

# 주문형 비디오에서 $m$ 배속 고속 재생을 위한 GoP 기반 동적 전송 스케줄 작성

정회원 이 재 홍\*

## A GoP-based Dynamic Transmission Scheduling for supporting Fast Scan Functions with $m$ -times playback rate in Video-On-Demand

Jae-Hong Lee\* *Regular Member*

### 요 약

주문형 비디오 서비스는 궁극적으로 사용자에게 VCR 기능과 같은 대화형 연산을 지원하여야 할 것으로 기대된다. 특히, VCR 기능 중에서 고속 전진 및 고속 후진 재생의 지원이 중요하다. 그러나 고속 재생을 지원하기 위해서는 매우 큰 시스템 자원을 필요로 하므로 이를 줄이기 위한 방안이 필요하다. MPEG 표준에서 GoP(Group-of-Pictures)는 독립적인 복원이 가능한 임의 접근 단위이며 비디오 스트림은 GoP 단위로 저장 및 전송되는 것이 효율적이므로 GoP 단위로 건너 뛰면서  $m$ 배속 고속 재생을 제공하는 것이 실용적이다. 본 논문에서는 저장된 비디오 스트림에 대하여  $m$ 배속 고속 재생 기능을 지원하기 위한 동적 전송 스케줄 작성 방안을 제안한다. 제안된 방안은 사용자가 고속 재생을 요청할 때마다 매  $m$ 번째 GoP들로 평활화 대상 GoP 집합을 구성하여 대역폭 평활화를 적용하여 대역폭 할당 스케줄을 구한 후에 새로운 전송 스케줄에 따라 클라이언트로의 데이터 전송을 재개한다. 본 논문에서는 고속 재생을 지원하기 위한 방안으로 "GoP 건너 뛰기"를 활용함으로써 재생 속도를  $m$ 배 빠르게 하는 효과를 가져 왔으며, GoP 단위로 대역폭 평활화를 적용함으로써 전송 스케줄을 구하기 위한 계산량을 크게 감소시킬 수 있었다.

### ABSTRACT

Video-On-Demand (VOD) is expected to provide the user with interactive operations such as VCR functions. In particular, fast scan functions like "Fast Forward" or "Fast Backward" for a certain speedup playback are required. Since they require a significant amount of system resources, schemes to reduce bandwidth requirements for the network or disk are needed. In MPEG standard, Group-of-Pictures (GoP) is a random access unit which can be decoded independently. Since storing and transmitting a video stream based on GoP is efficient, it is practical to support fast scan functions based on GoP. In this paper, we present a dynamic transmission scheduling scheme to support fast scan functions with  $m$ -times normal playback rate for a stored video. The proposed scheme writes a transmission schedule whenever user requests a fast scan function. That is, the scheme constructs the data set to be smoothed by skipping GoPs according to a given speedup factor, and then writes the transmission schedule by applying a bandwidth smoothing. Finally, the scheme restarts the transmission of video data to a client according to the new schedule. The proposed scheme results in speeding up the playback rate by utilizing "GoP skipping", and then reduces the computational overhead by applying a bandwidth smoothing based on GoP.

\* 충남대학교 컴퓨터공학과(jhlee@ece.chch.c.ac.kr)  
논문번호 : 99056-0218, 접수일자 : 1999년 2월 18일

## I. 서론

ATM 기반의 B-ISDN과 같은 고속 통신망의 출현과 더불어 MPEG과 같은 데이터 압축 기술의 표준화로 방대한 데이터 양을 처리해야 하는 비디오 관련 멀티미디어 응용의 실용화가 가능해졌다. 특히, 최근에 멀티미디어 응용 분야 중 주문형 비디오 (Video-On-Demand: VOD)에 대한 연구가 매우 활발하다<sup>[1,2,3]</sup>. VOD 시스템은 서버, 클라이언트 그리고 이들을 연결하는 망으로 구성된다. VOD 서버는 대용량 저장 장치에 압축된 비디오와 오디오 데이터를 유지하며 클라이언트의 요청에 따라 ATM망이나 CATV망을 통해 클라이언트로 이를 전송한다. VOD 클라이언트는 MPEG 복호화기를 갖춘 워크스테이션이나 Set-Top-Box로서 서버로부터 데이터를 수신할 수 있는 고정된 크기의 버퍼를 가진다.

비디오 데이터는 오디오 데이터에 비하여 그 양이 매우 많기 때문에 이를 망으로 직접 전송하는 것은 지나친 시스템 자원을 요구함으로써 적절하지 못하다. 따라서 저장 및 전송의 효율성과 복원할 경우 일정한 품질을 유지하기 위하여 비디오 데이터는 가변 비트율 (Variable Bit Rate : VBR) 모드의 MPEG 표준으로 부호화된다<sup>[4]</sup>. 그러나 압축된 VBR MPEG 비디오 데이터의 트래픽 특성은 프레임 사이의 부호화 방법의 차이로 인하여 매우 폭주적인 트래픽 특성을 보인다<sup>[5]</sup>. 그래서 이러한 트래픽을 그대로 망으로 전송하는 경우에 지나치게 큰 대역폭을 요구하게 되어 전송 비용을 높게 된다. 따라서 망 자원을 효율적으로 활용하기 위하여 전송 전에 폭주적인 트래픽을 평활화할 필요가 있다. 실시간 비디오와는 달리 저장된 비디오에서는 비디오 스트림의 통계적 특성을 전송 전에 알 수 있기 때문에 클라이언트에서 재생을 시작하기 전에 서버로부터 충분한 데이터를 선인출(prefetch)하여 전송하는 것이 가능하므로 최소한의 망 대역폭으로 비디오 스트림을 전송할 수 있다<sup>[6,7,8,9]</sup>.

