

# 분산 모바일 멀티미디어 시스템에서 재생 버퍼 수준에 기반한 동적 대역폭 할당 기법

김진환<sup>†</sup>

## 요약

분산 모바일 멀티미디어 시스템에서 클라이언트의 재생 버퍼 수준에 기반하여 통신망 대역폭을 동적으로 할당하는 기법이 본 논문에서 제시된다. 이 기법에서 비디오 요청을 서비스하기 위해 할당된 대역폭의 양은 요구한 클라이언트의 버퍼 수준에 따라 결정된다. 즉 클라이언트의 버퍼 수준이 일시적으로 낮아지거나 높아지면 재생 상황을 개선하기 위한 목적으로 더 많은 대역폭 또는 더 적은 대역폭이 할당되는 것이다. 또한 재생 버퍼 수준 기반 대역폭 할당 기법을 사용함으로써 클라이언트들에게 공평한 서비스가 제공될 수 있다. 고품질의 비디오 재생을 지원하기 위해서 비디오 프레임들은 반드시 재생 시간 전까지 클라이언트에 전송되어야 한다. 이 대역폭 할당 기법의 주목적은 전송이 중단되는 프레임의 수를 최소화하여 비디오 재생 성능과 서비스 품질을 향상시키며 동시에 진행되는 모든 비디오 요청 작업을 공평하게 서비스하는 것이다. 제시된 기법의 성능은 시뮬레이션을 통하여 다른 정적 대역폭 할당 기법과 비교 분석되었으며 프레임의 전송중단율은 버퍼수준에 따라 약 4%에서 9% 정도 감소된 것으로 나타났다.

키워드 : 분산 모바일, 비디오 재생, 버퍼 수준, 대역폭 할당, 서비스 품질

## A Dynamic Bandwidth Allocation Scheme based on Playback Buffer Level in a Distributed Mobile Multimedia System

Jinhwan Kim<sup>†</sup>

## ABSTRACT

In this paper, we propose a scheme for dynamic allocating network bandwidth based on the playback buffer levels of the clients in a distributed mobile multimedia system. In this scheme, the amount of bandwidth allocated to serve a video request depends on the buffer level of the requesting client. If the buffer level of a client is low or high temporarily, more or less bandwidth will be allocated to serve it with an objective to make it more adaptive to the playback situation of this client. By employing the playback buffer level based bandwidth allocation policy, fair services can also be provided to the clients. In order to support high quality video playbacks, video frames must be transported to the client prior to their playback times. The main objectives in this bandwidth allocation scheme are to enhance the quality of service and performance of individual video playback such as to minimize the number of dropped video frames and at the same time to provide fair services to all the concurrent video requests. The performance of the proposed scheme is compared with that of other static bandwidth allocation scheme through extensive simulation experiments, resulting in the 4-9% lower ratio of frames dropped according to the buffer level.

Keywords : Distributed Mobile, Video Playback, Buffer Level, Bandwidth Allocation, Quality of Service

## 1. 서론

지난 수년동안 모바일 통신망의 급격한 성장과 발전은 대역폭과 서비스 품질(QoS; Quality of Service) 보장 등 효과

적이며 효율적인 자원 활용에 대한 요구를 증가시켜 왔다 [1]. 분산 모바일 멀티미디어 시스템에서 모바일 클라이언트들은 모바일 셀룰라 통신망이나 무선 LAN과 같은 무선 통신망으로 비디오 서버에 연결된다[2]. 모바일 통신망과 무선 통신망의 대역폭이 과거보다 개선되고 있긴 하다. 그러나 고품질 비디오 화면의 해상도와 크기를 비롯하여 다양한 작업 부하 특성을 갖는 비디오 요청 작업들을 서비스하기 위하여 통신망 대역폭을 할당하는 방법은 분산 모바일 멀티

※ 본 연구는 2010년도 한성대학교 교내연구비 지원과제임.  
† 정희원 : 한성대학교 멀티미디어공학과 교수  
논문접수 : 2010년 5월 25일  
수정일 : 2010년 8월 18일  
심사완료 : 2010년 8월 30일

디어 시스템의 전체 성능과 각 비디오의 재생 품질에 있어 중요한 설계 요소로 작용한다[3].

유선 통신망의 분산 비디오 재생장치 시스템의 성능을 개선하기 위해 제시된 비디오 프레임 버퍼링, 피드백 제어 메카니즘, 비디오 스트림 스무딩 기법[4-5] 등은 모바일 시스템의 특성으로 인하여 모바일 비디오 재생장치 시스템에 동일하게 적용하기에는 여러 가지 제약 사항들이 있다[6]. 특히 휴대폰 등 모바일 단말기의 버퍼는 아직 노트북이나 데스크탑 컴퓨터에 비하면 버퍼 용량이 크지 않다. 이러한 모바일 단말기에 재생될 비디오 스트림을 위하여 모바일 통신망의 대역폭을 할당하는 기존의 스케줄링 기법들은[3, 6] 비디오 트래픽 크기의 변동성과 일시적인 과부하 현상을 효과적으로 해결하기 어려운 면이 있으며 또한 반드시 여분으로 사용할 수 있는 대역폭을 확보해야 한다는 문제점이 있다.

