

상이한 사용자 환경을 고려한 멀티미디어 데이터의 멀티캐스트 알고리즘

이 승 하*, 나 윤 주*, 김 성*, 남 지 승*, 전 형 국**, 조 창 식**
* 전남대학교 컴퓨터공학과, ** 한국 전자통신연구원
전화 : 062-530-0422 / 핸드폰 : 016-607-2467

Seung-Ha Lee, Yun Ju Na, Sung Kim, Ji Seung Nam
Dept. of Computer Engineering, Chonnam University
E-mail : shlee7@chonnam.ac.kr

Multicast Algorithm of Multimedia data over Heterogeneous User Environment

Abstract

Nowadays, Internet users are widespread internationally. Following the users requirements, server has to transfer real-time multimedia data to many clients. Multicast can significantly reduce the system overhead and the network resource waste, so it supports efficient mechanism when server has to transfer same data to many clients.

We can deliver multimedia data transfer in real-time using Patching Algorithm which has no delay for service request. And multicast algorithm can support different network bandwidth using Receiver-driven Layered Multicast under heterogeneous environment network. The purpose of this paper is to suggest an algorithm that can create new multicast group adequately and transfer real-time multimedia data efficiently under heterogeneous environment.

I. 서론

최근 인터넷 사용자의 증가와 네트워크 기술의 발전으로 인하여 서버에서 다수의 수신자에게 대용량의 데이터를 전송해 주어야 하는 실시간 멀티미디어 서비스가 점차 늘어나고 있다. 멀티캐스트는 다중의 수신자에게 동일한 데이터의 전송시 시스템의 부하를 줄이고, 네트워크 자원

낭비를 줄여주는 효율적인 메커니즘을 제공한다.

수신자가 요청한 데이터를 실시간으로 전송해 주기 위해서는 요청시마다 새로운 멀티캐스트 그룹이 생성되어야 하므로 멀티캐스트의 장점을 기대하기 어렵다. 또한 각기 다른 네트워크 대역폭을 갖는 인터넷 환경에서 단일 전송률을 유지하는 경우에는 수신자의 네트워크 상태에 따라 대역폭의 낭비나 데이터 손실등이 발생할 수 있다 [1]. 따라서 본 논문에서는 인터넷 환경에서 각기 다른 대역폭을 갖는 수신자가 실시간 멀티미디어 데이터를 요청하였을 때 적절한 멀티캐스트 그룹을 생성하여 효과적으로 전송해 주는 알고리즘을 제안하고자 한다.

II. 멀티캐스트 전송 기법

2.1 실시간 멀티미디어 데이터 전송

서버는 동일한 데이터에 대한 다중의 클라이언트의 요청에 대해 멀티캐스트를 이용함으로써 시스템의 작업처리량을 높일 수 있다. 데이터를 멀티캐스트로 전송해 주기 위해서는 사용자의 서비스 요청을 받아 일정시간 동안 대기시킨 후 대기중인 다수의 수신자에 대해 멀티캐스트 채널을 통해 일괄적으로 데이터를 전송해 주게 된다 [2]. 이러한 멀티캐스트 전송기법을 batching기법이라 한다. 그러나 일정시간 동안의 지연이 필요한 이 기법을 통해 실시간 전송 요청을 처리해 주기 위해서는 몇가지

문제점이 있다.

실시간 처리를 높이기 위하여 사용자의 waiting time을 짧게 갖는 경우 멀티캐스트를 통한 이점이 감소되며 반면 waiting time 길게 주는 경우 일찍 요청한 사용자는 그만큼의 지연을 겪게된다. 즉, 실시간 전송을 위해서는 사용자 각각의 요청시간에 따라 독립적으로 빨리 대응해 주어야 하며 각 멀티캐스트 그룹은 많은 수의 사용자에게 서비스를 제공해 주어야 한다는 매우 상반된 요구사항을 충족하여 주어야 한다.

Patching이라는 멀티캐스트 기법은 실시간 데이터 전송요청에 대한 지연없이 서비스를 제공해주는 것을 가능하게 한다[3].

Patching기법에서는 크게 두 개의 멀티캐스트 그룹으로 나눌 수가 있는데, 일정 시간내에 접속한 사용자에게 동일한 멀티캐스트 데이터를 전송을 하여주는 Regular 채널과 데이터의 처음부터 Regular채널을 통하여 받지 못한 부분까지의 전송을 담당하는 Patching채널이 있다. 이 Patching기법은 그림 1과 같다.

