

논문 2011-4-15

모바일 환경에서 실시간 비디오 스트림 전송을 위한 적응형 부하 조정 기법

Adaptive Load Balancing Scheme for Real-Time Video Stream Transmission in Mobile Environment

김진환*

Jinhwan Kim

요약 본 논문에서 실시간 비디오 스트림을 효율적으로 전송하기 위한 적응형 부하 조정 기법이 제시된다. 모바일 환경에서 비디오 요청을 하는 클라이언트의 재생 버퍼 수준은 일시적으로 높아지거나 낮아질 수 있다. 제시된 기법은 버퍼 수준이 낮은 클라이언트의 비디오 요청을 먼저 서비스하기 위하여 더 많은 통신망 대역폭을 할당하게 된다. 즉 분산 모바일 시스템에서 재생 버퍼 수준에 따라 통신망 대역폭의 양이 동적으로 할당된다. 제시된 부하 조정 기법은 비디오 재생시 서비스 품질과 실시간적 성능 향상을 위하여 재생시간 전까지 클라이언트에게 성공적으로 전송되는 프레임의 수를 최대화한다. 또한 재생 상황을 더욱 적합하게 유지함으로써 모든 클라이언트들에게 공평한 서비스가 제공될 수 있다. 제시된 부하 조정 기법의 성능은 시뮬레이션을 통하여 다른 정적 부하 조정 기법과 비교 분석되었으며 종료시한내에 성공적으로 전송되는 프레임의 비율이 더 높은 것으로 나타났다.

Abstract We propose an adaptive load balancing scheme to transport real-time video streams efficiently in this paper. The playback buffer level of a video requesting client is high or low temporarily in mobile environment. This scheme attempts to allocate more network bandwidth to serve a video request with the lower buffer level preferentially. In this scheme, the amount of network bandwidth is dynamically allocated to the requesting clients according to their playback buffer levels in a distributed mobile system. In order to improve the quality of service and real-time performance of individual video playback, the proposed load balancing scheme tries to maximize the number of frames that are transported successfully to the client prior to their playback times. Fair services can also be provided to all the concurrent clients by making their playback situation more adaptive. The performance of this load balancing scheme is compared with that of other static load balancing scheme through extensive simulation experiments, resulting in the higher ratio of frames transmitted successfully within given deadlines.

Key Words : adaptive, load balancing, buffer level, real-time, network bandwidth.

1. 서 론

모바일 환경에서 스마트폰이나 스마트패드를 이용하는 클라이언트들은 3G 통신망, WiFi 또는 WiBro 등의

무선 통신망으로 비디오 서버에 연결된다^[1]. 최근 무선 통신망의 급격한 성장과 발전은 동영상을 비롯한 멀티미디어의 서비스 품질(QoS; Quality of Service)을 향상시키고 있다. 현재 대역폭이 개선된 무선 통신망은 비디오, 음성, 데이터 등 멀티미디어 트래픽에 대한 서비스 품질을 효율적으로 제공하기 위한 효과적인 방법으로 집중식(centralized) 스케줄러를 사용하고 있다^[2]. 이 집중식 스

*정회원, 한성대학교 멀티미디어공학과
접수일자 2011.3.15, 수정일자 2011.6.25
게재확정일자 2011.8.12

케줄러는 트래픽의 양과 채널 조건에 따라 대역폭을 동적으로 할당할 수 있으며 스케줄링 서비스를 공평하게 수행할 수 있는 장점이 있다^[3]. 본 논문에서도 모바일 통신망 기반의 클라이언트-서버 시스템^[4]에서 비디오 서버가 비디오 스트림을 효율적으로 전송할 수 있도록 부하를 조정하는 집중식 스케줄러의 역할을 수행한다.

비디오 화면의 해상도와 품질로 인하여 다양한 작업 부하 특성을 갖는 비디오 요청 작업들을 서비스하기 위하여 비디오 서버는 작업 부하에 따라 무선 통신망의 대역폭을 할당하는 방법이 중요하며 이는 분산 모바일 멀티미디어 시스템의 전체 성능과 각 비디오의 재생 품질에 있어 중요한 설계 요소로 작용한다^[5]. 본 논문에서는 비디오 트래픽의 변동성과 일시적 과부하를 고려하여 각 클라이언트의 버퍼 수준에 따라 대역폭을 유연하게 할당하는 부하 조정 기법이 제시된다. 즉 서버는 통신망 대역폭이 부족할 경우 버퍼 수준이 낮은 클라이언트일수록 더 많은 대역폭을 우선적으로 확보하여 해당 비디오의 많은 프레임들을 전송함으로써 비디오 재생의 연속성을 최대한 유지할 수 있도록 한다.