VCR 기능에는 재생(Play), 정지(Stop), 일시 정지(Pause), 다시 시작(Resume), 그리고 영상과 음성이 없이 비디오 스트림 내 임의의 프레임 위치로 재생 위치를 변경하는 전진 이동(Jump Forward) 및 되감기(Jump Backward 또는 Rewind) 그리고 영상을 보여 주면서 고속 재생이 가능한 고속 전진 재생(Fast Forward) 및 고속 후진 재생(Fast Backward) 등이 있다<sup>[10]</sup>. VCR 기능은 VOD 서비스가 현재의

비디오 대역 서비스를 대체하기 위하여 반드시 지원하여야 할 매우 중요한 항목이 될 것이다. 망 대역폭 요구를 최소화하면서 VCR 기능을 지원하기 위하여 VOD 서버는 사용자가 VCR 기능을 요청한 프레임 위치로부터 비디오 데이터를 전송하기 위한 전송 스케줄을 작성하여야 한다. 특히, 고속 재생은 현재 재생중인 비디오의 재생 속도를 증가시킬 필요가 있기 때문에 VCR 기능들 중에서 가장 중요하게 고려되어야 하며, 이를 구현하는 데 있어서의 어려운 점은 대역폭 요구의 변화 변위가 크다는 점이다.

VCR 기능의 구현과 관련하여 많은 연구가 있었으나 임의 이동이나 정지/다시 시작만 지원하는 경우가 대부분이었다<sup>[11,12]</sup>. 이에 반해, Feng은 재생은 되었으나 클라이언트의 버퍼 내에 남아 있는 데이터를 대상으로 고속 후진 재생을 제공하였다<sup>[13]</sup>. 한편, 정상 재생이 가능한 비디오 스트림으로부터 "프레임 건너뛰기(Frame Skipping)"를 이용하여 얻은 프레임들을 새롭게 부호화함으로써 고속 재생용 복수의 스트림을 준비한 후에 사용자의 요청에 따라 정상 스트림과 고속 재생용 스트림 사이의 스트림 전환을 이용하여 고속 전진 재생을 제공하는 방안이 있다<sup>[14]</sup>.

$m$ 배속 고속 재생을 지원하기 위한 몇 가지 방안을 고려해 볼 수 있다.

### 1) $m$ 간격의 프레임 전송하는 방안

저장 시스템에서  $m$  간격의 프레임을 디스크로부터 읽어 전송하는 방안이다. 이 방안은 프레임간 예측 부호화를 이용하는 MPEG 데이터에는 적용하기 어렵고 각 프레임의 색인을 유지하여야 함으로 이로 인한 오버헤드가 따른다.

### 2) $m$ 간격의 I 프레임만을 전송하는 방안

독립적으로 복원이 가능한 I 프레임만을 전송하는 방안을 고려해 볼 수 있으나, I 프레임은 다른 프레임 유형에 비하여 데이터 양이 매우 크므로 높은 디스크 전송률 및 큰 망 대역폭을 요구한다.

### 3) 별도의 고속 재생용 스트림을 준비하는 방안

정상 재생용 스트림으로부터 고속 재생용 스트림을 별도로 준비한 후에 사용자 요청에 따라 정상 재생용 스트림과 고속 재생용 스트림사이를 전환하는 방안이 제안되었다<sup>[14]</sup>. 그러나 이 방안은 임의의 재생 속도에 따른 스트림을 준비할 수 없으며 복수의 고속 재생용 스트림들을 저장하는 데 있어서의 오버헤드와 스트림 전환시 동기화 문제가 발생한다.

MPEG 표준에서는 프레임 단위로 임의 접근은 어려우나 몇 개의 화면 데이터를 모아 구성한 GoP는 독립적인 복원이 가능하기 때문에, GoP 단위로 VCR 기능을 위한 임의 접근이 가능하다. 이를 위해 GoP 앞에 시퀀스 헤더를 삽입하고 있다<sup>[4]</sup>. 또한, 비디오 스트림은 GoP 단위로 디스크에 저장되고 망으로 전송되는 것이 효율적이므로<sup>[3,15]</sup> 고속 재생을 지원하기 위해서는 GoP 단위로 전송 스케줄을 작성하는 것이 실용적이다.

