaging filter)를 이용하여 재생될 비디오의 목표 화질을 달성하는 것으로 이루어진다. Dubois는 그의 연구에서 VBR (variable bit rate) 의 비디오 데이터의 트래픽 burstiness를 줄이는 것을 제안하였다[6]. 그는 ATM 환경 내에서 통계적 멀티 플렉싱 기법을 통해 burstiness의 감소를 실현시켰다. 그러나 Wu 또는 Dubios 등이 제안한 기법은 전송 비디오 전송율의 burstiness를 줄인 것으로, 각각의 다른 종류의 프레임이 화질에 끼치는 영향을 고려하지 않았으며 또한 네트워크 내에 존재하는 패킷 손실에 대한 고려가 부족하였다. 클라이언트가 느끼는 화질을 극대화하기 위해서는 각각의 프레임이 화질에 끼치는 영향을 고려할 필요가 있으며 중요한 프레임을 손실 없이 보내기 위한 기법이 필요하다. Argyriou는 유/무선 네트워크 환경에서 계층 통합 환적화 기법을 제안하였다[7]. 그는 종단 지연(end-to-end delay)과 패킷 손실에 대해서 automatic repeat request (ARQ) 와 forward error correction (FEC) 제어 함수 기법을 는 단 지연 계층과 링크 계층에 적용시켰다. 그는 제안된 계층 통합 기법으로 멀티미디어 패킷의 종단 지연을 줄이는 것과 멀티미디어의 화질 향상을 향상시키는 것에 성공하였으나, 각각의 프레임이 화질에 미치는 영향을 고려하는 것은 간과하였다. 는 종계층에서 프레임의 중요성을 계산하고 덴이터보를 링크 계층에서 패킷 재전송에 사용될 수 있다면 멀티미디어의 QoS는 더욱더 향상될 수 있을 것이다. Manour 등은 계층적 비디오 스트리밍을 서비스안하 때에 공동 전송 및 보호 (joint rate and protection) 기법을 제안하였다 [8]. 그들은 비디오 전송 알고리즘에서 패킷에 따른 비디오 화질의 왜곡을 고려한 전송률을 결정하는 것에 주목하였다. 패킷이 화질에 미치는 왜곡 정보는 프레임 종류와 화질에서 나타나는 화면의 PSNR 값이 저하되는 정도를 고려하였다. 제안된 패킷의 중요성을 측정하는 기법은 프레임간의 상호연관성(correlation)을 간과하는 방법이다. 이는 프레임이 손상되었을 경우 프레임의 해당 장면의 화질 뿐 아니라 프레임에 속해번는 Gop 내의 모든 화질 또는 프레임에 영향을 미치기 때문다. 따라서 패킷이 화질에 끼치는 왜곡정보를 정확하게 측정하지 못하였다. Miao 등은 각각의 프레임 왜곡 정보를 고려한 전송 할당 기법을 제안하였다 [9]. 그들의 실험에서 제안한 기법은 프레임 왜곡 정보를 고려하지 않은 기법에 비해 약 2.6 dB 이상의 화질 향상을 가져왔다. Frossard 등은 많은 스트리밍 서버가 비디오 데이터를 전송 시에 총체적인 (overall) 비디오 화질이 향상되는 최적화 프레임워크를 제안하였다[10]. [9, 10]에서 제안한 프레임의 중요성을 파악한 방법은 패킷의 크기와 패킷 손실에 따른 mean distortion error (MSE) 를 고려한 것이다. 이와 같은 방법은 프레임간의 의존성 (dependency)를 정확하게 고려하지 못할 뿐 아니라, 같은 크기를 가지는 패킷도 화질에 끼치는 영향이 크게 달라질 수 있다는 점을 간과한 방법이다. 본 논문에서는 패킷 QoS 모델과 패킷 스케줄링 알고리즘을 고려하는 프레임 워크를 제안한다. 이는 패킷의 중요성을 고려한 패킷의 선택과 선택된 패킷의 burstiness를 제거하는 것을 포함한다. 효과적인 패킷 선택 및 스케줄링을 위해 화질에 대해 미치는 영향을 수치화하는 개념인 패킷 Significance 를 제안하며, 패킷 Significance 를 바탕으로 클라이언트의 멀티미디어 QoS 를 극대화하기 위해 그리디(greedy) 기법을 사용하여, 네트워크 자원의 이용 효율을 높인다. 이후의 논문의 순서는 다음과 같다. 2절에서는 패킷 스케줄링 및 트래픽 스무딩 기법에 대해서, 3절에서는 패킷 Significance 기반 패킷 스케줄링 기법 (Significance-aware packet scheduling) 기법에 대해서 나타내었다. 본 논문에서 제안하는 기법의 성능 비교 및 평가 내용을 4절에 나타내었으며 5절에서는 본 연구에서 제안하는 기법의 오버헤드 (overhead) 및 실행 가능성 (practicality)이 조사되었다. 끝으로 6절에서 본 논문의 결론을 나타내었다.

Ⅱ. 패킷 스케줄링 및 트래픽 스무딩

2.1 패킷 전송 스케줄링

본 논문에서 패킷 스케줄링은 "무엇을 보내는가?" 와 "언제 보내는가?" 이 두 가지를 뜻한다. 기존의 패킷 스케줄링은 "언제 보내는가?" 만을 의미

했으나, 무선 네트워크 기술의 발달과 함께 이용 가능한 대역폭의 급변함은 계층적 인코딩 기술의 출현과 함께 “무엇을 보내는가?” 라는 문제를 대두시켰다. 스케줄러는 압축된 데이터 내에서 일정 부분만을 네트워크 자원이 허용하는 내에 선택해야 한다. 따라서 압축된 데이터 내에서 클라이언트의 QoS를 극대화하기 위해 보다 더 중요한 데이터를 먼저 선택해야 한다. 선택된 데이터를 올바르게 보내기 위해서 스케줄러는 보내고자 하는 트래픽의 burstiness를 최소화 시켜서 보내야 한다. 트래픽 스무딩 기법은 트래픽 내의 burstiness를 최소화 시켜 간접적으로 패킷 손실을 최소화하여 QoS를 향상시키고자 하는데 그 목적이 있다. 연속적인 오리지널 비디오 데이터 프레임은 패킷 타이징이라는 기법을 거쳐 패킷에 담아져서 전송된다. 이와 같은 패킷들에 대해서 패킷 스케줄러는 트래픽 스무딩 기법을 통해 트래픽의 burstiness를 줄일 수 있다. 트래픽 스무딩을 실행시키는 방법에는 두 가지가 있다. 하나는 크기 기반 스무딩 기법이고 다른 하나는 간격 기반 스무딩 기법이다. 크기 기반 트래픽 스무딩 기법에서는 패킷 스케줄러가 보내야 할 패킷의 크기를 조절하여 각각의 패킷의 크기를 비슷한 크기의 패킷으로 만들어서 균등한 간격으로 보낸다. 간격 기반 트래픽 스무딩 기법에서는 큰 패킷의 경우 간격을 크게 할당 하고 작은 패킷 사이의 간격은 좁게 만들어서 트래픽의 burstiness를 감소시킨다. 트래픽 내에서 burstiness를 제거하기 위해서 간격 기반 스무딩 기법 보다는 크기 기반 스무딩 기법이 더 자연스러워 보이는 것이 사실이다. 그러나 크기 기반 스무딩 기법은 여러 가지 이유로 현실적이지 못하다. 크기 기반 스무딩 기법은 하나의 패킷이 두 개 이상의 프레임 정보를 담도록 강제한다. MPEG 표준은 하나의 패킷이 둘 이상의 프레임 정보를 운반하지 말라는 규제는 없지만, 대부분의 스트리밍 시스템들은 하나의 패킷이 둘 이상의 프레임 정보를 저장해서 전송되는 것을 허락하지 않는다. 이는 만약 둘 이상의 프레임 정보가 저장된 패킷이 손실이 난다면, 이는 하나의 프레임이 아닌 다수의 프레임의 손실로 이어지게 되고 결국 화질 저화는 더 심해질 수밖에 없다. CPU 의 오버헤드 또한 크기 기반 스무딩 기법이 현실적이지 않은 또 하나의 이유이다. 하나의 패킷이 다수의 프레임 정보를 저장해서 전송된다면, 클라이언트의 시스템은 매 패킷을 받을 때마다 각 프레임의 정보를 정확히 구분하기 위해 각각의 분기점 (boundary)을 찾아야 한다. 각각의 프레임의 분기점을 찾는 것은 모바일 장치 (device) 와 같은 저 사양 장치에 심각한 CPU 오버헤드를 유발할 수 있다. 서버 측에서도 비슷한 문제가 발생한다. 실시간 스트리밍을 서비스를 할 때에, 화면에 뿌리는 시간 (presentation time stamp) 또는 디코드 시간 (decode time stamp), 크기 등의 정보가 각각의 프레임에 추가가 된다. 이러한 추가 정보를 프레임에 넣는 오버헤드를 줄이기 위해, 스트리밍 서비스를 위한 멀티미디어 정보들은 저장될 때 이러한 정보들을 처리한다. 이러한 정보들은 프레임들의 순서가 아닌 저장 타이밍과 디코딩 타이밍의 정보를 가지고 있는 일련의 패킷들로 구성된다. 그런데 하나의 패킷이 여러 프레임의 정보를 가지게 되면. 디코드 시간 등의 재생을 위한 추가 정보의 저장 방법 등에도 많은 영향을 미치게 된다. 따라서 대부분의 스트리밍 서버에서는 하나의 패킷이 하나의 프레임 정보만 보관하도록 하고 있다. 본 연구에서도 패킷 사이의 간격을 조절하여 burstiness를 줄이는 기법에 주목하였다.