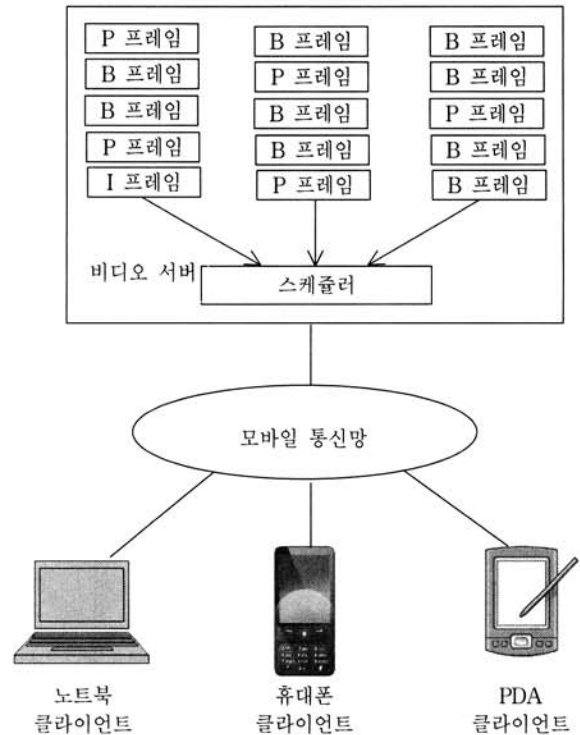
현재 무선 통신망은 비디오, 음성, 데이터 등 멀티미디어 트래픽에 대한 서비스 품질을 효율적으로 제공하기 위한 효과적인 방법으로 중앙 스케줄러를 사용하고 있다[2]. 이 중앙 스케줄러는 트래픽의 양과 채널 조건에 따라 대역폭을 동적으로 할당할 수 있으며 스케줄링 서비스를 공평하게 수행할 수 있는 장점이 있다. 본 논문에서도 모바일 통신망 기반의 클라이언트-서버 시스템[7-9]에서 서버가 중앙 스케줄러의 역할을 수행하며 비디오 스트림을 효율적으로 전송할 수 있도록 클라이언트의 버퍼 수준에 기반한 동적 대역폭 할당 기법이 제시된다.

서버는 각 클라이언트가 요청한 비디오의 평균 프레임 크기에 따라 모바일 통신망의 대역폭을 정적으로 할당하는 수락제어 단계를 먼저 수행한다. 그러나 실제 스케줄링 단계에서는 비디오 트래픽의 변동성과 일시적 과부하를 고려하여 각 클라이언트의 버퍼 수준에 따라 대역폭을 동적으로 할당하게 된다. 즉 서버는 통신망 대역폭이 부족할 경우 버퍼 수준이 높은 클라이언트에게 전송할 프레임들을 보유하며 확보한 대역폭을 버퍼 수준이 낮은 클라이언트에게 일시적으로 더 많이 할당함으로써 해당 프레임들을 전송하게 된다. 이 경우 버퍼 수준이 높은 클라이언트는 단말기 버퍼에 있는 현재 프레임들을 재생함으로써 비디오 재생의 연속성을 유지할 수 있게 된다.

본 논문은 2장에서 비디오 스트림 전송을 위한 모바일 통신망 기반의 클라이언트-서버 시스템 특성이 기술되며 3장에서 비디오 스트림을 전송하기 위하여 재생 버퍼 수준에 기반한 동적 대역폭 할당 기법을 기술한다. 그리고 4장에서 시뮬레이션을 통하여 다른 스케줄링 기법들과 비교된 성능을 분석하며 5장에서는 결론을 기술한다.

## 2. 분산 모바일 멀티미디어 시스템

본 논문에서는 모바일 통신망을 기반으로 하는 분산 모바일 멀티미디어 시스템을 고려한다(그림 1) 참조). 모바일 통신망에서는 기지국과 모바일 클라이언트들이 셀(cell) 단위를 기반으로 연결되며 기지국은 고속 통신망을 통하여 비



(그림 1) 분산 모바일 멀티미디어 시스템의 서버와 클라이언트

디오 서버와 연결된다. 모바일 클라이언트들은 기지국을 통하여 비디오 서버에 비디오 재생을 요청하게 된다. PDA, 휴대폰, 포켓 PC 등을 사용하는 thin 클라이언트들은 노트북 컴퓨터를 사용하는 thick 클라이언트에 비하여 비디오 버퍼 크기가 작으며 제한이 있게 된다[10]. 클라이언트들마다 작업 부하의 크기가 상이한 비디오를 요청하게 되며 예를 들어 고성능 PC를 이용하는 thick 클라이언트의 비디오 작업 부하량은 포켓 PC를 사용하는 thin 클라이언트에 비하여 훨씬 커지게 된다.

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 프레임들을 비디오 재생장치에 한 개씩 전송함으로써 수행된다. 그러나 비디오 프레임의 재생 시간을 놓치게 되면 해당 프레임은 즉시 재생이 취소되며 폐기된다. 이러한 현상은 끊임없는(seamless) 비디오 스트리밍(streaming) 유지에 심각한 영향을 미치게 되며 재생율(1초간 전송되는 프레임들의 수)에 관한 서비스 품질과 실시간적 성능도 저하시키게 된다[11].

비디오 서버가 다수의 클라이언트로부터 비디오 요청을 수신할 때 수락 제어 과정을 먼저 실시한 이후 다수의 비디오 요청 작업을 동시에 서비스할 수 있다. 서버가 비디오 요청의 수락을 허용한 경우에는 해당 비디오 파일을 저장장치에서 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) 시퀀스

로 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[12] 기지국과 모바일 통신망을 통하여 해당 모바일 클라이언트에게 전송된다. 비디오 패킷 전송시 통신 오류가 발생할 수 있으며 재전송 과정이 필요하게 된다. 본 논문에서는 모바일 통신망의 오류 발생에 대한 구체적 기술은 생략한다.

분산 모바일 멀티미디어 시스템은 유선 통신망에 기반한 분산 멀티미디어 시스템과 비교할 때 여러 가지 차이점을 가지고 있다. 우선 모바일 통신망과 무선 LAN의 대역폭이 점차 개선되고 있긴 하나 아직도 유선 통신망의 대역폭 크기에 비하면 작다. 또한 전송시 오류 발생 확률이 크고 통신망 내에서의 단절 현상이 빈번하며 모바일 통신망의 셀 단위에서는 상향 속도와 하향 속도가 비대칭적 특성을 가지고 있다. 모바일 통신망에서 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[11]. 고화질 비디오 재생을 위한 비디오의 가변적 비트율은 모바일 멀티미디어 시스템에서 해결하기 더욱 어려운 문제로 간주된다. 그리고 클라이언트마다 다른 해상도와 화질을 요구함에 따라 작업부하의 크기가 달라지고 일시적인 과부하가 자주 발생하므로 이를 공평하고 효율적으로 해결할 수 있는 기법들이 필요하게 된다. 본 논문에서는 제한된 모바일 통신망의 대역폭을 효과적으로 사용함으로써 모바일 클라이언트의 요청을 공평하게 서비스하는 효율적인 대역폭 할당 기법을 비디오 서버에 구성하고자 한다.

