

동적인 채널할당을 이용한 결합형 주문형 비디오 서비스

정회원 이석원*, 박승권**

Hybrid Video on Demand Using Dynamic Channel Allocation

Suk Won Lee*, Sung-Kwon Park** Regular Members

요 약

주문형 비디오 시스템의 설계에 있어 가장 중요하게 고려되고 있는 것은 사용되는 채널의 수를 일정하게 유지 하면서 자입자의 대기시간을 줄이는 방법이다. 이러한 연구의 한 시도로 근래에 실시간 주문형 비디오와 유사주문형 비디오의 장점을 결합하는 결합형 주문형 비디오에 대한 연구가 활발히 이루어지고 있다. 결합형 주문형 비디오에 대한 연구는 결국 어떻게 유니캐스트 채널을 사용하는 사용자들을 그룹화하여 멀티캐스트 채널을 사용하도록 할 것인가에 대한 연구이다. 본 논문에서는 동적인 채널을 이용하여 시청자의 대기시간을 기존의 방식에 비해 효율적으로 줄일 수 있는 결합형 주문형 비디오를 제안하였다. 모의실험 결과 제안된 방식은 10개의 비디오를 100의 채널을 이용하여 0.5 requests/second의 도착률로 서비스할 경우 Unified Video-on-Demand 비해 551%의 성능 향상을 보였다. 본 논문은 동적채널 할당을 이용하여 채널을 해제하고 다시 사용하는 과정, 성능 모델, 모의실험을 제시한다.

Key Words : Hybrid Video-on-demand, Dynamic channel allocation, Improving channel efficiency

ABSTRACT

In designing a video-on-demand (VoD) system, the major challenge may be how to reduce the channels concurrently used maintaining the client's waiting time. For this reason, the various architectures which integrate the multicast streams with the unicast streams were suggested in order to improve channel efficiency in recent years. In combining multicast with unicast, the ways to group the unicast channels together are important so that clients can share the multicast transmission channels. This paper proposes a hybrid video-on-demand system which gathers the unicast channels in new ways and shares multicast transmission channels efficiently by using dynamic channel allocation architecture. The numerical results demonstrate that the proposed architecture in some case achieves performance gain of 551% compared to existing architecture. This paper presents procedure of channel release and reuse, performance analysis, and simulation results of the dynamic channel allocation architecture.

I. 서 론

지난 수 년간에 걸친 디지털 신호처리 및 데이터 압축 기술의 발전과 광대역 가입자망의 대중화는 주문형 비디오(VoD: Video-on-Demand)와 같은 실시간

서비스를 가능하게 하였다. VoD는 시청자가 대형의 비디오 서버에 저장되어 있는 비디오 프로그램을 선택하여 네트워크를 통하여 시청이 가능한 시스템을 말한다. 비디오는 동영상, 음성 등의 대용량 데이터의 집합체로써 압축 이후에도 많은 데이터를 전송해야

* 한양대학교 전자통신전파공학과대학원 (sukwon213@ihanyang.ac.kr)

** 한양대학교 전자통신컴퓨터공학부 (sp2996@hanyang.ac.kr)

논문번호 : KICS2004-08-002, 접수일자 : 2004년 8월 2일

하므로, 이를 네트워크를 통하여 전송할 경우 대단히 큰 채널 대역폭이 필요하여 채널 대역폭을 확보하기 위해서 적지 않은 비용이 소요되므로 채널 대역폭의 효율적 활용이 필요하다.

VoD는 비디오를 전송하는 방식에 따라 분류할 경우 실시간 주문형 비디오 (TVoD: True-VoD) [1]와 유사 주문형 비디오(NVoD: Near-VoD) [2-12]로 크게 분류된다. TVoD는 시청자가 원하는 시간에 원하는 프로그램을 선택하여 시청할 수 있는 방식으로 빨리 돌리기, 되돌리기, 정지, 다시보기 등의 VCR(Video Cassette Recorders)과 같은 기능을 제공하는 대화형 서비스이다. TVoD는 대화형 서비스의 장점이 있지만 시청자가 비디오 서버에서부터 시청자단까지 전송채널을 점유하기 때문에 서비스의 제공에 많은 대역폭이 필요하고 따라서 많은 비용이 소요된다는 단점이 있다.

반면 NVoD는 대화형 서비스 기능과 시청자의 대기시간을 희생하는 대신, 하나의 비디오 프로그램을 여러 개의 방송채널을 통하여 순차적으로 반복하는 방식이다. NVoD는 하나의 비디오 채널을 이용하여 다수의 시청자들이 동시에 비디오를 시청할 수 있도록 하여 시스템 비용을 획기적으로 줄였다. NVoD가 TVoD에 비해 현저히 적은 채널을 필요로 하지만 많은 시청자들이 요청한 비디오를 즉시 볼 수 없다는 단점이 있다.