2.2 패킷 손실과 QoS

트래핑 스무딩 기법은 클라이언트가 느끼는 QoS를 크게 향상시킬 수 있다[11],[12]. 그러나 트래픽 스무딩 기법이 항상 패킷 손실을 감소시키는 것은 아니다. QoS가 크게 향상되는 것은 패킷 손실이 무조건 감소해서가 아니라 중요한 패킷의 손실이 감소한 것에 기인한다[13]. 즉 간격 기반 트래픽 스무딩 기법이 중요한 패킷과 그렇지 않은 패킷을 구분하여 중요한 패킷의 손실을 줄일 수 있다는 것이다. 이러한 한상은 두 가지 기술적인 요소에서 기인한다. 첫 번째는 비디오 프레임이 패킷 안에 저장되는 방식에서 기인한다. 이미 언급한 바와 같이, 하나의 패킷은 두 개 이상의 프레임 정보를 저장할 수 없다. 모바일 무선 비디오 스트리밍 방식의 비디오 재생은 비교적 저 화질(low bit-rate)버전이고, B 프레임의 경우 최고 전송 유닛(maximum transfer unit: MTU)에 비해 그 크기가 매우 작다. I 프레임의 경우 B 프레임에 비해 그 크기가 매우 크다. 트래픽 스무딩 기법에서는 트래픽의 burstiness를 감소시키는데 그 목적이 있으므로, I 프레임에는 넓은 간격(interval)을 B 프레임에는 좁은 간격을 할당하게 된다. 따라서 프레임 배치 측면에서 보면 I 프레임 보다는 B 프레임들이 더 촘촘히 배치되어 트래픽 운영 측면에서 bursty하게 된다. 따라서 이용 가능한 대역폭 등의 상황에 따라 bursty한 B 프레임이 I

프레임에 비해 패킷 손실의 위험이 더 높다.

트래픽 스무딩 기법이 중요한 패킷의 손실을 줄이는 그 두 번째는 운영체제(Operating System: OS)가 큐를 운용하는 방식에 있다. 패킷이 전송되면 해당 패킷이 메인 메모리로 복사되어 저장되고 운영체제는 이 패킷의 포인터를 포인터 큐에 저장한다. 각각의 포인터는 메인 메모리에 저장된 패킷의 메모리 주소를 가리키게 된다. 따라서 크기에 상관없이 하나의 포인터는 하나의 패킷을 가리키게 되므로 B 프레임의의 경우 I 프레임 패킷에 비해 훨씬 더 많은 큐 공간을 차지하게 된다. 이와 같은 현상은 결국 bursty한 B 프레임 패킷이 I 프레임에 비해 버퍼 오버 플로우 등으로 인한 패킷 손실에 노출되는 것을 의미한다. 트래픽 스무딩의 간격 할당 방식과 운영체제의 큐 관리 방식이 합쳐지면서 주목할 만한 결과가 나타나게 되는데 이는 다음과 같다. 간격 기반 트래픽 스무딩 알고리즘은 데이터 전송률을 고르게 만든다. 그러나 이것은 운영체제의 관점에서 보면 들어오는 패킷들이 트래픽 스무딩 기법으로 인해 더 bursty해 진다. 그리고 버퍼 오버 플로 등에 의해 I 프레임은 덜 그리고 B 프레임들은 더욱더 패킷 손실 위험에 처해진다. 트래픽 스무딩 기법은 간접적으로 I 프레임 등의 큰 크기 프레임을 선호하게 되고 이는 결국 클라이언트가 느끼는 멀티미디어의 QoS 등의 화질을 향상시키는 요인이 되는 것이다. 따라서 패킷 손실과 패킷이 QoS에 미치는 영향에 대해서 고려해볼만한 가치가 있다. 패킷은 인터넷을 통해 전송되면서 때때로 손실이 일어나고 이것은 피할 수 없는 필연적인 문제이다. 그러므로 패킷 손실이 최소화할 수 있도록 패킷 스케줄러는 보내야 할 패킷의 선택과 그리고 보낼 시에 손실이 발생하지 않도록 보호 조치를 행해야 한다. 본 연구는 패킷 스케줄러가 패킷의 중요성을 고려해서 패킷 스케줄링을 하는, 즉 중요한 프레임(또는 레이어)을 선택하고 전송될 패킷들의 burstiness를 조절하여 패킷 손실은 최소화하면서도 클라이언트가 느끼는 QoS는 최대화하는 것에 그 목적이 있다.

2.3 패킷 Significance

패킷의 중요함은 패킷이 담고 있는 프레임의 종류와 Group of pictures (GoP) 내의 프레임의 위치에 따라 다르다. 만약 I 프레임이 없다면 같은 GoP 내의 P 프레임이나 B 프레임을 성공적으로 전송받아도 소용이 없다. 무선 모바일 환경 내에서 실시간 비디오 스트리밍을 서비스할 때의 I 프레임은 고속 유선 네트워크 환경 내에서 스트리밍을 서비스할 때보다 훨씬 낮은 빈도수로 나타난다. 예를 들면 MPEG-4 의 인코더의 I 프레임 사이의 기본(default) 간격은 250 이다[14]. 이와 같은 현상은 압축 효율적인 측면에서 기인하는데, I 프레임이 많을수록 압축 효율이 떨어지기 때문이다. 패킷 Significance 는 패킷 속에 있는 프레임의 중요도를 정량화한다. 과거에 많은 연구들[15]-[17]이 패킷의 중요성을 정량화하여, 그 중요도에 따라 패킷 스케줄링 기법을 개발하였다. 패킷 중요도를 측정하기 위해 사용된 매개변수로는 프레임의 크기, 프레임 종류, 의존하는 프레임의 수, 디코드된 화면에서 MSE (mean square error) 왜곡 정도 등이 사용되었다. 패킷의 중요성을 정확히 측정하기 위해서 이와 같은 매개변수들은 다 같이 하나의 문맥에서 측정되어야 한다. 본 연구에서는 앞에서 언급된 것들을 포함할 뿐 아니라 패킷 손실로 인해 사용자에 나타나는 화질의 PSNR의 저하된 값을 정확히 측정하였다. 본 논문에서 비디오는 계층적으로 부호화되었다고 가정한다. 는 i 번째 GoP의 j번째 프레임의 k번째 레이어 정보를 뜻한다. 프레임의 의존도는 화살표(→)로 표시한다. 예를 들어 A→B 는 A 가 제대로 디코딩되기 위해서는 B 프레임 정보가 있어야 한다는 것을 의미한다. 이미지의 화질 정보를 위해 본 연구에서는 이미지 모델을 개발하였다. 이미지는 2차원의 픽셀들이 나열되어 있는 형태이며 픽셀의 수에 따라 화질의 급을 나눈다. 예를 들면 HD: 1024*768, CIF: 352*288 등이 있다. 각각의 비트는 대개 24bit 의 크기를 가진다. $f^i_{j,k}$ 가 제대로 전송되고 디코드 되었을 때 $f^i_{j,k}(x,y)$를 (x, y) 의 위치에서 픽셀 값이라고 하고, $f^i_{j,k}$ 가 제대로 전송되지 않거나 제대로 디코드 되지 않았을 때 $\widehat{f^i_{j,k}}(x,y)$를 (x, y)에서의 픽셀값이라고 하자. 패킷 Significance를 정의하기 위해 패킷의 기여도를 먼저 정의한다. 패킷의 기여도 $D(f^i_{j,k})$는 다음 식 (1)과 같이 정의된다[25].

$$10\log_{10}\frac{W\times H\times 255^2}{\sum_{x=0}^{W-1}\sum_{y=0}^{H-1}\left|\widehat{f^i_{j,k}}(x,y)-f^i_{j,k}(x,y)\right|^2} \quad (1)$$

H 와 W 는 각각 화면의 높이와 너비를 나타낸다. 패킷의 기여도 $D(f^i_{j,k})$는 프레임 또는 레이어

$f^i_{j,k}$가 손실이 났을 때, 나타나는 화질의 PSNR 이 저하된 값을 나타낸다. 이를 바탕으로 패킷 Significance, $Q(f^i_{j,k})$ 가 정의된다. $Q(f^i_{j,k})$ 는 프레임 또는 레이어 $f^i_{j,k}$ 가 손실이 났을 때 나타나는 화질의 저하된 값 (패킷의 기여도 $D(f^i_{j,k})$) 뿐 아니라, 해당 프레임 또는 레이어 $f^i_{j,k}$에 의존하는 프레임들 또는 레이어들의 화질 저화를 포함한다. 패킷 Significance는 식 (2)와 같이 정의된다.

$$Q(f^i_{j,k}) = \sum_{f^l_{m,n} \in C(f^i_{j,k})} D(f^i_{j,k}) + D(f^i_{j,k}) \qquad (2)$$

$C(f^i_{j,k})$는 $f^i_{j,k}$에 의존하는 프레임들 또는 레이어들을 의미한다. $f^i_{j,k}$ 가 제대로 전송되지 않거나 제대로 디코딩 되지 않으면 해당 프레임 또는 레이어 뿐 아니라 GoP 내의 모든 의존하는 프레임 또는 레이어들이 영향을 받는다. 즉, 패킷 Significance 값, $Q(f^i_{j,k})$는 패킷 그 자체의 기여도 값 $D(f^i_{j,k})$와 그에 의존하는 패킷들의 기여도 값 $\sum_{f^l_{m,n} \in C(f^i_{j,k})} D(f^i_{j,k})$ 들의 값들의 합으로 구성된다. 따라서 패킷 Significance 의 연산의 오버헤드는 패킷 기여도를 구하는 오버헤드와 같다고 할 수 있다. 그림 1은 각각의 레이어의 패킷 기여도 값과 패킷 Significance 값을 보여주는 간단한 예제이다. 그림 2에서 볼 수 있다시피 각각 I 프레임, P 프레임, B 프레임의 패킷 기여 값, $D(f^i_{j,k})$, 이 각각 5, 3, 1 이라고 하면 패킷 Significance, $Q(f^i_{j,k})$, 값은 (5+3+1)=9, (3+1)=4, 1 과 같다. 이와 같이 본 논문에서 제안하는 패킷 Significance 는 각각의 프레임과 레이어들의 패킷 기여도를 바탕으로 패킷이 화질에 미치는 영향을 제대로 파악할 수 있다.