### 3. 재생 버퍼 수준 기반 동적 대역폭 할당 기법

#### 3.1 평균 대역폭과 버퍼 수준

모바일 통신망의 비디오 서버는 클라이언트가 요청한 MPEG-4 비디오 스트림을 전송하기 전 수락 제어(admission control) 절차를 수행한다. 클라이언트들이 요청한 비디오 스트림 n 개의 평균 대역폭은  $M_1, M_2, \dots, M_n$ 으로 각각 표기되며 이들의 합은 수식 (1)에서  $M_{sum}$ 으로 정의된다. 임의의 스트림  $i(1 \leq i \leq n)$ 의 평균 대역폭  $M_i$ 는 비트 수로 나타나는 프레임의 평균 크기를 재생율과 곱한 값으로 결정된다. 예를 들어 재생율이 30이고 프레임의 평균 크기가 1000 바이트인 경우  $M_i$ 는 다음과 같이 계산된다.

$$\begin{aligned} M_i &= 30\text{프레임/초} \times 1000\text{바이트/프레임} \times 8\text{비트/바이트} \\ &= 240,000\text{비트/초} \\ &\approx 0.24\text{Mbps} \end{aligned}$$

모바일 통신망에서 기지국과 클라이언트들 간에 사용가능한 최대 대역폭을  $B_{total}$ 이라 하며 실제 사용가능한 통신망 대역폭을  $B_{avail}$ 로 정의한다. 즉  $B_{avail}$ 은  $B_{total}$ 에서 오류 발생시 재전송을 위한 대역폭과 비디오 스트림 전송 이외의 목적으로 사용되는 대역폭을 제외한 대역폭을 의미한다.  $M_{sum}, B_{avail}, B_{total}$ 들에 대해서는 수식 (1)의 관계가 적용된다.

$$\begin{aligned} M_{sum} &= M_1 + M_2 + \dots + M_n \quad (n > 0) \\ &= \sum_{i=1}^n M_i \\ M_{sum} &\leq B_{avail} \leq B_{total} \end{aligned} \quad (1)$$

새로운 클라이언트가 요청한  $n+1$ 번째 비디오 스트림의 평균 대역폭  $M_{n+1}$ 이 추가된 대역폭의 합  $M_{sum}$ 이 수식 (1)의  $M_{sum} \leq B_{avail}$  관계를 계속 만족하면 서버가 이를 수용하지만 그렇지 않은 경우에는 이를 거부할 수 있다. 그리고 거부된 비디오 스트림은 나중에 서버가 재수용할 수 있으나 이에 관한 구체적인 과정은 본 논문에서 기술하지 않는다.

수락 제어 이후 각 비디오 스트림의 실제 대역폭은 VBR 특성[13]에 따라 크기가 가변적이며 일정 주기마다  $i$ 번째 스트림이 요청한 프레임들의 크기를 비트로 표시한 실제 대역폭  $B_i$ 와 이들의 합인  $B_{sum}$ 은 수식 (2)의 관계로 성립된다.

$$B_{sum} = \sum_{i=1}^n B_i \quad (2)$$

비디오 서버는 매초마다 각 스트림이 요구한 프레임들을 저장장치에서 검색 후 서버의 버퍼에 저장하게 되며  $B_{sum}$ 은  $B_{avail}$ 을 초과하는 경우가 발생할 수 있다. 이 경우 통신망 대역폭의 한계로 인해 서버가 전송을 중단시켜야 하는 프레임들이 발생하게 된다. 전송이 중단되는 프레임들의 수가 커질수록 해당 클라이언트는 끊임 없는 비디오 스트리밍 유지가 어렵게 되고 재생을 및 서비스 품질(QoS)이 저하되는 현상이 발생한다. 본 논문에서는 이러한 현상을 최소화시키고자 서버가 각 클라이언트의 버퍼 수준을 고려하여 매주기마다 대역폭을 동적으로 할당한다. 각 클라이언트는 휴대폰 또는 PDA 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[8]. 그러나 버퍼 수준이 높을 경우 비디오 재생이 시작되는 시간이 늦어지고 기

역장치 활용도가 저하되는 문제가 있으며 모바일 단말기에 대용량의 버퍼를 유지하기 어려운 면이 있다.

i번째 비디오 스트림을 위한 단말기의 버퍼 수준을  $L_i$ 라 하며 시간 단위인 초를 이용하여 크기를 설정한다. 예를 들어  $L_i$ 가 5인 경우 해당 단말기의 버퍼에 5초간 재생할 수 있는 프레임들이 저장되어 있음을 의미한다. 본 논문에서 임의의 비디오 스트림을 재생하기 위한 모든 단말기의 버퍼 수준은 초기치가 1보다 크거나 같은 값을 가지는 것으로 가정한다. 실제로 단말기의 버퍼 수준이 1이라 하더라도 비디오 스트림에 따라 1초간 저장되는 프레임들의 평균 크기가 다를 수 있으므로 단말기 버퍼마다 버퍼의 실제 크기는 상이할 수 있다. 서버는 각 스트림의 버퍼 수준을 일정 주기(본 논문에서는 1초로 가정함)마다 파악하며 버퍼 수준의 합  $L_{sum}$ 을 다음과 같이 정의한다.

