

네트워크 및 버퍼 상태를 모두 고려한 혼합형태의 비디오 스트림 전송기법

(A Hybrid Transmission Scheme for Efficient Video Streaming)

이 선 현 * 정 광 수 **
(Sunhun Lee) (Kwangsue Chung)

요약 기존의 스트림 전송기법은 네트워크 안정화만을 추구한 나머지 사용자 관점에서 스트리밍 응용 프로그램의 특성을 간과하는 문제점을 가진다. 또한 스트리밍 서비스에 큰 영향을 미치는 미디어의 특성에 대한 고려가 없다는 한계를 가지고 있다. 본 논문에서는 기존 스트리밍 전송기법의 한계를 극복하기 위해서 네트워크 관점의 요구사항과 서비스 사용자의 요구사항을 동시에 고려한 혼합된 형태의 HAViS (Hybrid Approach for Video Streaming) 전송기법을 제안하였다. 제안하는 HAViS 전송기법은 기존 연구와 유사하게 네트워크 상태에 적합하도록 전송률을 조절함으로써 네트워크의 안정성을 향상시킬 뿐만 아니라, 수신단 버퍼의 underflow나 overflow를 효과적으로 예방하여 끊김없는 부드러운 스트리밍 서비스를 제공한다. 또한 서비스되는 미디어 스트림의 특성을 고려하여 설계하였다. 실험 결과를 통해서 제안한 HAViS 전송기법이 버퍼자원 사용의 효율성을 높이며 네트워크 안정성을 향상시키고, 사용자에게 끊김없이 부드러운 스트리밍 서비스를 제공할 수 있음을 확인할 수 있었다.

키워드 : 스트리밍 프로토콜, 전송률 조절, 혼잡 제어, 미디어 특성 인지

Abstract Existing streaming mechanisms have no consideration for the characteristics of streaming applications because they only consider network stability. In this paper, in order to overcome limitations of the previous work on video streaming, we propose a new video streaming mechanism called "HAViS(Hybrid Approach for Video Streaming)". The proposed mechanism includes more sophisticated features that consider both network and user requirements. Therefore, the HAViS mechanism improves the network stability by adjusting the sending rate of video stream based on the network state and it also provides the smooth playback by preventing the buffer underflow or overflow. Moreover, it is designed to take into consideration the streaming video content. Through the simulation, we prove that the HAViS mechanism efficiently uses the buffer resources and provides improved network stability and smooth playback.

Key words : Streaming protocol, Rate control, Congestion control, Media-aware

1. 서론

현재 인터넷에서의 주요 트래픽들은 전송 프로토콜로 대부분 TCP(Transmission Control Protocol)를 사용하고 있다. TCP는 종단간(end-to-end)의 신뢰적인 패킷 전달을 수행하며 혼잡 제어 메커니즘(congestion control mechanism)을 통해 송신단의 전송률을 직접 제어

하여 혼잡 상황으로 발생하는 데이터의 무분별한 손실을 막고 네트워크가 안정하게 동작할 수 있도록 한다[1].

최근 들어 오디오나 비디오 스트리밍과 같은 멀티미디어 트래픽이 증가하고 있다. 이러한 트래픽들은 지연에 민감하고 어느 정도의 패킷 손실을 허용하며 전송률을 기반으로 동작하는 특징을 가진다. 멀티미디어 트래픽의 이러한 본질적인 특징으로 인해 신뢰적인 패킷 전달을 수행하는 TCP는 멀티미디어 트래픽의 전송 프로토콜로는 적합하지 않다. 따라서 대부분의 멀티미디어 트래픽은 혼잡 제어를 수행하지 않는 UDP(User Datagram Protocol)를 기반으로 데이터를 전송하게 되었다. 하지만 UDP는 TCP와 같은 혼잡 제어 메커니즘이 없으므로 네트워크 혼잡 상황을 야기하여 네트워크를 붕

* 본 연구는 정보통신부 및 정보통신연구진흥원의 대학 IT연구센터 육성 지원사업의 연구결과로 수행되었음

† 학생회원 : 광운대학교 전자통신공학과
sunlee@adams.kw.ac.kr

** 종신회원 : 광운대학교 전자통신공학과 교수
kchung@kw.ac.kr

논문접수 : 2006년 2월 2일
심사완료 : 2007년 3월 22일

괴(congestion collapse)시킬 가능성이 있으며 기존의 주요 인터넷 트래픽인 TCP와도 친화적(TCP-friendly)으로 동작하지 않는다는 문제점을 갖는다[1,2].

1990년대 이후로 스트리밍 전송기법과 관련하여 네트워크 안정성을 향상시키기 위해, 혼잡 제어 메커니즘 적용에 대한 연구가 활발하게 진행되고 있다[2~13]. 이러한 연구들은 네트워크의 상태에 따라 전송률을 조절함으로써 무분별한 패킷 손실을 막고, 네트워크의 안정성을 향상시키는 것을 목적으로 하고 있다. 하지만 이러한 기존의 연구들은 네트워크상태의 안정화만을 고려하므로 스트리밍 응용 프로그램의 본질적인 특성에 대한 고려가 부족하다는 한계를 가진다. 또한 스트리밍 서비스의 기본적인 요소인 미디어 스트림의 특성에 대한 고려가 부족하다는 한계점을 가지고 있다.

본 논문에서는 현재의 네트워크 상태와 전송되는 비디오 데이터의 특성을 모두 고려하는 스트리밍 전송기법으로서, HAViS(Hybrid Approach for Video Streaming) 기법을 제안한다. HAViS는 네트워크 상태에 따라 전송률을 조절하며, 서비스되는 비디오 데이터의 특성을 고려하였다. 즉, 기존의 관련 연구들이 추구하는 네트워크 안정성 향상을 만족시키면서 사용자 관점에서 끊임없이 부드러운 비디오 재생을 위해 비디오 품질 및 수신단의 버퍼링 레벨을 조절하게 된다.

본 논문의 2장에서는 스트리밍 전송기법과 관련하여 기존 연구들을 분류 및 기술하였고, 3장에서는 제안하는 HAViS 기법의 알고리즘에 대해 상세히 기술하였다. 4장에서는 시뮬레이터를 이용하여 제안하는 알고리즘의 성능을 검증하고, 기존 스트리밍 전송기법과의 성능 비교 결과를 기술하였으며, 마지막으로 5장에서 결론을 맺었다.

2. 관련 연구

인터넷의 주요 트래픽에 사용되는 전송 프로토콜인 TCP는 멀티미디어 스트리밍 서비스의 요구사항을 만족시키지 못하므로 대부분의 스트리밍 트래픽은 UDP를 사용하게 된다. 하지만 혼잡 제어 메커니즘이 없는 UDP를 전송 프로토콜로 사용함으로써 네트워크의 불안정을 발생시키게 되며 TCP를 사용하는 트래픽과 네트워크 대역폭을 공평하게 사용하지 못하는 문제점을 가지고 있다. 1990년대 이후로 이러한 문제점을 개선하기 위해 혼잡 제어 메커니즘을 적용한 스트리밍 전송기법에 관한 연구가 활발하게 진행되고 있다. 본 장에서는 기존에 혼잡 제어 메커니즘을 적용한 기존의 스트리밍 관련 연구들에 대해 설명하고 분류하도록 한다. 그리고 기존 연구들이 가지는 문제점에 대해서 기술하도록 한다.

2.1 스트리밍 전송기법에 대한 관련 연구

혼잡 제어 메커니즘이 없는 UDP 기반의 스트리밍 서비스는 네트워크 불안정을 발생시킨다는 중요한 문제점을 갖는다. 이러한 문제점을 해결하기 위해 기존에 제안된 스트리밍 전송기법들은 주로 전송률 조절을 통한 네트워크 안정성 향상을 목적으로 하고 있다. 이러한 연구들은 네트워크 상태에 따라 전송률을 조절함으로써 네트워크의 안정성을 향상시키며, 부드러운 비디오 재생을 위해 전송률의 변화를 감소시키게 된다.