유선 통신망에서 비디오 스트림의 전송과 재생 성능을 개선하기 위해 제시된 비디오 프레임 버퍼링, 피드백 제어 메카니즘, 비디오 스트림 스무딩 기법^[6, 7] 등은 모바일 시스템의 특성으로 인하여 모바일 비디오 재생장치 시스템에 동일하게 적용하기에는 여러 가지 제약 사항들이 있다^[8]. 비디오 트래픽 크기의 변동성과 일시적인 과부하 현상을 효과적으로 해결하기 어려우며 또한 반드시 여분으로 사용할 수 있는 대역폭을 확보해야 한다는 문제점이 있다. 또한 휴대폰 등 모바일 단말기의 버퍼는 아직 노트북이나 데스크탑 컴퓨터에 비하면 버퍼 용량이 크지 않다.

본 논문은 2장에서 모바일 환경에서 비디오 스트림을 전송하기 위한 클라이언트-서버 시스템 특성이 기술되며 3장에서 비디오 스트림을 전송하기 위하여 재생 버퍼 수준이 낮을수록 많은 통신망 대역폭이 할당되는 적응형 부하 조정 기법을 기술한다. 그리고 4장에서 시뮬레이션을 통하여 다른 기법과 비교된 성능을 분석하며 5장에서는 결론을 기술한다.

II. 모바일 환경의 클라이언트-서버 시스템

1. 클라이언트-서버 시스템

모바일 환경에서 모바일 클라이언트와 비디오 서버는 셀(cell) 단위를 기반으로 하는 모바일 통신망이나 무선 LAN인 WiFi 또는 광역 무선 인터넷인 WiBro 등으로 연결된다. 모바일 클라이언트들은 이러한 무선 통신망을 이용하여 비디오 서버에 비디오 재생을 요청하게 된다. 휴대폰, 스마트폰 등을 사용하는 thin 클라이언트들은 노트북 컴퓨터나 스마트 패드를 사용하는 thick 클라이언트에 비하여 비디오 버퍼 크기가 작고 제한이 있게 된다^[9]. 클라이언트들마다 작업 부하의 크기가 상이한 비디오를 요청하게 되며 예를 들어 고성능 노트북을 이용하는 thick 클라이언트의 비디오 작업 부하량은 휴대폰을 사용하는 thin 클라이언트에 비하여 훨씬 커지게 된다.

모바일 환경에서 jitter 현상을 최소화하기 위해 클라이언트에 버퍼 기법이 필요하나 thin 클라이언트 경우 thick 클라이언트에 비하면 버퍼 용량이 충분하지 않다. 결국 버퍼 크기가 작은 클라이언트일수록 버퍼가 큰 클라이언트에 비하여 통신망 오류에 민감하게 되며 비디오 재생시 과부하에 따른 성능 저하가 발생하게 된다^[10]. 모바일 통신망과 무선 LAN의 대역폭이 점차 개선되고 있는 하나 아직도 유선 통신망에 비하면 대역폭 크기가 작고 전송시 오류 발생 확률이 크며 통신망 내에서의 단절 현상도 빈번하게 발생한다.

모바일 클라이언트는 서버로부터 수신한 프레임들을 자신의 버퍼에 저장하며 버퍼 수준이 사전에 정의된 수치에 도달하면 비디오의 재생을 시작하게 된다. 즉 복원 과정이 시작되는 것이며 실제 비디오의 재생은 복원된 프레임에 비디오 재생장치에 한 개씩 전송함으로써 수행된다. 그러나 비디오 프레임의 재생 시간을 놓치게 되면 해당 프레임은 즉시 재생이 취소되며 폐기된다. 이러한 현상은 끊김없는(seamless) 비디오 스트리밍(streaming) 유지에 심각한 영향을 미치게 되며 재생율(1초간 재생되는 프레임들의 수)에 관한 서비스 품질과 실시간적 성능도 저하시키게 된다^[10].

2. 비디오 서버

비디오 서버가 다수의 비디오 요청 작업을 동시에 서비스할 경우에는 해당 비디오 파일을 저장 장치에서 검색하며 비디오 프레임들을 요청별로 지정된 비디오 버퍼에 저장하게 된다. 본 논문에서 비디오 서버는 MPEG-4 표준으로 압축된 비디오를 가지고 있는 것으로 가정하며

각 비디오 스트림은 GOP(Group of Picture) 시퀀스로 구성되어 있다. 각 GOP는 I, P, B 프레임들로 구성되며 통상 I 프레임의 크기는 B 프레임이나 P 프레임보다 훨씬 크다. GOP마다 크기가 상이하며 비디오의 고정된 재생율에 따라 각 비디오 스트림의 대역폭 요건은 가변적 특성을 가지게 된다. 이 비디오 프레임들은 패킷으로 구성되며^[11] 무선 통신망을 통하여 해당 모바일 클라이언트에 게 전송된다.

III. 적응형 부하 조정 기법

1. 부하 조정을 위한 수락 제어

고화질 비디오 재생을 위한 비디오의 가변적 비트율은 모바일 환경에서도 해결하기 어려운 문제로 간주된다. 본 논문에서는 제한된 무선 통신망의 대역폭을 효과적으로 사용함은 물론 모든 모바일 클라이언트의 요청을 공평하게 서비스할 수 있는 효율적인 부하 조정 기법을 비디오 서버에 구성하고자 한다.