본 논문에서는 m배속 고속 전진 및 후진 재생을 지원하기 위한 GoP 기반의 동적 전송 스케줄 작성 방안을 제안한다. 사용자가 고속 재생을 요청할 때마다 서버는 동적으로 고속 재생을 지원하기 위한 전송 스케줄을 작성하고 이 스케줄에 따라 비디오 데이터를 클라이언트로 전송한다.

본 논문의 구성은 다음과 같다. II장에서는 GoP를 기반으로 VCR 기능을 지원하기 위한 방안을 기술한다. III장에서는 VOD 서버의 전송 스케줄과 대역폭 평활화를 정의하고 고속 재생을 지원하기 위한 GoP 기반 동적 스케줄링 알고리즘을 제안한다. IV장에서는 제안된 방안의 성능을 분석하기 위하여 MPGE-1으로 부호화된 비트열 데이터를 대상으로 실험을 수행하고 V장에서 결론을 맺는다.

## II. VCR 기능의 지원 방안

이 장에서는 GoP 단위로 고속 재생 및 검색과 같은 VCR 기능을 지원하기 위한 방안을 기술한다.

그림 1은 여러 가지 VCR 기능을 지원하기 위하여 전송을 위해 선택된 GoP들을 보여 준다. 그림 1에서 빗금 친 블록은 선택된 GoP를 나타낸다.

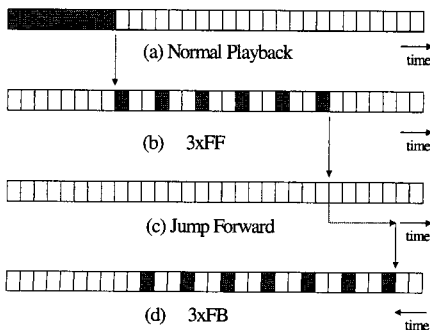


그림 1. VCR 기능을 지원하기 위하여 선택된 GoP들

그림 2는 그림 1의 VCR 기능들을 지원하기 위한 서버와 클라이언트 사이의 제어 절차를 보여 준다.

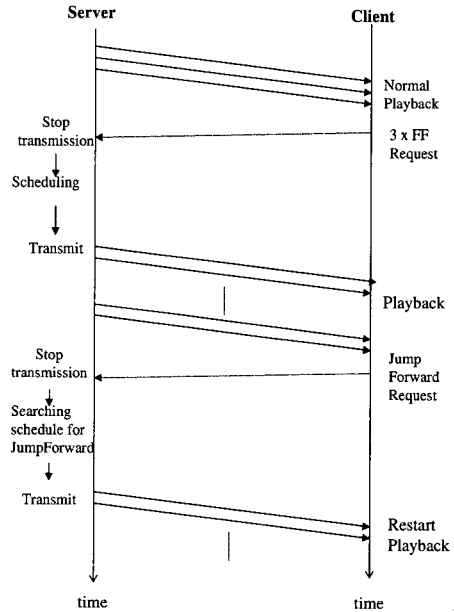


그림 2. VCR 기능을 지원하기 위한 제어 절차의 예

그림 1-(a)는 정상 재생시 서버가 클라이언트로 전송하여야 할 GoP들을 보여 준다. 클라이언트가 그림 1-(b)에서와 같이 3배속 고속 전진 재생을 요청하는 경우에 서버는 클라이언트의 비디오 데이터의 전송을 즉시 중단하고 새로운 전송 스케줄을 작성한다. 서버는 그림 1-(b)에서와 같이 클라이언트가 고속 전진 재생을 요청한 프레임 위치가 속한 GoP로부터 비디오 스트림의 마지막 GoP까지 매 3번째 GoP를 선택하여 평활화 대상 데이터 집합을 구성하고 대역폭 평활화를 적용하여 대역폭 할당 계획, 즉 전송 스케줄을 작성한다. 그리고 서버는 이 스케줄에 따라 비디오 데이터의 전송을 다시 시작한다. 서버가 전송 스케줄을 작성하는 동안, 클라이언트는 사용자가 고속 재생을 요청한 시점에 재생되었던 프레임을 계속 보여 주고 있다가 서버로부터 재생을 다시 시작하라는 제어 신호를 받으면 버퍼 내 데이터를 지우고 새로운 전송 스케줄에 따라 전송된 비디오 데이터의 재생을 시작한다. 이와 같이 본 방안에서는 “GoP 건너 뛰기(GoP Skipping)”를 이용함으로써 클라이언트는 재생 속도를 변경할 필요 없이 정상 재생과 동일한 방법으로 고

속 재생을 지원할 수 있다.

