

# RTP를 이용한 비디오 스트림의 실시간 멀티캐스트 전송 기법

정규수\* · 양종운\* · 나인호\*

\*군산대학교

## Real-Time Multicast Transmission Technique of Video Stream using RTP

Kyu-Su Jung\* · Jong-Un Yang\* · In-Ho Ra\*

\*Kunsan National University

E-mail : kyusu@kunsan.ac.kr

### 요 약

본 논문에서는 비디오 스트림을 실시간에 전송하기 위하여 RTP(Real-Time Protocol)을 이용하여 전송 방법에 대해 기술한다. 먼저 비디오 스트림을 연속적으로 전송을 보장하기 위해 네트워크의 종단간 특성을 분석하여 네트워크 상황을 파악하는 방법을 제시하고, 대역폭을 극복하기 위한 버퍼링 기법과 RTCP(Real-Time Control Protocol)를 이용하여 데이터의 전송 상황을 분석하여 실시간 멀티미디어 전송의 QoS를 보장하고 연속성을 유지할 수 있는 알고리즘을 제시한다.

### Abstract

In this paper, we describe a method for real-time transmitting video streams based on RTP. In order to guarantee synchronous video streams, we propose a method to grasp network situations by analyzing end-to-end network traffic. In addition, we present an algorithm for satisfying QoS requirements of real-time multimedia transmission and maintaining continuity of transmission. It describes a buffering method for overcoming bandwidth limitations and an analyzing method based on RTCP for grasping network traffic situation to resolve the problem of real-time transmission of video streams.

### 1. 서론

비디오와 같은 연속성 매체의 전송 요구 사항은 기존의 텍스트 위주의 데이터와는 판이하게 다르다. 텍스트 데이터는 전송의 실시간성을 요구하지 않지만 모든 데이터 비트가 손실이나 에러 없이 정확히 전송되어야 한다. 반면 어느 정도의 손실 또는 에러를 감수할 수 있으나 데이터 비트들을 실시간으로 전송되어야 한다. 또한 활성화(live) 시간이 비교적 길며 주기적으로 비트 스트림을 발생시키는데 매우 큰 대역폭을 요구하기 때문에 현재의 인터넷과 같이 가변적이고 제한된 대역폭 상에서는 실시간 전송 및 재생이 어려운 실정이다[1]. 따라서 실시간 전송을 위해서 프로토콜의 오버헤드를 줄이거나 네트워크 전송 상태 정보를 이해하고 대처하는 방법을 모색해야 한다.

인터넷에서 VOD(Video On Demand)서비스를 하기 위해서 기존의 Unicast 방식으로 전송할 때 접속하는 클라이언트 수가 증가함에 따라 데이터의 트래픽은 증가하게되며 라우터가 폭주할 수도 있다. 대역폭이 큰 동영상을 전송하는데 효과적으로 다수의

클라이언트를 서비스하기 위해서 대역폭을 적게 소모하며 클라이언트의 수가 증가에 따라 대역폭이 비례적으로 증가하지 않는 Multicast에 대해 많이 연구하고 있다[2].

본 논문의 구성은 다음과 같다. 먼저 II장에서는 실시간 전송을 관리할 수 있는 RTP/RTCP에 대해 소개하고, 네트워크의 자원 낭비를 막을 수 있으면서 브로드캐스트 방식과는 달리 네트워크 내의 특정 호스트에게만 서비스 할 수 있는 멀티캐스트에 대해 소개한다. 다음, III장에서는 동영상 전송을 위해 필요한 몇 가지 고려사항에 대해 기술한다. IV장에서는 실시간 전송을 위한 모델링 및 버퍼링 기법에 대해 기술하고, 마지막으로 V장에서는 결론 및 향후 연구 방향에 대해 기술한다.

### II. 관련연구

#### (1) RTP/RTCP

화상회의, VOD서비스에서 실시간 연속적인 멀티미디어 전송을 위하여 연구되어진 프로토콜이

RTP(Real-Time Protocol)로서 인터넷의 불규칙한 대역폭을 극복하기 위하여 네트워크의 정보를 제공하고 있다. 전송된 메시지가 만들어진 시간과 전체 메시지에서 몇 번째인지를 알 수 있는 정보와 비디오, 오디오 데이터에 대해서 코딩 방법이 무엇인지 명시되어 있다. 실시간 전송을 위해 네트워크 상태를 파악할 수 있도록 jitter, 전송 시간 등의 정보도 명시하도록 되어 있다.