$$L_{sum} = \sum_{i=1}^n L_i \quad (3)$$

### 3.2 동적 대역폭 할당

서버는 전송하고자 하는 실제 대역폭의 합  $B_{sum}$ 이 사용가능한 통신망 대역폭의 합  $B_{avail}$ 보다 클 경우 단말기 버퍼 수준을 고려하여 버퍼 수준이 높은 클라이언트일수록 비디오 스트림을 가급적 다음번에 전송하도록 대역폭을 조정한다. 즉 버퍼 수준이 가장 높은 클라이언트부터 프레임들의 전송이 보류되며 보류된 만큼의 대역폭은 버퍼 수준이 상대적으로 낮은 다른 클라이언트들을 위해 할당된다. 이번 주기에 보류된 프레임들은 다음 주기에 전송될 수 있다. 따라서 버퍼 수준이 높았던 클라이언트는 새로운 프레임들을 수신하지 못한 상태에서 버퍼에 있는 현재 프레임들을 계속 재생하게 되며 버퍼 수준이 감소되는 결과가 발생한다. 버퍼 수준이 낮았던 클라이언트는 서버로부터 새로운 프레임들을 수신함으로써 수신한 만큼 버퍼 수준이 증가하게 된다. 그리고  $B_{sum}$ 이  $B_{avail}$ 보다 작은 경우에는 서버가 모든 클라이언트의 프레임들을 전송한 후 여분의 대역폭을 버퍼 수준이 가장 낮은 클라이언트부터 추가로 할당하여 다음 주기에 전송할 프레임들을 이번 주기에 미리 전송함으로써 해당 클라이언트의 버퍼 수준이 추가로 증가된다. 실제 대역폭의 합  $B_{sum}$ 이  $B_{avail}$ 보다 클 경우와  $B_{sum}$ 이  $B_{avail}$ 보다 작거나 같을 경우의 동적 대역폭 할당 방법이 의사 언어로 (그림 2)와 (그림 3)에서 각각 기술되었다.

$B_{sum}$ 과  $B_{avail}$ 의 차는  $B_{lack}$ 으로 (그림 2)의 3행에서 기술되었으며  $B_{lack}$ 은 결국  $B_{sum}$ 을 전송할 때 부족한 대역폭을 의미한다.  $B_{lack}$ 이 0보다 클 경우(4행) 클라이언트들의 버퍼 수준 중 가장 큰 값인  $L_{max}$ 를 찾는 과정이 반복된다(6행).  $L_{max}$ 가 1.0을 초과하는 경우(7행)  $L_{max}$ 를 가진 클라이언트의 대역폭  $B_i$ 는  $B_{lack}$ 과 비교된다. 이  $B_i$ 가  $B_{lack}$ 보다 작거나 같을 경우(9행) 해당 클라이언트의 대역폭  $B_i$ 는 전송이 보류되며 이 대역폭을  $P_i$ 로(10행) 설정하고  $L_{max}$ 는 1.0이 차감된다(11행). 해당 클라이언트는 1초간 서버로부터 새로운 프레임의

```

1: if ( $B_{sum} > B_{avail}$ )
2:   begin
3:      $B_{lack} = B_{sum} - B_{avail}$ ;
4:     while ( $B_{lack} > 0$ )
5:       begin
6:          $L_i$ 가 가장 큰 값을  $L_{max}$ 로 설정함;
7:         if ( $L_{max} > 1.0$ )
8:           begin
9:              $L_{max}$ 에 해당하는  $B_i$ 가  $B_{lack}$ 보다 작거나 같으면
10:               $B_i$ 는 전송보류되며 이 대역폭은  $P_i$ 로 설정함;
11:               $L_{max}$ 를 1.0 차감하고  $B_{lack}$ 에서  $B_i$ 를 차감함;
12:               $L_{max}$ 에 해당하는  $B_i$ 가  $B_{lack}$ 보다 크면
13:                 $B_{lack}$ 만큼 전송보류되며 이 대역폭은  $P_i$ 로
                설정함;
14:                 $L_{max}$ 에서  $B_{lack}/B_i$  값을 차감하고  $B_{lack}$ 은 0
                으로
                설정함;
15:              end
16:            else //  $L_{max} \leq 1.0$ 
17:              begin
18:                 $B_i$ 에서 전송보류된  $P_i$ 를 차감한 값을  $B_i'$ 로
                설정함;
19:                 $B_{sum}' = \sum_{i=1}^n B_i'$ ;
20:                 $B_{lack}' = B_{sum}' - B_{avail}$ ;
21:                for  $i=1$  to  $n$  do
22:                  begin
23:                     $P_i' = B_{lack}' \times B_i / B_{sum}'$ ;
24:                     $D_i = B_i - P_i - P_i'$ ;
25:                  end
26:                 $B_{lack}$ 은 0으로 설정함;
27:              end
28:            end

```

(그림 2)  $B_{sum} > B_{avail}$  경우의 동적 대역폭 할당 알고리즘

전송이 보류되며 대신 단말기 버퍼에 있는 프레임들을 1초간 재생함으로써 버퍼 수준이 감소되는 것이다. 이와 같이 서버는 버퍼 수준이 낮은 다른 클라이언트들의 프레임들이 더 많이 전송될 수 있도록 버퍼 수준이 높은 클라이언트의 프레임들의 전송을 보류하게 된다.  $B_{lack}$ 은 전송이 보류된  $B_i$ 만큼 차감하게 되며(11행) 이는 필요한 통신망 대역폭의 크기를 감소시킨 것을 의미한다. 9행과는 달리  $L_{max}$ 를 가진 클라이언트의 대역폭  $B_i$ 가  $B_{lack}$ 보다 클 경우(12행)  $B_i$ 에서  $B_{lack}$ 을 차감한 대역폭은 전송이 가능하며  $B_{lack}$ 만큼만 전송이 보류되며  $P_i$ 로 설정된다(13행). 이 경우 해당 클라이언트는  $B_{lack}/B_i$  시간(단위 초)동안은 단말기 버퍼에 있던 프레임들을 재생하게 되므로 버퍼 수준도 이 비율만큼 차감된다(14행). 서버가 해당 클라이언트에게 전송하는 프레임들의 재생 시간은  $1.0 - B_{lack}/B_i$ 이 되는 것이다. 서버는 더 이상 보류할 수 있는 대역폭이 없으므로  $B_{lack}$ 은 0이 되고(14행) 4행의 반복문을 벗어나므로 새로운  $L_{max}$ 를 찾는 과정은 중지된다.

