

# 인터넷 방송에서 멀티 스트림을 이용한 비디오 스트림 전송

\*강 경 원\*, 정 태 일\*, 류 권 열\*\*, 문 광 석\*, 김 문 수\*

\*부경대학교 전자컴퓨터정보통신공학부

\*\*위덕대학교 멀티미디어공학과

## Video Stream Delivery for Using Multi-stream in the Webcasting

Kyung W. Kang\*, Tae I. Jeong\*, Kwon R. Ryu\*\*, Kwang S. Moon\*, Moon S. Kim\*

\*School of Electronics, Computer and Telematic Engineering, Pukyong National University

\*\*Department of Multimedia Engineering, Uiduk University

e-mail: kangkw@mail1.pknu.ac.kr

### 요 약

최근 인터넷은 WWW의 대중화에 힘입어 디지털 오디오, 비디오 데이터 전송에 대한 요구가 증가하고 있다. 본 논문에서는 QoS가 보장되지 않는 현재의 인터넷 인프라 환경하에서 클라이언트가 가진 대역폭을 최대한 활용하여 최선의 서비스를 제공할 수 있는 멀티 스트림을 이용한 새로운 비디오 스트림 방법을 제안한다. 제안한 방법은 사용자 측에서 멀티 스트림을 통해 동시에 클라이언트 측으로 전송함으로써 네트워크 트래픽에 의해 발생하는 전송지연에 민감하지 않을 뿐만 아니라, 클라이언트 측의 대역폭을 최대한 이용할 수 있게되어 최선의 서비스를 제공할 수 있다.

### 1. 서 론

현재 인터넷의 확산과 WWW(World Wide Web)의 대중화에 힘입어 인터넷 사용 인구가 급속도로 증가하고 있다. 이와 추세를 맞추어 인터넷으로 동영상, 오디오 등의 멀티미디어 정보를 실시간으로 전달할 수 있는 스트리밍 기술의 개발과 네트워크 인프라의 확충으로 인터넷과 멀티미디어가 결합한 인터넷 방송(Webcasting)이 대두되고 있다. 인터넷 방송은 스트리밍 기술을 이용한 VOD(Video on Demand) 서비스를 통해 기존 지상파 방

송이 가지고 있는 단방향적 매스 미디어적인 성격과 시간적 제약성을 극복할 수 있을 뿐만 아니라 현재 추진 중인 디지털 방송과 병행할 수 있다. 그러나 현재의 인터넷 환경하에서는 기존의 전화망에서 제공되고 있는 보장형 서비스와는 달리 전송시간, 대역폭, 지연변이 등과 같은 성능에 대한 QoS(Quality of Service)은 보장할 수 없고, 단지 클라이언트에게 신뢰성 있는 데이터 전송만을 제공할 수 있다[1,2]. 현재 인터넷 방송에서 가장 많이 사용되고 있는 Real Network사의 Real Server G2[3]나 Microsoft사의 Netshow[4]와 같은 스트리밍 미디어 서버를 이용하여 VOD 서비스를 위한 비디오 스트림을 클라이언트들에게 전송할 경우 단일 스트림의 형태로 전송되기 때문에, 이 스트림에 대한 어떠한 QoS도 보장할 수 없다. 따라서 네트워크 트래픽의 급증으로 혼잡이 발생하여 전송 지연이 발생할 경우 클라이언트는 그 영향을 그대로 받게되어 전송 성능이 떨어지는 단점을 가지고 있다.

따라서 본 논문에서는 QoS가 보장되지 않는 현재의 인터넷 인프라 환경하에서 클라이언트가 가진 대역폭을 최대한 활용하여 최선의 서비스를 제공할 수 있는 멀티 스트림을 이용한 새로운 비디오 스트림 방법을 제안한다. 제안한 방법은 클라이언트가 서버에 접속시 멀티스레드를 이용하여 다중으로 접속한다. 이렇게 생성된 멀티 스트림들은 독립적으로 다른 네트워크 경로를 통

해 동시에 미디어 서버에서 클라이언트 측으로 전송되기 때문에, 단일 스트리밍으로 전송되는 경우보다 네트워크 트래픽의 증가에 의해 발생하는 전송 지연에 덜 민감할 뿐만 아니라, 클라이언트 측의 대역폭을 최대한 이용할 수 있어 클라이언트 측면에서 보다 최선의 서비스를 제공할 수 있다.