RTP 프로토콜은 실시간 데이터의 특성에 중점을 두어 제정한 프로토콜이라 할 수 있다. RTP는 멀티캐스트 또는 유니캐스트 상에서 음성, 화상 또는 모의 데이터와 같은 실시간 데이터를 전송하는데 적합한 단대단 전송기능을 제공한다. RTP는 자원 예약에 대한 내용은 다루지 않으며, 특히 즉시 데이터 전송(timely delivery), QoS 보장, 뒤바뀐 순서 전송 방지와 같은 기능을 제공하지 않는다. RTP 패킷은 UDP를 이용하여 전달한다[3][4].

RTCP는 RTP의 컨트롤 프로토콜로서 세션에 참가한 모든 참가자들의 전송상태에 대한 정보를 주기적으로 전송한다. RTCP는 SR(Sender Report), RR(Receiver Report), SDES(Source Description) BYE(goodBYE) 그리고 APP(APPLICATION defined)의 5가지 메시지를 교환하여 다음과 같은 네가지 기능을 수행한다.

첫째, 주된 기능으로는 데이터 전송 상태에 대하여 피드백을 참가자들에게 주는 것이다. 이 기능은 아주 중요한 기능으로 흐름과 충돌에 대한 컨트롤 기능과 관련이 있다. 일례로, 송신자가 네트워크의 대역폭을 상대방에 알려주어 고품질의 화상 또는 음성 데이터 전송을 요청할 수 있다.

둘째, RTCP는 CNAME이라고 불리는 트랜스포트 레벨의 identifier를 전송한다. 이 기능은 프로그램을 다시 시작하였을 때 SSRC(소스 ID)가 다른 것과의 충돌로 인하여 바뀌더라도 CNAME에 의해 같은 사용자임을 알 수 있게 하여준다.

셋째, RTCP 패킷들은 주기적으로 세션의 모든 참가자에게 전송되기 때문에 세션에 얼마나 많은 참가자들이 있는지를 알려 준다. 참여자수의 수에 따라 주기를 조정해야 할 필요가 있다. 만일 세션의 참가자들이 많아짐에도 불구하고 주기가 빠른 상태를 계속 유지한다면 많은 대역폭이 RTCP 패킷에 의하여 사용되기 때문이다.

마지막으로 RTCP는 옵션을 통해 사용자의 이름, e-mail 주소, 전화번호, 지역, 사용도구, 상대방에게 전달하고 싶은 메시지 등을 전송한다.

(2) Multicast

웹서비스에서 사용하는 Unicast 방식은 접속자 수가 많을수록 비례적으로 서비스하는 대역폭이 증가되어야 한다. 하나의 클라이언트와 서버 사이에 일대일 관계를 유지하기 때문에 다수의 클라이언트의 접속 시 일대다 관계로서 데이터를 전송 시켜야한다. 대역폭이 증가하므로 서버나 라우터의 과부하로 인하여 다른 서비스까지 나쁜 영향을 미칠 수 있다. 이에 반해, Multicast 방식은 Unicast에서 1:N 접속

이 이루어졌을 때 서버에서 N회 동일한 데이터를 전송시켜야 했던 것을 서버에서 전송된 데이터가 라우터에 복사되어 라우터 내의 동일한 네트워크에서 그룹에 참가하고 싶은 클라이언트가 있는지 알아보고, 그룹에 참가하고자 하는 클라이언트에게만 데이터를 전송하여 서버와 라우터의 트래픽이 클라이언트 수에 따라 증가하지 않게 된다[5]. 따라서, VOD 서비스, 화상회의 등의 실시간 전송에 효과적이다.

III. 고려사항

비디오 데이터의 방대한 줄이기 위해 MPEG, H.261 과 같은 압축표준을 사용하는데 영상의 공간 상관율에 따라 가변적인 압축율이 나오게 된다.