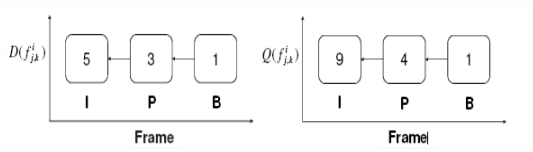


그림 1. 패킷 기여도와 Significance
Fig. 1. Packet contribution & significance

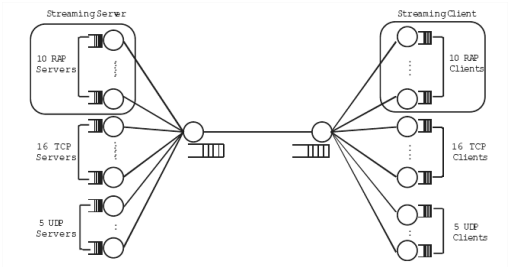


그림 2. 네트워크 토폴로지[25]
Fig. 2. Network topology[25]

Ⅲ. 패킷 Significance 기반 패킷 스케줄링

3.1 패킷 선택

패킷을 선택하는 과정은 한정된 자원 즉 사용가능한 대역폭 또는 노드의 큐의 크기 등에 맞추어 전체 비디오 데이터 중 보낼 수 있는 양만큼 선택을 해서 전송을 하는 것이다. $\delta(f^i_{j,k})$를 이전 패킷 이후 다음 패킷을 전송하기까지의 전송 간격이라고 가정하고, $S(f^i_{j,k})$를 패킷의 크기로 가정한다. 현재의 사용가능한 대역폭은 스트리밍 서버나 CDN 등에게 여러 가지 기법을 통해 [9]-[19] 미리 알려져 있다고 가정한다. 본 논문의 목적은 패킷 스케줄링을 통하여 사용자로 하여금 최상의 멀티미디어 화질을 보게 하고자 함에 있다. $I(f^i_{j,k})$를 주어진 자원에 맞추어 선택된 패킷이라고 가정하고 $E(f^i_{j,k})$를 선택된 패킷 중에 전송에 실패하거나 제대로 디코드 되지 않은 패킷이라고 가정한다. 이때 고려할 것은 주어진 자원 내에서 최상의 화질 $\xi(f^i_{j,k})$을 위해서 가장 중요한 패킷부터 먼저 고려해야 한다는 것이다. 그를 위한 다음 식 (3)과 같다 [3].

$$\underbrace{\sum_{f^i_{j,k} \in I(f^i_{j,k})} D(f^i_{j,k})}_{A} - \underbrace{\sum_{f^i_{j,k} \in E(f^i_{j,k})} Q(f^i_{j,k})}_{B} \qquad (3)$$

식 (3)에서 A항은 전송을 위해 선택된 패킷들의 패킷 기여도의 합을 나타내며 B항은 제대로 전송되지 않거나 디코드 되지 않은 패킷들로 인한 화질의 저화를 의미한다. 따라서 A항을 최고치로 그리고 B항을 최저로 만들면 화질 $\xi(f^i_{j,k})$은 극대화 될 수 있다. 패킷을 선택하는 문제는 배낭 문제 (knapsack problem)와 동일한 문제로 볼 수 있는데, 이는 패킷의 크기는 배낭에 넣을 아이템의 무게, 그리고 패킷 Significance, $Q(f^i_{j,k})$는 아이템의 값어치와 일맥 상통하다는 점에 기인한다. 그리고 배낭의 사용가능

용량은 현재의 이용 가능한 대역폭의 크기와 같다. 대역폭 예측은 1-2초 이내로 하기 때문에 본 논문에서 스케줄링 단위는 매 GoP 단위로 이루어진다. 다음 식 (4)는 매 스케줄링 단위 (GoP) 마다 패킷 선택을 위해 사용가능한 대역폭을 계산하는 과정을 나타낸다.

$$U = \int_{t_0}^{t_0+w} \rho(t)dt \tag{4}$$

t_0, w 그리고 $\rho(t)$는 각각 매 스케줄링 시의 윈도우 시작 시간, 길이, 그리고 그때의 대역폭 이용 가능한 대역폭의 크기를 나타낸다. 본 논문에서는 패킷을 선택함에 있어서 그리디 기법 (greedy approach)를 사용하였다. 패킷 Significance를 패킷의 크기로 나눈 값 $\varepsilon(f_{j,k}^i) = \frac{Q(f_{j,k}^i)}{S(f_{j,k}^i)}$ 에 따라 높은 순서대로 주어진 자원, 즉 대역폭을 초과하지 않도록 차례대로 선택한다. 이는 다음 식 (5)에 나타나있다.

$$\sum_{f_{j,k}^i \in I(f^i)} S(f_{j,k}^i) \le U \tag{5}$$

3.2 패킷 전송

전송할 패킷이 정해지면, 선택된 패킷을 안전하게 클라이언트에게 전송할 필요가 있다. 선택된 패킷의 전송 스케줄을 정하는 것의 전송 사이의 간격을 정하는 것과 일맥상통하다. 선택된 패킷이 대역폭 사용량을 넘지 않도록 패킷 선택 프로세스에서 이미 이루어졌지만 여전히 패킷의 손실 가능성의 존재한다. 패킷이 bursty하게 송할 대역이는 버퍼 오버플로 등으스케줄을 패킷 손실로 이어진다. 간격을 서버에서는 패킷 손실선택된소화하기 위해 패킷의 burstiness를 된소화하여야 한다. 본 연구에서는 패킷 사이의 간격을 이용하기 위해 패킷 Significance을 이용하였다. 핵심rst디어는 중요을 패킷일수록 더 큰 간격을 할당 받는한다.것이다. $\delta(f_{j,k}^i)$를 이전 패킷 이후 다음 패킷을 송할하기까지의 전송 간격이라고 가정한다. 예를 대역$\delta(f_{j,k}^i)$ 값이 30 msec 라고 한다대역이는 이전 패킷 $f_{j,k-1}^i$을 보낸 뒤 30 msec 시간이 지난 뒤에 $f_{j,k}^i$ 패킷을 보내다.것을 송 사이의 각 레이어의 $\delta(f_{j,k}^i)$를 구하는 식은 다음 식 (6)과 같다 [25].

$$\frac{S(f_{j,k}^i) \times Q(f_{j,k}^i)}{\sum_{F(f_{j,k}^i) \in I(f^i)} S(f_{j,k}^i) \times Q(f_{j,k}^i)} \times GoP_{(\sec)} \tag{6}$$

프레임 또는 레이어가 MTU (maximum transmit unit) 보다 클 경우에는 여러 개의 패킷으로 나뉘어져 전송된다. 이와 같은 경우 각각의 프레임 또는 레이어가 할당 받은 간격을 다시 패킷의 수로 나뉘어 동일한 프레임 또는 레이어를 이루는 각 패킷은 동일한 간격을 할당받는다.

Ⅳ. 성능 평가

4.1 실험 환경

본 연구에서는 3가지 패킷 스케줄링 알고리즘을 비교한다. 패킷 Significance를 고려한 패킷 스케줄링 (Significance Aware Packet Scheduling: SAPS), 크기 기반 패킷 스케줄링 (Size-based Packet Scheduling: SBPS), 그리고 최선 기반 패킷 스케줄링 알고리즘 (Bit-rate based best effort packet scheduling: Best-Effort)의 성능을 비교 평가한다. 각각의 패킷 스케줄링 알고리즘은 패킷의 선택과 전송 방법에 있어서 각기 다른 척도를 가지고 이를 수행한다. 각 패킷 스케줄링 알고리즘은 3가지 프로세스 (레이어 선택 프로세스, 패킷 선택 프로세스 그리고 간격 할당 프로세스) 로 나누어질 수 있다. 레이서 선택 프로세스는 SAPS, SBPS, 그리고 Best-Effort 모두 동일하다. 레이서 선택 프로세서는 계층적 인코딩 (scalable encoding) 기법에서만 필요한 과정인데, 다계층(multi-layer)으로 인코딩된 비디오 파일에서 이용 가능한 대역폭에 맞추어서 최대 몇 계층까지 보낼지를 결정하는 과정을 말한다. 예를 들면 비디오가 1계층부터 4계층까지 계층적 인코딩 기법을 사용하여 인코딩 되었다고 가정하자. 1계층부터 2계층까지 전송하는데 필요한 대역폭이 250KByte/sec 이고 1계층부터 3계층까지 보내는데 필요한 대역폭이 320KByte/sec 라고 가정하자. 그리고 현재 이용 가능한 대역폭은 300KByte/sec 이라 하면, 패킷 스케줄러는 3계층까지 보내는 것으로 결정한다. 그리고 선택된 레이어는 계층단계인 패킷 선택 프로세스에서 이용 가능한 대역폭에 더 세세