네트워크 상태에 따라 전송률을 조절하는 기존 연구들은 (1) TCP의 혼잡 제어 알고리즘의 수정이나 (2) TCP 트래픽의 평균 전송률 계산, 또는 (3) 네트워크 가용 대역폭 측정과 같은 방법을 이용하여 비디오 스트림의 전송률을 조절하게 된다.

TCP의 윈도우 기반의 혼잡 제어 알고리즘인 AIMD(Additive Increase and Multiplicative Decrease)를 수정하여 전송률을 조절하는 기법으로는, TCP Vegas의 전송률 조절 기법을 사용하는 SCP(Streaming Control Protocol), 네트워크상태에 따라 패킷의 전달 주기를 조절하는 전송률 기반의 혼잡 제어 알고리즘인 RAP(Rate Adaptation Protocol)와 TLFC(TCP-Like Flow Control), TCP의 흐름 제어(flow control)기법을 수신단에서 에뮬레이션하는 TEAR(TCP Emulation At Receiver), TCP의 혼잡 제어 알고리즘에서 발생하는 전송률의 변화크기를 감소시킨 SQRT등이 있다[2-6].

두 번째로 TCP 트래픽의 평균 전송률 계산을 이용하여 전송률을 조절하는 기법들은 네트워크 상태를 고려한 TCP 친화적인 전송률 조절을 위해 경쟁하는 TCP 트래픽의 평균 전송률을 추정하여 비디오 스트림의 전송률을 조절한다. 식 (1)은 TCP 트래픽의 평균 전송률을 간단하게 계산하는 방법으로, S는 패킷의 크기, RTT(Round Trip Time)는 종단간의 지연, p는 패킷 손실률을 의미한다. Padhye는 중복된 ACK(Acknowledgement) 패킷의 수신이나 전송 타이머의 타임아웃(timeout)값을 고려하여 TCP의 전송률을 보다 정확하게 계산하였다. 식 (2)는 Padhye가 제안한 TCP의 전송률 모델링 공식을 보여준다[14].

$$T = \frac{S}{RTT\sqrt{p}} \quad (1)$$

$$T = \frac{S}{t_{RTT}\sqrt{\frac{2p}{3}} + t_{RTO}(3\sqrt{\frac{3p}{8}})p(1+32p^2)} \quad (2)$$

t_{RTT} 는 종단간 지연, t_{RTO} 는 재전송 타이머의 값인, RTO(Retransmission Time-Out)를 의미한다. TCP 트래픽의 평균 전송률 계산을 이용하여 전송률을 조절하는 기법들은 식 (1)이나 식 (2)를 사용하여 비디오 스트림

의 전송률을 조절하며, 대표적으로 TFRC(TCP-Friendly Rate Control Protocol)와 TFRC(TCP-Friendly Rate Control)가 있으며, 이밖에 MSTFP(Multimedia Streaming TCP-Friendly transport Protocol)나 LDA(Loss Delay based Adaptation)등이 있다[7-10].

네트워크 가용 대역폭 측정을 사용하는 스트림 전송 기법에는 네트워크의 가용 대역폭을 주기적으로 측정하고, 네트워크의 혼잡 상태가 발생할 경우, 측정된 가용 대역폭의 값으로 전송률을 조절하는 SMCC(Streaming Media Congestion Control)와 전송률뿐만 아니라 서비스되는 비디오의 품질도 함께 조절하는 VTP(Video Transport Protocol)가 있다. SMCC나 VTP의 가용 대역폭 측정 알고리즘은 TCP Westwood에서 제안한 패킷쌍(packet-pair) 기반의 가용 대역폭 측정 알고리즘을 그대로 사용하게 된다[11,12].

네트워크 상태에 따라 전송률을 조절하는 기존 연구들은 네트워크 상태가 혼잡할 경우, 전송률을 낮춤으로써 네트워크의 안정성 향상을 추구한다. 이와 다른 관점으로 멀티미디어 스트리밍의 특성을 고려하여, 수신단에서 끊임없이 부드럽게 재생되는 것을 목적으로 하는 전송기법으로 Buffer-driven scheme이 있다. 일반적으로 멀티미디어 스트리밍 응용은 수신단에 일정 크기의 버퍼를 사용하여 네트워크에서 발생하는 지연이나 지터(jitter)를 최소화한다. Buffer-driven scheme은 수신단의 현재 버퍼상태를 송신단에서 예측하여 버퍼의 underflow나 overflow가 발생하지 않도록, 서비스되는 비디오의 품질을 조절함으로써 서비스 사용자들에게 끊임없이 부드러운 비디오 재생을 가능하게 한다[13].

그림 1은 앞서 설명한 멀티미디어 스트리밍 전송 기법에 관련된 기존 연구들을 전송률 조절 방법에 따라 분류한 것이다. 본 논문에서 제안하는 HAViS 전송기법은 대부분의 기존 스트리밍 전송 기법이 목적으로 가지는 네트워크 안정성 향상을 추구함과 동시에 실질적으로 스트리밍 응용에서 유용하게 사용되는 버퍼 관리 기법을 사용하여 끊임없이 부드러운 스트리밍 서비스를 제공하는 것을 목적으로 한다. 즉, 네트워크 관점과 사용자 관점을 모두 고려하는 혼합된 형태를 가진다. 또한 서비스되는 비디오의 품질 특성을 고려한 버퍼 관리 기

법을 통해 효율적인 비디오 스트리밍 서비스를 제공하게 된다.

2.2 기존 연구의 문제점

지연에 민감하고 어느 정도의 패킷 손실을 허용하며, 전송률 기반으로 동작하는 스트리밍 서비스의 특성으로 인해, 일반적으로 멀티미디어 스트리밍은 UDP 기반으로 동작하게 된다. 하지만 UDP는 혼잡 제어 메커니즘의 부재로 인해 네트워크 혼잡 상황을 유발한다는 중요한 문제점을 가지며 이러한 문제점을 개선하기 위해 2.1절에서 기술한 다양한 관련 연구들이 수행되었다. 이러한 연구들은 TCP의 혼잡 제어 메커니즘을 수정하거나 TCP의 평균 전송률 또는 네트워크의 가용 대역폭으로 비디오 스트림의 전송률을 조절하게 된다. 하지만 이러한 연구들은 네트워크의 안정성만을 고려하므로 비디오 스트리밍의 기본적인 요구사항에 대한 고려가 부족하다고 할 수 있다. 즉, 네트워크 상태에 따른 전송률 조절을 통해 패킷 손실이 줄어들고, 네트워크가 안정화되는 장점은 있으나, 이것이 스트리밍 서비스를 제공받는 사용자에게 있어서의 서비스 품질에 대한 향상을 보장하지는 않는다.

사용자에게 끊임없이 부드러운 스트리밍 서비스를 제공하기 위해서는 전송률의 변화폭이 작아야하며, 수신단에 존재하는 버퍼의 underflow나 overflow를 효과적으로 예방할 수 있어야 한다. 이러한 요구사항을 고려한 관련 연구로, 2.1절에서 소개한 Buffer-driven scheme이 있다. Buffer-driven scheme은 수신단의 버퍼 상태를 송신단에서 예측하여 서비스되는 비디오의 품질을 조절하게 된다. 비디오의 품질 조절을 통해 수신단의 버퍼가 underflow나 overflow가 발생하는 것을 예방하게 되며 이를 통해 사용자에게 끊임없이 부드러운 비디오 전송을 제공하게 된다. 하지만 Buffer-driven scheme은 사용자 관점에서, 버퍼의 상태만을 고려하므로 네트워크 상태에 대한 고려가 부족하다는, 기존 UDP 기반의 스트리밍 프로토콜의 단점을 계승하게 된다.

식 (3)은 Buffer-driven scheme이 가지는 시간 t 에서의 송신단의 버퍼상태 예측 알고리즘을 나타낸 것이다. s 는 송신단 버퍼에 들어가는 데이터, α_s 는 송신단의 전송률, $B_s(0)$ 는 송신단의 초기 버퍼링 상태를 의미한다.