한편 7행과 달리  $L_{max}$ 가 1.0이하인 경우(16행) 모든 클라

이언트들의 버퍼 수준이 초기값 1.0 이하이므로 버퍼 수준에 여유가 없음을 의미하게 된다. 이 경우 서버는 각 클라이언트  $i$ 마다 실제 대역폭  $B_i$ 에서 전송이 보류된 대역폭  $P_i$ (10행과 13행에서 설정됨)를 차감한 대역폭의 합  $B_{sum}$ '을 구하고(19행)  $B_{sum}$ '에서  $B_{avail}$ 을 차감한 대역폭을  $B_{lack}$ '으로 정의한다(20행).  $B_{lack}$ '은  $B_{sum}$ '을 전송하기에 여전히 부족한 대역폭을 의미한다. 따라서 서버는 각 클라이언트마다 전송을 보류해야 하는 대역폭을 한번 더 산정하게 된다. 스트림  $i$ 마다  $B_i'$  대역폭을  $B_{sum}'$ 으로 나눈 비율과  $B_{lack}'$ 을 곱하여 전송을 보류해야 하는 대역폭  $P_i'$ 가 결정된다(24행). 각 스트림  $i$ 의 실제 대역폭  $B_i$ 에서 전송이 1차로 보류된  $P_i$ 와 2차로 보류된  $P_i'$ 를 모두 차감한 대역폭은  $D_i$ 로 설정되며(24행) 이는 실제로 전송이 중단되는 대역폭을 의미한다. 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[14] 이후  $P$  프레임,  $I$  프레임 순으로 결정됨으로써 중요도가 큰 프레임일수록 전송이 중단되는 경우가 최소화되도록 하였다. 그리고 26행에서 서버는 새로운  $L_{max}$ 를 찾는 과정을 중지하고자  $B_{lack}$ 을 0으로 설정하여 4행의 반복문을 벗어나게 된다.

실제 대역폭의 합  $B_{sum}$ 이  $B_{avail}$ 보다 작거나 같을 경우  $B_{avail}$ 에서  $B_{sum}$ 을 차감한 대역폭은  $B_{suff}$ 로 정의되며((그림 3)의 31행) 이는  $B_{sum}$ 을 전송하고 남는 대역폭을 의미한다. 서버는  $B_{suff}$ 가 0보다 클 경우 클라이언트 버퍼 수준에서 최소값  $L_{min}$ 을 찾는다(34행). 즉 서버는 버퍼 수준이 최소인 클라이언트를 찾고 남는 대역폭을 이 클라이언트에 추가로 배정하여 다음 주기에 전송할 프레임들을 이번 주기에 미리 전송할 수 있도록 하는 것이다. 버퍼 수준이  $L_{min}$ 인 클라이언트가 다음 주기에 전송할 실제 대역폭을  $B_{i,next}$ 로 정의하며 이  $B_{i,next}$ 가  $B_{suff}$  이하일 경우(35행)  $B_{i,next}$ 는 모두 전송될 수 있으므로 해당 클라이언트의 버퍼 수준은 1.0이 증가되고  $B_{suff}$ 는  $B_{i,next}$ 만큼 감소된다(36행). 35행과 달리  $B_{i,next}$ 가  $B_{suff}$ 보다 클 경우(37행)  $B_{suff}$ 만큼만 이번 주기에 전송되며

$B_{i,next}$ 에서  $B_{suff}$ 을 차감한 대역폭은 원래 주기인 다음 주기에 전송될 수 있다. 따라서 해당 클라이언트의 버퍼 수준은  $B_{suff}/B_{i,next}$  비율만큼 증가되며  $B_{suff}$ 는 0이 되어(38행) 그림 3의 32행에 있는 반복 과정은 종료된다.

#### 4. 성능 분석 및 평가

##### 4.1 버퍼 수준과 프레임 전송중단율

본 논문에서는 압축률이 우수한 VBR(Variable Bit-Rate) 방식으로 부호화된 MPEG-4 비디오 스트림 “Jurassic Park One”, “Silence of the Lambs”, “Star Wars IV” 등[15]을 대상으로 실험을 수행하였다. 이 비디오 스트림들은 모두 QCIF 형식이며 GOP내 프레임 수는 12이고 GOP 패턴은 IBBPBBPBBPBB이다. 세 비디오 스트림들의 총 프레임 수는 각각 108000개이며 프레임 크기를 분석한 결과는 <표 1>과 같다.

세 비디오 스트림들에 대하여 프레임의 최대값을 평균값으로 나눈 비율은 각각 10.49, 13.36, 17.29로 분석되었다.  $i(1 \leq i \leq n)$ 번째 클라이언트가 요청하는 각 비디오 스트림의 재생율을 24로 동일하게 적용할 경우 전송에 필요한 평균 대역폭과  $n(=20)$  개의 모바일 클라이언트에게 전송할 경우 평균 대역폭의 합  $M_{juristicsum}$ ,  $M_{silencesum}$ ,  $M_{starwarssum}$ 은 다음과 같이 설정된다.

$$\begin{aligned} \text{“Jurassic Park One”의 평균 대역폭과 평균 대역폭의 합} \\ M_{juristic_i} &= 276.875\text{바이트/프레임} \times 24\text{프레임/초} \times 8\text{비트/바이트} \\ &\approx 53160\text{비트/초} \end{aligned}$$

$$M_{juristicsum} = \sum_{i=1}^{20} M_{juristic_i} = 1,063,200\text{비트/초}$$

$$\begin{aligned} \text{“Silence of the Lambs”의 평균 대역폭과 평균 대역폭의 합} \\ M_{silence_i} &= 214.0\text{바이트/프레임} \times 24\text{프레임/초} \times 8\text{비트/바이트} \\ &\approx 41088\text{비트/초} \end{aligned}$$

$$M_{silencesum} = \sum_{i=1}^{20} M_{silence_i} = 821,760\text{비트/초}$$

$$\text{“Star Wars IV”의 평균 대역폭과 평균 대역폭의 합}$$

```

29: else // Bsum ≤ Bavail
30: begin
31:   Bsuff=Bavail-Bsum;
32:   while (Bsuff > 0)
33:     begin
34:       Li가 가장 작은 값을 Lmin로 설정함;
35:       Lmin에 해당하는 Bi,next가 Bsuff보다 작거나 같으면
36:         Lmin을 1.0 증가시키고 Bsuff에서 Bi,next를 차감함;
37:       Lmin에 해당하는 Bi,next가 Bsuff보다 크면
38:         Lmin에 Bsuff/Bi,next 값을 증가시키고 Bsuff를 0으로
           설정함;
39:     end
40: end
    
```