근래에 NVoD의 성능개선을 위해 많은 연구가 진행되어 왔으며, 이 연구는 크게 피라미드(Pyramid) 방식 [2-7]과 하모닉(Harmonic) 방식 [8-11]으로 분류된다. 이 두 가지 방식은 비디오 데이터의 분할과 시청자단의 저장장치를 이용한 방식이다. 피라미드 방식은 데이터의 크기가 증가하도록 분할하고 분할된 데이터를 동일한 대역폭을 가진 채널들을 이용하여 전송하는 방식이다. 하모닉 방식은 데이터의 크기가 동일하도록 분할하고 분할된 데이터를 전송할 때 채널의 대역폭을 감소시키는 방식이다. 이 방식들은 데이터의 분할을 이용하여 채널대역폭의 효율을 높이거나 혹은 시청자의 대기시간을 줄였으나 시스템의 복잡성을 매우 증가시키는 원인이 되었다.

다른 한편, NVoD의 대역폭 이득의 장점과 TVoD의 짧은 대기시간의 장점을 결합한 결합형 비디오 (Hybrid VoD) [13-15]에 대한 연구가 진행되었다. UVoD(Unified VoD) [13, 14]가 대표적인 결합형 비디오 방식인데, 전체 채널 중의 일부는 멀티캐스트(NVoD)로 사용하고, 일부는 유니캐스트(TVoD)로 사용한다. 이때 임계값(Admission threshold)을 미리 정

해 놓고 시청자의 요청이 도착한 시간과 멀티캐스트 비디오의 시작을 기다리는 시간의 차이가 임계값보다 작을 경우 멀티캐스트를 사용한다. 그리고 이 시간이 임계값보다 클 경우 유니캐스트를 사용하는데, 이때 멀티캐스트로 전송되어 오는 데이터를 사용자의 저장 장치에 저장한다. 일정시간 후에 저장된 데이터를 사용하여 유니캐스트를 해제 할 수 있도록 하여 시스템 성능을 향상시켰다.

본 논문에서는 동적채널 할당방식을 이용하여 NVoD와 TVoD를 결합하는 방식에 대하여 제안하였다. 동적 채널할당 방식에서 전체 채널은 동적 채널과 고정 채널로 나누어진다. 동적 채널은 유니캐스트와 멀티캐스트 양쪽 모두 사용이 가능하며, 고정 채널은 멀티캐스트 용도로 사용이 가능하다. 동적 채널은 시청자단의 저장장치를 이용하여 일정 시간이 흐른 후 해제되어 다른 동적채널로 재사용이 가능한 구조로 되어 있다. 모의실험에서 시청자의 평균 대기시간을 기준으로 성능을 비교한 결과 동적 채널할당 방식이 기존의 방식에 비해 월등히 좋은 성능을 보였다.

본 논문의 구성은 2장에서 동적인 채널할당 방식에 대하여 살펴보고, 3장에서 성능을 분석하기 위해 수치적인 계산을 하겠다. 4장에서 모의실험을 통하여 성능분석을 하고, 5장에서 결론을 맺는 것으로 하겠다.

II. 동적채널 할당 방식

1. 동적채널 할당 방식의 기본 개념

VoD 시스템에서 N 개의 채널이 사용가능 하다고 생각해보자. 본 제안에서는 N 개의 전체 채널을 N_D 개의 동적채널과 $N_{SM} = N - N_D$ 개의 고정 멀티캐스트 채널로 분류한다. N_{SM} 개의 고정 멀티캐스트 채널은 해제되지 않는 채널이며, 비디오 데이터를 처음부터 끝까지 전송하는데 사용된다. 반면 N_D 개의 동적채널은 비디오 데이터의 일부를 전송하고 일정시간이 흐른 후 해제하는 용도로 사용된다.

N_D 개의 동적채널은 다시 N_{DU} 개의 동적 유니캐스트 채널과 N_{DM} 개의 동적 멀티캐스트 채널로 나누어진다. 동적 유니캐스트 채널은 TVoD와 같이 사용되며, 동적 멀티캐스트 채널은 멀티캐스트를 위해서 사용된다. 결국 $N_M (= N_{SM} + N_{DM})$ 개의 고정 멀티캐스트 채널과 동적 멀티캐스트 채널은 종래의 NVoD와 같은 방식으로 일정시간 간격으로 비디오 데이터를 전송하

도록 한다. 두 개의 인접한 고정 멀티캐스트 채널 사이에는 d 개의 동적 멀티캐스트 채널이 존재할 수 있다.

