

이동 환경에서 프락시를 사용한 효과적인 미디어 스트림 전송 기법

남 동 훈, 박 승 규
아주대학교 정보 및 컴퓨터공학부
E-mail: ndh@madang.ajou.ac.kr

An Effective Method of Media Stream Transmission using Proxy in Mobile Environments

Dong-Hoon Nam, Seung-Kyu Park
Division of Information and Computer Engineering, Ajou University

요 약 이동 컴퓨팅 환경에서의 멀티미디어 스트림을 전송하는 경우에는 무선 네트워크의 높은 오류 발생률(error rate)과 잦은 단절(disconnection)로 인하여 데이터의 도달시간이 불규칙하게 되므로 데이터가 자연스럽게 재생되지 못한다. 따라서 시간의 영향을 많이 받는 멀티미디어 데이터를 자연스럽게 재생하기 위해서는 데이터 도달 지연시간의 영향을 최소화하는 것이 필요하다. 본 논문에서는 이동 컴퓨팅 환경에서 효과적으로 멀티미디어 스트림을 전송하기 위한 기법을 제안한다. 이를 위해 프락시를 사용한 효과적인 버퍼링 방법을 사용하여 데이터의 도달시간을 일정하게 유지시키는 방법과, 사용자가 요구하는 서비스 품질(Quality of Service: QOS)을 만족시키면서 프레임의 지연에 좀 더 유연하게 대처할 수 있는 재생 제어방법을 제안하였다.

1. 서론

최근의 컴퓨팅 환경은 어느 곳에서나 작업을 끊임없이 계속할 수 있는 이동 컴퓨팅 환경으로 발전하고 있다. 이러한 환경은 기술의 발달에 따라 앞으로 더욱 급속도로 진행될 것으로 보인다. 또한 사용자들의 데이터에 대한 요구도 이전의 문자 위주의 정보가 아닌 그림이나 소리, 동화상등을 포함하는 종합적인 정보를 요구하는 추세이다. 따라서 앞으로는 이동 환경에서 멀티미디어 서비스를 제공하는 수준이 될 것이다.

하지만 현재의 이동 컴퓨팅 환경에서는 무선 네트워크의 높은 전송 에러율과 잦은 연결 단절(disconnection)로 인하여 멀티미디어 서비스를 제공하기 힘들다. 이는 무선 네트워크에만 한정되는 것이므로 고정된 유선 망으로 연결되어 있다면 충분히 멀티미디어 서비스가 가능하다. 특히, 유선 망에서는 정상적으로 전송된 데이터가 무선망을 거치게 되면서 에러가 발생하면, 재전송을 하게 되는데, 이때에는 무선망으로만 재전송이 일어나는 것이 아니라, 다시 서버로부터 데이터를 재전송 받아야 하기 때문에 네트워크의 낭비일 뿐만 아니라 전송 지연이 더욱 길어지게 된다.

2. 시스템 구조

그림 1에서는 이동 컴퓨팅 환경의 구조를 보여주고 있다. 본 연구에서는 프락시를 이용한 버퍼링 방법을 제안하였다. 프락시는 서버와 이동 호스트 사이의 유선망과 무선망을 연결해 주는 곳에 위치하게 된다. 따라서 기존의 MSS(Mobile

Support Station)에 위치하게 된다. 이것은 이동 호스트의 대리자(agent) 역할을 하며, 서버에서 오는 스트림을 버퍼링하여 가지고 있으면서 주기적으로 이동 호스트에게 전송을 한다.

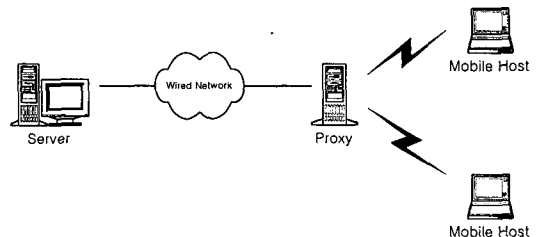


그림1. 무선 네트워크 환경의 동기화 모델

데이터가 도착하는 시간이 일정하지 않거나 요구되는 전송률을 유지하기 힘든 경우는 주기적으로 데이터를 재생해야 하는 멀티미디어 응용에는 적합하지 않다. 따라서 데이터의 전송률을 유지시켜서 데이터의 도달시간을 비교적 일정하게 만들어주는 것이 필요하다. TDMA와 같은 몇몇의 다중 접속 프로토콜(multiple access protocol)에서는 채널에 자원을 할당하는 방법을 사용하여서 일정하게 데이터의 전송률을 유지시킬 수 있다. 만약 네트워크의 급격한 부하증가나 데이터 전송 오류가 없다면, 이렇게 데이터의 전송률을 유지시키는 것으로도 데이터가 비교적 고정된 시간 범위(jitter bound) 내에 도착하게 된다. 하지만 이렇게 자원을 할당하는 방법으로 데이터의 전송률을 유지하는 것은 best effort 방법을 사용하는 인터넷 망 환경이나, 지역 네트워크 환경(LAN)과 같은 유선 망에서는 구현하기 힘들고, 또한 이러한 방법은 자원의 낭비를 가져올 수도 있다. 본 논문

본 연구는 한국과학재단 목적기초연구 (KOSEF 96-01-01-05-1-3) 지원으로 수행되었음.