혼잡제어 기법 수정	TCP 전송률 모델링공식	가용대역폭 측정	버퍼 상태기반 품질 조절
<ul style="list-style-type: none"> • SCP [2] • RAP [3] • TLFC [4] • TEAR [5] • SQRT [6] 	<ul style="list-style-type: none"> • TFRC [7] • TFRC [8] • MSTFP [9] • LDA [10] 	<ul style="list-style-type: none"> • SMCC [11] • VTP [12] 	<ul style="list-style-type: none"> • Buffer-driven scheme [13]

그림 1 기존 연구 분류

송신단의 버퍼상태는 송신단에서 직접적인 버퍼상태 모니터링으로 예측가능하다. 마찬가지로 식 (4)는 시간 t 에서의 수신단의 버퍼상태 예측 알고리즘을 나타내고 있다. a_r 은 수신단이 받은 수신률, d 는 응용프로그램에 의해 소비되는 데이터, $B_r(0)$ 는 수신단의 초기 버퍼링 상태를 의미한다. 현재 수신단의 버퍼상태 예측은 네트워크의 지연이나 패킷 손실을 무시할 때, 식 (3)과 식 (4)를 조합하여 식 (5)와 같이 전개될 수 있다. f_j 는 j 번째 비디오 프레임의 바이트수를 의미한다. 전송되는 비디오의 품질 조절을 통한 전송률 조절은 수신단의 버퍼상태 예측을 기반으로 하며 수신단의 버퍼상태는 송신단에서 전송한 패킷 수와 비디오 인코딩율에 따라 예측 가능하다[13].

$$B_s(t) = \sum_{j=1}^t s(j) - a_s(j) + B_s(0) = B_s(t-1) + s(t) - a_s(t) \quad (3)$$

$$B_r(t) = \sum_{j=1}^t a_r(j) - d(j) + B_r(0) = \sum_{j=1}^t a_r(j) - f_j + B_r(0) \quad (4)$$

$$B_r(t) = \sum_{j=1}^t s(j) - f_j + B_s(0) + B_r(0) - B_s(t) \quad (5)$$

인터넷 환경에서 효율적인 비디오 스트리밍 서비스를 위해서는 네트워크 상태를 고려하여 전송률을 조절함으로써 네트워크의 안정성을 향상시키고 동시에, 사용자에게 끊임없이 부드러운 비디오 재생을 제공할 수 있어야 한다. 따라서 네트워크 상태나 버퍼 상태, 하나만을 고려하는 기존의 연구들은 한계가 있으며 이를 해결하기 위해서는 네트워크 상태에 따라 전송률을 조절함으로써 네트워크 혼잡 상황을 회피하고, 서비스되는 비디오의 특성에 따른 적절한 버퍼 관리 기법을 통해 비디오 스트림을 전달하는 효율적인 전송 기법이 필요하다.

3. HAViS 전송기법

본 장에서는 2장에서 지적한 기존 스트리밍 전송기법들의 문제들을 개선하기 위해서 새롭게 제안한 HAViS (Hybrid Approach for Video Streaming) 전송기법에 대해 기술한다. 제안하는 HAViS 전송기법은 네트워크 상태에 따라 비디오 스트림의 전송률을 조절하며 사용자에게 끊임없이 부드러운 스트리밍 서비스를 제공하기 위해 송신단에서 예측한 수신단의 버퍼 상태를 기반으로 서비스되는 비디오의 품질을 조절한다. 또한 비디오의 품질 변화율이 줄이기 위해 미디어의 특성을 고려한 버퍼 관리 기법을 적용하였다. 이를 통해 제안하는 HAViS 전송기법은 기존 스트리밍 전송기법의 문제점을 개선하고 효율적인 비디오 스트리밍 서비스를 제공하게 된다.

3.1 HAViS 개요

기존 스트리밍 전송기법들은 UDP 기반의 멀티미디어 전송 프로토콜이 가지는 혼잡 제어 메커니즘의 부재라는 문제를 개선하기 위해 주로 혼잡 제어 메커니즘 적용을 통한 네트워크 안정성 향상을 목적으로 하고 있다. 이와는 다른 관점으로, 일반적으로 스트리밍 응용에서 사용되는 버퍼 관리 기법을 통한 전송 기법이 제안되었으나, 네트워크 상태에 대한 고려가 부족하다는 한계를 갖는다. 또한 두 가지 접근 방법 모두 멀티미디어 스트리밍에서 서비스되는 미디어의 특성에 대한 고려가 부족하다는 공통적인 한계를 갖는다.

본 논문에서는 효율적인 스트리밍 서비스를 제공하기 위한 새로운 전송기법으로 HAViS를 제안한다. HAViS 전송기법은 네트워크 관점에서 네트워크 상태에 따라 전송률을 조절함으로써 네트워크의 안정성을 향상시키며, 사용자에게 제공되는 서비스 관점에서 부드러운 비디오 재생을 지원하기 위해 비디오 품질 조절을 통해 버퍼의 underflow와 overflow를 예방한다. 또한 비디오 스트림의 비트율(bit-rate)로 대표되는 미디어의 특성을 고려해서 효과적인 버퍼 관리를 수행하게 된다.

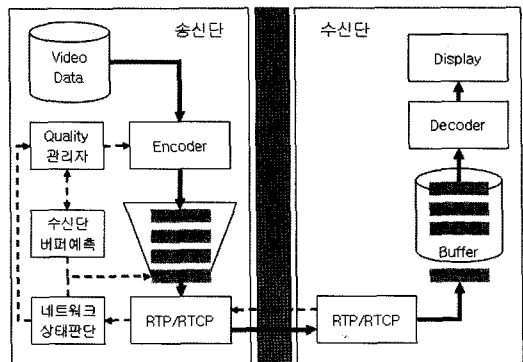


그림 2 HAViS 전송기법의 전체 구조

그림 2는 HAViS의 종단간 전체 구조를 나타낸 것이다. HAViS 전송기법은 기본적으로 IETF(Internet Engineering Task Force)에서 제정한 표준 스트리밍 프로토콜인 RTP/RTCP(Realtime Transport Protocol/Realtime Transport Control Protocol) 모델을 사용하게 된다. 제어 프로토콜인 RTCP는 패킷 손실율이나 RTT등의 현재 네트워크 상태에 대한 정보를 주기적으로 송신단에 알려준다[15]. 이러한 RTCP 패킷의 네트워크 상태 정보를 바탕으로 HAViS 전송기법은 현재의 네트워크 상태를 혼잡상태나 안정상태로 판단하게 된다.

수신단의 RTCP 수신자 보고(RR:Receiver Report) 패킷의 정보를 통해 네트워크 상태를 판단한 후, HAViS

는 수신단의 버퍼상태를 예측하게 된다. 종단간 지연이 큰 네트워크 환경까지 고려하기위해 RTT 이후의 수신단 버퍼상태를 예측하며, 이렇게 예측된 수신단의 버퍼상태와 현재의 네트워크 상태에 따라 스트리밍되는 비디오의 전송률 및 품질을 조절하게 된다. RTT 이후의 수신단 버퍼상태가 underflow나 overflow가 예상될 경우, 전송률의 변화없이, 스트리밍되는 비디오의 품질만을 조절함으로써 끊김없이 부드러운 재생을 제공하게 되며 버퍼의 underflow나 overflow의 위험이 없을 경우에는 비디오의 품질 변화없이 네트워크 상태에 따라 전송률만을 증가시키거나 감소시키게 된다.

이후 3.2절에서는 RTCP 패킷의 피드백 정보에 기반한 네트워크 상태 판단 알고리즘에 대해 설명하며 3.3절에서는 HAViS 전송기법이 가지는 수신단의 버퍼상태 예측 알고리즘 및 관리 기법에 대해 상세하게 설명한다. 그리고 3.4절에서는 네트워크 상태와 버퍼 상태를 모두 고려하는 HAViS의 전송률 및 비디오 품질 조절 알고리즘에 대해 설명하도록 한다.