앞서 언급했듯이 N_{DM} 개의 동적 멀티캐스트 채널은 N_D 개의 동적채널의 일부로 일정 시간이 흐른 후 해제되어 재사용이 가능한데, 그 방식은 UVoD [13, 14]와 같이 임계값 δ 를 미리 정하였으며, 시청자단의 장치와 비디오 서버는 임계값을 모두 알고 있다고 가정한다. 비디오 서버는 시청자 요청의 도착률(Request arrival rate)에 따라 임계값을 조정하며, 또한 N_{DU} , N_{SM} , N_{DM} 의 비율을 조정할 수 있다. 임계값의 조정과 N_{DU} , N_{SM} , N_{DM} 의 비율은 시스템의 성능을 좌우하는 매우 중요한 요소로 구체적인 수치는 제 3장에서 제시하겠다.

시청자의 요청이 도착하는 시간과 이어서 서비스되는 멀티캐스트 채널이 시작되는 시간의 차이를 β 라 하자. 시청자의 요청이 도착하면 서버는 β 를 체크한다. 만약 β 가 δ 보다 작을 경우 서버는 시청자가 다음에 시작되는 멀티캐스트 채널을 이용하여 비디오를 시청하도록 한다. 이러한 시청자들을 이 논문에서 멀티캐스트 사용자라 한다. 이때 만약 δ 가 크지 않을 경우 시청자의 대기 시간은 짧을 것이다. 반면 β 가 δ 보다 클 경우 서버는 시청자가 강제로 동적 유니캐스트 채널을 기다리도록 한다. 이러한 시청자들을 이 논문에서 유니캐스트 사용자라 한다. 일정시간이 흐른 후 동적 멀티캐스트 채널과 동적 유니캐스트 채널은 해제될 수 있다. 멀티캐스트 사용자와 유니캐스트 사

용자의 채널을 해제하고 재사용하는 과정은 앞으로 상세히 설명하겠다.

2. 멀티캐스트 사용자

앞서 β 가 δ 보다 작을 경우 사용자는 멀티캐스트 채널을 사용하게 되며 이를 멀티캐스트 사용자라 하였다. 멀티캐스트 사용자의 동작과정은 그림 1과 2에서 설명하였다. 멀티캐스트 사용자의 동작과정은 사용하게 될 채널이 고정 멀티캐스트 채널인가 혹은 동적 멀티캐스트 채널인가에 따라 차이가 있다. 앞서 언급했듯이 인접한 고정 멀티캐스트 채널 사이에 동적 멀티캐스트 채널들이 존재하는데 인접한 고정멀티 캐스트 채널 사이에 존재하는 동적멀티캐스트 채널들의 숫자는 d 이며, d 는 가변적이다. 본 논문의 제 3장에서는 가장 좋은 효율을 보이는 최적의 동적 멀티캐스트 채널의 숫자를 제시하겠다.

그림 1과 2에서 C_i^e 는 채널을 표시하기 위한 것이다. 파라미터 a 는 고정 멀티캐스트 채널의 표시를 위한 것으로 $a=0, 1, 2, 3, \dots$ 의 값을 가진다. 파라미터 k 는 $0 \leq k \leq d$ 사이의 임의의 숫자이며 인접한 두 개의 고정채널들 사이에 위치한 동적채널을 표시하기 위한 것이다. 예를 들면 C_0^0 은 0 번째 고정 멀티캐스트 채널을 의미한다. C_1^0 은 0 번째 고정 멀티캐스트 채널 바로 다음에 서비스되는 1번째 동적 멀티캐스트 채널을 의미한다. C_0^0 채널의 시작 시점이 t_s^0 이고 비디오가 반복되는 주기는 T 라 가정하면, C_1^0