에서는 자원 할당을 통한 채널의 할당을 무선네트워크 환경에서 만 사용하고, 무선 연결(wireless link)은 고정된 전송률을 가진다고 가정한다.

때때로 jitter나 네트워크 지연은 전송되는 동안 거치게 되는 장치들의 수에 비례한다. 유선 네트워크 환경에서 데이터 스트림은 많은 네트워크 장치들 - 예를 들어서 스위치나 라우터와 같은 - 을 통하여 전송되고 이러한 것들이 전송지연과 같은 것을 만들어서 jitter나 네트워크 지연의 주요한 원인이 되고 있다. 물론 전송되는 동안 외부 환경에 의하여 데이터가 변하여 발생하는 데이터 오류 등으로 재전송을 하여도 지연이 생길 수는 있지만 이것은 유선 네트워크의 환경에서는 매우 적은 경우이므로 여기서는 고려하지 않기로 한다.

3. 프락시를 사용한 버퍼링 모델

본 논문에서는 서버와 이동 호스트 사이에 [그림 1]과 같이 프락시를 두어서 서버로부터 전송되는 미디어 스트림을 버퍼링 하도록 만들었다. 프락시는 feedback을 통한 버퍼레벨 제어 방법을 사용하여 버퍼에 있는 데이터의 양을 일정하게 유지시킴으로써 버퍼 overflow나 underflow를 막고, 유선 네트워크 등을 통해서 발생할 수 있는 전송 지연이나 jitter에 대해서도 안정시킬 수 있다.

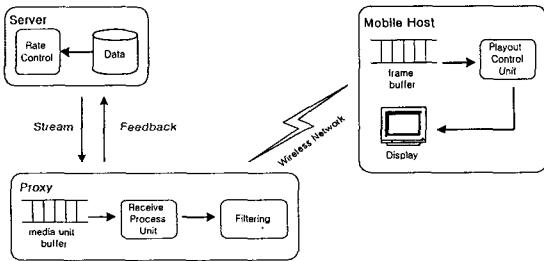


그림 2. 버퍼링 모델

프락시에 미디어 유닛 버퍼를 두는 주된 이유는 네트워크를 통하여 데이터가 전송되는 동안 발생할 수 있는 네트워크 지터(jitter)의 영향을 줄이는데 목적이 있다. 네트워크를 통하여 도달하는 데이터는 도달시간이 일정하지 않으므로 버퍼를 사용하여 이러한 효과를 감소시켜야 한다.

또 다른 이유는 버퍼 레벨을 측정하여 현재의 네트워크 상태를 알아내기 위한 것이다. 서버에서 보내는 데이터는 미디어 유닛 버퍼를 일정한 전송 주기로 전송하지만, 네트워크 내에서의 데이터 지연 등으로 도달시간은 일정하지가 않게 된다. 이러한 결과로 버퍼의 수준이 일정한 수준을 유지하다가 계속해서 떨어지게 되고, 현재 네트워크에서 지연이 많이 발생하는 것을 알 수 있다.

이동 호스트의 프레임 버퍼는 사용자에게 자연스러운 재생을 제공하기 위한 재생 제어를 하기 위한 것이다. 미디어 유닛 버퍼에 있는 미디어를 데이터 프로세스 유닛에서 프레임 단위로 분리한 후에 이를 압축 해제하여 프레임 버퍼에 집어넣는다. 프레임 버퍼도 미디어 유닛 버퍼처럼 버퍼의 레벨을 측정하여 프레임의 화면 표시시간을 결정하게 된다.

재생 제어 유닛은 프레임의 재생 시간에 맞추어 데이터를 재생

하는 것과 다른 미디어와의 동기화, 즉 인터미디어 동기화를 담당한다.

feedback을 사용하여 버퍼의 양을 일정하게 유지시키는 방법은 프락시가 버퍼의 상태를 계속해서 감시하면서 네트워크의 상태를 파악하고, 이 정보를 서버에게 feedback하게 된다. 그러면 서버는 feedback받은 정보를 기초로 데이터의 전송주기를 변경시켜서 클라이언트의 버퍼를 일정하게 유지시킬 수 있는 것이다. 이때, 서버에게 feedback을 계속해서 보내는 것이 아니라, 버퍼의 상한경계(high threshold)와 하한경계(low threshold)를 정하고 이 범위를 벗어난 경우에만 전송하게 된다. 서버에게 주는 정보는 전체 버퍼의 크기 중에서 현재 버퍼가 점유하고 있는 비율을 보내주게 되면, 서버는 그것을 가지고 전송 주기를 변화시킨다.