VCR 기능들 중 다른 예로, 사용자는 그림 1-(c)와 같이 전진 이동을 요청할 수 있다. 이 기능은 재생 위치를 비디오 스트림 내 임의의 프레임 위치로의 이동시켜 주며 영상을 보여 주지는 않는다. 한편, 사용자가 그림 1-(d)와 같이 3배속 고속 후진 재생을 요청할 때, 서버는 재생 중인 프레임으로부터 역 방향으로 건너 뛰면서 매 3번째 GoP들로 평활화 대상 데이터 집합을 구성하여 대역폭 평활화를 적용하여 동적으로 전송 스케줄을 작성한다. 일반적으로 정상 재생과는 달리 고속 재생은 영상을 자세히 보고자 하는 것이 아니라 특정 프레임을 검색하기 위하여 사용되므로 클라이언트에서의 복호 과정을 복잡하게 하면서까지 프레임들을 역 방향으로 재생할 필요는 없다. 따라서, 본 논문에서는 고속 후진 재생시 클라이언트는 전송된 GoP내의 프레임들을 역 방향이 아니라 순 방향으로 재생을 함으로써 클라이언트의 구조를 간단하게 할 수 있었다. 한편, GoP가 독립적으로 복원이 가능한 임의 접근 단위이지만, 그림 3에서와 같이 GoP내 마지막 두 B 프레임, 즉 B<sub>11</sub>과 B<sub>12</sub>는 두 GoP에 걸쳐 있으므로 GoP 단위로 고속 재생을 지원하는 경우에는 이들 B 프레임의 복원에 다음 GoP의 I<sub>13</sub> 프레임 정보가 필요하다. 따라서, 고속 재생시 GoP 단위로 복원할 때 약간의 화질 저하가 발생한다. 이를 막기 위해 GoP 내의 프레임들이 다른 GoP로부터 독립적인 복원이 가능하도록 부호화되어야 한다.

I<sub>1</sub> B<sub>2</sub> B<sub>3</sub> P<sub>4</sub> B<sub>5</sub> B<sub>6</sub> P<sub>7</sub> B<sub>8</sub> B<sub>9</sub> P<sub>10</sub> B<sub>11</sub> B<sub>12</sub> I<sub>13</sub> .....

(부호화 순서)

I<sub>1</sub> P<sub>4</sub> B<sub>2</sub> B<sub>3</sub> P<sub>7</sub> B<sub>5</sub> B<sub>6</sub> P<sub>10</sub> B<sub>8</sub> B<sub>9</sub> I<sub>13</sub> B<sub>11</sub> B<sub>12</sub> .....

(전송 순서)

그림 3. 부호화 및 전송 순서

### III. 고속 재생을 위한 동적 스케줄 알고리즘

이 장에서는  $m$ 배속 고속 재생을 위한 GoP 기반 동적 스케줄 작성 방안을 제안한다. 제안된 방안은 사용자가 고속 재생을 요청한 프레임 위치로부터 비디오 스트림의 전송 스케줄을 구하기 위하여 GoP 단위로 대역폭 평활화를 적용한다.

본 논문에서는 GoP 내 프레임들의 누적된 데이터 양의 크기를 이용할 수 있다고 가정한다.

#### 1. 전송 스케줄과 대역폭 평활화

전송 스케줄  $TS$ 는 식 (1)과 같이 세그먼트들에 대한 정보로 이루어 진다. 세그먼트는 동일한 대역폭이 할당되는 GoP들로 구성된다.  $i$ 번째 세그먼트  $S_i$ 에 대한 정보는 시작 GoP 번호  $F_i$ , 끝 GoP 번호  $L_i$ 와 이 세그먼트에 할당될 대역폭  $R_i$ 로 이루어 진다.

$$TS = \{S_1, S_2, \dots, S_i, \dots, S_K\} \quad (1)$$

$$S_i = \{F_i, L_i, R_i\}$$

$TS$ 를 이용하여 서버는 대역폭  $R_i$ 를 확보한 후에  $S_i$ 를 이루는 GoP들에 속한 프레임들을 클라이언트로 전송한다. 세그먼트  $S_{i+1}$ 의 프레임들을 전송하기 전에 서버는  $S_{i+1}$ 에 필요한 대역폭  $R_{i+1}$ 을 확보하기 위하여 망과 대역폭 협상을 시도한다.

서버는 망에 대한 대역폭 요구를 줄이기 위하여 클라이언트에서 재생을 시작하기 전에 비디오 데이터를 선인출하는 대역폭 평활화를 활용한다. 대역폭 평활화는 최대 비트율, 대역폭 변화 변위와 대역폭 변경 횟수 등의 평활화 조건들을 만족하면서 클라이언트에서 버퍼 오버플로우와 버퍼 언더플로우 없이 비디오 스트림을 전송하기 위한 최적의 대역폭 할당 계획을 구하기 위하여 사용된다<sup>[6,7,8,9]</sup>.

#### 2. 스케줄링 알고리즘

표 1에 본 절에서 사용되는 기호들을 요약하였다.

표 1. 기호 요약

기호	의 미
$L$	GoP내 프레임 수
$N$	비디오 스트림의 총 프레임 수
$m$	고속 재생 속도 인자
$g(i)$	$i$ 번째 GoP내 프레임들의 누적된 데이터 양
$G$	고속 재생을 지원하기 위한 평활화 대상 GoP 집합
$n_{FF}$	고속 전진 재생을 위한 평활화 대상 GoP 수
$n_{FB}$	고속 후진 재생을 위한 평활화 대상 GoP 수
$P_G(n)$	$G$ 에 대하여 시작 GoP로부터 $i$ 번째 GoP까지 클라이언트가 재생한 누적 데이터 양
$U_G(n)$	$G$ 에 대하여 시작 GoP로부터 $i$ 번째 GoP까지 클라이언트가 수신할 수 있는 누적 데이터 양
$A_G(n)$	$i$ 번째 GoP의 재생 시작에서 이용할 수 있는 누적 데이터 양

본 논문에서는  $m$ 배속 고속 전진 및 후진 재생을 나타내기 위하여  $mxFF$ 와  $mxFB$ 로 표기한다.