## 2. 멀티 스레드를 이용한 멀티 스트림 전송

현재 인터넷 방송에서 가장 많이 사용되고 있는 비디오 스트림들은 클라이언트와 미디어 서버간에 단일 스트림 형태로 접속이 된다. 따라서, 네트워크의 트래픽 증가에 의해 전송 지연이 발생할 경우 클라이언트는 그 영향을 그대로 받게 되어 전송 성능이 떨어지는 단점이 있다. 따라서 본 논문에서는 네트워크에서 발생하는 전송 지연에 효율적으로 대처할 수 있을 뿐만 아니라 클라이언트 측의 대역폭을 최대한 활용할 수 있는 멀티 스레드를 이용한 멀티 스트림을 제안한다.

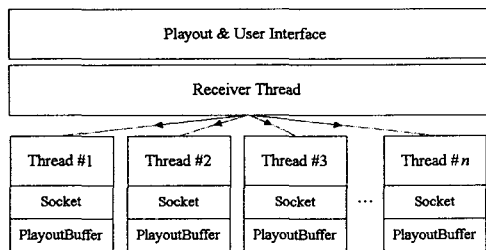


그림 1. 클라이언트의 수신단 구조

그림 1은 제안한 멀티 스트림을 생성하는 클라이언트의 수신단 구조를 나타낸 것이다. 그림 1에서 보는 바와 같이 수신단은 화면 구성 및 재생을 위한 부분과 멀티 스트림을 하나의 스트림으로 변환하는 수신 스레드, 그리고 개별적으로 서버에 접속하여 비디오 스트림을 받아오는 소켓 스레드들로 구성되어 있다.

수신 스레드는 생성된 다중 소켓들에 의해 전송된 버퍼에 있는 데이터들을 하나의 스트림으로 생성하기 위해 동기화 과정은 필수적이다. 그림 2는 수신 스레드와 멀티 스트림사이의 동기화 과정을 설명한 것이다. 각 소켓 스레드들은 독립적으로 미디어 서버에 접속하여 비디오 스트림을 전송 받아 자신이 가진 버퍼에 저장한 후 수신 스레드에게 알린 후 자신은 스트림상태

로 들어간다. 수신 스레드는 버퍼가 채워져지면 데이터를 읽은 후 소켓 스레드에게 다시 동작할 것을 통지하고 라운드 로빈 방식으로 다른 소켓 스레드를 검사한다.

네트워크를 통해 데이터를 읽는 과정은 네트워크 트래픽 상황에 매우 의존적이기 때문에 네트워크 소켓에서 데이터를 읽을 때 걸리는 시간은 수신 스레드가 재생을 위해 데이터를 읽는 시간에 비해 많이 걸린다. 그러나 그림 2에서 보는 바와 같이 멀티 스트림을 사용할 경우 다소의 전송 지연이 발생하더라도 수신 스레드는 전송지연없이 일정한 속도로 비디오 스트림을 재생할 수 있다.

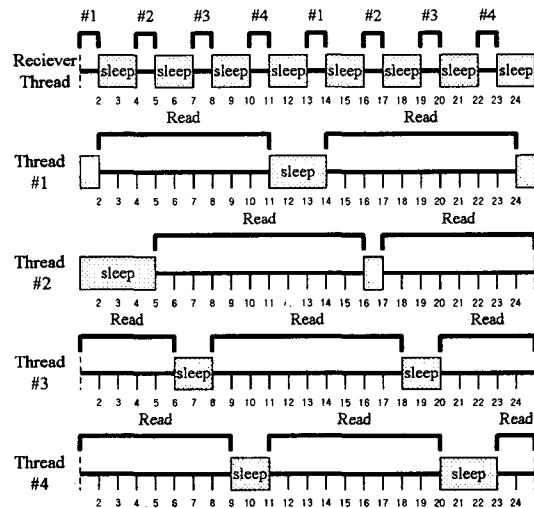


그림 2. 수신스레드와 멀티 스트림사이의 동기화

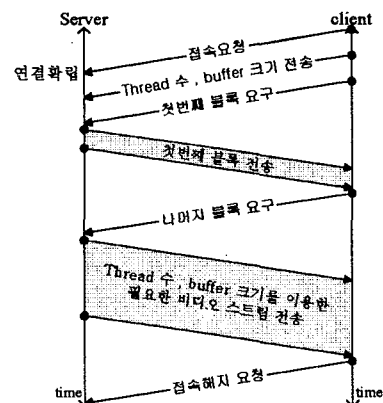


그림 3. 클라이언트와 서버사이의 전송제어절차

그림 3은 서버와 클라이언트사이의 전송제어절차를 나타낸 것이다. 클라이언트와 서버가 접속이 성립되면 클라이언트는 전체 스투드 수와 버퍼 크기를 서버쪽으로 전송한다. 서버는 이 정보를 이용해 식 (1)과 같이 비디오 스트림을 읽어 하나의 스트림을 생성하여 전송한다.

$$\text{Stream}_p = (B_s \times T_n) \times i + B_s \times p \quad (1)$$

여기서  $B_s$ 는 버퍼의 크기이며,  $T_n$ 은 전체 스투드 수를 의미한다. 그리고  $p \in \{0, 1, \dots, T_n\}$ ,  $i \in \{0, 1, \dots, N\}$ 를 의미한다. 첫 번째 블록은 초기 버퍼링을 위해 별도로 먼저 전송한다.