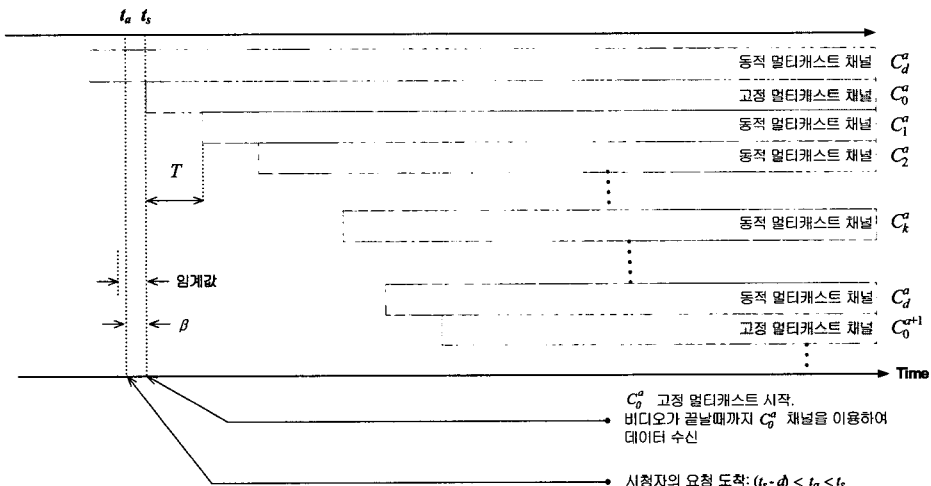


그림 1. 고정채널 멀티캐스트 사용자의 타이밍 절차

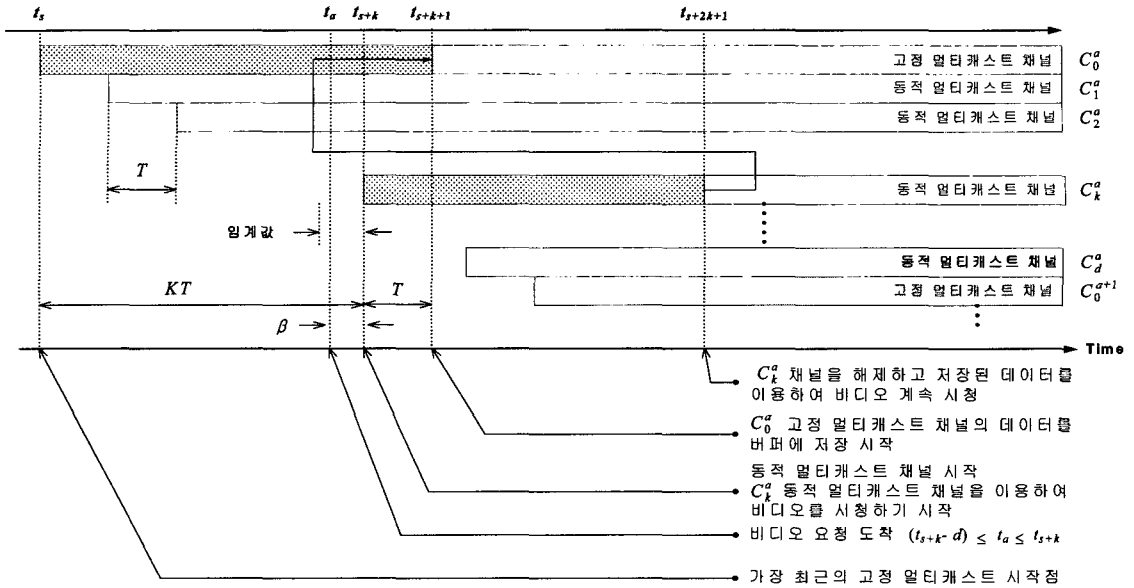


그림 2. 동적채널 멀티캐스트 사용자의 타이밍 절차

채널의 시작 시점은 $(t_s + T)$ 이다. C_0^a 는 0 번째 고정 채널 다음에 할당되는 k 번째 동적 멀티캐스트 채널을 의미하며, 이 채널은 $(t_s + kT)$ 시점에 시작된다.

그림 1은 사용하게 될 채널이 고정 멀티캐스트 채널일 경우를 설명하였다. 그림 1에서 처럼 사용자의 요청이 $(t_s - \delta) \leq t_a \leq t_s$ 에 도착하면 시청자는 다음에 서비스될 고정 멀티캐스트 채널을 기다렸다가 그 채널을 이용하여 비디오를 시청한다. 여기서 t_s 는 C_0^a 고정 멀티캐스트 채널이 시작되는 시간이다. 이 경우 시스템은 저장장치를 사용하지 않고 비디오가 종료될 때까지 고정 멀티캐스트 채널을 이용하여 비디오를 시청하게 된다.

그림 2는 사용하는 채널이 동적 멀티캐스트 채널일 경우를 설명하였다. 그림 2에서는 사용자의 요청이 $(t_{s+k} - \delta) \leq t_a \leq t_{s+k}$ 에 도착하는데, 이때 사용자는 C_k^a 동적 멀티캐스트 채널을 사용하게 된다. 여기서 $t_{s+k} = t_s + kT$ 는 C_k^a 동적 멀티캐스트 채널의 시작 시간이다.