식 (2)의  $g(i)$ 는  $i$ 번째 GoP 내 프레임들의 누적된 데이터 양을 나타내며,  $L$ 과  $x_j$ 는 GoP 내 프레임 수와 비트 단위의  $j$ 번째 프레임의 크기를 나타낸다.

$$g(i) = \sum_{j=L \times i}^{L \times (i+1) - 1} x_j \quad (2)$$

한편, 클라이언트는 하나의 GoP 내 프레임들을 저장할 고정된 크기의 버퍼 크기  $B_c$ 를 가지며 식 (3)을 만족하여야 한다.

$$B_c = \max_i g(i) \quad (3)$$

고속 재생을 지원하기 위한 전송 스케줄을 구하기 위한 절차는 다음과 같다.

첫 번째, 시작 GoP의 번호  $s$ 를 구한다.  $s$ 를 구하기 위하여 비디오 스트림 내 GoP의 수  $N_{GoP}$ 를 필요로 하며 식 (4)에 의해 구한다. 식 (4)에서  $N$ 은 비디오 스트림의 총 프레임 수이다.

$$N_{GoP} = \lceil \frac{N}{L} \rceil \quad (4)$$

다음에 식 (5)에 의해  $s$ 를 구할 수 있다.

$$s = \lfloor \frac{f}{L} \rfloor \quad (5)$$

두 번째, 재생 방향과 재생 속도 증가 인자  $m$ 에 따라 평활화 대상 GoP 수  $n$ 을 구한다.  $n$ 은 고속 전진 재생의 경우에  $n_{FF}$ , 고속 후진 재생의 경우에  $n_{FB}$ 로 표기하며, 각각 식 (6)과 식 (7)에 의해 구할 수 있다.

$$n_{FF} = \lfloor \frac{N_{GoP} - s}{m} \rfloor \quad (6)$$

$$n_{FB} = \lfloor \frac{s}{m} \rfloor \quad (7)$$

세 번째,  $n$ 을 이용하여 식 (8)과 같이 평활화 대상 GoP 집합  $G$ 를 구성한다.  $s \pm m$ 은  $s$ 번째 GoP로부터 시작하여 순 방향 (+) 또는 역 방향 (-)으로  $m$  개씩 건너 뛰면서 얻어진 GoP를 나타낸다. 식 (8)에서  $n$ 은 평활화 대상 GoP의 수를 나타낸다.

$$G = \{s, s \pm m, s \pm 2m, \dots, s \pm i \times m, \dots, s \pm n \times m\} \quad (8)$$

네 번째, 평활화 대상 GoP 집합  $G$ 에 대역폭 평활화를 적용하여 전송 스케줄을 구한다.

기존 대역폭 평활화 알고리즘들<sup>[6,7,8,9]</sup>은 프레임을 평활화 단위로 하고 있으나 제안된 고속 재생을 위한 스케줄링 방안은 GoP를 평활화 단위로 한다.

식 (9)에서  $P_G(i)$ 는 평활화 대상 GoP 집합  $G$ 에 대하여 시작 GoP  $s$ 로부터  $i$ 번째 GoP까지 클라이언트가 재생한 누적 데이터를 나타낸다.  $i$ 번째 GoP의 재생 시각에 클라이언트의 버퍼에  $P_G(i)$  이상의 데이터를 이용할 수 없는 경우에 버퍼 언더플로우가 발생하며 더 이상의 재생이 불가능하게 된다. 따라서 서버는 비디오 스트림의 연속적인 재생을 보장하기 위해서  $i$ 번째 GoP의 재생 시작 전에  $P_G(i)$  이상의 데이터를 전송하여야 한다.

$$P_G(i) = \begin{cases} 0 & , i = 0 \\ \sum_{j=1}^i g(s + (j-1) \times m) & , 1 \leq i \leq n \end{cases} \quad (9)$$

식 (10)에서  $U_G(i)$ 는 평활화 대상 GoP 집합  $G$ 에 대하여 시작 GoP  $s$ 로부터  $i$ 번째 GoP까지 클라이언트가 수신할 수 있는 최대 데이터 양을 나타내며,  $i$ 번째 GoP 내 프레임들을 재생할 시각에 클라이언트가  $U_G(i)$  이상의 데이터를 수신하게 되면 버퍼가 넘치게 되며 버퍼 용량  $B_c$ 이상의 데이터는 폐기된다.

$$U_G(i) = \begin{cases} B_c & , i = 0 \\ B_c + P_G(i-1) & , 1 \leq i \leq n \end{cases} \quad (10)$$

따라서  $i$ 번째 GoP의 재생 시각에서 이용할 수 있는 누적 데이터를 나타내는  $A_G(i)$ 는 식 (11)에서와 같이  $P_G(i)$ 와  $U_G(i)$  사이에 위치하여야 한다.

$$P_G(i) \leq A_G(i) \leq U_G(i) \quad (11)$$

서버는 두 곡선  $P_G(i)$ 와  $U_G(i)$ 에 의해 둘러 쌓인 영역에서 평활화 조건들을 만족하는 곡선  $A_G(i)$ 의 기울기를 이용하여 각 세그먼트에 할당될 대역폭을 구할 수 있다.