하게 맞추어지는 과정을 거친다. 즉 레이어 선택 프로세스는 현재의 이용 가능한 대역폭을 넘는 가장 낮은 계층을 결정하는 과정을 말하는 것이다. 단일 계층으로 인코딩된 비디오를 전송하는 스트리밍 서비스에서는 이와 같은 과정은 생략되어진다. 레이어 선택 프로세스 이후에는 SAPS 그리고 SBPS 알고리즘은 각각 이용 가능한 대역폭에 맞추어 패킷 선택 프로세스를 수행한다. SAPS 기법은 GoP 단위로 패킷의 중요성을 바탕으로 자원이 허락하는 내에 패킷을 선택하는 반면에 SBPS 기법은 GoP 시작의 패킷부터 디코딩 순서대로 대역폭이 허락하는 만큼 패킷을 선택한다. Best-Effort(BE) 기법은 해당 레이어 계층 모두를 전송한다. SAPS 기법 및 SBPS 기법은 레이어 선택 프로세스 이후 패킷 선택 프로세스를 통해 이용 가능한 대역폭의 사용량을 초과하지 않기 위해 대처하지만 BE 기법은 해당 레이어 전체를 보내기 때문에 보내어진 패킷은 네트워크 상황에 따라 무작위 패킷 손실의 위험에 처해진다. 전송되어야 할 패킷이 정해지면, 알고리즘은 각각의 척도를 가지고 패킷 전송 간격을 결정한다. SAPS 기법은 패킷의 패킷 Significance 값에 따라 전송 간격을 결정하는 반면에 SBPS 기법과 BE 기법은 패킷 크기의 크기에 비례해서 패킷 전송 간격을 할당한다. 즉, 큰 크기의 패킷일수록 작은 크기의 패킷에 비해 큰 간격을 할당받는다. 시뮬레이션은 네트워크 시뮬레이터 (NS version 2) 를 이용해서 실행되었으며 [20], 네트워크 토폴로지는 그림 2 에 나타나 있다.

그림 2에서 보다시피 31개의 서버들과 31개의 클라이언트들이 나타나있다. 총 16개의 TCP 노드와 5개의 UDP 노드가 10개의 스트리밍 서버와 함께 링크를 공유하고 있다. 처음 10개의 노드들은 스트리밍 서비스를 위한 서버이며 각각 연결된 클라이언트들을 위해 스트리밍을 서비스한다. 스트리밍 서비스는 RAP(rate adaptive protocol) 프로토콜을 통해 대역폭을 예측하여 실행되었다[19]. 나머지 노드들은 백그라운드위해트래픽 (TCP와 UDP 트래픽)을 발생시켜 실제와 유사한 상황을 나타내었다. 백그라운드 트래픽은 FTP 와 HTTP 서비스가 TCP 세션 위에 실행되었으며, CBR(constant bit rate)가 UDP 세션 위로 실행되었다. 본 연구에서는 실험을 위해 실제 비디오 스트리밍 파일을 사용하였으며 해당 비디오 클립 및 관련 트레이스들은 [21]에서 공개적으로 이용가능하다. 본 논문에서는 실험을 위해 여러 비디오 클립을 사용하였으며, 각 비디오들은 색깔 히스토그램 (color histogram), motion dynamics 등 각기 다른 특징을 가진다. 본 실험에서는 SAPS 알고리즘이 다양한 비디오 클립 및 실험환경에서 어떠한 성과를 보일지 조사하였다. 비디오 클립은 계층적 인코딩 기법)(하나의 기본 레이어, base layer,와 3개의 고급 레이어, advanced layer) 또는 싱글 레이어로 (single layer) 인코딩되었다. 표 1은 기본적인 비디오 클립의 정보를 나타내었다.

표 1. 비디오 클립 기본 정보
Table 1. Basic information of video clips

File	MotherandDaughter	King
Compression type	MS MPEG-4(MP43)	H.264
Frame rate ($frame/sec$)	30	
Picture size ($pixel$)	176 * 144	1,920 * 1,080
Number of Frames	930	2,801
Time (sec)	31	934
Bit-rate($kbit/sec$)	306	16,-
Mean ($byte$)	1,275	12,-
Variance ($byte^2$)	1,528,-	8,615,-
GoP structure	GoP(30,1), Single layer	GoP(30,3), Multi-layer
Encoder	MEncoder [14]	

그림 3은 실험된 비디오의 샘플 화면을 보여준다. 비디오 클립들은 MEncoder 로 인코딩 되었다[14].

본 연구의 목적은 클라이언트가 느끼는 멀티미디어의 화질을 극대화하는 것이다. 본 연구는 패킷이나 프레임 손실을 줄인다거나 이용 가능한 대역폭을 확장하려는 것이 아니다. 따라서 시뮬레이션 기반의 네트워크상에서 전송된 패킷을 제대로 재건축 및 디코딩하는 것은 아주 중요하다. 본 연구에서는 클라이언트 노드 상에서 프레임 디코딩 엔진을 실행하였다. 디코딩 엔진은 네트워크를 통해 들어오는 패킷을 다시 조합하고 이미지를 구축한다. 구축된 이미지는 화질 평가를 위해 전송된 비디오 파일과 비교하여 PSNR 값을 도출한다. PSNR 값은 각각의 디코딩된 프레임마다 계산되어지며 최종 성능은 도출된 PSNR 값의 평균으로 평가되어진다. 클라이언트는 스트리밍 서비스가 시작된 후 2초 후에 화면에 재생을 시작한다. 만약 전송된 패킷의 순서가 재생 순서와 맞지 않을 경우나 또는 프레임을 조합하

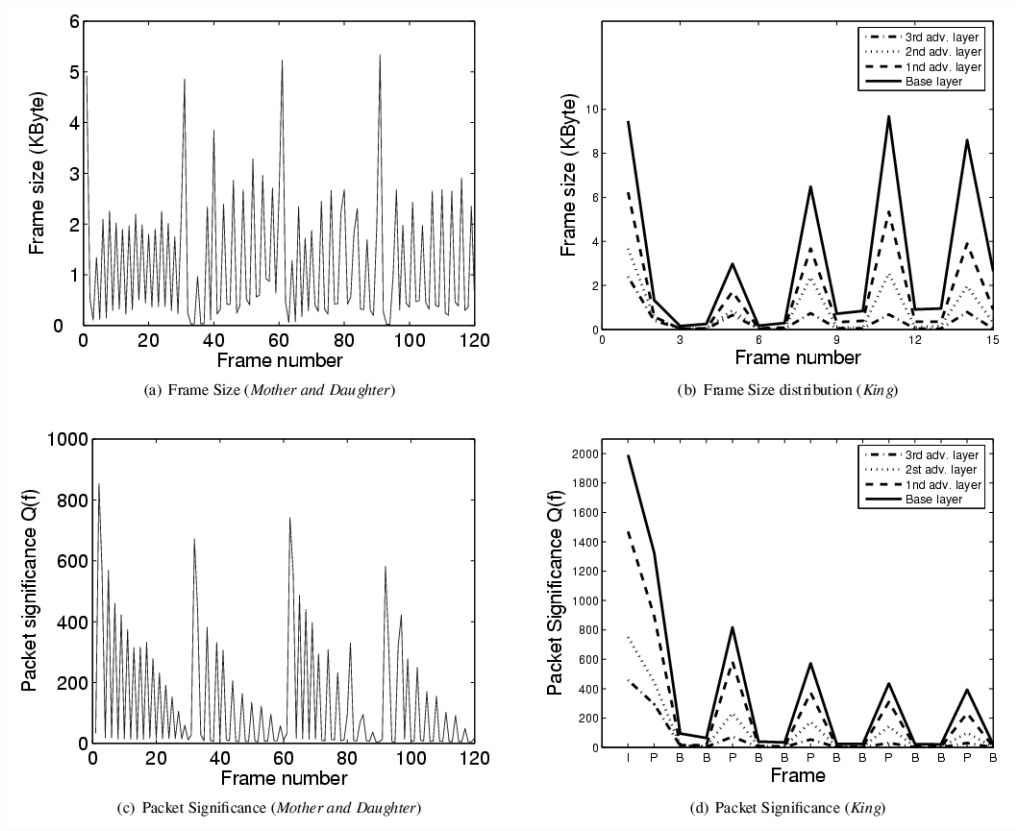
(a) Mother and daughter (b) King

그림 3. 실험에 사용된 비디오 화면 샘플
Fig. 3. Sample video scene

기 위한 패킷들이 다 도착하지 않은 경우 패킷들은 데드라인까지 큐에 머물러 있게 되며, 그 이후에는 버리게(drop) 된다. 또한 하나 이상의 패킷을 이루는 프레임의 패킷이 전송되지 않거나 또는 손상되어 전송되어서 제대로 디코딩 과정을 거칠 수 없는 경우 프레임을 구성하는 전송된 전체 패킷 모두를 버리게 된다. 그리고 해당 프레임이 손상되거나 데드라인을 넘어서 전송된 경우 즉 클라이언트에서 해당 프레임이 재생 시간까지 도착하지 않을 경우 이전 프레임을 이용한 손실 숨기기 기법(previous frame concealment scheme)이 디코더 엔진에서 사용된다[22].