시청자는 C_k^a 동적 멀티캐스트 채널을 이용하여 비디오 시청을 시작한다. 동시에 T 시간이 지나면 가장 최근에 서비스되어진 고정 멀티캐스트 채널을 찾아서 해당 고정 멀티캐스트 채널로부터 비디오 데이터를 전송받아 버퍼에 저장한다. 그림 2에서는 C_0^a 고정 멀티캐스트 채널로부터 데이터를 수신하여 저장하는

데 이는 고정 멀티캐스트 채널의 비디오가 끝날 때까지 지속한다. C_k^a 동적 멀티캐스트 채널을 이용하여 비디오를 시청하던 시청자는 C_k^a 를 통하여 시청하는 비디오가 버퍼에 저장되어 있는 비디오의 시작점과 같아질 경우 동적 멀티캐스트 채널을 해제하고 버퍼에 저장되어 있는 비디오 데이터를 이용하여 연속적으로 비디오를 시청할 수 있다.

그림 2를 예를 들어 구체적으로 설명하면, 시청자는 $t_{s+k} = t_s + kT$ 시간에 C_k^a 동적 멀티캐스트 채널을 이용하여 비디오를 시청하기 시작한다. 그리고 T 시간이 지난 후, $t_{s+k+1} = t_s + (k+1)T$ 시간부터 C_0^a 고정 멀티캐스트 채널로부터 비디오 데이터를 버퍼에 저장한다. 이때 버퍼에 저장되는 비디오 데이터는 비디오의 시작부터가 아니라 비디오의 시작에서 $(k+1)T$ 시간만큼 지난 후부터 저장된다. 따라서 C_k^a 채널은 $(k+1)T$ 시간 이후에 해제가 가능하다. 그림에서 보면 $t_{s+2k+1} = t_s + (2k+1)T$ 시간에 C_k^a 이 해제될 수 있다. 시청자는 C_k^a 채널이 해제된 후, $(k+1)T$ 시간부터 C_0^a 로부터 저장되었던 데이터를 이용하여 계속하여 비디오를 시청할 수 있다. 만약에 NVoD만 고려할 경우 C_k^a 채널의 시작시점부터 C_0^a 의 데이터를 저장하고 $t_{s+2k} = t_s + 2kT$ 시간에

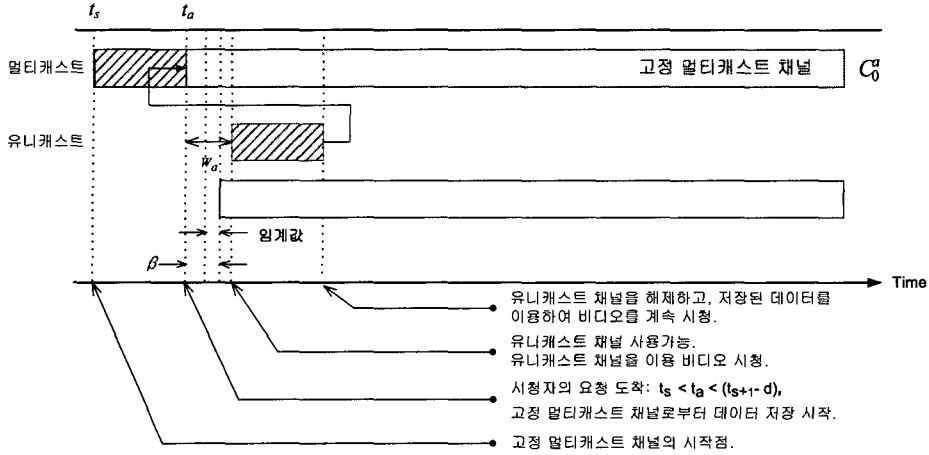


그림 3. 고정채널 유니캐스트 사용자의 타이밍 절차

C_k^o 채널을 해제하는 것이 가장 이상적이다 [12]. 그러나 C_k^o 채널이 서비스 된 이후에 유니캐스트 사용자가 비디오 시청을 요구할 경우를 생각해보면, 요청 시간이 C_k^o 채널이 시작되는 시점부터 최소 δ , 최대 T 시간이 지난 이후라는 것을 알 수 있다. 따라서 유니캐스트 사용자가 C_k^o 채널이 시작되는 시점부터 C_k^o 의 데이터를 저장하는 것이 불가능하며, 유니캐스트 사용자도 동일한 멀티캐스트 스트림을 재사용할 수 있도록 하기 위하여 고정 멀티캐스트 채널의 저장 시점을 T 만큼 지연시켰다.

C_k^o 채널이 해제되어 어떤 채널로 다시 사용될 수 있는지, 동적 멀티캐스트 채널과 고정 멀티캐스트 채널의 비율은 어떻게 조정해야 하는지, 이를 통하여 어느 정도의 채널 효율의 이득을 볼 수 있는지는 3장에서 구체적으로 살펴보겠다.