세그먼트들은 대역폭 평활화 알고리즘에 의해 구해지며, 각 세그먼트는  $A_G(i)$ 의 동일한 기울기를 가진다. 또한 세그먼트는 재생 속도 인자  $m$ 에 의해

선택된 GoP들로 구성된다. 서버는  $i$ 번째 세그먼트  $S_i$ 의 시작과 끝 GoP 번호만으로  $S_i$ 에 소속된 GoP들을 구할 수 있으며, 이를 이용하여 GoP내 프레임들을 디스크로부터 검색하여 전송한다.  $S_i$ 에 소속된 일련의 GoP들  $I_i$ 는 재생 방향과 재생 속도 인자  $m$ 에 따라 식 (12)와 같이 표현된다. 식 (12)에서  $k$ 는 재생 방향에 따라 ( $F_i \pm j \times m$ )의 값이 세그먼트내 마지막 프레임 번호  $L_i$ 보다 크지 않은 최대값 또는  $L_i$ 보다 작지 않은 최대값을 의미한다.

$$I_i = \{F_i, F_i \pm m, F_i \pm 2m, \dots, F_i \pm k \times m, L_i\} \quad (12)$$

$$\therefore k = \begin{cases} \max_{0 \leq j \leq n_{FF}} \{j \mid (F_i + j \times m) < L_i\} & \text{for FF} \\ \max_{0 \leq j \leq n_{FB}} \{j \mid (F_i - j \times m) > L_i\} & \text{for FB} \end{cases}$$

서버는 식 (13)을 이용하여  $i$ 번째 세그먼트에 대하여  $R_i$  (단위는 초당 비트수)의 대역폭을 할당하고, 이 정보를 전송 스케줄 정보  $TS$ 에 유지한다. 식 (13)에서  $R_{play}$ 는 클라이언트에서의 재생 속도를 나타내며 보통 초당 24~30 프레임이다.

$$R_i = \frac{A_G(L_i) - A_G(F_i)}{L_i - F_i} \times \frac{1}{L} \times R_{play} \quad (13)$$

그림 4는 정상 재생과 3배속 고속 전진 재생 (3xFF)을 지원하기 위하여 동적으로 전송 스케줄을 작성하는 예를 보여 준다.

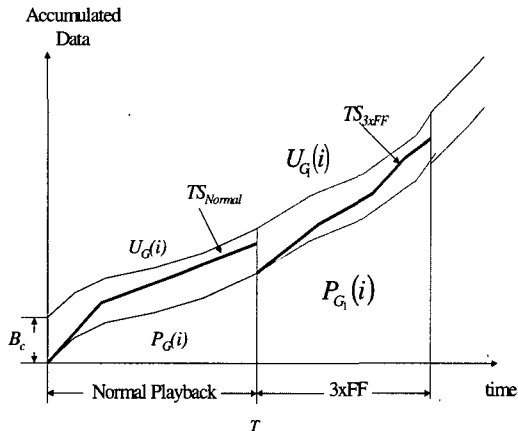


그림 4. 정상 재생과 3xFF를 위한 동적 스케줄링

정상 재생시 서버는 정상 스케줄  $TS_{Normal}$ 에 따라

비디오 데이터를 전송한다. 사용자가  $T$  시각에 3xFF을 요청하는 경우에 서버는 클라이언트로의 데이터 전송을 즉시 중단하고 평활화 대상 GoP 집합  $G_i$ 를 재구성하고 이를 기반으로  $U$ 를 구하여 대역폭 평활화를 적용함으로써 새로운 스케줄  $TS_{3xFF}$ 를 작성한다.

그림 5는 위에서 기술한 동적 스케줄링 알고리즘을 보여 준다. 그림 4에서  $G(i)$ 는 평활화 대상 GoP 집합  $G$ 의  $i$ 번째 GoP의 번호를 나타낸다.