### 3. 인터넷 트래픽 특성

실제 인터넷 환경하에서 실험은 제어하기가 어렵다. 본 논문에서는 제어 가능한 실험을 위해 인터넷 트래픽 모델 시뮬레이션을 사용하였다. 기존에는 일반적으로 포아송(Poisson) 또는 마코비안(Markovian)을 기본으로 한 모델을 사용하였으나, 이 모델은 비교적 넓은 범위의 시간 스케일로 평균을 구했을 경우, 비스트한 특성이 없어진다.[5,6]. 따라서 본 논문에서는 그림 4와 같이  $\alpha$ -stable self-similar 트래픽 모델을 사용하였다[7].  $x$ 축은 시간 슬롯을 의미하며,  $y$ 축은 각 시간 슬롯 당 전송되는 데이터 양을 의미한다.

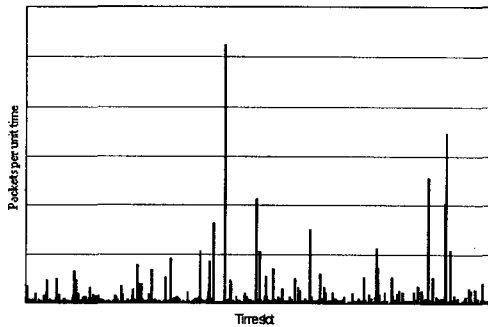


그림 4.  $\alpha$ -stable self-similar 트래픽 모델 ( $\alpha=1.28$ ,  $H=0.8333$ )

그림 5는 인터넷 트래픽 시뮬레이션이 그림 4와 같은 전송량을 가지도록 서버에 적용 될 지연 모델을 나타낸 것이다. 즉, 단위 IP 패킷(1500bytes)를 전송 후 그림과 같이 지연 시키므로써 전체 전송속도를 그림 4와

같은 형태가 되도록 하였다.

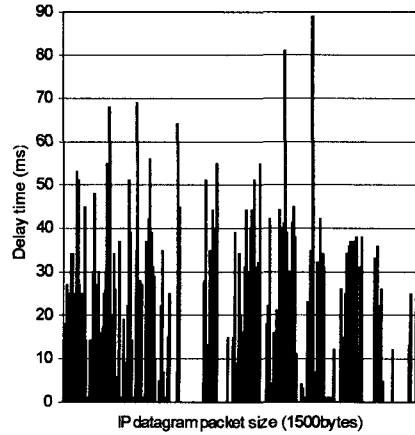


그림 5. 지연 모델

### 4. 실험 및 고찰

본 논문의 실험을 위한 서버와 클라이언트 프로그램들은 windows 98 환경하에서 자바로 구현하였으며, 비디오 데이터는 기존에 산재해 있는 동영상 표준 비디오 파일인 MPEG 파일을 사용하였다. 표 1은 본 논문에서 사용한 비디오 데이터의 특성을 나타낸 것이다.

표1. 실험대상 비디오 데이터의 특성

프레임 패턴	IBBPBBPBBPBBPBB
평균 프레임 크기	4577 bytes
화면 크기	352×240
VBV	20

그림 6과 그림 7은 평균 전송속도가 약 200kbps, 100kbps에 대해 제안한 4개의 스투드를 이용한 방법과 기존의 단일 스트림 방법의 성능을 비교한 것이다. 버퍼의 전체 크기는 모두 VBV(video buffer verifier)에서 지정한 최소 크기인 40Kbyte로 지정하였다. 단일 스트림을 이용한 방법은 전송지연이 커짐에 따라 그 영향이 두드러지가 나타났지만, 4개의 스투드를 이용한 방법은 200Kbps에서는 일정한 GOP 처리 속도를 보였으며, 100Kbps에서도 전송 지연에 민감하지 않음을 알 수 있었다.

그림 8은 버퍼의 크기가 40k이고, 전송속도가 100kbps 일 때 스투드 수에 따른 성능을 비교한 것이

다. 전체적으로 스레드 수가 많을수록 전체적인 좋은 성능은 얻을 수 있었다.