(그림 3)  $B_{sum} \leq B_{avail}$  경우의 동적 대역폭 할당 알고리즘

<표 1> 프레임 분석 결과

| 프레임 크기     | Jurassic Park One | Silence of the Lambs | Star Wars IV |
|------------|-------------------|----------------------|--------------|
| 최소 프레임 크기  | 8바이트              | 9 바이트                | 10바이트        |
| 최대 프레임 크기  | 2903바이트           | 2858바이트              | 4279바이트      |
| 평균 프레임 크기  | 276.875바이트        | 214.0바이트             | 247.5바이트     |
| 최대값/평균값 비율 | 10.49             | 13.36                | 17.29        |
| 평균 대역폭     | 53160비트/초         | 41088비트/초            | 47520비트/초    |
| 평균 대역폭의 합  | 1063200비트/초       | 821760비트/초           | 950400비트/초   |

$$M_{starwars_i} = 247.5 \text{바이트/프레임} \times 24 \text{프레임/초} \times 8 \text{비트/바이트}$$

$$\approx 47520 \text{비트/초}$$

$$M_{starwars_{sum}} = \sum_{i=1}^{20} M_{starwars_i} = 950,400 \text{비트/초}$$

모바일 통신망의 최대 대역폭  $B_{total}$ 을 2Mbps로 가정하며 비디오 스트림 전송에 사용되는 평균 대역폭의 합( $M_{jurassic_{sum}}$ ,  $M_{silence_{sum}}$ ,  $M_{starwars_{sum}}$ )과 비디오 스트림 전송시 서버가 사용할 수 있는 최대 대역폭( $B_{jurassic_{avail}}$ ,  $B_{silence_{avail}}$ ,  $B_{starwars_{avail}}$ )은 각각 동일한 것으로 가정할 때 수식 (1)에서 기술된 다음 관계가 성립된다.

$$M_{jurassic_{sum}} \leq B_{jurassic_{avail}} \leq B_{total}(=2Mbps)$$

$$M_{silence_{sum}} \leq B_{silence_{avail}} \leq B_{total}(=2Mbps)$$

$$M_{starwars_{sum}} \leq B_{starwars_{avail}} \leq B_{total}(=2Mbps)$$

본 논문에서 모바일 통신망의 통신 장애로 인한 프레임 재전송에 필요한 대역폭은  $B_{jurassic_{avail}}$ ,  $B_{silence_{avail}}$ ,  $B_{starwars_{avail}}$ 에 각각 포함되지 않는 것을 가정하였다.

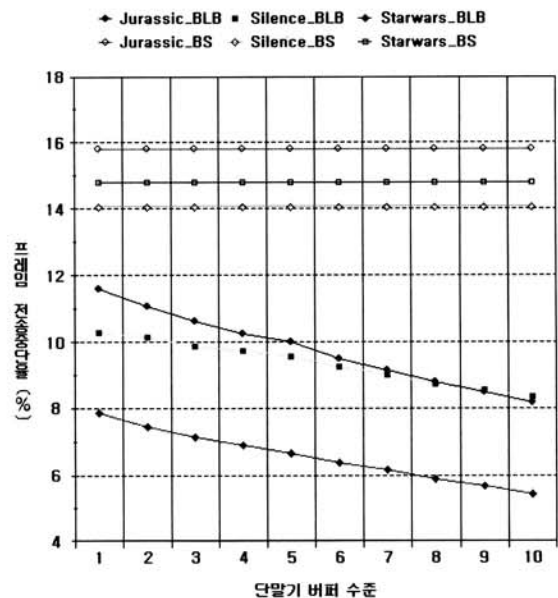
단말기 버퍼 수준을 사전에 파악하여 비디오 스트림별로 통신망 대역폭을 정적으로 할당하는 BS(Buffer Sensitive) 대역폭 할당 기법[6]과 제시된 BLB(Buffer Level-based) 대역폭 할당 기법의 성능이 비교 분석되었다. BS 대역폭 할당 기법은 이용가능한 통신망 대역폭을 스트림의 수로 나눈 후 이 값을 각 스트림의 평균 대역폭과 비교하여 더 작은 값을 해당 스트림의 실제 대역폭으로 결정한다. 이후 모든 스트림의 실제 대역폭의 합이 이용가능한 통신망 대역폭보다 작을 경우에만 여분의 통신망 대역폭이 발생하며 이 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 BS 대역폭 할당 기법은 스트림별로 통신망 대역폭을 정적으로 할당한 후 전송 중 발생하는 단말기 버퍼 수준을 별도로 고려하지 않기 때문에 이러한 사항을 고려하여 동적으로 대역폭을 조절하는 BLB 대역폭 할당 기법과 성능을 직접 비교하기는 사실상 어렵다. 본 실험에서는  $M_{jurassic_{sum}}$ ,  $M_{silence_{sum}}$ ,  $M_{starwars_{sum}}$ 이  $B_{jurassic_{avail}}$ ,  $B_{silence_{avail}}$ ,  $B_{starwars_{avail}}$ 과 각각 동일하기 때문에 BS 기법과 BLB 기법의 수락 제어 단계에서 통신망의 여유 대역폭은 전혀 없는 것으로 가정한다. 두 기법은 동일한 실험 환경에서 시뮬레이션이 4200초간 수행되었으며 이 결과를 토대로 두 기법의 성능이 비교 분석되었다.