*Procedure Dynamic\_Scheduling*( $B_c, m, n_{FF}, n_{FB}, s$ )

*begin*

*if* (FAST\_FORWARD)  $n = n_{FF}$

*else*  $n = n_{FB}$

$i = 1$

*while* ( $i < n$ ) *begin*

*if* (FAST\_FORWARD)  $G(i) = s + m \times i$

*else*  $G(i) = s - m \times i$

$i = i + 1$

*end while*

$P_G(0) = 0, U_G(0) = B_c, i = 1$

*while* ( $i < n$ ) *begin*

$P_G(i) = P_G(i-1) + g(G(i))$

$U_G(i) = P_G(i-1) + B_c$

$i = i + 1$

*end while*

$TS = \text{Smoothing}(P_G, U_G)$

*Output*  $TS$

*end procedure*

그림 5. 동적 스케줄링 알고리즘

#### IV. 실험 및 고찰

제안된 방안의 성능을 분석하기 위하여 MPEG-1으로 부호화된 비트열 데이터<sup>[16,17]</sup>를 이용하여 GoP 단위로 누적시켜 평활화 대상 GoP의 데이터 양으로 활용하였다. 실험에 사용된 데이터의 트랙백 특성을 표 2에 보였다. 실험 데이터의 GoP 내 프레임 수  $L$ 은 12이며, 하나의 GoP의 길이는 대략 0.5초이다.

표 2. 실험에 사용된 비트열 데이터의 트래픽 특성

데이터	평균 비트율 (Mbps)	최대 비트율 (Mbps)	폭주성 (burstiness)	프레임 수
starwars	0.374	4.446	11.88	174,136
lambs	0.183	3.356	18.38	40,000

그림 6은 “starwars”<sup>[16]</sup>의 GoP 단위의 누적 데이터 양을 보여 준다. 그림 6에서 볼 수 있듯이 VBR 모드의 MPEG 표준으로 부호화된 비디오 스트림의 GoP 단위의 데이터 양도 매우 폭주적인 트래픽을 보인다.

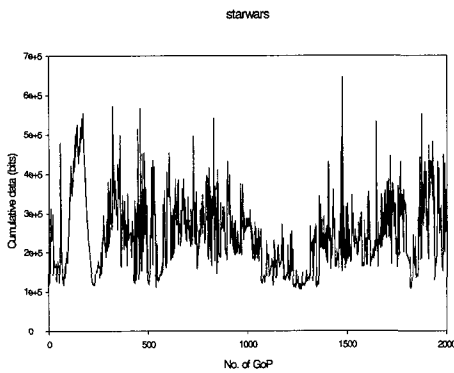


그림 6. GoP 단위의 누적 데이터 양의 변화

그림 7은 “starwars” 데이터를 대상으로 “majorization”을 이용하여 최대 비트율과 대역폭 변화 변위를 최소화하는 대역폭 평활화 알고리즘인 OSA(Optimal Smoothing Algorithm)<sup>[6]</sup>을 프레임과 GoP 단위로 적용한 경우의 계산량을 보여 준다.

$n$ 이 평활화 대상 GoP 수일 때 제안된 동적 전송 스케줄링 방안의 계산 복잡도는  $O(n^2)$ 으로, 그림 7에서 볼 수 있듯이 제안된 방안은 프레임 단위의 대역폭 평활화 방안을 이용한 전송 스케줄링 방안과 비교하여 계산량을  $1/L^2$ 으로 대폭 줄일 수 있었다. 따라서 서버에서의 전송 스케줄 작성으로 인한 지연만을 고려하였을 때 클라이언트에서의 재생 시작 지연은 매우 작을 것으로 예상되므로 사용자가 고속 재생을 요청할 때마다 제안된 방안을 이용하여 동적으로 전송 스케줄을 작성할 수 있을 것으로 기대된다.

그림 8은 “starwars”와 “lambs”<sup>[17]</sup>에 대하여 첫 번째 GoP로부터 고속 전진 재생이 이루어진다고

가정한 경우에 정상 재생과 2~6배속에 따른 평활화 대상 GoP 수  $n_{FF}$ 를 보여 준다.

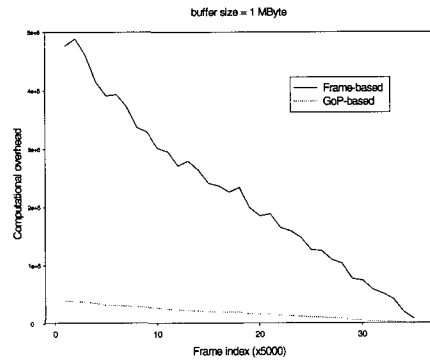


그림 7. 프레임과 GoP 단위로 대역폭 평활화가 적용된 경우의 계산량 비교

그림 8에서 볼 수 있듯이 “starwars”의 경우에  $n_{FF}$ 가 정상 재생시 14,511개에서 6배속 재생시 2,419개로, “lambs”의 경우에 3,333개에서 6배속 재생시 556개로 대폭 감소하고 있다. 따라서, 고속 재생을 지원하기 위한 평활화 대상 GoP 수는 재생 속도가 증가함에 따라 대폭 줄어들게 되어 매우 적은 계산 지연으로 전송 스케줄을 온라인으로 작성할 수 있다.

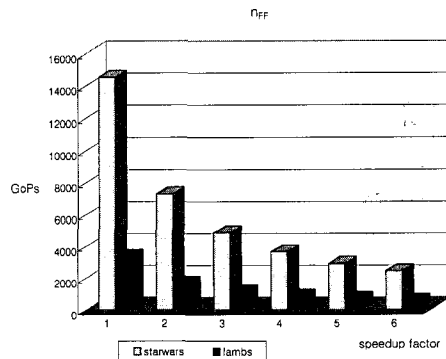


그림 8. 고속 재생시 평활화 대상 GoP 수

그림 9와 그림 10은 사용자가 “starwars”의 20,000번째 프레임 위치에서 사용자가  $m$ 배속 고속 재생을 요청한 경우, 제안된 방안의 버퍼 크기에 따른 최대 비트율과 대역폭 변경 횟수를 보여 준다.