무선 통신망의 비디오 서버는 클라이언트가 요청한 MPEG-4 비디오 스트림을 전송하기 전 클라이언트들이 요청한 각 비디오 스트림의 평균 비트율을 이용하여 수락 제어(admission control) 절차를 수행한다. 이 평균 비트율은 1초동안 재생되는 프레임의 수인 재생율과 프레임의 평균 크기(비트 단위)를 곱한 값으로 결정된다. 예를 들어 재생율이 30프레임/초이고 프레임의 평균 크기가 10000 비트인 경우 평균 비트율은 300000비트/초가 된다. 클라이언트들이 실제 사용할 수 있는 무선 통신망의 최대 대역폭을 B_{max} 라 정의할 때 모든 클라이언트들의 평균 비트율의 합은 항상 B_{max} 보다 작거나 같아야 한다. 비디오 패킷 전송시 통신 오류가 발생할 수 있으며 재전송 과정이 필요하게 되므로 B_{max} 이외에 별도의 대역폭이 필요하나 본 논문에서는 무선 통신망의 오류 발생에 대한 구체적인 기술은 생략한다.

새로운 클라이언트가 비디오 서버에게 비디오 스트림의 재생을 요청할 경우 서버는 이 비디오 스트림의 평균 비트율을 포함하여 평균 비트율의 합을 다시 산정하게 된다. 이때 평균 비트율의 합이 B_{max} 보다 클 경우 비디오 서버는 새로운 클라이언트의 요청을 거부할 수 있다. 거부된 비디오 스트림은 나중에 서버가 재수용할 수 있으나 이에 관한 구체적인 과정은 본 논문에서 기술하지 않

기로 한다.

2. 적응형 부하 조정 기법

수락 제어 이후 각 비디오 스트림의 실제 비트율은 VBR 특성^[12]에 따라 매초 크기가 변동되어 평균 비트율과 차이가 발생한다. 크기가 가변적인 $i(1 \leq i \leq n)$ 번째 스트림의 실제 비트율은 다음 수식에서 B_i 로 표기되며 이들의 합은 B_{sum} 으로 표기된다.

$$B_{sum} = \sum_{i=1}^n B_i$$

비디오 서버는 매주기(본 논문에서는 1초로 가정함)마다 n 개의 스트림이 요구한 실제 비트율의 합 B_{sum} 과 B_{max} 를 비교하게 된다. B_{sum} 이 B_{max} 를 초과하는 경우 통신망 대역폭의 한계로 인해 서버가 전송을 중단시켜야 하는 프레임들이 발생하게 된다. 전송이 중단되는 프레임들의 수가 많아질수록 해당 클라이언트는 끊임없는 비디오 스트리밍 유지가 어렵게 되고 재생율 및 서비스 품질(QoS)이 저하되는 현상이 발생한다. 본 논문에서는 이러한 현상을 최소화시키고자 서버가 각 클라이언트의 버퍼 수준을 고려하여 매초마다 대역폭을 동적으로 할당한다. 각 클라이언트는 휴대폰 또는 스마트폰 등(이하 단말기로 기술함)에 설치된 버퍼에 재생율을 고려한 일정 시간 동안 프레임들이 서버로부터 도달하면 이때부터 버퍼에 있는 프레임을 디코딩하여 재생장치로 전달하게 된다. 따라서 버퍼 수준이란 클라이언트의 단말기가 일정 시간 동안 전송된 프레임들을 저장할 수 있는 버퍼의 크기를 의미한다. 클라이언트의 버퍼 수준이 높을수록 실제 버퍼의 용량이 큰 것을 의미하는 것이다.

모바일 클라이언트는 단말기의 특성상 기억장치의 크기가 상이하며 실제 버퍼의 수준도 상이하게 된다. 버퍼 수준이 높을수록 오랜 시간 동안 재생할 수 있는 많은 수의 프레임들을 버퍼에 유지하게 되며 통신망의 일시적 장애나 오류 발생시에도 덜 민감하게 반응할 수 있는 장점이 있다^[13]. 그러나 버퍼 수준이 높을 경우 비디오 재생이 시작되는 시간이 늦어지고 기억장치 활용도가 저하되는 문제가 있으며 모바일 단말기에 대용량의 버퍼를 유지하기 어려운 면이 있다.

i 번째 비디오 스트림을 위한 단말기의 버퍼 수준을 $Level_i$ 라 하며 시간 단위의 초를 이용하여 크기를 설정한다. 예를 들어 $Level_i$ 가 3인 경우 해당 단말기의 버퍼에 3초간 재생할 수 있는 프레임들이 저장되어 있음을 의미