비디오 스트림을 전송시키기 위해서 목적에 따라 고정 비트율(CBR, Constant Bit Rate)과 가변 비트율(VBR, Variable Bit Rate)의 두 가지 방식으로 전송시킬 수 있다. CBR의 경우, 고정된 대역폭을 확보 하므로서 연속적인 스트림을 재생하는데 네트워크 자원뿐만 아니라 시스템의 자원을 효율적으로 사용할 수 있고, 연속적인 서비스를 보장할 수 있는 장점이 있다. 반면, 영상의 압축율이 떨어지고 네트워크의 대역폭이 불규칙할 경우 화질이 균등하지 않는 단점이 있다[6].

표 1. 멀티미디어 데이터의 QoS값

QoS	max. delay (s)	max. delay jitter (ms)	ave throughput (Mbit/s)
voice	0.25	10	0.064
video(TV quality)	0.25	10	100
Compressed video	0.25	1	2~10
real-time data	0.001~1	-	< 10

비디오 스트림을 인터넷상에서 실시간에 전송하기 위해서 몇 가지 조건들을 충족 시켜야만 한다.

첫째, 표 1과 같이 멀티미디어 데이터의 QoS 값을 충족시켜줄 수 있어야 하며, 둘째, 지터 발생으로 인한 비디오 스트림의 출력량을 조절할 수 있어야 한다. 셋째, 실시간성 보장하기 위한 방법 등을 고려해야 한다.

수신측에서는 지터를 줄이고, 송신측과 수신측이 서로 다르게 동작하는 시점이 다르기 때문에 버퍼링을 해야만 한다. 과도한 지터는 수신측에서 최대 지연시간내에 패킷이 도착하지 않는 결과가 생기므로 손실될 가능성이 크다. 수신측의 지연시간 허용값과 데이터 손실에 대한 처리 방법을 명확하게 해야한다.

IV. 모델링 및 구현

(1) 모델링

RTP를 이용하여 실시간 멀티캐스트 전송을 구현하기 위하여 그림 1과 같이 객체 모델, 이벤트 모델, 버퍼링 모델을 제안한다.

객체 모델에는 다양한 미디어의 특성을 고려한

정보를 담고 있는 MediaObject를 응용 프로그램에서 요구하게 된다. MPEG, H.261, JPEG 등 미디어 타입과 미디어의 시간, 우선 순위 등의 정보를 포함하고 있다. 미디어 타입에 따라 미디어 특성에 적합한 특정 처리가 필요하지만 추상적인 처리를 위해 MediaObject에서 PacketObject에 범용적으로 담겨질 DataObject 로 변환되어 PacketObject에 담긴다.

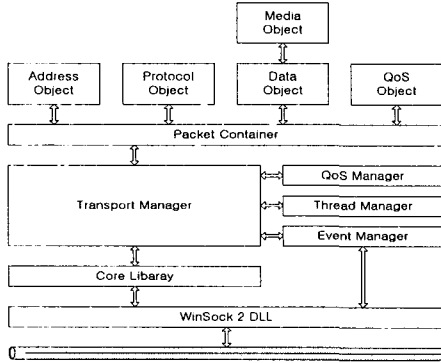


그림 1. 객체 모델

AddressObject는 호스트 주소나 호스트 이름, 포트 번호, 서비스 번호를 관리하며, ProtocolObject에서는 프로토콜에 따른 소켓 초기화 작업을 담당하는 객체이다. 미디어의 특성과 프로토콜에 따라 응용프로그램에서 QoS를 적용할지 여부에 따라 QoS 파라미터를 정보를 담은 객체이다.

```

struct _flowspec
{
    u_long TokenRate;
    u_long TokenBucketSize;
    u_long PeakBandWidth;
    u_long Latency;
    u_long DelayVariation;
    u_long Servicetype;
    u_long MaxSDUSize;
    u_long MinPoliceSize;
} FLOWSPEC;
struct _QoS
{
    FLOWSPEC Send;
    FLOWSPEC Receive;
    WSABUF Provider;
} QOS;
    
```

그림 2. QoS 구조체

이렇게 Transport Manager는 PacketContainer에 담긴 객체들 중 DataObject와 QoSObject를 이용하여 미디어의 특성 정보를 얻고, 전송지연 허용시간을 얻어 RTP의 초기화 작업을 한다. 다수의 클라이언트의 접속을 가능하도록 PacketContainer에 하나의 쓰레드가 할당되며 ThreadManager에서 WorkThread를 생성하게 된다. WorkThread에서는 QoS 처리, 버퍼 처리를 위하여 QoSObject를 이용하여 Container에 있는 소켓에 QoS 옵션을 할당하게

된다. 그림 1에서 ThreadManager는 WorkThread에서 사용하게 될 버퍼를 할당하게 되며 WorkThread 자신에게 할당된 버퍼에만 액세스가 가능하며 버퍼가 full이거나 empty일 경우, BufferManager에서 Event를 발생시키게 된다.