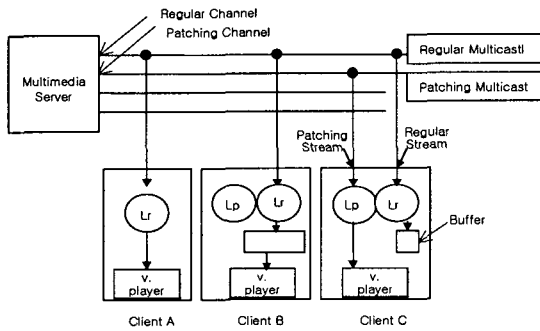


그림 1. Patching 기법

수신자는 Patching채널이 있는 경우 데이터 전송을 받은 즉시 미디어 데이터를 재생기를 통하여 재생하며, Regular채널의 데이터는 수신측의 버퍼에 저장해 놓은 후 Patching채널의 전송이 완료된 후 저장 버퍼에 있는 데이터를 재생한다.

2.2 가입자들의 이질성을 고려한 멀티캐스트 전송

일반적으로 데이터 전송시 서버는 수신자들의 네트워크 상태를 고려하여 평균적인 전송율로 데이터를 전송하게 된다.

그러나, 각기 다른 네트워크 대역폭을 갖는 인터넷 환경에서 단일 전송률을 유지하는 경우에는 수신자의 네트워크 상태에 따라 대역폭의 낭비나 데이터 손실등이 발생할 수 있다. 다양한 네트워크 상태를 고려하여 전송해 주

기 위한 여러 기법 중 McCane에 의해 제안된 RML (Receiver-driven Layered Multicast)은 송신자가 멀티미디어 데이터를 여러개의 데이터 스트림으로 나누어 계층화된 코딩(layered coding)된 형태로 전송한다[4]. 하나의 데이터 프레임을 여러 개의 데이터 스트림으로 나누어 전송하고 각 수신자는 자신의 네트워크 상태에 따라 전송받을 데이터 스트림의 개수를 결정한다. 수신자는 이 데이터 스트림을 많이 받을수록 양질의 데이터를 제공할 수 있다.

계층적 비디오 코딩(layered video coding)의 기술을 실시간 멀티미디어 통신에 적용하여, 압축된 비디오 정보를 영상의 기본적인 정보를 갖는 하나의 기본계층과 고품질의 정보를 갖는 복수 개의 상위 계층으로 나누어 각 계층을 서로 신뢰도가 다른 채널로 전송함으로써 이질적 통신망 환경의 다양성에 무관하게 사용자 단말에서 자신의 환경에 적합하게 각 계층을 취사선택 할 수 있도록 한다.

수신자는 네트워크 상태를 파악하여 혼잡이 발생했을 때에는 한 계층을 내려 그룹에서 탈퇴하고, 무 부하일 때에는 한 계층을 올려 그룹에 가입한다. 이와같이 수신자는 새로운 그룹에 가입을 하거나 탈퇴를 함으로서 자신의 능력에 맞는 서비스를 제공받을 수 있다. RLM에서 송신자는 계층화된 신호를 대응되는 멀티캐스트 그룹에게 전송한다.

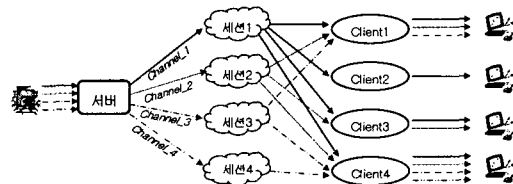


그림 2. 이질적인 통신망에서의 Multicast 서비스를 통한 데이터 송수신 관계

III. 멀티캐스트 그룹 생성 알고리즘

본 장에서는 실시간 멀티미디어 데이터 전송을 위한 Patching기법과 이질적인 통신망에서의 수신자 네트워크 상태를 고려한 RLM기법을 이용하여 두 조건을 만족하는 알고리즘을 제안하고자 한다.

3.1 사용자의 요청 처리

사용자가 원하는 멀티미디어 데이터 정보와 자신의 IP Address를 가지고 서버에 요청하면 서버는 서버의 멀티캐스트 관리 테이블을 참조하여 Patching_ID와

상이한 사용자 환경을 고려한 멀티미디어 데이터의 멀티캐스트 알고리즘

Regular_ID의 값을 돌려준다.