한다. 실제로 단말기의 버퍼 수준이 3이라 하더라도 비디오 스트림에 따라 3초간 저장되는 프레임들의 평균 크기가 다를 수 있으므로 단말기 버퍼마다 버퍼의 실제 크기는 상이할 수 있다.

```

Btrans: 전송될 비트율의 합
B'i: 전송될 대역폭이 조정된 스트림 i의 비트율
Di: 전송이 중단된 스트림 i의 비트율

Btrans=0
do
  Leveli가 가장 작은 값을 찾고 Levelmin으로 설정
  if Levelmin=0
  then
    Leveli가 0인 n'개 대역폭들의 합 B'sum 설정
    for each i from 1 to n'
      if B'sum > Bmax
      then
        B'i=Bmax/B'sum * Bi
      else
        B'i=Bi
    else
      if Btrans+Bi > Bmax
      then
        B'i=Bmax-Btrans
      else
        B'i=Bi
    Btrans=Btrans+B'i
  while ( Btrans < Bmax)
  for each i from 1 to n
    Leveli=Leveli+B'i/Bi
    if Leveli < 1.0
    then
      Di=(1.0-Leveli)*Bi
      Leveli=0
    
```

그림 1. $B_{sum} > B_{max}$ 경우의 부하 조정 기법
Fig. 1 Load Balancing Scheme for $B_{sum} > B_{max}$

서버는 전송하고자 하는 실제 비트율의 합 B_{sum} 이 B_{max} 보다 클 경우 단말기 버퍼 수준이 가장 낮은 비디오 스트림부터 먼저 전송하도록 통신망 대역폭을 할당하는 부하 조정 기법이 적용되며 이 알고리즘은 그림 1에 의사 언어로 기술되어 있다. 따라서 버퍼 수준이 상대적으로

높은 단말기는 새로운 프레임들을 수신하지 못한 상태에서 버퍼에 있는 현재 프레임들을 계속 재생하게 되며 버퍼 수준이 감소되는 결과가 발생할 수 있다. 버퍼 수준이 낮았던 단말기는 서버로부터 새로운 프레임들을 수신함으로써 수신한 만큼 버퍼 수준이 증가하게 된다. 그리고 B_{sum} 이 B_{max} 보다 작거나 같은 경우에는 서버가 모든 단말기의 프레임들을 전송한 후 여분의 대역폭을 버퍼 수준이 가장 낮은 단말기부터 추가로 할당하여 다음 주기에 전송할 프레임들을 이번 주기에 미리 전송함으로써 해당 단말기의 버퍼 수준이 추가로 증가된다. 이 경우에 대한 알고리즘은 그림 2에 기술되어 있다.

```

Bnext: Levelmin을 갖는 스트림의 1초후 비트율
B''i: 추가로 전송되는 스트림 i의 비트율
for each i from 1 to n
  B'i=Bi
  Leveli=Leveli+1.0
Btrans=Bsum
do
  Leveli가 가장 작은 값을 찾고 Levelmin으로 설정
  if Btrans+Bnext > Bmax
  then
    B''i=Bmax-Btrans
    Leveli=Leveli+B''i/Bi
  else
    B''i=Bnext
    Leveli=Leveli+1.0
  Btrans=Btrans+B''i
  while (Btrans ≤ Bmax)
    
```

그림 2. $B_{sum} \leq B_{max}$ 경우의 부하 조정 기법
Fig. 2. Load Balancing Scheme for $B_{sum} \leq B_{max}$

그림 1에서 B_{trans} 는 버퍼수준이 최소값인 단말기부터 전송되는 비트율의 합을 의미하며 매초마다 서버는 초기치를 0으로 설정한다. 서버는 $Level_i$ 가 최소인 값을 찾은 후 이를 $Level_{min}$ 으로 설정한다. 이때 $Level_{min}$ 이 0이면 버퍼수준이 0인 모든 단말기들에 대하여 동일한 비율로 스트림을 전송하게 된다. 이 단말기의 수를 n' 라 하고 해당 비트율의 합을 B'_{sum} 으로 정의한다. B'_{sum} 이 B_{max} 보다 큰 경우 버퍼수준이 0인 단말기로 전송될 대역폭 B_i 는 B_{max}/B'_{sum} 에 비례한 만큼만 전송될 수 있으며 이는 B'_i

로 표기된다. B_{trans} 는 B'_i 만큼 증가된다. B_i 에서 B'_i 를 차감한 대역폭은 일단 전송이 보류되며 1초후 전송될 수 있다. B'_{sum} 이 B_{max} 이하일 경우에는 B_i 가 모두 전송되며 B'_i 와 B_i 는 같고 결국 B_{trans} 는 B_i 만큼 증가된다.

버퍼수준의 최소값 $Level_{min}$ 이 0보다 클 경우 서버는 이 값을 갖는 단말기로 보낼 대역폭 B_i 와 B_{trans} 의 합을 B_{max} 와 비교한다. B_{trans} 와 B_i 의 합이 B_{max} 보다 클 경우 결과적으로 B_{max} 에서 B_{trans} 를 차감한 만큼만 전송될 수 있으므로 B'_i 는 $B_{max}-B_{trans}$ 가 된다. B_{trans} 와 B_i 의 합이 B_{max} 이하일 경우에는 B_i 가 모두 전송될 수 있으므로 B'_i 는 B_i 가 된다. B'_i 가 결정될 때 마다 B_{trans} 값이 증가되며 증가된 B_{trans} 가 B_{max} 보다 작을 경우에는 동일한 과정들이 반복된다.