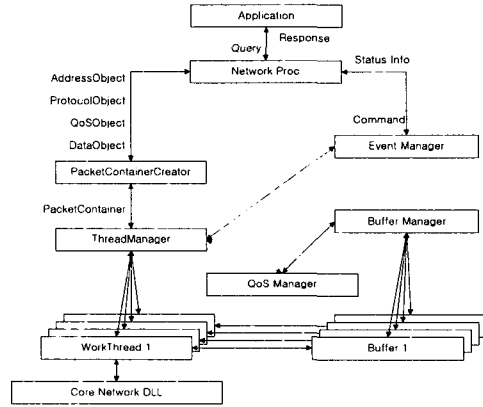


그림 3. 이벤트 모델

(2) 동적 흐름 제어

실시간 비디오 데이터를 동적흐름제어(dynamic flow control)하기 위해 다음과 같은 방식을 사용한다.

첫 번째는 수신자가 경험하는 패킷 손실을 측정하여 네트워크 상태를 판단한다. 두 번째 방식은 수신자의 네트워크의 상태를 송수신자간의 패킷의 왕복시간(round trip time)으로 판단한다. 세 번째 방식은 수신자의 네트워크 상태를 패킷간의 도착지연 시간(jitter)으로 판단한다. 수신자의 수신상태가 판단 되면 송신자는 적절한 대역폭을 예측하여 출력되는 데이터 량을 동적으로 조절하면 된다.

RTCP에서는 수신자가 주기적으로 RTCP의 패킷을 통해 수신할 패킷의 개수, 수신된 패킷의 개수, 송신자와 수신자의 평균 왕복시간 등의 정보를 송신자에게 보내게 된다. 이 패킷을 통해 송신자는 수신자의 수신상태와 가용 대역폭을 예측할 수 있다. 대역폭을 예측하기 위한 수식은 다음과 같다. 네트워크의 체중도 SPLR(Smooth Packet Loss Rate), λ로 나타낸다[3].

$$\lambda = (1 - \alpha) * \lambda + \alpha * \beta \quad (\text{식-1})$$

(0 ≤ α ≤ 1, β 각 주기마다 새로운 패킷 분실율)

RTCP 데이터 패킷의 지터에 근거하여 네트워크의 상태를 측정하고 측정된 네트워크 상태에 따라 전송률을 변화시키는 기법을 설명한다. 데이터 패킷의 지터 J는 한 쌍의 패킷에 대해 송신자로부터 보내지는 시간 간격과 수신자에게 도착하는 시간 간격의 차이를 말한다. 만약 i 번째 패킷의 RTP timestamp가 Si 이고 i 번째 패킷에 대한 RTP 도착 timestamp가 Ri라고 한다면 두 개의 패킷의 i와 j의 시간 간격의 차이 D는 다음과 같이 주어진다[3].

$$D(i, j) = (Rj - Ri) (Sj - Si) = (Rj - Sj) - (Ri - Ri) \quad (\text{식-2})$$

i번째 패킷과 i-1번째 패킷의 시간간격의 차이값 D를 사용하여 송신자로부터 받은 각 데이터 i 패킷에 대한 지터는 계속적으로 다음과 같이 계산되어진다.

$$J = J + (|D(i-1, i) - J|) / 16 \quad (\text{식-3})$$

수신자측에서는 RTCP report를 보낼 때마다 계산된 현재의 지터값을 RTCP 패킷에 실어서 송신자에게 전송한다. 송신자측에서는 수신자로부터 받은 RTCP report의 패킷 필드에서 지터값을 읽어서 네트워크의 상태를 파악한다. 파악된 네트워크의 상태에 따라 수신자의 능력에 맞는 데이터량을 제공하기 위하여 적절한 전송량을 결정한다.

(3) 알고리즘

RTP/RTCP를 이용한 기본적인 알고리즘은 그림 5와 같다.