4.2 프레임 크기 및 패킷 Significance 분포

이번 장에서는 프레임 크기와 패킷 Significance의 상관관계에 대해서 조사한다. 다양한 측면에서 패킷 크기 및 Significance를 조사하기 위해 각기 다른 인코딩 기법 (마이크로 소프트 MPEG-4 버전3 (MS-MP43), H.264 [14])와 다양한 GoP 구조가 사용되었다. 그림 4는 프레임 크기 및 해당 프레임의 패킷 Significance 값을 보여준다. 각각의 GoP 구조는 Mother & Daughter: GoP(30,1), King: GoP(30,3) & 계층적 인코딩과 같다. 그림 4(a), (c)에서 볼 수 있다시피 I 프레임은 매 30개의 프레임마다 분포되어 있으며 가장 큰 패킷 Significance 값을 가진다. P 프레임은 하나의 GoP 내에서 I 프레임에 가까이 위치할수록 큰 값을 가지는 경향이 있으며 멀리 떨어질수록 패킷 Significance 값이 감소한다. 이는 하나의 GoP 내에서 뒤에 위치할수록 해당 프레임에 의존하는 프레임의 수가 감소하기 때문이다. 그림 4의 (b)와 (d)는 계층적으로 인코딩된 동영상 King 비디오 클립의 프레임 크기 및 해당 프레임의 패킷 Significance 값을 보여준다. 4개의 각기 다른 값이 표현된 이유는 1계층부터 4계층까지 각기 다른 값을 가지기 때문이다. 매 30 프레임마다 I 프레임이 분포하고 있으며 다른 프레임에 비해 월등히 높은 패킷 Significanc 값을 가진다. King 비디오 클립에서도 Mother and Daughter 비



(a) Frame Size (*Mother and Daughter*)
(b) Frame Size distribution (*King*)
(c) Packet Significance (*Mother and Daughter*)
(d) Packet Significance (*King*)

그림 4. 프레임 크기 및 패킷 Significance 분포
Fig. 4. Frame size and Packet Significance distribution

디오 클립과 마찬가지로 GoP 내에서 앞에 있는 P 프레임일수록 값이 큰 경향이 있으며 이는 비록 사이즈가 작다 할지라도 의존하는 프레임의 수가 많으면 큰 사이즈의 P 프레임보다 더 중요할 수 있다는 것을 보여준다. B 프레임의 경우 GoP 내에서 앞에 있던지 뒤에 있던지 그 값이 매우 작으며, 이는 크기 뿐 아니라 의존하는 프레임의 수 또한 매우 작기 때문이다. 인코딩 전송 속도는 약 900 Kbit/sec 이며 각 레이어 당 해당 요구 대역폭은 각각 76.8, 153.6, 230.4, 439.2 Kbit/sec 와 같다.

4.3 이용 가능한 대역폭 적응 및 QoS

이번 장에서는 이용 가능한 대역폭의 크기에 따라 해당 논문에서 제안하는 기법의 성능을 평가한다. 병목 노드의 큐 크기는 100,000 이며 각 3가지 기법: 패킷 Significance 기반 패킷 스케줄링 기법(SAPS), 크기 기반 패킷 스케줄링 기법(SBPS), Bit-rate 기반 최선 기법(BE)의 성능이 비교, 평가되었다. 그림 5는 하나의 레이어로 인코딩된 비디오 클립과 계층적 인코딩 기법의 결과를 각각 PSNR 값으로 나타낸다. 본 논문의 기법은 하나의 레이어로 인코딩된 비디오 뿐 아니라 다계층 레이어(multi-layer)로 인코딩된 비디오를 가지고도 성능평가를 하는데 이는 본 논문에서 제안하는 기법의 유용성 (flexibility) 를 보여주기 위함이다.

그림 5의 (a)와 (b)는 대역폭의 크기의 변화에 따른 3가지 기법의 성능 평가의 결과를 보여준다. 병목 지점에 있는 노드의 이용 대역폭은 6 Mbits/sec 에서 51.3 Mbits/sec 로 변화시켰으며, 클라이언트의 해당 이용 가능한 평균 대역폭은 128 Kbits/sec 에서 990 Kbits/sec 이다. 그림 5의 (a)에서 볼 수 있다시피 SAPS 기법이 가장 높은 QoS를 보여주는데 이는 SAPS 기법이 다른 2가지 기법보다 더 중요한 패킷을 더 많이 성공적으로 클라이언트에 전달했음을 의미한다. 그러나 이용 가능한 대역폭이 늘어날수록 SAPS 기법과 SBPS 그리고 BE 기법과의 차이는 줄어든다. 이는 대역폭 같은 자원이 풍부할 때에는 모든 패킷이 다 성공적으로 전송되어 각 기법의 차이가 없어지는 것에 기인한다.

그림 5의 (b)는 다계층 레이어 (layer) 으로 인코딩된 비디오의 PSNR 결과를 보여준다. 병목 링크의 노드의 이용 가능한 대역폭이 높을 때는 SAPS, SBPS, BE 기법에서 QoS (PSNR) 값의 차이가 거의 없다. 병목 링크의 대역폭이 줄어들수록 SAPS 기법의 이용 가능한 대역폭의 적응능력이 점차 두드러지기 시작한다.

그림 6은 대역폭의 변화에 따른 3가지 기법의 패킷 전송 결과의 통계치를 보여준다. 그림 6의 (a), (b), (c), (d)는 Mother and Daughter 동영상 클립의 성능 결과를 보여준다. X 축은 이용 가능한 대역폭의 크기이고, Y 축은 패킷 전송 결과를 퍼센트(%)로 보여준다. Sent(Success) 는 성공적으로 보낸 패킷의 퍼센트를 뜻하며, Sent(Fail) 은 전송을 위해 선택되었지만, 전송에는 실패한 퍼센트를, 그리고 Not_Sent 는 선택되지 않은 퍼센트를 각각 나타낸다. 각각의 병목 링크의 이용 가능한 대역폭에 따라 그림에 SAPS, SBPS, BE의 결과를 3개의 막대그래프가 각각 나타낸다. 그림 6의 (a), (b), (c), (d)는 하나의 레이어로 인코딩된 동영상이기 때문에 BE 기법의 경우 Not_Sent가 0%다. 노드의 대역폭의

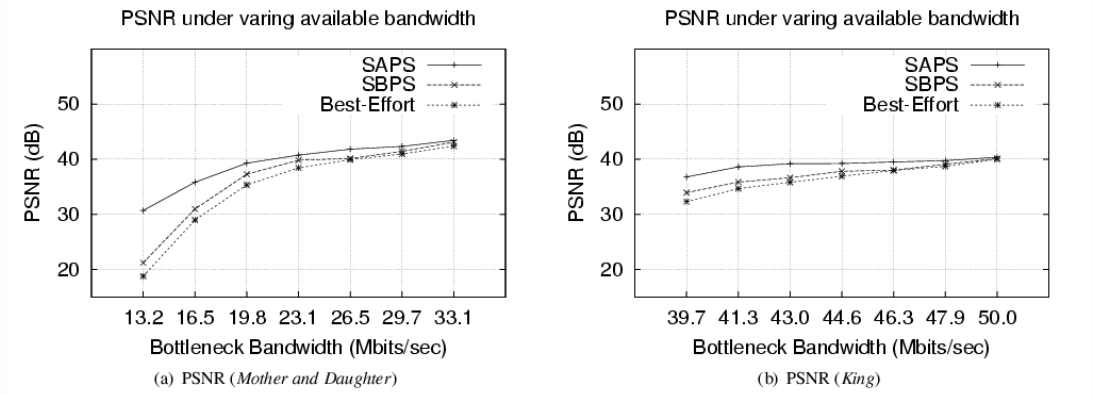
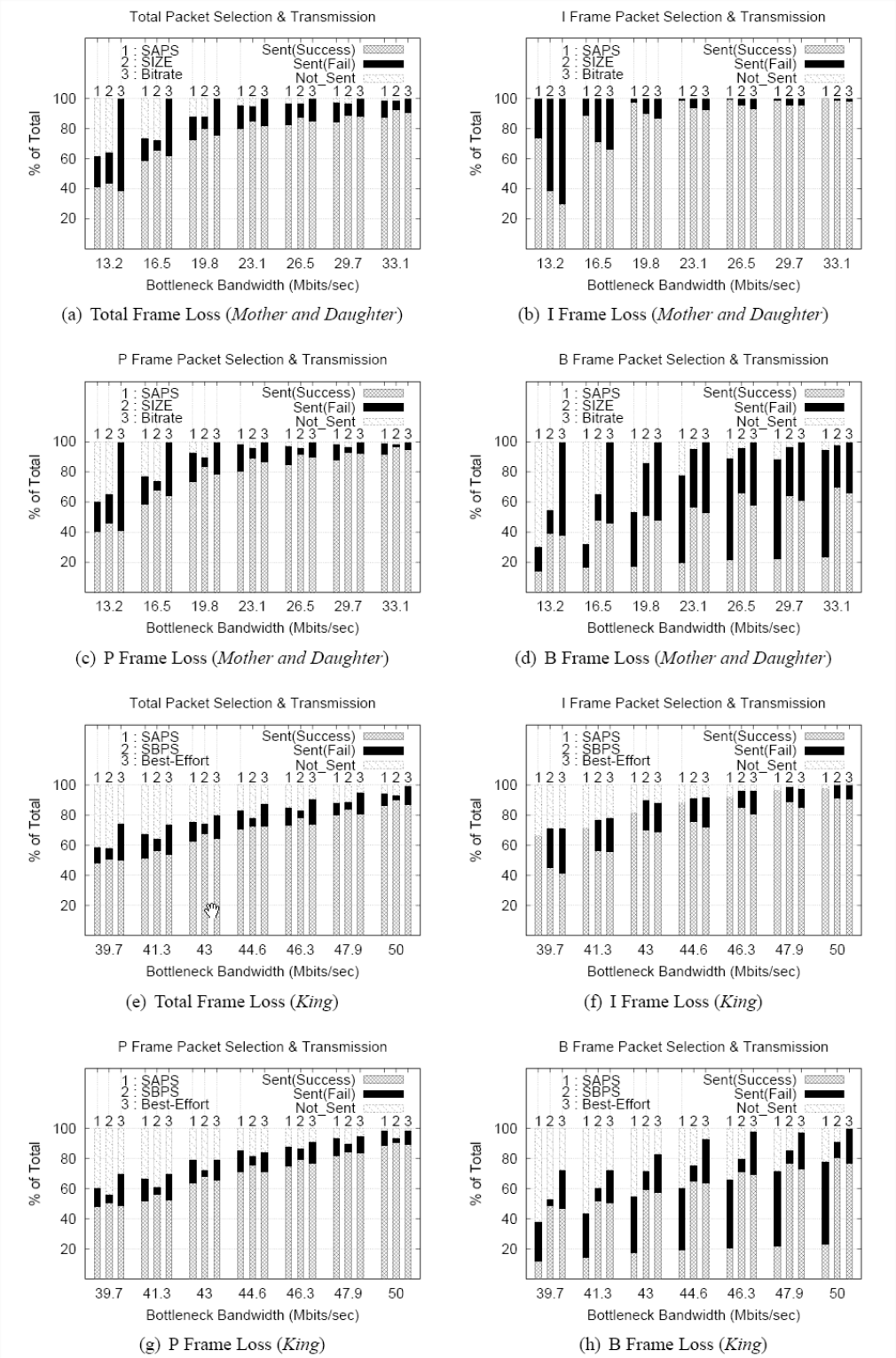