3.2 네트워크 상태 판단

HAViS 전송기법은 패킷 손실률에 기반하여 현재의 네트워크 상태를 판단한다. 패킷 손실률은 RTCP 패킷의 수신자 보고서에서 "Cumulative Number of Packets Lost" 필드 정보를 통해 얻어진다[15].

$$Packet\ Loss\ Rate = 1 - \frac{Number\ of\ Received\ Packets}{Number\ of\ Sent\ Packets} \quad (6)$$

식 (6)은 RTCP 패킷의 수신자 보고를 통해 패킷 손실률에 대한 정보를 송신단에 제공하기 위해, 수신단에서 패킷 손실률을 계산하는 공식을 기술하였다. 즉, 패킷 손실률은 RTCP 패킷의 전달 주기동안 전체 전달된 패킷의 수와 손실된 패킷의 수에 대한 비율로 계산할 수 있다.

계산된 패킷 손실률을 기반으로 HAViS는 현재의 네트워크 상태를 그림 3과 같이 판단하게 된다. 패킷 손실률이 0보다 큰 경우, 네트워크 혼잡에 의한 패킷 손실 발생으로 간주하고 네트워크 상태를 혼잡상태(congestion state)로 판단하며 반대로 패킷 손실이 발생하지 않는 경우에는 현재의 네트워크 상태를 안정상태(stable state)로 판단하게 된다.

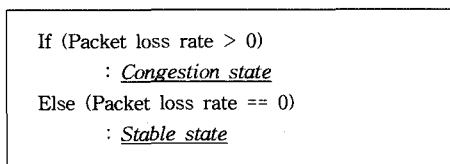


그림 3 네트워크 상태 판단

3.3 수신단 버퍼상태 예측 및 관리 기법

HAViS 전송기법은 스트리밍 서비스 사용자에게 끊김없는 재생 서비스를 제공하기 위해 버퍼의 underflow나 overflow를 예방한다. 버퍼의 underflow나 overflow를 예방하기 위해서는 수신단의 버퍼상태를 수신단의 피드백 정보나, 송신단에서의 예측을 통해 알아야 한다. 수신단의 피드백 정보는 현재 수신단의 버퍼상태를 정확하게 알려줄 수 있다는 장점을 가지지만, 추가적인 제어 메시지가 필요하거나 기존의 RTCP와 같은 제어 메시지의 수정이 필요하다는 단점을 가진다. 또한 제어 메시지가 송신단에 알려질 때의 시점에선 이미 단방향 지연(one-way delay) 이전의 상태 정보가 된다는 한계를 가진다. 이러한 한계를 개선하기위해, 제안하는 HAViS 전송기법에서는 추가적인 제어 메시지의 사용없이, 송신단에서 수신단의 버퍼상태를 예측할 수 있는 버퍼상태 예측 기법을 사용하게 된다. 또한 높은 지연을 가지는 네트워크 환경에서의 버퍼의 underflow나 overflow까지 효과적으로 예방하기 위해서, HAViS는 RTT이후의 수신단의 버퍼상태를 예측하는 기법을 사용한다.

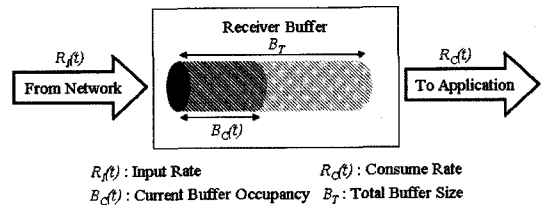


그림 4 수신단 버퍼 상태

그림 4는 임의의 시간에서 수신단의 버퍼상태를 나타내고 있다. B_T 는 수신단의 전체 버퍼크기를 나타내며 $B_C(t)$ 는 현재 수신단의 버퍼상태를 의미하게 된다. $R_r(t)$ 는 네트워크로부터 수신단 버퍼에 들어오는 데이터율을 의미한다. 송신단과 수신단 사이의 네트워크 지연을 무시할 경우, $R_r(t)$ 는 송신단에서의 전송률로 대체할 수 있다. $R_c(t)$ 는 수신단 버퍼로부터 스트리밍 애플리케이션에 의해 소비되는 데이터 소비율을 의미하며 현재 스트리밍되는 비디오 데이터의 비트율로 표현할 수 있다. 그림 4에서 HAViS 전송기법이 가지는 RTT 이후의 수신단 버퍼상태 예측 알고리즘은 식 (7)과 같이 전개할 수 있다.

$$B_E(t+1) = B_C(t) + R_r(t) - R_c(t) \times RTT \quad (7)$$

where, $B_E(t) \approx B_B(t)$

$$R_r(t) = \frac{Number\ of\ Packets \times Packet\ Size}{RTCP\ Period}$$

$$R_c(t) = Encoding\ Rate\ at\ Time(t)$$

HAViS 전송기법은 송신단에서 식 (7)의 알고리즘을 이용하여 RTT 이후의 수신단의 버퍼상태를 예측한다. 이러한 수신단 버퍼상태 예측은 수신단에서 보고되는 RTCP주기마다 주기적으로 수행된다. RTT 이후의 버퍼 점유도 증가, 감소는 네트워크로부터 수신단 버퍼에 유입되는 데이터율인, $R_r(t)$ 과 스트리밍 애플리케이션에 의해 소비되는 데이터 소비율인, $R_c(t)$ 의 차에 RTT를 곱한 것으로 계산된다. RTT 이후의 수신단 버퍼상태 예측 값인, $B_E(t+1)$ 은 현재의 버퍼상태인 $B_C(t)$ 와 RTT 이후의 버퍼 점유도 증가, 감소 성분을 더하여 계산된다. 현재의 수신단 버퍼상태, $B_C(t)$ 는 이전 주기에 구해진 버퍼상태 예측값, $B_E(t)$ 와 근사값을 가진다. 네트워크상태와 버퍼상태에 따른 전송률 및 비디오 품질 결정 이후, $B_C(t)$ 는 새로운 전송률 및 비디오 품질을 적용하여 다시 계산된다. 이것은 다음 주기에서 수행되는 버퍼상태 예측 알고리즘의 정확성을 위한 것이다.

HAViS 전송기법은 수신단 버퍼상태에 대한 정보를 얻기 위해 RTT 이후의 수신단 버퍼상태 예측 알고리즘을 사용한다. RTT 이후, 즉 미래시점의 수신단 버퍼상태를 예측하는 이유는 현재 수신단 버퍼상태를 예측할 경우, 높은 지연을 가지는 네트워크 환경에서는 여전히 버퍼의 underflow나 overflow가 발생할 가능성이 있기 때문이다. 하지만 HAViS가 가지는 RTT 이후의 수신단 버퍼상태 예측 알고리즘은 송신단에서 수신단의 버퍼상태를 예측하므로 추가적인 제어 메시지를 사용하지 않으면서, 높은 지연을 가지는 네트워크 환경에서도 버퍼의 underflow나 overflow를 효과적으로 예방할 수 있다.

수신단의 버퍼상태 예측 기법과 함께 HAViS 전송기법은 스트리밍 서비스에서 서비스되는 비디오나 오디오와 같은 미디어의 특성에 대한 고려를 포함한다. 미디어의 특성에는 여러 가지가 있을 수 있으나, 가장 대표적으로 압축 기법에 따른 비트율을 들 수 있다. 서비스되는 비디오의 비트율은 네트워크 환경이나 수신단의 버퍼 관리에 큰 영향을 미치거나, 받게 된다. 따라서 이러한 미디어의 특성에 따른 수신단의 버퍼 관리가 필요하게 된다.