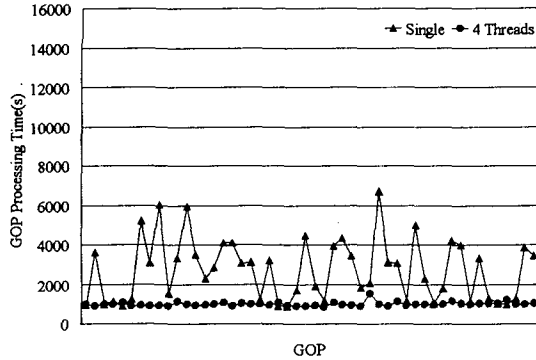


그림 6. 전송속도 200Kbps에서 기존의 방법과 제안한 방법 성능 비교

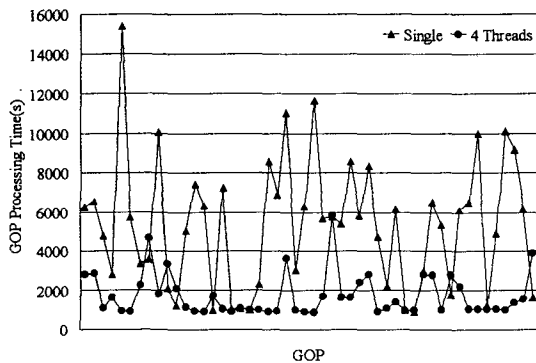


그림 7. 전송속도 100Kbps에서 기존의 방법과 제안한 방법 성능 비교

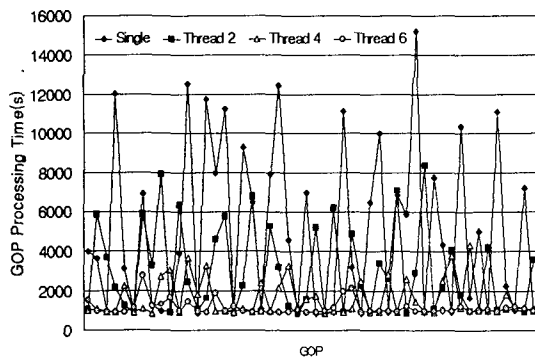


그림 8. 스레드 수에 따른 성능비교

## 5. 결론

본 논문에서는 QoS가 보장되지 않는 현재의 인터넷 인프라 환경하에서 클라이언트가 가진 대역폭을 최대한 활용하여 최선의 서비스를 제공할 수 있는 멀티 스트림을 이용한 새로운 비디오 스트림 방법을 제안하였다. 다양한 인터넷 트래픽 조건하에서 실험한 결과, 기존의 단일 스트림 전송보다 제안한 멀티 스트림을 이용한 전송이 네트워크 트래픽에 의해 발생하는 전송 지연에 대해서 기존의 방법보다 민감하지 않아 보다 최선의 서비스를 제공할 수 있음을 확인하였다.

향후 전송 속도에 따른 비디오 스트림에 대한 스칼라와 에러 은닉에 관한 연구, 비디오 스트림에 관한 프로토콜(RTP, RSVP)이 된다면 보다 효율적인 비디오 스트림 전송이 가능하리라 사료된다.

## 참고 문헌

- [1] P. Ferguson and G. Huston, *Quality of Service: Delivering QoS on the Internet and in Corporate Networks*, Wiley Computer Publishing, 1998.
- [2] R. Steinmetz and K. Nahrstedt, *Multimedia: Computing, Communications and Applications*, Prentice Hall PTR, 1995.
- [3] Real Video G2 of Real Network, available at <http://www.real.com>.
- [4] Netshow of the Microsoft, available at <http://www.microsoft.com/netshow>.
- [5] V. Paxson and S. Floyd, "Wide Area Traffic: The Failure of Poisson Modeling, *IEEE/ACM Transactions on Networking*", Vol. 3, No. 3, June 1995, p. 226-244
- [6] W. E. Leland, M. S. Taqqu, W. Willinger, and D. V. Wilson, "On the Self-similar Nature of Ethernet Traffic (Extended Version), *IEEE/ACM Transactions on Networking*", Vol, 2, No. 1, February 1994, pp. 1-15
- [7] J.R. Gallardo, D. Makrakis, and L. Orozco-Barbosa, "Use of Alpha-Stable Self-Similar Stochastic Processes for Modeling Traffic in Broadband Networks", in the *Proceedings of 1998 SPIE Performance and Control Conference*, October 1998, pp. 218-296.