그림 5. 병목 대역폭에 따른 PSNR
Fig. 5. PSNR over bottleneck bandwidth variation



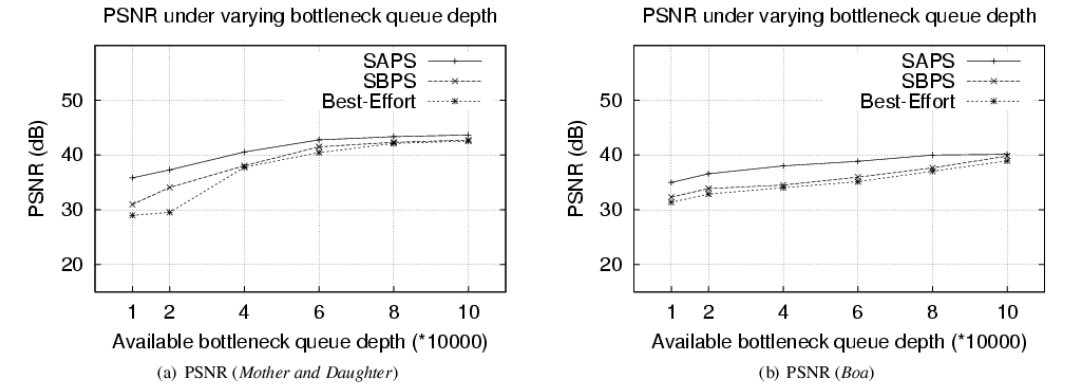
(a) Total Frame Loss (*Mother and Daughter*)
(b) I Frame Loss (*Mother and Daughter*)
(c) P Frame Loss (*Mother and Daughter*)
(d) B Frame Loss (*Mother and Daughter*)
(e) Total Frame Loss (*King*)
(f) I Frame Loss (*King*)
(g) P Frame Loss (*King*)
(h) B Frame Loss (*King*)

그림 6. 병목 노드 대역폭에 따른 패킷 손실
Fig. 6. Frame loss statistics over the bottleneck bandwidth

16.5 Mbits/sec 일 때, SAPS한 SBPS는 전송을 위해 레이어 선택 프로세스한 패킷 선택 프로세스를 통해 전체 패킷에서 약 70%의 패킷을 선택한다. BE 스케줄링 기법은 스케줄링 기법은 SAPS 기법과 SBPS 기법에 비해 약 30% 정도 더 많은 패킷을 전송하려고 시도하지만 성공적으로 도착한 패킷의 양, 즉 Sent(Success) 값은 SAPS, SBPS, BE 기법체 두가 비슷하거 약BE 기법이 오히려 낮다. 그리고 성공적으로 보내어진듯이 프레임의 비율을 보면 각 기법에 따라 현저한 차이를 보인다. 그림 6(b), (c), (d)는 의 일 때, SAPS 기법에 비해 월등히 높은 PSNR 값을 보이는 이유를 설명한다. 그림 6(b), (c), (d)는 3가지 기법에 따른 I 프레임, P 프레임, B 프레임의 전송 성공 또는 실패의 비율을 보여준다. 이용 가능한 대역폭이 16.5 각각 나타낸3개의 일 때, 그림 6(a)는 SAPS 기법이 약 70%, SBPS 기법이 약 40% 그리고 BE 기법이 약 30% 정도의 I 프레임의 전송 성공률을 보여준다. 이는 SAPS 기법이 성공적으로 트래픽 내의 burstiness를 조절하여 I 프레임의 손실 비율을 효과적으로 낮춘 것에 기인한다. SBPS 기법은 BE 기법에 조금더 높은 전송 성공 비율을 보이는데, 이는 BE 기법은 패킷 선택 프로세스를 걸리지 않기 때문에 SBPS 기법에 비해 이용 가능한 대역폭에 미세하게 적응을 하지 못하기 때문이다. BE 따라서 BE 기법의 I 프레임은 다른 두기법에 비해 가장 패킷 손실의 위험에 노출되어 있는 것이다. 그림 6(d)가 이를 구체적으로 뒷받침한다. 그림에서 보다시피 그림 6(d)는 B 프레임 패킷의 선택 및 전송 성공 실패 여부의 결과를 보여준다. 이용 가능한 대역폭이 16.5 Mbits/sec 일 때, 각 SAPS, SBPS, BE 기법은 전체 B 프레임의 정보에서 각각 30%, 55%, 100%의 B 프레임을 선택해서 전송을 시도한다. 그리고 각 3가지 기법의 성공 비율은 17%, 49%, 46% 이다. 여기서 알수있듯이, SAPS 기법은 가장높은 B 프레임 손실 비율을 보여준다. 요약하자면 SAPS 기법은 멀티미디어의 Qos를 극대화하기 위해 더 중요한 패킷을 선택해서 안전히 보내는 기법으로 다른 기법에 비해 I 프레임은 높은 전송 성공률을 그리고 B 프레임은 낮은 성공률을 보인다. 그림 6의 (e), (f), (g), (h)는 King 비디오 클립의 패킷 전송 결과를 보여준다. King 비디오 클립은 다계층 레이어 인코딩 기법을 사용하였으므로, BE 기법도 레이어 선택 프로세스를 통해 모든 데이터를 전송 하지 않는다. 그림 6(e)에서 볼 수 있다시피, 총 프레임 손실은 SAPS 기법이 다른 2가지 기법에 비해 비슷하거나 더 높은 값을 보인다. 그러나 그림 5(b)에서 볼 수 있었듯이, SAPS 기법은 다른 두 가지 기법에 비해 더 높은 QoS를 보여준다. 그림 6의 (f), (g), (h)에서 그 이유를 찾을 수 있다. 그림 6의 (f)는 각 기법에 따라 I 프레임의 전송 성공 결과를 보여준다. 그림에서 볼 수 있다시피 SAPS 기법은 레이어 선택 프로세스, 패킷 선택 프로세스, 간격 할당 기법을 통해 99% 이상 선택된 I 프레임을 성공적으로 전송하였다. 그러나 SBPS와 BE 기법은 비록 선택된 I 프레임의 퍼센트는 같지만, 월등히 낮은 전송 성공률을 보인다. 이는 SAPS 기법이 다른 두 가지 기법에 비해 더 중요한 패킷을 더 안전하게 보내려고 한 것에 기인한다. 그림 6(h)는 이와 반대되는 현상을 보인다. 그림 6(h)는 각 3가지 기법에 따라 B 프레임의 전송 성공 결과를 보여준다. 그림에서 볼 수 있다시피 SAPS 기법의 경우 가장 낮은 퍼센트의 B 프레임을 선택했고, 그리고 가장 낮은 성공률을 보여준다. 예를 들면, 이용 가능한 대역폭이 41.3 일 때, SAPS 기법은 약 40% 를, SBPS 기법은 약 60%, 그리고 BE 기법은 약 70%의 B 프레임을 선택한다. 그리고 B 프레임의 성공률은 SAPS 기법이 약 10%, SBPS 기법이 약 49% 그리고 BE 기법이 약 46% 정도이다. 이용 가능한 대역폭이 늘어날수록 SBPS 기법과 BE 기법의 경우 B 프레임의 성공률이 눈에 뜨게 늘어나지만, SAPS 기법의 경우 아주 서서히 조금씩 증가할 뿐이다. 이는 SAPS 기법이 보다 더 중요한 패킷, 예를 들면 I 프레임과 높은 패킷 Significance 값을 가지는 P 프레임 등을 위해 보다 덜 중요한 패킷을 더욱더 bursty하게 만들기 때문이다.