HAViS는 수신단 버퍼의 underflow나 overflow를 예방하기 위해 버퍼 점유도에 대한 임계값(threshold)을 설정한다. 이러한 임계값이 너무 큰 값으로 설정이 될 경우, 버퍼상태 변화에는 능동적으로 반응하지만 네트워크상태 변화에 대해 소극적으로 반응하게 된다. 반대로 임계값이 너무 작은 값으로 설정이 될 경우, 네트워크상태 변화에는 능동적으로 반응하지만 버퍼상태 변화에 대해 소극적으로 반응하게 된다. 이처럼 임계값 설정은

네트워크 상태 변화에 반응하는 구간과 버퍼상태 변화에 반응하는 구간으로 구분하는 역할을 수행하므로 중요한 기준이 된다. 비디오 스트림의 비트율을 임계값 설정에 반영하지 않을 경우, 송신단에서 수행하는 수신단 버퍼의 underflow나 overflow 예측이 부정확할 수 있다는 문제점을 갖는다. 이러한 문제는 불필요한 전송률 조절이나 비디오 품질의 조절로 인해 사용자에게 제공되는 스트리밍 서비스의 품질을 저하시킬 수 있다는 문제점으로 연결된다. 이러한 문제를 개선하기 위해, HAViS 전송기법에서는 임계값 설정의 기준으로, 현재 서비스되는 비디오 스트림의 비트율을 사용한다. 즉, 현재 서비스되는 비디오 스트림의 비트율에 따라 underflow나 overflow 예측에 대한 기준이 되는, 임계값을 결정함으로써 버퍼 자원의 효율성을 증가시킬 수 있으며 불필요한 전송률 및 비디오 품질 조절을 줄일 수 있다.

식 (8)은 HAViS의 미디어 특성을 고려한 임계값 설정을 보여준다. 버퍼의 underflow를 예방하기 위한 최소 임계값, MinTH는 현재 서비스되는 비디오 스트림의 인코딩 비트율, RENC로 1초동안 소비될 수 있을 버퍼상태를 의미하며 전체 버퍼 크기, BT에 대한 비율로 나타난다. 버퍼의 overflow를 예방하기 위한 최대 임계값, MaxTH는 전체 100%에서 MinTH의 차로 계산된다.

$$\begin{aligned} \text{MinTH}(\%) &= \frac{R_{ENC}(\text{tps})/8}{B_T(\text{bytes})} \times 100 \\ \text{MaxTH}(\%) &= 100 - \text{MinTH} \end{aligned} \quad (3)$$

3.4 전송률 및 비디오 품질 조절 알고리즘

HAViS 전송기법은 네트워크 상태와 미디어 특성을 고려한 버퍼상태에 따라 서비스되는 비디오 스트림의 전송률 및 비디오 품질을 조절한다. 이처럼 네트워크 관점의 요구사항과 사용자에게 서비스되는 관점을 모두 고려하는 혼합된 형태의 HAViS는 일차적으로 3.2절에서 설명한 패킷 손실률에 근거해서 네트워크 상태를 혼잡상태와 안정상태로 구분하게 된다. 네트워크 상태가 결정된 후에는 3.3절에서 설명한 버퍼 상태 예측 및 관리 기법에 의해 RTT 이후의 수신단의 버퍼상태를 예측하고 각각의 경우에 대해서 서비스되는 비디오 스트림의 전송률 및 비디오 품질을 조절하게 된다.

그림 (9)는 HAViS 전송기법의 전송률 및 비디오 품질에 대한 조절을 보여준다. 네트워크상태가 혼잡상태로 판단될 경우, 전송률을 줄임으로써 네트워크를 안정화시킬 필요성이 있다. 하지만 RTT 이후의 수신단의 버퍼상태가 ($C \sim \text{MinTH}\%$)로 예측되는 경우, 버퍼의 underflow가 발생할 가능성이 있으므로 이를 예방하기 위해 서비스되는 비디오의 품질을 낮추게 된다. 마찬가지로 ($\text{MaxTH} \sim 100\%$)로 예측되는 경우, 버퍼의 overflow가 발생할 가능성이 있으므로 이를 예방하기 위해 비디오

스트림의 품질을 높게 된다. 버퍼의 underflow나 overflow가 예상될 경우, 현재의 전송률은 그대로 유지하고 서비스되는 비디오의 품질을 조절함으로써 underflow 예상 경우에는 버퍼링 레벨을 높게 되며, overflow 예상 경우에는 버퍼링 레벨을 낮추게 된다. 버퍼의 underflow나 overflow의 위험이 없는 {MinTH ~ MaxTH%} 구간에서는 비디오 품질의 변화없이, 네트워크의 안정성 향상을 위해 전송률을 α 만큼 감소시키게 된다.

네트워크상태가 안정상태로 판단되는 경우, 혼잡상태와 마찬가지로 RTT 이후의 수신단 버퍼상태가 (0 ~ MinTH%)로 예측되는 경우에는 비디오 품질을 낮추며, {MaxTH ~ 100%}로 예측되는 경우에는 비디오의 품질을 높게 된다. 혼잡상태와 마찬가지로, 이 두 경우에는 전송률의 변화없이 서비스되는 비디오 품질만을 조절하게 된다. 하지만 버퍼의 underflow나 overflow의 위험이 없는 {MinTH ~ MaxTH%} 구간에서는 혼잡상태에서 감소한 전송률의 회복을 위해 전송률을 β 만큼 증가시키게 된다.

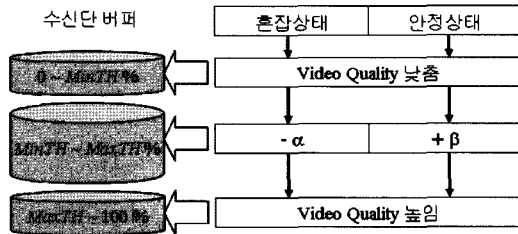


그림 5 전송률 및 품질 조절

버퍼의 underflow나 overflow 위험이 없는 경우, HAViS 전송기법은 네트워크 안정화를 위해 전송률을 증가 또는 감소시키게 된다. 전송률을 조절하는 감소, 증가 파라미터인 α, β 는 경쟁하는 TCP 트래픽과의 형평성을 고려하기위해 TCP의 혼잡 제어 메커니즘을 모방하여 결정한다. 네트워크가 혼잡상태일 경우, 다음 RTCP 주기의 전송률은 현재의 전송률, T_C 에서 α 만큼 감소하게 되며, α 는 패킷 손실률, p 의 제곱근에 비례해서 전송률을 감소시키는 TCP의 성질을 이용하여 식 (9)와 같이 계산된다. 전송률 증가 파라미터인 β 도 α 와 마찬가지로 TCP의 혼잡 제어 메커니즘을 모방하여 결정한다. 기존의 TCP에서 전송률의 선형적인 증가는 RTT마다 하나의 패킷만큼 증가시킨다. RTP/RTCP 모델 기반의 HAViS 전송기법은 전송률의 조절주기가 RTT가 아닌 하나의 RTCP 주기가 되므로 TCP의 전송률 조절주기에 맞추어 비디오 스트림의 전송률을 조절할 필요가 있다. 따라서 하나의 RTCP 구간에서 증가시킬 패킷의 수

인, P_{Inc} 를 우선적으로 구하고, 이를 중단간 지연, t_{RTT} 로 나누어 전송률 증가성분, β 를 구하며 이렇게 얻어진 전송률 증가성분을 현재의 전송률과 합하여 다음 RTCP 주기에서 서비스되는 비디오 스트림의 전송률을 조절하게 된다. 이러한 전송률 감소, 증가 파라미터 α 와 β 를 사용해서 전송률을 조절함으로써 HAViS는 네트워크 상태에 따라 전송률을 조절하여 네트워크의 안정성을 향상시키며, 전송률의 변화폭을 줄임으로써 안정적인 스트리밍 서비스를 제공하게 된다.