B_{trans} 가 B_{max} 와 같게 되면 서버는 각 단말기의 버퍼수준을 산정한다. 실제 전송될 비트율 B'_i 와 전송하려했던 비트율 B_i 의 비율을 현재 버퍼수준 $Level_i$ 에 더하면 새로운 $Level_i$ 가 결정된다. 즉 B'_i 와 B_i 가 동일할 경우 $Level_i$ 는 1.0이 증가되는 것이며 B'_i 가 B_i 의 1/2일 경우 0.5가 증가하게 된다. 다시 산정한 $Level_i$ 가 1.0 미만인 경우 해당 단말기가 재생할 수 있는 시간이 1초미만인 것을 의미한다. $Level_i$ 가 0.5라면 해당 단말기는 0.5초 동안만 비디오 스트림을 재생할 수 있으며 나머지 0.5초는 새로운 프레임들을 재생할 수 없게 된다. 따라서 서버는 종료시한내에 재생할 수 없는 나머지 0.5초 분량에 해당하는 프레임들의 전송을 중단하게 된다. 이와 같이 서버에 의해 전송 중단되는 스트림 i 의 대역폭을 D_i 로 표기한다. 전송이 중단되는 프레임의 수가 많아질수록 서비스 품질과 실시간 성능이 저하된다. 각 단말기에서 프레임들이 재생된 시간만큼 버퍼수준은 감소되며 본 논문에서는 1초 단위로 감소되는 것을 가정한다. 그러나 버퍼수준 $Level_i$ 가 1.0미만인 경우에는 버퍼에 있는 모든 프레임들이 재생되며 $Level_i$ 는 0으로 재설정된다.

전송이 보류된 대역폭에 해당하는 프레임들은 다음 주기에 전송될 수 있으나 전송이 중단된 대역폭에 속한 프레임들은 서버에 의해 폐기되는 것을 의미한다. 본 논문에서는 전송이 중단되는 프레임들에 대하여 우선순위 정책을 적용한다. 즉 중요도가 가장 낮은 B 프레임들이 먼저 중단되며^[11] 이후 P 프레임, I 프레임 순으로 결정됨으로써 중요도가 큰 프레임일수록 전송이 중단되는 경우가 최소화되도록 하였다.

실제 비트율의 합 B_{sum} 이 B_{max} 보다 작거나 같을 경우

모든 스트림들의 비트율은 해당 단말기로 전송되며 $Level_i$ 는 1.0만큼 증가된다. 이후 B_{max} 에서 B_{sum} 을 차감한 비트율을 활용하기 위해 $Level_i$ 가 가장 작은 스트림 대역폭부터 추가로 전송한다. 즉 서버는 다음 주기에 전송할 프레임들을 이번 주기에 미리 전송할 수 있도록 하는 것이다. $Level_{min}$ 을 갖는 스트림의 1초후 비트율 B_{next} 를 B_{sum} 과 동일한 B_{trans} 에 더한 값이 B_{max} 를 초과하면 B_{next} 비트율중 $B_{max}-B_{trans}$ 만큼만 전송되며 이는 B''_i 로 표기된다. 해당 단말기의 버퍼 수준은 B''_i/B_i 만큼 증가된다. $B_i-B''_i$ 대역폭은 서버가 전송을 보류하며 1초후 정상적으로 전송하게 된다. B_{trans} 와 B_{next} 의 합이 B_{max} 보다 작은 경우에는 B_{next} 가 모두 추가로 전송되며 B''_i 는 B_{next} 가 되고 $Level_i$ 는 1.0만큼 증가된다. 증가된 B_{trans} 가 B_{max} 이하일 경우에는 최소 버퍼수준을 찾아서 대역폭을 추가로 전송하는 과정이 반복되며 그렇지 않은 경우에는 서버의 부하 조정 기법이 종료된다.

IV. 성능 분석

본 논문에서는 압축률이 우수한 VBR(Variable Bit-Rate) 방식으로 부호화된 MPEG-4 비디오 스트림 "Star Wars IV"와 "Jurassic Park One"들을^[14] 대상으로 실험을 수행하였다. 이 비디오 스트림들은 모두 QCIF 형식이며 비율 제어(rate control)없이 I, P, B 프레임에 대한 양자화 계수가 30으로 동일하게 설정된 시간적 scalable 방식으로 압축되었다. 한 GOP내 프레임 수는 12이고 GOP 패턴은 IBBPBBPBBPBB이다. 두 비디오 스트림들의 총 프레임 수는 각각 108000개이며 "Star Wars IV"와 "Jurassic Park One"의 프레임의 평균 크기는 각각 1980.016비트와 2215비트로 분석되었다. 비디오 스트림의 재생율을 24로 동일하게 적용할 경우 "Star Wars IV"와 "Jurassic Park One"은 각각 스트림당 47760.384비트/초와 53160비트/초의 비트율이 평균 비트율로 산정되며 $n(=20)$ 개의 클라이언트를 대상으로 할 경우 평균 비트율의 합이 각각 955207.68비트/초와 106320비트/초가 되며 이는 통신망의 최대 전송 대역폭 B_{max} 와 각각 크기가 같다고 가정한다.