4.4 병목 노드의 큐 크기 및 QoS

이번 장에서는 병목 노드의 큐 크기에 따라 세 가지 기법의 성능평가를 수행한다. 병목 노드의 큐 크기는 10,000부터 100,000 까지 변화시켰으며, 이용 가능한 대역폭은 충분한 값으로 고정 시켜 큐 변화에 따른 기법의 적응성을 제대로 판단할 수 있게 하였다. 그림 7(a), (b)는 병목 노드의 큐 크기 변화에 따른 3가지 기법의 PSNR 값을 보여준다. 병목 노드의 큐 크기가 충분히 클 경우 세 가지 기법의 PSNR 값은 거의 동일하다. 그러나 큐 값이 줄어들수록 세 가지 기법의 차이는 두드러지기 시작한다. 예를 들면, 병목 노드의 큐 사이즈가 20,000 일 때, SAPS, SBPS, BE 기법의 PSNR 값은 각각



(a) PSNR (*Mother and Daughter*) (b) PSNR (*Boa*)

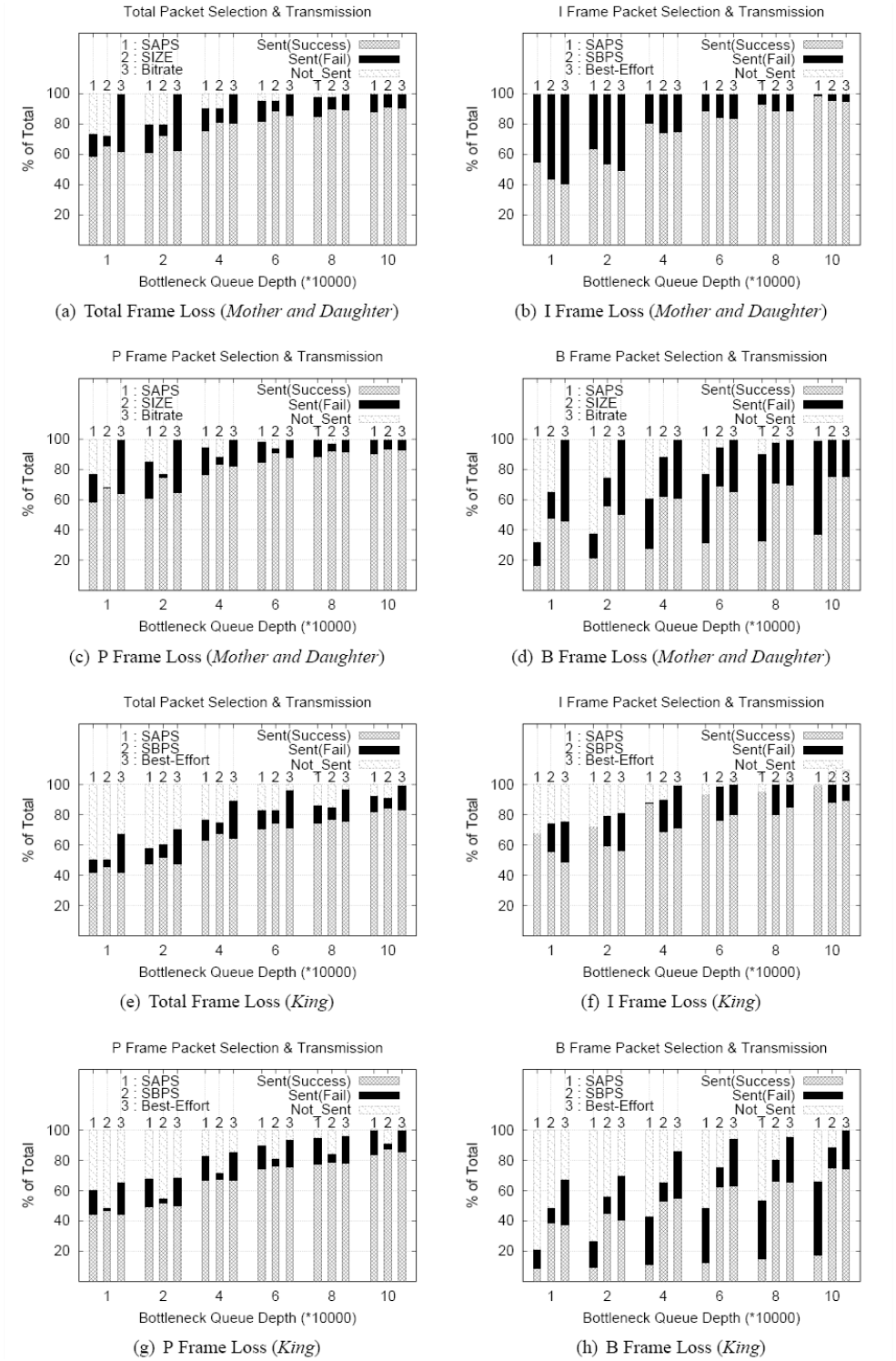
그림 7. 병목 노드 크기에 따른 PSNR
Fig. 7. PSNR over bottleneck queue depth

(Mother and Daughter: 37, 34, 30), (King: 36, 33, 32)이다.

그림 8(a)는 병목 노드의 큐 크기에 따른 Mother and Daughter 비디오 클립의 각각의 패킷 전송 결과를 보여준다. 그림에서 볼 수 있듯이 병목 노드의 크기가 커질수록 SAPS 기법과 SBPS 기법이 더 많은 패킷을 선택하며, BE 기법은 항상 모든 패킷을 보내므로 Not_sent 의 비율이 0% 인 것을 볼 수 있다. SAPS 기법과 SBPS 기법은 거의 비슷하게 패킷을 선택하는 것을 볼 수 있으며, SBPS 기법이 SAPS 기법에 비해 약간 더 높은 총 패킷 전송 성공률을 보인다. 그러나 더 많은 패킷의 전송 시도가 더 높은 PSNR 값의 결과로 이어지지는 않는다. 그림 8(b)은 세 가지 기법의 I 프레임 전송 성공 결과를 보여준다. SAPS 기법은 다른 두 가지 기법에 비해 항상 높은 I 프레임 패킷 성공률을 보인다. SBPS 기법과 BE 기법은 비록 I 프레임의 선택은 SAPS 기법과 같거나 더 많은 양의 I 프레임을 선택해서 전송을 시도하지만 SAPS 기법에 비해 전송 성공 비율이 낮은 것을 알 수 있다. 이는 SBPS 기법과 BE 기법이 패킷의 크기에 비례하는 간격을 할당한 것과 달리 SAPS 기법은 QoS에 영향을 끼치는 더 중요한 패킷에 더 큰 간격을 할당함으로써 I 프레임과 그리고 일부 높은 패킷 Significanc 값을 가지는 P 프레임의 패킷 손실 비율을 성공적으로 낮추었기 때문이다. 그림 8(d)는 B 프레임 패킷의 전송 결과를 보여준다. SAPS 기법의 경우 다른 두 가지 기법에 비해 가장 낮은 퍼센트의 B 프레임을 선택하고, 또한 성공률 또한 매우 낮음을 알 수 있다. 그림 8(e), (f), (g), (h)는 다계층으로 인코딩 된 King 비디오 클립의 병목 노드 큐의 크기에 따른 패킷 전송 결과를 보여준다. 그림 8(e)에서 볼 수 있듯이 다계층으로 인코딩 된 경우 BE 기법도 레이어 선택 프로세스를 통해 이용 가능한 대역폭에 적응을 하고 있는 것을 알 수 있다. 그림 11(b)는 각 기법의 I 프레임 패킷의 전송 결과를 보여준다. 그림에서 볼 수 있듯이 SBPS 와 BE 기법과 달리 SAPS 기법은 99% 이상의 전송 성공률을 볼 수 있는데, 이는 SAPS 기법이 다계층으로 인코딩 된 비디오 클립에서도 효과적으로 잘 동작하는 것을 보여준다. 그림 8(c), (d)는 P 프레임과 B 프레임 패킷의 전송 결과를 보여준다. P 프레임의 경우 SBPS 기법이 가장 높은 성공률을, 그리고 SAPS 기법과 BE 기법이 약간 낮게, 비슷한 성공률을 보여준다. B 프레임의 경우 SAPS 기법은 가장 낮은 성공률을, SBPS 기법과 BE 기법이 월등히 높은 성공률을 보여준다. SAPS 기법은 의도적으로 낮은 중요성을 가지는 패킷을 더욱 bursty하게 만들고 또한 더 중요한 패킷들을 덜 bursty하게 만든다.

V. 패킷 Significance 의 연산 오버헤드

실시간 멀티미디어 응용프로그램에서 비디오 프레임들은 온라인상에 실시간으로 생성되고 전송되므로 본 연구에서 제안하는 기법을 실제로 사용하기 위해서는, 각 패킷 Significance 또한 실시간으로 계산되어질 필요가 있다. 이번 장에서는 패킷 Significance 를 계산하는데 걸리는 시간 오버헤드 및 본 연구에서 제안하는 기법의 실현 가능성에 대해 조사한다. 패킷 Significance 의 연산 오버헤드는 GoP 구조와 길이에 따라 다르다. 본 연구에서는 실제 동영상을 가지고 실험을 하였다. 해상도가 176*144 및



(a) Total Frame Loss (*Mother and Daughter*) (b) I Frame Loss (*Mother and Daughter*)

(c) P Frame Loss (*Mother and Daughter*) (d) B Frame Loss (*Mother and Daughter*)

(e) Total Frame Loss (*King*) (f) I Frame Loss (*King*)

(g) P Frame Loss (*King*) (h) B Frame Loss (*King*)

그림 8 병복 노드의 크기에 따른 패킷 손실

Fig. 8 Frame loss statistics over the bottleneck queue depth

1920*1080 두 개의 비디오 클립을 이용하였으며, MPEG-4/FGS 및 H.264/AVC [14] 로 인코딩을 수행하였다. 패킷 Significance 를 구하는 식은 식 (2)에 나타나있다. 2절에서 설명했다시피, 패킷 Significance 의 연산의 오버헤드는 패킷 기여도를 구하는 오버헤드와 같다고 할 수 있다. 패킷 기여도를 구하는 식은 식 (1)에 나타나있다. 패킷 기여도를 구하는 오버헤드는 각 프레임을 디코딩하는 오버헤드와 같다. 프레임을 디코딩 하는 것은 파싱 (parsing), IDCT (inverse discrete cosine transform), 재건축 (reconstruction), Dithering 등 여러 가지 단계를 거친다[23]. 펜티엄 3 600-MHz 컴퓨터에서 H.264 baseline 으로 인코딩된 Foreman [21] 비디오 클립(355*288, 326 Kbps)을 디코딩하는데 걸린 속도는 67 frames/sec 와 같다 [24]. 대부분의 비디오 프레임들이 30 frames/sec 이내의 속도로 인코딩 된다는 점을 감안할 때, 본 논문에서 제안하는 중요성 기반 패킷 스케줄링 기법은 충분히 실행 가능하다는 것을 알 수 있다.