$$\alpha = T_C \times \sqrt{p} \tag{9}$$

$$\beta = \frac{P_{\in c}}{t_{RTT}}, P_{\in c} = \frac{RTCP \text{ period}}{t_{RTT}} \tag{10}$$

```

If (Packet loss rate > 0) // Congestion state
  If (Buffer occupancy < MinTH)
    Video quality 감소
    MinTH, MaxTH 재설정
  Else if (MinTH < Buffer occupancy < MaxTH)
    Sending rate -=  $\alpha$ 
  Else (Buffer occupancy > MaxTH)
    Video quality 증가
    MinTH, MaxTH 재설정

Else (Packet loss rate == 0) // Stable state
  If (Buffer occupancy < MinTH)
    Video quality 감소
    MinTH, MaxTH 재설정
  Else if (MinTH < Buffer occupancy < MaxTH)
    Sending rate +=  $\beta$ 
  Else (Buffer occupancy > MaxTH)
    Video quality 증가
    MinTH, MaxTH 재설정
    
```

그림 6 HAViS 전송기법

그림 6은 지금까지 설명한 HAViS 전송기법이 가지는 스트리밍 비디오의 전송률 및 비디오 품질 조절 알고리즘을 정리한 것이다. 먼저 패킷 손실률에 근거하여 네트워크 상태를 혼잡상태와 안정상태로 판단하게 되며 RTT 이후의 수신단의 버퍼상태 예측을 기반으로 서비스되는 비디오 스트림의 전송률 및 비디오 품질을 조절하게 된다. 그리고 서비스되는 미디어의 특성, 즉 비디오 스트림의 비트율을 고려한 버퍼 관리를 위해 비디오 품질이 변화할 경우, 수신단 버퍼의 underflow나 overflow 가능성을 판단하는 기준 임계값을 재설정하게 된다.

4. 실험 및 성능 평가

본 장에서는 새로 제안한 HAViS(Hybrid Approach for Video Streaming) 전송기법의 성능 평가를 위해

LBNL(Lawrence Berkely National Laboratory)의 ns-2(network simulator)를 사용하여 다양한 실험을 수행하였다[16].

4.1 실험 환경

제한한 HAViS 전송기법의 성능을 평가하기 위해서 그림 7과 같은 실험 환경을 구성하여 성능 실험을 수행하였다. HAViS 전송기법의 성능 및 기존 연구와의 성능 비교를 위해 배경 트래픽으로 TCP 트래픽을 사용하였다. TCP를 배경 트래픽으로 사용하는 이유는 스트리밍 데이터가 전달되는 인터넷 환경에서 대부분의 주요 트래픽들이 TCP를 기반으로 동작하기 때문이다.

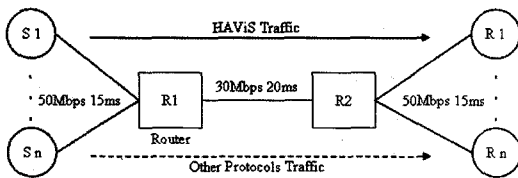


그림 7 실험 환경

사용하는 스트리밍 비디오의 품질은 20Mbps에서 4Mbps까지 5가지 품질을 사용하였으며, 각 비디오 품질의 차이는 4Mbps로 구분하였다. 수신단의 전체 버퍼 사이즈는 8Mbytes로 설정하였으며 비디오가 재생되기 전에 미리 버퍼링되는 초기 버퍼 점유도는 50%로 가정하고 실험하였다.

4.2 버퍼상태 예측 및 임계값 설정

본 절에서는 HAViS 전송기법의 수신단의 버퍼상태 예측의 정확성과 서비스되는 비디오 품질에 따른 임계값의 설정에 대한 실험 결과를 설명한다. HAViS 전송기법에서 버퍼상태 예측과 미디어 특성을 고려한 임계값 설정은 서비스되는 비디오 스트림의 전송률 및 비디오의 품질을 조절하는 중요한 기준이 되므로 그 정확성에 대한 검증이 선행되어야 한다.

그림 8은 그림 7의 실험 환경에서 HAViS 전송기법이 가지는 RTT 이후의 수신단의 버퍼상태에 예측기법에 대한 정확성을 검증한 실험 결과이다. 정확성 검증을 위해 실제 수신단에서 버퍼상태의 변화를 기록하여 송신단에서 예측한 수신단 버퍼상태 예측값과 비교하였다. 결과를 통해 HAViS 전송기법의 가장 중요한 알고리즘인, RTT 이후의 수신단 버퍼상태의 예측기법이 버퍼 점유도가 증가하거나 감소하는 것에 상관없이 송신단의 예측값과 실제 수신단의 측정값이 거의 일치하는 것을 확인할 수 있다. 즉, HAViS 전송기법이 가지는 RTT 이후 수신단의 버퍼상태 예측기법에 대한 정확성을 검증할 수 있다.

그림 8의 실험 결과는 버퍼상태 예측에 대한 결과와

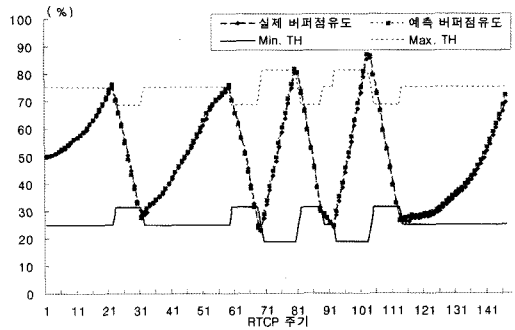


그림 8 버퍼예측 기법의 정확성

함께 현재 서비스되는 미디어의 특성, 즉 비디오 데이터의 비트율에 따른 적응적인 임계값 설정 결과를 보여준다. 3.3절에서 설명하였듯이, 미디어 특성을 고려한 최소 임계값과 최대 임계값의 설정은 버퍼의 underflow나 overflow를 효과적으로 예방함과 동시에 버퍼 자원의 효율적인 사용을 가능하게 한다. 실험 결과는 현재 서비스되는 비디오 스트림의 비트율을 기반으로 HAViS 전송기법이 적응적으로 최소 임계값(MinTH)와 최대 임계값(MaxTH)을 설정하는 것을 확인시켜 준다. 또한 미디어의 특성을 고려한 동적인 임계값 설정 기법이 수신단 버퍼의 underflow나 overflow를 효과적으로 예방함으로써 사용자에게 끊임없이 부드러운 스트리밍 서비스를 제공할 수 있음을 확인할 수 있다.

4.3 HAViS 전송기법의 성능 실험

4.2절의 실험 결과를 통해 제한한 HAViS 전송기법이 가지는 수신단 버퍼상태 예측기법의 정확성을 검증하였고, 서비스되는 비디오 스트림의 특성에 따라 임계값을 적응적으로 설정할 수 있음을 검증하였다. 본 절에서는 TCP 트래픽과 경쟁하는 상황에서 HAViS 전송기법의 기본적인 성능을 실험을 통해 확인한다.

그림 9는 TCP 트래픽과 경쟁하는 상황에서 HAViS 전송기법의 기본적인 성능에 대한 실험 결과를 보여준다. 그림 9(a)는 네트워크 상태 및 수신단 버퍼상태 예측에 따른 HAViS의 전송률 변화 및 서비스되는 비디오 품질의 변화 나타낸 것이다. 시뮬레이션의 시작에서 HAViS 전송기법은 기본적으로 16Mbps의 비트율을 가지는 비디오 스트림을 전송하게 되며 TCP 트래픽과 경쟁하는 상황에서 네트워크 상태가 혼잡상태로 판단되는 경우, 네트워크 안정화를 위해 전송률을 감소시키고, 네트워크 상태가 안정상태로 판단되는 경우, 전송률을 증가시키게 된다. 실험 결과를 통해 확인할 수 있듯이, HAViS 전송기법은 TCP 트래픽과 비교하여 전송률의 변화폭을 줄임으로써 안정적인 스트리밍 서비스를 제공할 수 있다.