결론적으로 말하면, 그림 1과 2에서 멀티캐스트 사용자의 대기 시간은 멀티캐스트 채널의 종류에 상관없이 시청자의 요청이 도착한 시간과 다음 멀티캐스트 채널의 시작시간의 차이, 즉 β 가 된다.

3. 유니캐스트 사용자

시스템은 β 가 임계값 δ 보다 클 경우 사용자는 동적 유니캐스트 채널을 사용하게 하며 이를 유니캐스트 사용자라 하였다. 만약 유니캐스트 사용자는 사용 가능한 동적 유니캐스트 채널이 있을 경우 즉시 비디오 시청을 시작하며, 그렇지 않을 경우 사용 가능한 동적 유니캐스트 채널을 기다렸다가 해당 동적 유니캐스트 채널을 이용하여 비디오를 시청한다. 이 경우

유니캐스트 사용자는 사용자의 요청이 도착하기 바로 전에 서비스된 멀티캐스트 채널이 고정 멀티캐스트 채널인가 혹은 동적 멀티캐스트 채널인가에 따라 다른 형태의 동작을 하게 되며 이를 그림 3과 4에 나타내었다.

그림 3는 고정 멀티캐스트 채널이 앞서 서비스된 경우를 나타내었다. 그림 3에서 유니캐스트 사용자의 요청이 $t_s < t_a < (t_{s+1} - \delta)$ 에 도착하였으며, 고정 멀티캐스트 채널은 시간 t_s 에 시작되고 W_a 는 사용 가능한 동적 유니캐스트 채널을 유니캐스트 큐(Queue)에서 기다리는 시간이다. 유니캐스트 사용자는 비디오를 요청하고 사용 가능한 동적 유니캐스트 채널을 기다린다. 그와 동시에 시청자는 앞서 서비스된 고정 멀티캐스트 채널의 비디오 데이터를 버퍼에 저장하기 시작한다. 이 비디오 데이터의 저장은 고정 멀티캐스트 채널을 통해 서비스되는 비디오가 끝날 때 까지 계속된다. 만약 동적 유니캐스트 채널이 사용 가능해지면 시청자는 동적 유니캐스트 채널을 이용하여 비디오 시청을 시작한다. 시청자는 동적 유니캐스트 채널을 통하여 서비스되는 비디오가 버퍼에 저장되어 있는 비디오 데이터의 시작점과 같아지는 순간 동적 채널을 해제할 수 있다. 왜냐하면 시청자는 버퍼에 저장되어 있는 비디오 데이터를 이용하여 지속적으로 비디오를 시청할 수 있기 때문이다.

그림 4는 동적 멀티캐스트 채널이 앞서 서비스된 경우를 나타내었다. 그림에서 시청자의 요청은 $t_{s+k} < t_a < (t_{s+k+1} - \delta)$ 에 도착하였으며, C_k^o 동적 멀티캐스트 채널은 t_{s+k} 에 시작하며, 가장 최근에 서비스되었던 고정 멀티캐스트 채널은 t_{s+k} 에 시작한다. 이