단말기 버퍼 수준을 사전에 파악하여 비디오 스트림 별로 통신망 대역폭을 정적으로 할당하는 BSLB(Buffer Sensitive Load Balancing) 기법^[8]과 버퍼수준의 최소값부터 우선적으로 전송하도록 대역폭을 조정하는 본 논문

의 적응형 부하 조정기법인 ALB(Adaptive Load Balancing) 기법의 성능이 비교 분석되었다. BSLB 기법은 이용가능한 통신망 대역폭을 스트림의 수로 나눈 후 이 값을 각 스트림의 평균 대역폭과 비교하여 더 작은 값을 해당 스트림의 실제 대역폭으로 결정한다. 이후 모든 스트림의 실제 대역폭의 합이 이용가능한 통신망 대역폭보다 작을 경우에만 여분의 통신망 대역폭이 발생하며 이 여분의 대역폭을 스트림별로 단말기 버퍼 수준을 고려하여 추가 할당하게 된다. 즉 버퍼 수준이 낮을수록 버퍼 수준이 높은 스트림에 비하여 여유 대역폭을 더 많이 할당받은 후 프레임들을 전송하게 된다. 그러나 BSLB 기법은 스트림별로 통신망 대역폭을 정적으로 할당한 후 전송 중 발생하는 단말기 버퍼 수준을 별도로 고려하지 않기 때문에 이러한 상황을 고려하여 동적으로 대역폭을 조절하는 ALB 기법과 성능을 직접 비교하기는 사실상 어렵다. 본 실험에서는 BSLB 기법과 ALB 기법의 수락 제어 단계에서 통신망의 여유 대역폭은 전혀 없는 것으로 가정한다. 두 기법은 동일한 실험 환경에서 시뮬레이션이 70분간 수행되었으며 이 결과를 토대로 두 기법의 성능이 비교 분석되었다.

서버는 동일한 비디오 스트림을 20개의 클라이언트 단말기들에 대하여 평균도착시간이 30초이며 1초부터 601초 사이의 임의의 시간에 전송이 시작되도록 하였다. 본 논문에서는 두 기법의 성능을 비교하기 위하여 프레임 전송성공율이 측정된다. 서버가 해당 클라이언트 단말기로 전송해야 하는 프레임들의 수와 단말기가 실제로 수신한 후 성공적으로 재생한 프레임들의 수의 비율이 전송성공율로 정의된다. 즉 서버가 임의의 클라이언트 단말기로 100개의 프레임을 전송해야 하나 통신망 대역폭의 부족으로 10개의 프레임을 중단하고 90개의 프레임만 전송하여 단말기가 재생하였다면 전송성공율은 90%가 되는 것이다. 즉 전송성공율이 클수록 클라이언트 단말기가 재생하는 프레임들의 수가 증가하는 것이다.

20개 스트림들의 전송이 모두 시작된 이후인 25분부터 10분 간격으로 65분까지 전송성공율이 측정되었다. 모든 단말기 버퍼 수준의 초기치가 각각 1.0, 5.0, 9.0으로 설정된 상태에서 측정된 두 기법의 프레임 전송성공율이 표 1과 표 2에 나타나 있다.

표 1. "Star Wars IV"의 전송성공율(%)

Table 1. Ratio of frames that are transported successfully for "Star Wars IV"

기법	25분	35분	45분	55분	65분
BSLB	84.67	83.06	83.72	83.85	83.87
ALB-1	95.56	96.03	96.95	97.42	97.22
ALB-5	95.86	96.01	96.98	97.47	97.26
ALB-9	96.27	96.27	97.15	97.63	97.40

표 2. "Jurassic Park One"의 전송성공율(%)

Table 2. Ratio of frames that are transported successfully for "Jurassic Park One"

기법	25분	35분	45분	55분	65분
BSLB	80.33	82.41	83.72	83.85	83.87
ALB-1	95.96	92.36	92.57	93.95	94.68
ALB-5	95.75	92.16	92.41	93.44	93.96
ALB-9	95.92	92.22	92.39	93.55	93.40