서버는 동일한 비디오 스트림을 20개의 클라이언트 단말기들에 대하여 1초부터 120초 사이의 임의의 시간에 전송을 시작한다. 본 논문에서는 두 기법의 성능을 비교하기 위하여 프레임 전송중단율이 측정된다. 즉 서버가 각 클라이언트 단말기로 전송할 때 대역폭 부족으로 전송을 중단시켜야 하는 프레임의 수를 서버가 각 클라이언트 단말기로 전송해

야 하는 프레임의 수로 나눈 값이 프레임 전송중단율로 정의된다. 이 비율이 클수록 클라이언트들이 재생하지 못하는 프레임들의 수가 증가하는 것이다. 모든 단말기 버퍼 수준의 초기치를 1.0부터 10.0까지 1.0씩 차례로 증가시킨 상태에서 두 기법의 프레임 전송중단율을 각각 비교한 결과가 (그림 4)에 나타나 있다. “Jurassic Park One”의 경우 BS 기법은 여분의 통신망 대역폭이 없으므로 버퍼 수준의 크기에 상관없이 프레임 전송중단율이 동일하며 약 15.80%로 분석되었다. 그러나 BLB 기법의 경우에는 버퍼 수준이 1.0인 경우 프레임 전송중단율이 약 11.58%로 나타났으며 버퍼 수준의 초기치를 1.0씩 증가시킬 때마다 약 0.2%에서 0.4% 정도로 프레임 전송중단율이 감소되었으며 버퍼 수준이 10.0인 경우에는 프레임 전송중단율이 약 8.37%로 분석되었다.

여분의 통신망 대역폭이 없는 상태에서 버퍼 수준에 따라 통신망 대역폭을 동적으로 조정하여 스트림의 프레임들을 전송하는 BLB 기법의 경우 대역폭을 정적으로 할당하는 BS 기법보다 프레임 전송중단율이 감소되는 결과가 나타났으며 버퍼 수준의 초기치를 증가시킬수록 근소한 차이로 프레임 전송중단율이 감소되는 결과도 나타났다. “Silence of the Lambs”와 “Star Wars IV” 경우 BS 기법의 프레임 전송중단율은 각각 14.06%와 14.81%로 나타났으며 BLB 기법의 프레임 전송중단율은 버퍼 수준에 따라 10.26%에서 8.34%로 감소된 결과와 7.86%에서 5.43%로 감소된 결과가 각각 나타났다. 프레임의 최대값을 평균값으로 나눈 비율은 “Star Wars IV”가 17.29로 가장 크지만 프레임 전송중단율이 가장 낮게 나타났으며 이는 평균대역폭을 초과하는 빈도수가 다른 비디오 스트림들에 비하여 상대적으로 낮게 발생한 것이 원인으로 분석되었다.

BLB 기법에서는 전송을 중단시켜야 하는 프레임들에 대해서 우선순위 정책을 사용하여 B 프레임, P 프레임, I 프레



(그림 4) BLB 기법과 BS 기법의 프레임 전송중단율

임 순서로 중단된 비율이 감소하는 결과가 측정되었다. BS 기법의 경우는 원래 우선순위 정책이 적용되지 않았지만 본 실험에서 우선순위 정책을 동일하게 적용하여 프레임별 전송중단율을 측정하였다. 버퍼 수준 1.0에서 "Jurassic Park One", "Silence of the Lambs", "Star Wars IV" 비디오 스트림들에 대한 BS 기법의 프레임 전송중단율은 각각 15.80%, 14.06%, 14.81%이었으며 BLB 기법의 프레임 전송중단율은 각각 11.58%, 10.26%, 7.86%로 나타났다. 이때 전송중단된 프레임들을 분석하여 프레임별로 전송중단율이 비교된 결과가 표 2에서 기술되었다. BS 기법의 경우 프레임 중단시 우선순위 정책이 적용되지 않는다면 중요도가 가장 높은 I 프레임이 중단될 수 있고 이로 인하여 동일한 클라이언트 단말기에서 재생될 수 없는 GOP내 P 프레임과 B 프레임들이 모두 전송 중단되기 때문에 전체 프레임 전송중단율은 실제로 더 증가될 수 있다.

〈표 2〉 프레임별 전송중단율(%)

| 전송중단율        | I 프레임 | P 프레임 | B 프레임 |
|--------------|-------|-------|-------|
| Jurassic_BS  | 0.0   | 2.21  | 97.79 |
| Jurassic_BLB | 0.01  | 0.91  | 99.08 |
| Silence_BS   | 0.01  | 2.80  | 97.19 |
| Silence_BLB  | 0.0   | 0.90  | 99.10 |
| Starwars_BS  | 0.0   | 2.12  | 97.88 |
| Starwars_BLB | 0.0   | 0.03  | 99.97 |

4.2 공평성과 프레임 전송 중단율

각 비디오 스트림을 요청한 20개의 클라이언트 단말기들 중 동일한 버퍼 수준을 갖는 클라이언트들을 2개씩 설정하였으며 버퍼 수준의 초기치는 1.0에서 10.0까지 1.0씩 증가시키며 설정되었다. 즉 버퍼 수준의 초기치가 1.0인 클라이언트도 2개, 2.0인 클라이언트도 2개이며 초기치가 10.0인 클라이언트도 마찬가지로 2개임을 의미한다. 이에 대한 실험 결과가 <표 3>에 나타났다. 버퍼 수준의 초기치가 상이한 클라이언트들에 대하여 측정된 전체 프레임 전송중단율은 "Jurassic Park One"이 9.59%이며 "Silence of the Lambs"가 9.28%, "Star Wars IV"가 6.40%로 각각 나타났다. 각 비디오 스트림에 대하여 버퍼 수준별로 측정된 프레임 전송중단율은 거의 동일하게 나타나고 있다. 즉 버퍼 수준의 초기치가 모두 동일하게 설정된 상태에서는 버퍼 수준이 클수록 근소한 차이로 프레임 전송중단율이 감소되지만 버퍼 수준의 초기치가 다양하게 설정된 상태에서는 버퍼 수준에 따른 프레임 전송중단율의 차이가 거의 없으며 이는 버퍼 수준이 상이한 클라이언트들에 대한 공평성이 유지됨을 의미한다.