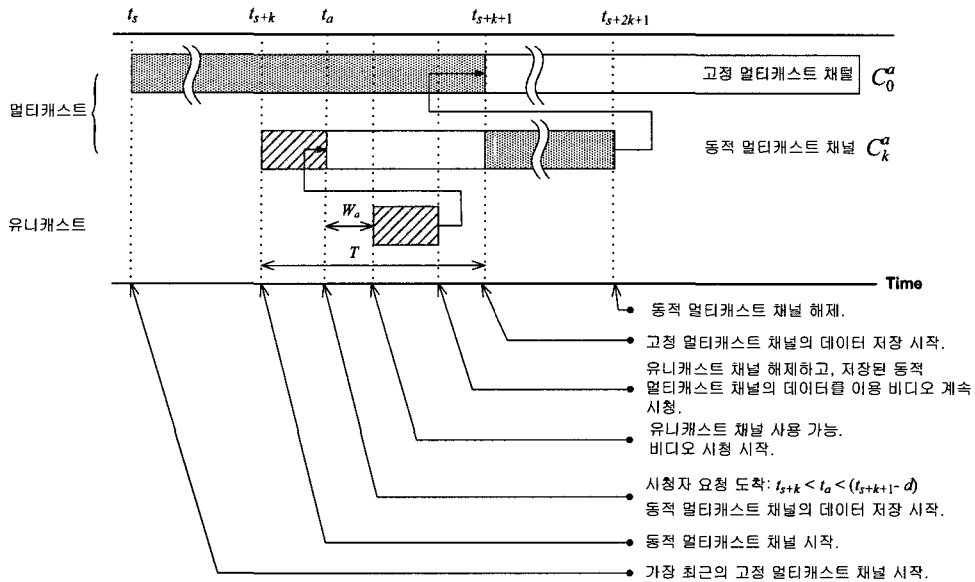


그림 4. 동적채널 유니캐스트 사용자의 타이밍 절차

경우에는 시스템의 동작이 그림 3의 경우와 그림 2의 경우가 연속적으로 나타나는 형태를 보인다. 우선 그림 3과 같이 요청이 도착하자마자 시청자는 앞서 서비스된 동적 멀티캐스트 채널로부터 비디오 데이터를 저장하면서 사용가능한 동적 유니캐스트 채널을 기다린다. 이때 동적 멀티캐스트 채널로부터 데이터의 저장은 고정 멀티캐스트 채널의 저장과 달리 $t_s + (2k+1)T$ 시간까지 한다. 그 이유는 동적 멀티캐스트 채널이 $t_s + (2k+1)T$ 에 해제되기 때문이다. W_0 시간 후, 시청자는 사용가능한 동적 유니캐스트 채널을 이용하여 비디오를 시청한다. 시청자가 버퍼에 저장한 비디오 데이터의 시작점이 현재 동적 유니캐스트 채널로 서비스되고 있는 비디오 데이터와 같아질 경우 동적 유니캐스트 채널은 해제된다.

이와 동시에 $t_{s+k+1} = t_s + (k+1)T$ 시간이 되면, 시스템은 동적 유니캐스트 채널의 해제 여부와 상관없이 동적 멀티캐스트 채널의 해제를 위한 과정에 들어간다. 이 과정은 그림 2에서 설명한 과정과 동일한 방법으로 진행하여 C_k^a 동적 채널을 해제하고 C_0^a 고정 채널을 사용하게 된다. 결국 그림 4에서 시청자는 처음에는 동적 유니캐스트 채널을 사용하고, 다음에는 동적 유니캐스트 채널을 해제하고 C_k^a 동적 멀티캐스트 채널을 사용하게 되며, 마지막으로 C_k^a 동적 멀티캐스트 채널을 해제고 C_0^a 고정 멀티캐스트 채널만

사용하게 된다. 따라서 일정 시간이 흐른 후, 동적 유니캐스트 채널과 동적 멀티캐스트 채널은 다른 용도로 재사용이 가능하게 된 것이다.

결론적으로 그림 3과 4에서 시청자가 비디오를 시청하기 위해 기다리는 대기시간은 앞서서 서비스된 채널이 동적 멀티캐스트 채널 혹은 고정 멀티캐스트 채널 여부에 상관없이 사용가능한 동적 유니캐스트 채널을 기다리는 시간 W_0 가 된다.

III. 성능 모델링

참용성이 없는 시청자의 경우 요청한 비디오를 시청하기 위한 대기시간이 길어질 경우 요청을 취소할 수 있으므로, VoD 서비스 제공자의 관점에서 보면 각 시청자의 정확한 대기시간은 매우 중요하다. 그러나 각 시청자의 정확한 대기시간과 시청자 각각의 행동양식을 알기란 불가능하다. 따라서 본 논문에서는 시청자의 평균 대기시간을 VoD 시스템의 성능을 평가하는 기준으로 하겠다.

본 논문에서는 성능을 모델링 하는 과정에서 계산상의 편의를 위하여 해제된 동적 멀티캐스트 채널은 동적 멀티캐스트 채널로만 사용이 가능하고, 해제된 동적 유니캐스트 채널은 동적 유니캐스트 채널로만 사용이 가능하다고 가정하겠다. 따라서 동적 멀티캐스트 채널과 동적 유니캐스트 채널이 동적 멀티캐스트 혹은 동적 유니캐스트 채널로 상호 교환하면서 사용

이 가능한 실제 시스템의 성능은 본 논문에서 수식적으로 모델링한 성능보다 우수할 것이다.