4. 재생 제어

멀티미디어 데이터의 재생시에는 화면에 보여질 데이터가 준비 되면 바로 화면으로 보여주는 것이 아니라, 그 프레임이 재생 되어야 하는 시간에 맞추어서 화면에 디스플레이 되어야 하는데, 이렇게 재생 시간에 맞추어 데이터를 재생하는 것을 재생 제어라고 한다. 동기화를 위하여 버퍼제어만을 사용하면 전체 시간을 못맞출 수 있다. 따라서 이러한 재생제어를 함으로써 미디어가 재생되어야 할 시간을 맞추는 것이다.

본 논문에서 제안하는 재생 제어 방법은 프레임 버퍼의 레벨에 따라 프레임이 화면에 보여지는 기간(display duration)을 계산하여 화면에 디스플레이 되는 속도를 점진적으로 높이고 낮추는 방법을 사용하여 재생 속도를 조절한다. 이렇게 함으로써 프레임 버퍼에 프레임들이 쌓일 수 있는 시간을 확보한다.

제안된 재생제어 방법에서 i번째 프레임의 화면표시 유지시간은 다음과 같다. 전체 버퍼에서 버퍼의 중간 수준과 현재의 버퍼레벨과의 차이를 비율로 나타낸 것이 다음 수식이다.

$$\Delta t = \frac{B/2 - L_{current}}{B} \quad \text{수식-1}$$

그리고 위 수식을 사용하여 조정된 화면 표시 유지시간을 얻는다.

$$C = \frac{1 + \Delta t}{R} \quad \text{수식-2}$$

하지만, 프레임의 화면 유지시간이 급격하게 변하지 않도록 하기 위한 다음과 같은 smoothing 함수를 사용한다.

$$D_i = \alpha C + (1 - \alpha) D_{i-1} \quad \text{수식-3}$$

여기서, $L_{current}$ 는 현재 프레임 버퍼의 수준이고, B는 프레임 버퍼의 크기이다. 그리고 R은 이 스트림의 재생 주기이며, D_i 는 i번째 프레임의 화면 유지시간이다. 또한 α 는 바로 전 프레임과 현재 프레임의 화면 유지시간이 급격하게 변하지 않도록 하기 위한 smoothing factor이다.

전체 버퍼 크기의 1/2 - 유지하고자 하는 버퍼 레벨 - 에서 현재 버퍼의 레벨을 빼면 버퍼 레벨을 유지하기 위한 프레임 수가 나온다. 이것을 전체 버퍼의 크기로 나누게 되면, 버퍼 크기에 대한 비율이 된다. 미리 규정된 재생 주기에 지속시간을 곱한 것을 더하게 되면, 프레임의 화면 지속 시간이 계산된다. 하지만, 갑작스러운 지속시간의 변화가 아니라 점진적인 지속시간의 변화가 이루어지기 위해서, 바로 전 프레임과의 지속시간 차이를 줄여야 한다. 수식-4는 이러한 완만한 지속시간의

변화를 위한 식이다. 여기서 α 는 앞 뒤 프레임의 지속시간 차이를 줄이기 위한 smoothing factor로써 0에서 1 사이의 값을 가지며, 보통 0.6정도를 준다.

$$D_i = \frac{\alpha(1.5 - \frac{L_{current}}{B})}{R} + (1 - \alpha)D_{i-1} \quad \text{수식-4}$$

수식-1, 2, 3을 전개하면 수식-4를 얻을 수 있다. 위의 수식을 사용하여 i 번째 프레임의 화면 유지시간을 구한다.

5. 모의 실험

실험에는 초당 30프레임의 해상도를 가지는 10분 길이의 비디오 클립을 사용하였다. 평균 전송 지연은 약 300ms 정도로 하였고, 프록시에 있는 버퍼의 크기는 약 2초 정도의 용량을 가지며, 이동 호스트에 있는 버퍼는 1초 정도의 재생을 할 수 있는 프레임울 보관할 수 있는 크기를 가지도록 하였다. 또한, 연결 단절에서도 제안한 프록시 버퍼가 유효한가를 실험하기 위해 인위적으로 시작시간에서 14초 후에 1초 동안의 단절이 발생하도록 하였다.