"Star Wars IV"의 경우 BSLB 기법은 여분의 통신망 대역폭이 없으므로 버퍼 수준의 크기에 상관없이 프레임 전송성공율(S-BSLB)이 동일하며 약 80.33%에서 86.32%로 분석되었다. 그러나 ALB 기법의 경우에는 버퍼 수준의 초기치가 1.0인 경우(ALB-1) 프레임 전송성공율이 약 95.56%에서 97.42%로 나타났으며 버퍼 수준의 초기치가 5.0과 9.0인 경우(ALB-5와 ALB-9)에 전송성공율이 거의 비슷한 수준으로 측정되었다(표 1 참조). 버퍼수준의 초기치가 크다고 해서 전송성공율이 이에 반드시 비례하여 커지는 것은 아닌 것으로 분석되었다. "Jurassic Park One"의 경우에도 BSLB 기법의 전송성공율은 83.06%에서 84.67% 범위인 반면 ALB 기법은 버퍼 수준의 초기치가 1.0일때(ALB-1) 전송성공율은 92.36%에서 95.96% 범위로 나타났으며 버퍼수준의 초기치가 5.0과 9.0(ALB-5와 ALB-9)일때도 결과의 차이는 거의 없는 것으로 분석되었다(표 2 참조). 여분의 통신망 대역폭이 없는 상태에서 버퍼 수준에 따라 부하를 조정하여 통신망 대역폭을 동적으로 할당하는 ALB 기법의 경우 대역폭을 정적으로 할당하는 BSLB 기법보다 프레임 전송성공율이 증가되는 결과가 나타났으며 버퍼 수준의 초기치 증가가 프레임 전송성공율의 증가를 의미하지 않는 결과가 나타났다. 그리고 프레임의 최대값을 평균값으로 나눈 비율은 "Star Wars IV"가 17.29로 10.49인 "Jurassic Park One"보다 크지만 프레임 전송성공율이 약 2-4% 정

도 더 높게 나타났으며 이는 평균대역폭을 초과하는 빈도수가 "Jurassic Park One"보다 상대적으로 낮게 발생한 것이 원인으로 분석되었다.

ALB 기법에서는 전송을 중단시켜야 하는 프레임들에 대해서 우선순위 정책을 사용하여 B 프레임, P 프레임, I 프레임 순서로 프레임의 전송이 중단된다. "Star Wars IV"와 "Jurassic Park One" 스트림들을 버퍼수준의 초기치가 1.0인 상태에서 전송중단된 프레임들 중 I, P, B 프레임들이 차지하는 비율을 측정된 결과 이는 우선순위 순서대로 각각 0.002%, 1.759%, 97.219%와 0.03%, 1.37%, 98.60%로 나타났다. 프레임의 전송을 중단시켜야 할 경우 우선순위가 가장 낮은 B 프레임들을 먼저 중단시킨 결과로 분석된다. BSLB 기법의 경우는 프레임별 우선순위 정책이 적용되지 않았기 때문에 비교 실험을 생략한다.

ALB 기법에서는 버퍼수준의 초기치가 설정된 이후 시간에 따른 평균 버퍼수준 값이 측정되었다. 그림 3과 4는 각각 "Star Wars IV"와 "Jurassic Park One" 스트림들의 평균 버퍼수준의 변동값을 25분부터 65분까지 10분 간격으로 나타내고 있다. "Star Wars IV"의 경우 55분부터 65분 사이에 스트림의 전송량이 크게 증가하여 평균 버퍼수준이 가장 낮게 나타났으며 35분부터 55분 사이에 전송량이 감소하여 평균 버퍼수준이 다시 증가한 것으로 분석되었다. "Jurassic Park One"의 경우 25분부터 35분 사이에 평균 버퍼 수준이 약 0.01로 가장 낮게 나타났으며 실제 45분부터 55분 사이에 평균 버퍼수준이 증가하였다가 55분과 65분 사이에 다시 감소하는 것으로 분석되었다.

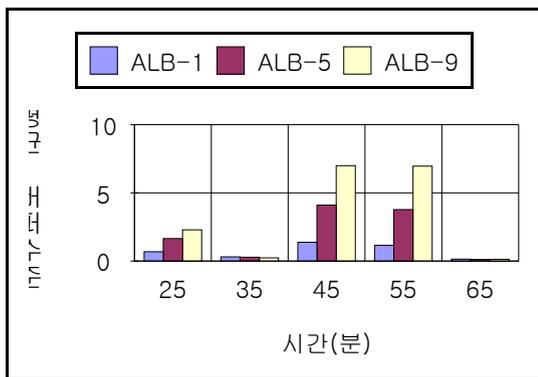


그림 3. "Star Wars IV"의 평균 버퍼수준
Fig. 3. Mean buffer level of "Star Wars IV"

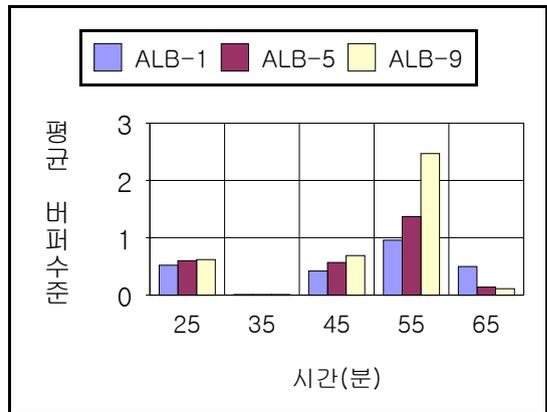


그림 4. "Jurassic Park One"의 평균 버퍼수준
Fig. 4. Mean buffer level of "Jurassic Park One"