CRTPSession 객체에서는 Remote와 Local간의 접속을 유지하고 CRTP, CRTCP의 패킷이 중복되거나 다중세션을 지원할 경우 충돌 없이 패킷을 전송, 수신하는데 필요한 객체이다. 세션을 열고 원격지 호스트와의 지연을 측정하기 위해 임시 CRTCP만을 전송하여 Jitter를 계산하고, 전송 데이터 CRTP 패킷을 전송한 후, 다시 RTCP 패킷을 전송시킨다. 원격지에서 보내온 패킷이 있는지 검사하고, 다음에 전송할 RTP 패킷에 이전에 보낸 시간과 지금 보내는 시간의 지연시간을 (식-2), (식-3)으로 구하고 다시 전송을 한다.

(4) 결과

군산대(Pentium II-350, 256MB, Windows 2000, 203.234.48.125)와 고리넷(Pentium III-750, 256MB, Linux6.1, 211.48.254.242, 서울)의 두 시스템 왕복시간이 분당 8~290ms 정도 소요되는 상태에서 시뮬레이션 하였다. 최대 지연시간 허용 범위를 최대 지연시간에 대한 현재 패킷의 지연시간의 비율로서 5단계로 나누어 기본적으로 전송되어야 할 전송폭을 0.5배씩 증가, 감소시켜 버퍼링하였다. 지연율이 25% 이하일 경우, 패킷을 손실한 것으로 보고 다음 패킷을 수신하도록 하였다.

그 결과, TCP/IP전송 방식보다 RTP 전송 효율이 10%정도 높다는 것을 알 수 있었다. 지연시간에 의한 패킷 손실율은 기존의 전송방법 비슷하게 약 3%정도로 나타났다. 이것은 RTP와 QoS 알고리즘이 더욱 최적화된다면 전송효율을 더 향상시킬 수 있음을 의미한다.

V. 결론

본 논문에서는 실시간에 비디오 데이터를 전송하기 위하여 RFC 1889와 1890을 기반으로 RTP/RTCP를 이용해서 고품질의 비디오 데이터를 멀티캐스트하는 방법을 연구하였다.

실시간 전송을 위해 RTP/RTCP는 이용하여 원격지와 로컬간의 패킷 지연시간을 스트림을 지속적으로

로 계산하고, 지연 시간을 5단계로 나누어 데이터의 전송폭을 유동적으로 변화시킬 수 있도록 하였다. 또한 동일한 비디오 스트림에 대해 서비스를 요구하는 클라이언트가 증가할 때 전송대역폭을 과부하를 초래하지 않도록 멀티캐스트를 이용하여 전송하도록 하였다.

향후 연구로 다중 세션을 처리, 버퍼관리, 다중그룹관리의 오버헤드를 막기 위해서 지금보다 더 효율적이고 최적화된 알고리즘 개발과 RTP 패킷 손실에 대한 보상 방법 등을 추가적으로 연구해야 할 것이다.

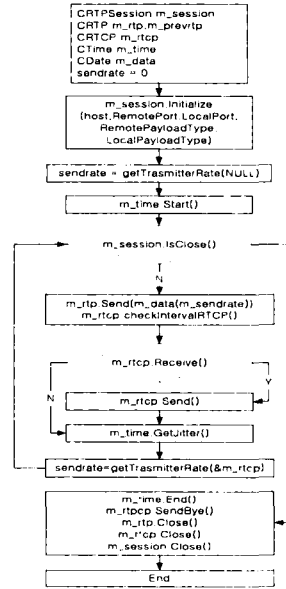


그림 4 RTP/RTCP 구현 알고리즘

참고문헌

- [1] 윤원용, "인터넷에서 저장형 MPEG 비디오 데이터의 효과적인 전송방법", 1998
- [2] Macedonia, M Brutzman D., "MBone provides Audio and Video across the Internet", IEEE Computer vol. 27, No.4, April 1994.
- [3] S. Casner, R. Frederick, V. Jacobson, "RTP : A Transport Protocol for Real-Time Applications", RFC1889, Jan 1996
- [4] H. Schulzrinne, "RTP Profile for Audio and Video Conferences with Minimal Control", RFC1890, Jan 1996
- [5] James F. Kurose, Keith W. Ross "Computer Networking", Addison Wesley, 2001
- [6] 강상욱, "A Study on Real-Time Media Synchronization in Multimedia Networks", 포항공대 석사학위논문, 1993