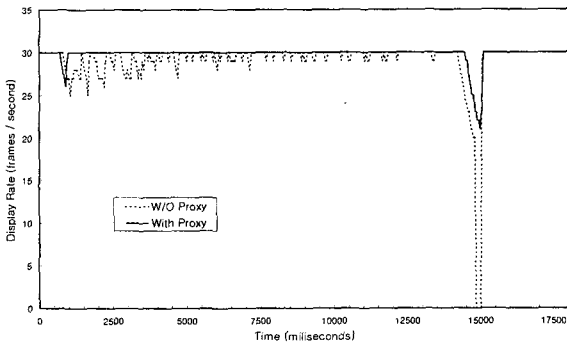


그림 3. 재생률 비교

그림3은 프록시를 사용한 경우와 사용하지 않은 경우의 재생률의 변화를 보이고 있다. 프록시를 사용한 경우 전반적으로 초당 30프레임 정도의 재생률을 꾸준히 유지하는 반면, 프록시를 사용하지 않은 경우는 프레임을 건너뛰는 스킵(skip) 등이 발생하여 재생률의 변화가 발생하는 것을 볼 수 있다.

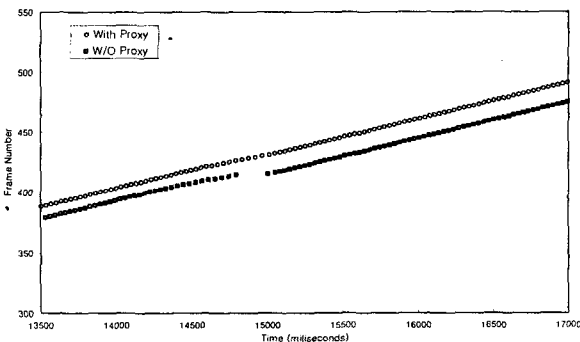


그림 4. 단절기간이 1초일 경우의 재생 비교

그림4는 프록시를 사용한 경우, 단절이 발생한 동안에도 끊김

없이 재생되어지는 것을 보인다. 반면에 프록시를 사용하지 않았을 경우에는 14.8초에서 재연결이 이루어진 15.1초 사이에 재생이 원활하게 이루어지지 않고 끊기는 현상이 발생하였다. 이것은 이동 호스트에 항상 1초정도 재생할 수 있는 분량의 프레임이 적제되어있지 않기 때문이며, 무선 연결이 재연결 되었음에도 불구하고, 서버에서 프레임을 가져오는데 네트워크 지연 시간이 더 걸리기 때문에 발생하는 것으로 보인다.

6. 결론

본 논문에서는 이동컴퓨팅 환경에서 멀티미디어 데이터의 재생 시에 발생하는 데이터의 지연 문제를 해결하기 위하여 프록시를 두어 버퍼링하는 구조를 제시하고 있다. 또한 작은 단절이 발생하는 환경에 적합한 재생 제어 방법을 제안하였다.

모의실험 결과 프록시와 제안한 재생제어 방법을 사용한 경우 짧은 기간동안의 단절에 대해서도 끊김 없이 재생할 수 있음을 보여주고 있으며, 전체적으로 안정된 재생율을 보여줌으로써 프레임의 지연에 좀 더 유연하게 대처할 수 있음을 알 수 있다.

본 연구는 주로 비디오 정보에 대해 다루고 있고, 오디오 정보는 앞의 연구와 조금 다른 특성이 있으므로 이를 보완하는 연구가 앞으로 이루어져야 할 것이다. 또한 본 논문에서 제안한 재생제어 방법은 단지 사용자의 입장에서 보았을 때 보다 자연스럽게 재생되는 것을 고려하였기 때문에 전체 재생시간을 지키는 것이 더 중요한 응용에서는 다른 방법을 사용해야 한다.

References

- [1] Bruce Zenel, Dan Duchamp, "A General Purpose Proxy Filtering Applied to the Mobile Environment", Columbia University.
- [2] Akihiro Hokimoto and Tatsuo Nakajima, "Handling Continuous Media in Mobile Computing Environment", Japan Advance Institute of Science and Technology.
- [3] Jochen Seitz, Nigel Davies, Michale Ebner, Adrian Friday, "A CORBA-based Proxy Architecture for Mobile Multimedia Applications", Proceedings of the 2nd IFIP/IEEE International Conference on Management of Multimedia Networks and Service(MMNS'98), November, 1998.
- [4] Suresh Singh, "Quality of Service Guarantee in Mobile Computing", Department of Computer Science, University of South Carolina Columbia.
- [5] Anupam Joshi, Sanjiva Weerawarana and Elias N. Houstis, " On Disconnected Browsing of Distributed Information", University of Missouri Columbia, Department of Computer Sciences Purdue University.
- [6] B. R. Badrinath, Arup Acharya and Tomasz Imielinski, "Designing distributed algorithms for mobile computing networks", Department of Computer Science, Rutgers University.