상이한 대역폭을 갖는 수신자를 위한 계층적 수신을 처리하기 위해서는 수신자 측에서 네트워크 정보를 통해 QoS를 처리하여 자신의 네트워크 상황에 따라 서버에 추가적인 계층 전송을 요청한다. 이에 대한 계층적 전송은 각 계층들이 서로 다른 IP multicast address를 사용하여 전송되게 된다.

서버가 전송한 네트워크 Report 패킷에 대하여 수신자가 이 패킷을 수신하여 네트워크 상태를 분석한다. 이들 패킷의 정보를 이용하여 네트워크 대역폭의 정보를 알 수 있다. 수신측에서는 네트워크 상황을 무부하, 부하, 혼잡인 세가지의 상태로 나누어 수신자는 이 값을 통해 추가적인 데이터 채널을 전송받는다. 네트워크가 혼잡 상태인 경우 가장 최근에 전송받은 상위 레이어 채널을 닫고, 패칭 채널 역시 데이터 수신이 종료됨과 함께 채널을 닫는다. 수신측에서 처리되는 알고리즘은 그림 3과 같다.

```
Req_Data(Client_ID, Multimedia_ID)
Begin
    request data to server;
    it receives a token (P_ID, R_ID);
    /* Data Transfer */
    if P_ID is not null {
        receives Patching Multicast data and play;
        receives Regular Multicast data and save buffer;
    } else
        /* first request client only receive
           regular channel */
    it receives Regular Multicast data and play;
    /* Control Message */
    check enhance_ID from client_session table;
    if NetworkQoS is unload
        request enhance_ID[last+1] to server;
    else QoS is congestion
        request dispatch enhance_ID[last];
End;
```

그림 3. 수신자 알고리즘

표 1. 수신자 멀티캐스트 채널 관리 테이블

구 성	기 능
P_ID	현재 수신되고 있는 Patching 채널
R_ID	현재 수신되고 있는 Regular 채널
Enhance_ID[i]	수신중인 Enhancement 채널

3. 2 송신자의 멀티캐스트 데이터 전송

송신자에게 멀티미디어 데이터를 요청한 수신자가 이미 전송중인 R_ID에 포함되는 경우 새로운 P_ID 데이터 채널을 전송받아야 하며 그렇지 않은 경우 R_ID가 새로 생성되어 하나의 데이터 채널만 전송받는다.

레이어드 코딩 기법은 일정 시간동안에 전송이 완료되는 Patching 채널에 대하여는 고려치 않으며 Regular 채널에 대해서만 적용토록 한다. Regular 채널은 레이어드 코딩에서 가장 기본적인 정보만을 갖고 있는 Basic Layer이며 수신자의 네트워크 상황에 따라 송신자에게 추가로 요청하는 데이터 채널은 Enhancement Layer이다. Patching 채널은 수신자의 접속요청 시간이 다름에 따라 각기 다른 크기의 데이터 전송이 이루어진다. 따라서 이는 유니캐스트로 전송되며 Regular 채널을 전송하는 부분은 해당 멀티캐스트 그룹에 Join요청을 하는 것으로 처리된다. 송신측에서의 처리되는 알고리즘은 그림 4와 같다.

```
Send_Data(Client_ID, Multimedia_ID)
Begin
    receive request from receiver in waiting queue;
    if it has free channel
        Initialize the service a token as (P_ID=null,
            R_ID=null);
        /* select R_ID */
    if no regular channel of Multimedia_ID
        set RID = FreeChannel
    else
        set PID=Freechannel and RID=LatestRegular
    /* Send Data */
    if (patching databuffer = null)
        dispatch P_ID;
    else send patching channel;
    if (regular databuffer = null)
        dispatch R_ID;
    else send regular channel;
    /* Send Enhancement channel */
    if request Enhancement channel
        join Enhancement Multicast channel;
    else request dispatch Enhancement channel
        leave Enhancement Multicast channel;
    /* Send Network QoS check Packet */
    send Control Packet;
End;
```