그림 4의  $Smoothing()$  함수의 대역폭 평활화 알고리즘으로 OSA<sup>[6]</sup>을 사용하였다. 대역폭 평활화를 적용하지 않고 재생 시작 지연 없이 고속 재생을

지원하기 위해서는 "starwars"의 경우에는 1.54 Mbps, "lamb"의 경우에는 983 Kbps의 대역폭을 필요로 한다.

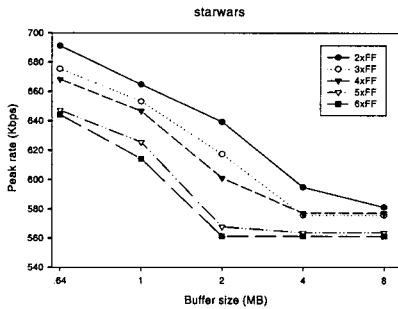


그림 9. 버퍼 크기에 따른 최대 비트율의 변화

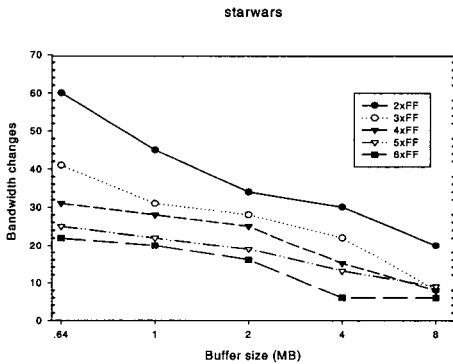


그림 10. m배속 고속 재생시 버퍼 크기에 따른 대역폭 변경 횟수

그러나 대역폭 평활화를 적용하여 고속 재생을 하는 경우에는 재생 속도와 버퍼 크기가 커짐에 따라 최대 비트율이 "starwars"의 경우 691.19 Kbps에서 561.29 Kbps, "lamb"의 경우에 457.48 Kbps에서 230.7 Kbps로 작아진다. 또한, 그림 10에서와 같이 재생 속도와 버퍼 크기가 커짐에 따라 대역폭 변경 횟수도 작아짐을 볼 수 있다.

### V. 결론

본 논문에서는 m배속 고속 전·후진 재생을 위한 GoP 기반의 동적 전송 스케줄 작성 방안을 제안하였다. 제안된 방안에서는 고속 재생을 지원하기 위해 재생 속도에 따른 "GoP 건너 뛰기"를 이용하여

사용자가 고속 재생을 요청한 프레임이 속한 GoP로부터 비디오 스트림의 잔여 GoP들에 대하여 평활화 대상 GoP 집합을 구하고 여기에 대역폭 평활화를 적용하여 대역폭 할당 계획을 작성한다. 본 논문에서 제안한 동적 전송 스케줄 작성 방안은 "GoP 건너뛰기"를 이용하여 정상 재생과 동일한 방법으로 데이터를 복원할 수 있게 함으로써 클라이언트의 구조를 단순화시키면서 고속 재생 효과를 얻을 수 있으며, GoP 단위로 전송 스케줄을 작성함으로써 스케줄 작성시 계산상의 오버헤드를 크게 줄일 수 있기 때문에 허용할 수 있는 재생 시작 지연 범위 내에서 사용자가 요청할 때마다 온라인으로 비디오 데이터를 전송할 수 있을 것으로 기대된다.

### 참고 문헌

- [1] T. D. C. Little, D. Venkatesh, "Prospect for Interactive Video-On-Demand," *IEEE Multimedia*, 1(3), pp. 14-24, 1994.
- [2] Daniel Minoli, *Video dialtone technology: digital video over ADSL, HFC, FTTC and ATM*, McGraw-Hill, 1995.
- [3] 권택근, 최성연, "대화형 VOD 서비스를 위한 멀티미디어 서버의 설계", *JCCI'96*, pp. 853-857, 1996.
- [4] 정제창 역, *그림으로 보는 최신 MPEG*, 교보문고, 1996.
- [5] Naohisa Ohta, *Packet video: modeling and signal processing*, Artech House, 1994.
- [6] J. D. Salehi, Z. L. Zhang, J. F. Kruose and D. Towsley, "Supprting stored video: Reducing rate variability and end-to-end resource requirements through optimal smoothing," *Proc. of ACM SIGMETRICS*, pp. 222-231, May 1996.
- [7] Wu-chi Feng and S. Sechrest, "Critical Bandwidth Allocation for Delivery of Compressed Video," *Computer Communications*, 18(10), pp. 709-717, Oct. 1995.
- [8] W. Feng and J. Rexford, "Performance Evaluation of Smoothing Algorithms for Transmitting Prerecorded Variable-Bit-Rate