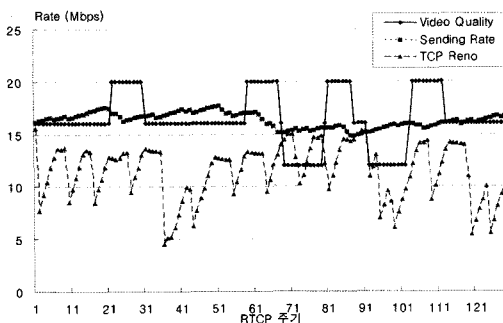
네트워크 상태를 기반으로 하는 전송률 조절 과정에서, RTT이후의 수신단 버퍼상태가 최대 임계값보다 커지는 21, 56, 80, 102번째 RTCP 주기에서는 버퍼의 overflow를 예방하기 위해서 스트리밍되는 비디오의 품질을 높여주게 된다. 이와 다르게, RTT이후의 수신단 버퍼상태가 최소 임계값보다 작아지는 30, 68, 85, 91, 112번째 RTCP 주기에서는 버퍼의 underflow를 예방하기 위해서 스트리밍되는 비디오의 품질을 낮추게 된다.

그림 9(b)는 기존의 표준 스트리밍 프로토콜인 RTP와 제안하는 HAViS 전송기법을 적용한 트래픽을 각각 인터넷의 주요 트래픽인 TCP와 경쟁시켰을 경우, 발생하는 패킷 손실의 분포를 비교한 실험 결과이다. 주로 UDP를 사용하는 RTP는 네트워크 상황에 대한 고려가 없으므로 네트워크가 혼잡한 상태에서도 전송률 조절을 하지 않는다. 하지만 HAViS 전송기법은 네트워크 상태를 고려하여 전송률 조절 기반의 혼잡 제어를 수행하므로 패킷 손실 발생이 크게 줄어들었음을 확인할 수 있다. HAViS 전송기법은 기존 RTP와 비교해서 약 40% 정도 패킷 손실 발생이 감소하였으며 이러한 결과를 통해 HAViS 전송기법이 네트워크의 안정성을 향상시킬

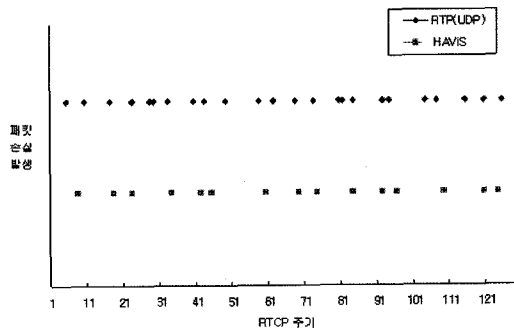
수 있음을 확인할 수 있었다.

HAViS 전송기법은 네트워크 안정성 향상뿐만 아니라, 스트리밍 서비스 사용자에게 끊임없이 부드러운 비디오 재생을 제공하기 위해 버퍼의 underflow와 overflow를 미리 예방하게 된다. 버퍼의 underflow, overflow를 예방하기 위해서 수신단 버퍼에 대한 최소, 최대 임계값을 설정하게 되며 미디어의 특성을 고려하기 위해 전송되는 비디오 데이터의 비트율에 기반하여 이러한 임계값을 결정하게 된다. 미디어 특성을 고려한 임계값 설정은 기본적으로 버퍼의 underflow나 overflow를 예방하고 버퍼 자원의 효율성을 증가시키는 장점을 가진다. 또한, 미디어 특성을 고려한 HAViS 전송기법은 필요한 비디오 품질의 변화를 최소화함으로써 사용자에게 제공되는 서비스의 품질을 향상시키게 된다.

그림 10은 미디어 특성을 고려하지 않고 임의의 값으로 최소, 최대 임계값을 설정했을 경우의 비디오 품질의 변화를 나타낸 실험 결과이다. (a)는 최소 임계값과 최대 임계값을 각각 40%, 60%로 설정하였을 경우의 실험 결과이며 (b)는 최소 임계값과 최대 임계값을 각각 10%, 90%로 설정하였을 경우의 실험 결과이다. 임계값

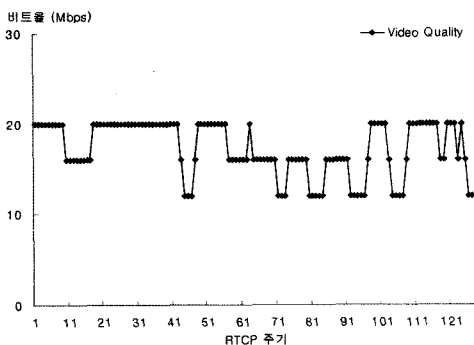


(a) 전송률 및 비디오 품질 조절

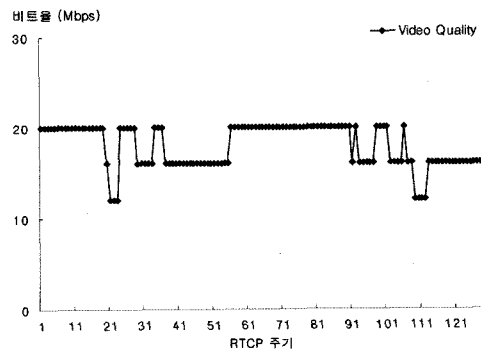


(b) 패킷 손실 발생

그림 9 HAViS 전송기법의 성능



(a) MinTH=40%



(b) MinTH=10%

그림 10 임계값 설정에 따른 성능 비교

의 결정은 HAViS 전송기법이 네트워크 상태 변화에 반응하는지, 수신단 버퍼 상태 변화에 반응하는지를 구분하므로 HAViS 전송기법의 성능에 직접적인 영향을 미친다. (b)의 경우, (a)에 비교해서 네트워크 상태 변화에 반응하는 구간이 넓어지므로 서비스되는 비디오 품질의 변화가 줄어드는 것을 확인할 수 있다. (a)와 (b)의 경우, 횡수의 차이는 있지만 그림 9에서 확인한 HAViS 전송기법의 비디오 품질 변화에 비해 더 많은 비디오 품질의 변화를 발생하게 된다. 결과를 통해, 버퍼의 underflow나 overflow를 예방하기 위한 임계값 설정에 있어, 미디어의 특성을 고려함으로써 사용자에게 제공되는 비디오 품질의 변화를 효율적으로 조절할 수 있음을 확인할 수 있다.

4.4 기존 전송기법과의 성능 비교 실험

본 실험은 기존에 제안된 스트리밍 전송기법과, 제안하는 HAViS 전송기법의 성능을 비교하기 위한 것이다. 성능 비교 대상으로는 2장에서 설명한 Padhye의 전송률 모델링 공식을 사용한 TFRC와 버퍼 상태를 기반으로 비디오 품질을 조절하는 Buffer-driven scheme을 사용하였다.

그림 11은 제안한 HAViS 전송기법과 기존의 연구인 TFRC와 Buffer-driven scheme의 성능을 버퍼 점유도와 패킷 손실 발생 분포를 통해 검증한 결과이다. TFRC는 네트워크 상태를 고려하여 전송률을 조절함으로써 패킷 손실 발생을 줄이는 장점을 가지지만 스트리밍 응용에서 일반적으로 사용되는 버퍼에 대한 고려가 없고, 비디오 품질 조절 기법을 적용하지 않고 전송률만을 조절하므로 (a)의 결과와 같이 버퍼의 underflow가 지속적으로 발생하게 된다. 이것은 사용자에게 비디오의 재생에 있어 끊김없이 부드러운 재생을 제공하지 못하는 문제점으로 나타난다. 반면, 버퍼 점유도를 기반으로 서비스되는 비디오의 품질을 조절하는 Buffer-driven scheme은 효과적으로 버퍼의 underflow와 overflow를

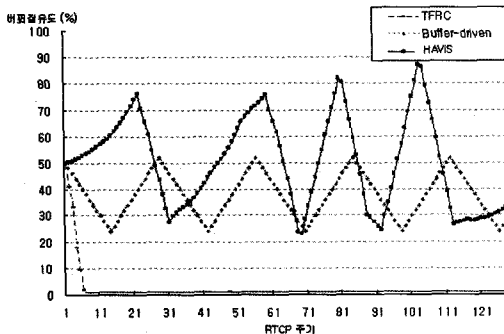
예방하지만 네트워크 상태에 대한 고려가 없으므로 패킷 손실이 많이 발생하게 된다. 따라서 사용자에게 끊김 없이 부드러운 비디오 재생을 제공할 수 있지만, 네트워크의 안정성을 위협할 수 있는 문제점을 갖는다.