V. 결론

본 논문에서 제시된 ALB 기법은 단말기 버퍼 수준이 가장 낮은 스트림부터 제한된 통신망 대역폭을 먼저 할당함으로써 성공적으로 전송되는 프레임의 수를 최대화한다. 프레임의 평균 크기와 재생율에 따라 평균 비트율이 결정되는 각 비디오 스트림은 VBR 특성상 실제 비트율의 크기가 가변적이다. 따라서 전송 중인 스트림의 비트율 합은 이용가능한 통신망 대역폭을 초과하는 기회가 자주 발생하게 된다. 이때 서버는 버퍼수준이 가장 낮은 단말기부터 우선적으로 프레임을 전송하고 버퍼수준이 큰 단말기일수록 다음 주기에 전송할 수 있도록 전송을 보류하는 적응형 조정 기법을 사용한다.

또한 전송 중인 스트림의 비트율 합이 이용가능한 통신망 대역폭보다 작을 경우에는 남은 대역폭을 버퍼 수준이 작은 클라이언트부터 먼저 추가로 할당하여 더 많은 프레임들이 전송되도록 한다. 본 논문의 ALB 기법은 성능 비교 결과 비디오 스트림의 서비스 품질과 끊임없는 비디오 스트리밍 효과를 향상시킬 수 있는 것으로 분석되었다. 따라서 제시된 ALB 기법은 통신망 대역폭이 한정된 모바일 환경의 멀티미디어 시스템에서 실시간적 성능과 서비스 품질을 더욱 향상시킬 수 있는 것으로 기대된다.

참 고 문 헌

- [1] S. Al-Majeed and M. Fluery, "Multi-connection TFRC video streaming in a concatenated network: latency and video quality", Proc. of Wireless and Mobile Network, pp. 34-45, Jun. 2010.
- [2] H S Guruprasad and H D Maheshappa, "Dynamic bandwidth allocation in distributed VoD using agent technology", European Journal of Scientific Research 33(3), pp. 536-545, 2009.
- [3] L. Xu, X. Shen and J. W. Mark, "Dynamic fair scheduling with QoS constraints in multimedia wideband CDMA cellular networks", IEEE Trans. on Wireless Communications 3(1), pp. 60-73, 2004.
- [4] H S Guruprasad and H D Maheshappa, "Dynamic load balancing architecture for distributed VoD using agent technology", Int'l Journal of Computer Science and Security 1(5), pp. 13-22, 2008.
- [5] K. Y. Lam, J. Yuen, S. H. Son, and E. Chan, "Scheduling video stream transmissions for distributed playback over mobile cellular networks," Proc. of Int'l Conf. on Parallel and Distributed Systems, Dec. 2002.
- [6] S. Rao and A. M. K. Cheng, "Scheduling and routing of real-time multimedia traffic in packet-switched networks," IEEE International Conference on Multimedia, July. 2000
- [7] S. Cen, C. Pu, and R. Staehli, "A Distributed Real-time MPEG Video Audio Player," International Workshop on Network and Operating System Support of Digital Audio and Video, 1995.
- [8] J. Yuen, K. Y. Lam, and E. Chan, "A Fair and Adaptive Scheduling Protocol for Video Stream Transmission in Mobile Environment," IEEE International Conference on Multimedia and Expo, Aug. 2002.
- [9] S. Mohapatra and N. Venkatasubramanian, "Proactive energy-aware video streaming to mobile handheld devices," IEEE Conference on Mobile and Wireless Communications Networks, 2003.
- [10] X. Cheng, P. Mohapatra, S. Lee, and S. Banerjee, "Performance evaluation of Video streaming in multihop wireless mesh networks," Network and Operating Systems Support for Digital Audio and Video, May, 2008.
- [11] S. Park, S. Lee, and J. Kim, "Adaptive high definition MPEG-2 Streaming using frame-based prioritization over IEEE 802.11a WLAN," SPIE, pp. 1880-1890, 2005.
- [12] M. Krunz, "Bandwidth Allocation Strategies for Transporting Variable-Bit-Rate Video Traffic," IEEE Communications Magazine, Jan. 1999.
- [13] X. Hei, Y. Liu, and K. W. Ross, "Inferring Network-Wide Quality in P2P Live Streaming Systems," IEEE Journal on Selected Areas in Communications 25(9), pp. 1640-1654, Dec. 2007.
- [14] <http://trace.eas.asu.edu/cgi-bin/main.cgi>

저자 소개

김 진 환(정회원)



- 1986년 서울대학교 컴퓨터공학과 졸업(학사)
- 1988년 서울대학교 컴퓨터공학과 졸업(석사)
- 1994년 서울대학교 컴퓨터공학과 졸업(박사)
- 1994년-1995년 서울대학교 컴퓨터

신기술공동연구소 특별연구원

• 1995년-현재 한성대학교 멀티미디어공학과 교수

<주관심분야 : 멀티미디어 시스템, 분산 실시간 시스템>

※ 본 연구는 한성대학교 교내연구비 지원과제입니다.