〈표 3〉 버퍼 수준별 프레임 전송중단율(%)

| 버퍼 수준        | 1.0  | 2.0  | 3.0  | 4.0  | 5.0  | 6.0  | 7.0  | 8.0  | 9.0  | 10.0 |
|--------------|------|------|------|------|------|------|------|------|------|------|
| Jurassic_BLB | 9.77 | 9.65 | 9.62 | 9.60 | 9.56 | 9.72 | 9.67 | 9.51 | 9.53 | 9.54 |
| Silence_BLB  | 9.13 | 9.15 | 9.20 | 9.21 | 9.32 | 9.37 | 9.40 | 9.37 | 9.36 | 9.30 |
| Starwars_BLB | 6.36 | 6.37 | 6.41 | 6.40 | 6.42 | 6.42 | 6.42 | 6.43 | 6.41 | 6.40 |

BLB 기법의 경우 버퍼 수준에 따라 대역폭을 조정하여 프레임 전송중단율을 감소시키는 물론 다양한 버퍼 수준을 갖는 클라이언트들에 대하여 공평성을 유지할 수 있다.

5. 결 론

분산 모바일 멀티미디어 시스템에서 클라이언트가 요청한 비디오 스트림을 서버가 전송할 때 단말기 버퍼 수준을 활용하여 모바일 통신망 대역폭을 동적으로 할당하는 기법이 본 논문에서 제시되었다. 제시된 BLB 대역폭 할당 기법은 단말기 버퍼 수준을 고려하여 제한된 통신망 대역폭을 효율적으로 사용함으로써 서버가 대역폭 부족으로 인하여 전송을 중단시켜야 하는 프레임 수를 최소화한다. 프레임의 평균 크기와 재생율에 따라 평균 대역폭이 결정되는 각 비디오 스트림은 VBR 특성상 실제 대역폭의 크기가 가변적이다. 따라서 전송 중인 스트림의 대역폭 합은 이용가능한 통신망 대역폭을 초과하는 기회가 자주 발생하게 된다. 이때 서버는 전송이 중단되는 프레임 수를 최소화할 수 있도록 버퍼 수준이 큰 클라이언트의 프레임들은 다음 주기에 전송할 수 있도록 보류하고 보류로 인해 확보된 대역폭은 버퍼 수준이 작은 클라이언트가 활용할 수 있도록 동적으로 조정하게 된다.

그리고 전송 중인 스트림의 대역폭 합이 이용가능한 통신망 대역폭보다 작을 경우에는 모든 클라이언트들의 프레임을 전송하고 남은 대역폭은 버퍼 수준이 작은 클라이언트들에게 추가로 할당하여 더 많은 프레임들이 전송되도록 함으로써 버퍼 수준별 프레임 전송중단율의 차이를 최소화할 수 있는 공평성이 유지된다. 본 논문의 BLB 기법은 성능 비교 결과 비디오 스트림의 서비스 품질과 끊임없는 비디오 스트리밍 효과를 향상시킬 수 있는 것으로 분석되었다. 따라서 제시된 BLB 대역폭 할당 기법은 통신망 대역폭이 한정된 모바일 환경의 분산 멀티미디어 시스템에서 실시간적 성능과 서비스 품질을 더욱 향상시킬 수 있는 것으로 기대된다.

참 고 문 헌

[1] M. Ravichandran, P. Sengottuvelan, and A. Shanmugam, "An Approach for Admission Control and Bandwidth Allocation in Mobile Multimedia Network Using Fuzzy Logic," *Int'l Journal of Recent Trends in Engineering*, Vol.1, No.1, May 2009.

[2] 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*, Vol.3, No.1, pp.60-73, 2004.
- [3] K. Y. Lam<sup>1</sup>, J. Yuen<sup>1</sup>, S. H. Son, and E. Chan, “Scheduling Video Stream Transmissions for Distributed Playback over Mobile Cellular Networks,” *Int’l Conf. on Parallel and Distributed Systems*, Dec. 2002.
- [4] 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-Aug. 2000.
- [5] 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.
- [6] 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*, Lausanne, Switzerland, August 2002
- [7] M. Burza, J. Kang, and P. van der Stok, “Adaptive Streaming of MPEG-based Audio/Video Content over Wireless Networks,” *Journal of Multimedia*, Vol.2, No.2, pp.17-27, 2007.
- [8] X. Hei, Y. Liu, and K. W. Ross, “Inferring Network-Wide Quality in P2P Live Streaming Systems,” *IEEE Journal on Selected Areas in Communications Volume 25, Issue 9*, pp. 1640-1654, Dec. 2007.
- [9] H. Espeland, and et al, “Transparent Protocol Translation and Load Balancing on a Network Processor in a Media Streaming Scenario,” *Network and Operating Systems Support for Digital Audio and Video*, May, 2008.
- [10] S. Mohapatra and N. Venkatasubramanian, “Proactive energy-aware video streaming to mobile handheld devices,” *IEEE Conference on Mobile and Wireless Communications Networks*, 2003.
- [11] 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.
- [12] 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.
- [13] M. Krunz, “Bandwidth Allocation Strategies for Transporting Variable-Bit-Rate Video Traffic,” *IEEE Communications Magazine*, Jan. 1999.
- [14] 신동규, 신동일, 박세영, “대역폭 감소를 적용한 MPEG-4 미디어 전송시의 압축화 기법 연구”, 정보처리학회 논문지B, 제15권 제6호, pp. 575-584, 2008.
- [15] <http://trace.eas.asu.edu/cgi-bin/main.cgi>

### 김진환



e-mail : kimjh@hansung.ac.kr

1986년 서울대학교 컴퓨터공학과(학사)

1988년 서울대학교 컴퓨터공학과(석사)

1994년 서울대학교 컴퓨터공학과(박사)

1994년~1996년 서울대학교 컴퓨터신기술

공동연구소 특별연구원

1995년~현 재 한성대학교 멀티미디어공학과 교수

관심분야: 멀티미디어 시스템, 분산실시간 시스템