그림 4. 송신자 알고리즘

표 2. 송신자 멀티캐스트 채널 관리 테이블

구 성	기 능
Client_ID	데이터 채널을 전송받고 있는 수신자를 식별하기 위한 ID
Data_ID	서버가 제공해 줄수 있는 멀티미디어 데이터에 대한 식별자
P_ID[i]	송신되고 있는 모든 Patching 채널에 대한 식별자
R_ID[i]	송신되고 있는 모든 Regular 채널에 대한 식별자
Enhance_ID[i]	송신되고 있는 모든 Enhancement 채널에 대한 식별자

IV. 인터넷상에서 시스템 구현

앞서 설명한 알고리즘을 구현하기 위하여 실시간 전송 프로토콜인 RTP를 이용하여 데이터 전송을 하며, 네트워크 QoS는 RTCP프로토콜을 사용하여 정보를 얻는다. 실시간 전송은 데이터의 정확도보다 시간지연에 더 민감하므로 UDP를 이용한 멀티캐스트 전송을 하며, 멀티캐스트 채널은 크게 Regular채널과 Enhancement채널이며 추가요청한 상위계층에 대하여 멀티캐스트 데이터가 추가로 전송된다. Patching채널은 UDP를 이용한 유니캐스트 전송을 하며, 사용자의 전송요청 등의 메시지 처리 및 RTCP 메시지는 TCP를 이용한 유니캐스트 전송으로 구현한다.

서버, 클라이언트간 시스템 구성도는 아래 그림 5와 같다.

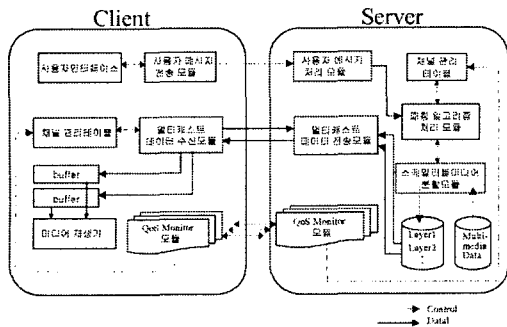


그림 5. 시스템 구성도

V. 결론 및 향후 연구 방향

본 논문은 최근 들어 이질적인 통신망 환경으로 연결된 인터넷을 통한 비디오나 오디오 같은 대용량 멀티미디어 데이터의 서비스가 증가함에 따라 이를 실시간으로 멀티캐스트 전송을 통해 제공해 주고자 하는 알고리

즘에 대하여 기술하였다.

실시간 멀티캐스트 전송을 위한 Patching 알고리즘과 통신망과 단말기의 QoS(Quality of Service) 상태에 따라 서비스 품질형태를 다양하게 지원하고 효율적인 네트워크 자원을 사용하기 위해 계층적 코딩(Layered Coding)기법을 사용하였다.

본 논문에서 제안한 이 알고리즘을 통하여 대용량의 멀티미디어 데이터를 전송하고자 하는 요구에 대하여 적절한 대응을 할 수 있을 것이다. 단, 제시한 알고리즘이 효율적으로 사용되기 위해서는 보다 많은 수신자의 요구시에 큰 성능을 기대할 수 있으리라 생각한다.

앞으로의 연구과제는 새로운 Regular채널의 생성을 효과적으로 지원하기 위한 연구가 지속되어야 한다.

참고문헌(또는 Reference)

- [1] Elan Amir, "An Agent-based Approach to Real-time Multimedia Transmission over Heterogeneous Environments", Ph.D. Thesis, University of California, Berkeley, 1998
- [2] A.Dan, D. Sitaram, and P. Shahabuddin. "Scheduling policies for an on-demand video server with batching", In Proc. of ACM Multimedia, pp.15-23, San Francisco, California, October 1994.
- [3] K. A. Hua, Y. Cai, and S. Sheu, "Patching : A multicast technique for true video-on-demand services", in Proc. of ACM Multimedia, pp.191-220, September 1998.
- [4] McCanne. S, Jacobson. V, Vetterli. M, "Receiver-driven Layered Multicast," ACM SIGCOMM '96, Stanford, CA, August 1996, pp. 117-130
- [5] Li. X, Paul. S, Ammar. M. H, "Layered Video Multicast with Retransmissions(LVMR): Evaluation of Hierarchical Rate Control," IEEE INFOCOM 98, March 1998.
- [6] 나승구, 고민수, 안중석, "인터넷상에서 원격회의를 위한 두 가지 흐름 제어 기법", 정보과학회 논문지, 제 26권 제 8호, pp.975-983, 1999.8