기존의 관련 연구와 다르게, 제안한 HAViS 전송기법은 네트워크 상태를 고려해서 전송률을 조절하고, 버퍼 상태 예측기법을 기반으로 버퍼의 underflow나 overflow의 위험이 있을 경우, 서비스되는 비디오의 품질을 조절하게 된다. 그림 11의 실험 결과를 통해 HAViS 전송기법은 버퍼의 underflow나 overflow를 효과적으로 예방함으로써 사용자에게 끊김없이 부드러운 스트리밍 서비스를 제공할 수 있으며, 동시에 네트워크 상태에 따라 적절히 전송률을 조절함으로써 네트워크 안정성을 향상시키는 것을 확인할 수 있다. (b)의 결과에서, HAViS 전송기법은 Buffer-driven scheme에 비해서 패킷 손실 발생이 크게 줄었으며, 네트워크 상태만을 고려한 TFRC와도 거의 비슷한 패킷 손실이 발생하는 것을 확인할 수 있다. 이러한 결과를 통해 제안한 HAViS 전송기법이 네트워크 안정성 측면만을 고려한 대부분의 기존 연구와 유사한 성능을 보이는 것을 확인할 수 있다.

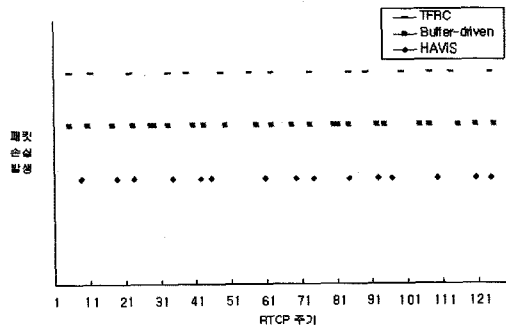
5. 결론 및 향후 과제

스트리밍 전송 프로토콜의 표준인 RTP는 대부분 혼잡 제어 메커니즘이 없는 UDP를 기반으로 하므로 네트워크의 안정성을 저해하는 원인이 된다. 이러한 문제점을 개선하기 위해 1990년대 이후로 네트워크의 안정성을 고려한 스트리밍 전송기법에 관한 연구가 활발하게 진행되고 있다. 하지만 네트워크의 안정성만을 고려한 스트리밍 전송기법은 멀티미디어 스트리밍의 기본적인 요구사항에 대한 고려가 부족하다는 한계를 갖는다. 또한 스트리밍 서비스의 품질에 큰 영향을 미치는 미디어의 특성에 대한 고려가 부족하다는 한계를 갖는다.

본 논문에서는 기존 스트리밍 전송기법들이 가지는



(a) 버퍼 점유도 변화 비교



(b) 패킷손실발생 비교

그림 11 기존 연구와의 성능 비교

한계를 극복하는 HAViS(Hybrid Approach for Video Streaming) 전송기법을 제안하였다. HAViS 전송기법은 멀티미디어 스트리밍과 관련된 기존 연구들처럼 하나의 관점에서의 성능향상만을 추구하지 않고 네트워크 상태와 사용자에게 서비스되는 품질을 모두 고려하는 혼합된 형태의 접근방법을 제시하고 있다. 제안한 HAViS 전송기법은 패킷 손실률에 기반하여 네트워크를 안정상태와 혼잡상태로 구분하고 각각의 상태에서 RTT 이후의 수신단 버퍼상태 예측을 기반으로 서비스되는 비디오 스트림의 전송률이나 품질을 조절하게 된다. 그러므로 HAViS 전송기법은 네트워크 상태에 따라 전송률을 조절함으로써 네트워크 안정성을 향상시키며, 수신단 버퍼의 underflow나 overflow를 효과적으로 예방하여 사용자에게 끊김없이 부드러운 비디오 재생을 보장하게 된다. 또한 서비스되는 미디어의 특성을 고려하여 버퍼를 관리함으로써 시스템의 자원을 보다 효율적으로 사용하고, 비디오의 품질 변화를 적절하게 조절하게 된다.

시뮬레이터를 이용한 실험을 통해서 HAViS 전송기법의 기본적인 성능을 검증하고 관련 연구와의 성능 비교를 수행하였다. 실험 결과를 통해 제안한 HAViS 전송기법이 사용자에게 끊김없이 부드러운 재생 서비스를 제공할 수 있는 동시에, 네트워크 상태를 고려함으로써 네트워크의 안정성을 향상시킬 수 있음을 확인할 수 있었다. 향후, 제안한 HAViS 전송기법이 현재 네트워크의 주요 트래픽인 TCP와 친화적으로 동작할 수 있도록 하는 방법에 대한 연구가 수행되어야 하고, 멀티캐스트 환경 및 무선 환경에서의 적용에 대한 연구도 같이 수행되어야 할 것이다.

참 고 문 헌

- [1] S. Floyd and F. Kevin, "Router mechanisms to support end-to-end congestion control," Technical Report, LBL-Berkeley, 1997.
- [2] S. Cen, C. Pu, and J. Walpole, "Flow and congestion control for internet streaming applications," *Multimedia Computing and Networking*, pp. 250-264, 1998.
- [3] R. Rejaie, M. Handley, and D. Estrin, "RAP: An end-to-end rate based congestion control mechanism for real-time streams in the Internet," *IEEE INFOCOMM*, 1999.
- [4] S. Na and J. Ahn, "TCP-like flow control algorithm for real-time applications," *IEEE ICON*, pp. 99-104, 2000.
- [5] I. Rhee, V. Ozdemir, and Y. Yi, "TEAR: TCP emulation at receivers - flow control for multimedia streaming," Technical Report, NCSU, 2000.
- [6] D. Bansal, and H. Balakrishnan, "Binomial congestion control algorithms," *IEEE INFOCOMM*,

pp. 631-640, 2001.

- [7] J. Padhye, J. Kurose, D. Towsley, and R. Koodli, "A model based TCP-friendly rate control protocol," *NOSSDAV*, 1999.
- [8] S. Floyd, M. Handley, J. Padhye, and J. Widmer, "Equation-based congestion control for unicast applications," *ACM SIGCOMM*, 2000.
- [9] Q. Zhang, Y. Zhang, and W. Zhu, "Resource allocation for multimedia streaming over the Internet," *IEEE Transactions on Multimedia*, Vol. 3, pp. 339-355, 2001.
- [10] D. Sisalem, and H. Schulzrinne, "The loss-delay based adjustment algorithm: A TCP-friendly adaptation scheme," *NOSSDAV*, 1998.
- [11] N. Aboobaker, D. Chanady, M. Gerla, and M. Sanadidi, "Streaming media congestion control using bandwidth estimation," *IFIP/IEEE MMNS*, pp. 89-100, 2002.
- [12] A. Balk, D. Maggiorini, M. Gerla, and M. Sanadidi, "Adaptive MPEG-4 video streaming with bandwidth estimation," *LNCS*, Vol. 2601, pp. 525-538, 2003.
- [13] D. Ye, X. Wang, Z. Zhang, and Q. Wu, "A buffer-driven approach to adaptively stream stored video over Internet," *IEEE High Speed Networks and Multimedia Communications*, pp. 81-85, 2002.
- [14] J. Padhye, V. Firoiu, D. Towsley, and J. Kurose, "Modeling TCP throughput: A simple model and its empirical validation," *ACM SIGCOMM*, pp. 303-314, 1998.
- [15] H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson, "RTP: A transport protocol for real-time applications," *IETF RFC 1889*, 1996.
- [16] The network simulator ns-2, <http://www.isi.edu/nanam/ns/>

이 선 현

정보과학회논문지 : 정보통신
제 34 권 제 2 호 참조

정 광 수

정보과학회논문지 : 정보통신
제 34 권 제 2 호 참조