Ⅵ. 결 론

한정된 자원 내에서 고성능 연산을 요구하는 응용프로그램인 멀티미디어 서비스의 QoS를 향상시키는 것은 결코 끝나지 않는 과제 중에 하나이다. 많은 연구가 트래픽 스무딩 또는 쉐이핑 (traffic smoothing or shaping)기법을 통해 한정된 큐 사이즈 내에서 패킷 손실을 줄이려고 하였고, 계층적 인코딩 기법을 통해 이용 가능한 대역폭에 대해 적응하는 기법을 개발하였다. 또한 패킷의 중요도에 따라 FEC(forward error correction) 기법 및 RDOP(rate distortion optimization protocol) 기법 등을 통해 QoS를 향상 시키려고 하였다. 그러나 각각의 기법은 각기 다른 계층(layer)에서 진행되었고, 한 계층에서의 성능향상이 전체적인 QoS의 향상으로 항상 이어지는 결과를 낳는 것은 아니었다. 트래픽 스무딩 기법은 Transport 계층 및 MAC 계층에서 burstiness를 줄임으로써 패킷 손실을 줄이려고 하는 문제이고, 계층적 인코딩 기법은 응용프로그램 계층에서 계층적 인코딩 기법을 통해 네트워크상에 이용 가능한 대역폭에 적응하려는 기법이다. 본 논문에서는 하나의 프레임 워크 내에서 계층 통합 최적화 기법을 통해 패킷 Significance 기반 패킷 스케줄링 기법을 제안하였다. 멀티미디어의 QoS를 극대화시키기 프레임의 중요도를 파악하는 "패킷 Significance" 개념을 제안하였으며 이는 해당 프레임이 손실이 난 경우 PSNR 값이 떨어지는 것을 양자화 한 것이다. "패킷 Significance" 는 프레임 사이의 의존도 뿐 아니라 프레임과 레이어의 상호 관계까지 포괄한다. 패킷 Significance 를 통해 패킷의 중요도를 파악하고, 이를 바탕으로 트래픽 스무딩 기법에 적용하였다. 본 논문에서 제안하는 SAPS (Significance-Aware Packet Scheduling) 기법은 레이어 선택 프로세스, 패킷 선택 프로세스, 패킷 간격 할당이라는 3가지 프로세스를 통해 이루어지며 패킷 스케줄링의 근본적 문제인 "무엇을 보내야 할까"와 "언제 보내야 할까" 의 문제를 해결한다. 본 논문에서 제안하는 기법은 계층적 인코딩 기법과 하나의 레이어 기법 둘 다에 적용할 수 있음을 실험을 통해 증명하였다. NS-2 (Network Simulator Version 2) 를 통해 실험을 수행하였으며 SBPS (Size-based Packet Scheduling) 와 BE (Bit-rate based Best Effort Packet Scheduling) 기법과 비교를 통해 제안하는 기법의 성능의 우월성을 입증하였다. SAPS 기법은 이용 가능한 대역폭 및 큐 크기에 적절히 패킷 선택 및 전송을 수행하여 한정된 자원을 더욱 효율적으로 사용, 멀티미디어의 QoS를 극대화 하였다. 또한 해당 기법의 연산 오버 헤드 및 실행 가능성에 대해서도 조사하였다.

참 고 문 헌

〔1〕 J. Zipper, C. Stoger, G. Hueber, R. Vazny, W. Schelmbauer, B. Adler, and R. Hagelauer, "A single-chip dual-band cdma2000 transceiver in 0.13 muhboxm cmos," IEEE Journal of Solid-State Circuits, 42(12), pp.2785-2794, Dec., 2007.

〔2〕 W. Li, "Overview of fine granularity scalability in MPEG-4 video standard," IEEE Trans. on Circuits and Systems for Video Technology, 11(3), pp. 301-317, Mar., 2001.

〔3〕 Y. Won, J. Jung, Y. Jun, I. Chang, and S. Hong, "Qos weighted scheduling: Real-time streaming of multi-resolution video," in Proc. of Graphics and Visualization in Engineering (GVE 2007), Clearwater, Florida, USA, Jan., 2007.

〔4〕 C. M. Aras, J. Kurose, D. S. Reeves, and H. Schulzrinne, "Real-time communication in packet-switched networks." in Proc. of the

IEEE, 82, pp.122-139, Jan., 1994.

〔5〕 J. Wu, J. Cai, and C. Chen, “Single-pass rate-smoothed video encoding with quality constraint,” IEEE Trans. on Signal Processing Letters, 14(10), pp.715-718, Oct., 2007.

〔6〕 J. Dubois, “Burstiness reduction of a doubly stochastic ar-modeled uniform activity vbr video,” Trans. on WORLD ACADEMY OF SCIENCE, ENGINEERING AND TECHNOLOGY, 23, pp.454-458, Aug., 2007.

〔7〕 A. Argyriou, “Cross-layer error control for multimedia streaming in wireless/wireline packet networks,”. IEEE Trans. on Multimedia, 10(6), pp.1121-1127, Sept., 2008.

〔8〕 H. Mansour, P. Nasiopoulos, and V. Krishnamurthy, “Real-time joint rate and protection allocation for multi-user scalable video streaming,” IEEE 19th International Symposium on Personal, Indoor and Mobile Radio Communications, pp.1-5, NY, USA, Sept., 2008.

〔9〕 P. Chou and Z. Miao, “Rate-distortion optimized streaming of packetized media,” IEEE Trans. on Multimedia, 8(2), pp.390-404, Apr., 2006.

〔10〕 J. Chakareski and P. Frossard, “Distributed packet scheduling of multiple video streams over shared communication resources,” in proc. on Multimedia Signal Processing, IEEE, pp.1-4, Shanghai, China, Oct., 2005.

〔11〕 S. S. Lam, S. Chow, and D. K. Yau, “An algorithm for lossless smoothing of mpeg vide,” Trans. on SIGCOMM Comput. Commun. Rev. 2(4), pp.281-293, 1994.

〔12〕 J. D. Salehi, Z. Zhang, J. Kurose, and D. Towsley, “Supporting stored video: Reducing rate variability and end-to-end resource requirements through optimal smoothing,” IEEE Trans. on Networking, 6(4), pp.397-410, Aug., 1998.

〔13〕 Y. Won and B. Shim, “Empirical study of VBR traffic smoothing in wireless environment,” In Proc. of the second international workshop on Innovative Internet Computing Systems, 2346, pp.193-204. Springer, June, 2002.

〔14〕 MEncoder. Program for encoding video+audio [online] available: http://www.mplayerhq.hu/.

〔15〕 P. Chou and Z. Miao, “Rate-distortion optimized streaming of packetized media,” IEEE Trans. on Multimedia, 8(2), pp.390-404, Apr. 2006.

〔16〕 K. Mayer-Patel, L. Le, and G. Carle, “An mpeg performance model and its application to adaptive forward error correction,” In Proc. of the tenth ACM international conference on Multimedia, pp. 1-10, NY, USA, 2002

〔17〕 I. Politis, M. Tsagkaropoulos, T. Pliakas, and T. Dagiuklas, “Distortion optimized packet scheduling and prioritization of multiple video streams over 802.11e networks,” Advances in Multimedia, 2007(1), pp.1-11, April, 2007.

〔18〕 원유집, 전영균, 박동주, 정제창, “다계층 멀티미디어 스트리밍을 위한 의미기반 패킷 스케줄링,” 정보과학회논문지 : 시스템 및 이론, 제33권 제10호, 722-733쪽 2006년 10월

〔19〕 R. Rejaie, M. Handley, and D. Estrin, “RAP: An end-to-end rate-based congestion control mechanism for realtime streams in the internet,” In Proc. of IEEE INFOCOM, 3, pp.1337-1345, Mar. 1999.

〔20〕 The network simulator - ns-2 Information Sciences Institute [online] available: http://nsnam.isi.edu/nsnam/index.php/user information.

〔21〕 Video traces file [online] available: http://www.dmclab.hanyang.ac.kr/data/mpeg2data/video_traces.htm.

〔22〕 Y. Chen, Y. Hu, O. Au, H. Li, and C. Chen, “Video error concealment using spatio-temporal boundary matching and partial differential equation,” IEEE Trans. on Multimedia, 10(1), pp. 2-15, Jan., 2008.

〔23〕 J. Mitchell, W. Pennebaker, C. Fogg, and D. LeGall, “MPEG video compression standard,” Champman and Hall, 1996.

〔24〕 M. Horowitz, A. Joch, F. Kossentini, and A. Hallapuro, “H.264/avc baseline profile decoder complexity analysis,” IEEE Trans. on circuits and systems for video technology, 13(7), pp.704-716, 2003.

[25] Sungwoo Hong, Youjip Won, “SAPS: Significance-Aware Packet Scheduling for Real-time Streaming of Layer Encoded Video,”

The IEEE International Workshop on Data Semantics for Multimedia Systems and Applications (DSMSA2008), Berkeley, California, USA Dec., 2008

[26] S. Giordano, M. Pagano, R. Pannocchia, and F. Russa., "A new call admission control scheme based on the self similarnature of multimedia traffic," In Proc. of IEEE Communications, Conference Record, Converging Technologies for Tomorrow's Applications, No.3, pp.1612-1618, Dallas, TX, USA, June, 1996.

홍 성 우 (Sung-woo Hong) 정회원

2005년 2월 한양대학교 전자통신컴퓨터공학과

2005년~현재 한양대학교 전자통신컴퓨터공학과 박사과정

<관심분야> 멀티미디어, 무선 네트워크, 운영체제

원 유 집 (You-jip Won) 정회원

1990년 2월 서울대학교 전산학과 학사

1992년 2월 서울대학교 전산학과 석사

1997년 2월 미네소타 대학교 전산학과 박사

1999년 현재 한양대학교 교수

<관심분야> 운영체제, 파일 시스템, 네트워크