1. 멀티캐스트 채널의 수

전통적인 NVoD에서 각각 L 길이를 가진 M 개의 비디오를 반복되는 주기 T 로 전송하기 위해 필요한 채널의 수 N_N 은 다음과 같다.

$$N_N = M \times \frac{L}{T} \quad (1)$$

앞서 언급했듯이, 동적 채널할당 방식 시스템에서는 N_D 개의 동적채널은 다시 N_{DU} 개의 동적 유니캐스트 채널과 N_{DM} 개의 동적 멀티캐스트 채널로 나누어지며, $N_M (= N_{SM} + N_{DM})$ 개의 멀티캐스트 채널(고정 멀티캐스트 채널 + 동적 멀티캐스트 채널)이 종래의 NVoD와 같은 방식으로 일정시간 간격으로 비디오 데이터를 전송하도록 한다. 파라미터 d 는 서로 인접한 고정 멀티캐스트 채널 사이에 있는 동적 멀티캐스트 채널의 숫자이므로, 주기 T 를 가지는 시스템에서 길이 L 인 M 개의 비디오를 서비스할 경우 고정 멀티캐스트 채널의 수, N_{SM} , 은 다음과 같이 계산된다 [12].

$$N_{SM} = M \times \left\lceil \frac{L}{T(d+1)} \right\rceil \quad (2)$$

2장에서 동적 멀티캐스트 채널은 시간 $t_s + 2k + 1 = t_s + (2k + 1)T$ 에 해제가 가능하다고 하였다. 이를 다시 말하면 C_k^a 채널이 C_{2k+1}^a 로 재사용이 가능하다는 의미이다. 일반적으로 C_k^a 와 C_{2k+1}^a 사이에 고정 멀티캐스트 채널이 없다면, C_k^a 은 C_{2k+1}^a 로 재사용이 가능하다. 그러나 C_k^a 와 C_{2k+1}^a 사이에 고정 멀티캐스트 채널 C_{k+1}^a 이 존재하면 C_k^a 은 다른 방식으로 사용해야 한다. 시스템의 성능을 고려하면 인접한 고정 멀티캐스트 채널 사이에 있는 동적 멀티캐스트 채널의 숫자 d 가 짝수일 경우가 성능이 좋을 것이다. 따라서 d 가 짝수인 경우만 생각하면 채널을 다음과 같은 방법으로 재사용할 수 있다.

<채널 재사용의 법칙>

- 1) $k < d/2$ 이면, C_k^a 는 C_{2k+1}^a 로 재사용된다.

- 2) $k = d/2$ 이면, C_k^a 는 C_1^{a+1} 로 재사용된다.

- 3) $k > d/2$ 이면, C_k^a 는 $C_{2(k-d/2)}^{a+1}$ 로 재사용된다.

이와 같은 채널 재사용의 법칙을 사용할 경우, 동적 멀티캐스트 채널이 해제되고 재사용됨으로 인하여 동시에 사용되는 동적 멀티캐스트 채널의 수, N_{DM} , 은 다음과 같다.

$$N_{DM} = M \times \left(\frac{d}{2} + 1 \right), \quad d \geq 2 \quad (3)$$

M 개의 비디오를 주기 T 로 반복해서 전송하기 위한 전체 채널 수, N_M , 은 고정 멀티캐스트 채널의 수와 동시에 사용되는 동적 멀티캐스트 채널의 수의 합이 된다. 따라서 전체 채널수 N_M 은 다음과 같다.

$$N_M = N_{SM} + N_{DM} = M \times \left(\left\lceil \frac{L}{T(d+1)} \right\rceil + \left(\frac{d}{2} + 1 \right) \right) \quad (4)$$

식 (4)에서, 만약 L/T 가 일정하다면, 시스템은 가장 작은 N_M 을 가질 때 가장 효율이 좋을 것이다. 그러므로 N_M 을 최소화 하는 d 를 찾는 것이 중요하다. 파라미터 d 는 최소값 0을 가질 경우, 동적채널이 존재하지 않음을 의미하며, 이 경우 시스템은 전통적인 NVoD 시스템이 된다. 그리고 만약 $d = (L/T - 1)$ 일 경우, 영화가 상영되는 길이 L 시간 사이에 하나의 고정채널이 존재함을 의미한다. 이를 수식으로 표시하면 다음과 같이 된다.

$$\left(\frac{L}{T} - 1 \right) \geq d \geq 0. \quad (5)$$

식 (4)는 $\lceil \cdot \rceil$ 때문에 불연속한 값을 가지므로 미분이 불가능하며, 따라서 d 는 L/T 를 변화시키면서 컴퓨터 시뮬레이션을 통하여 구하였다. 만약 N_M 을 최소화하는 d 의 값이 두개 이상 존재할 경우 가장 작은 d 를 선택하는 것이 좋다. 이유는 d 가 작을수록 버퍼에 저장해야 하는 비디오의 데이터량이 작아지기 때문이다. 이를 수식으로 나타내면 다음과 같다.

$$\text{Optimum } d = \{d \mid \text{Minimizes } N_M, (L/T - 1) \geq d \geq 0, d = \text{even}\} \quad (6)$$

그림 5는 N_M 을 최소화 하는 최적의 d 를 구하여 그래프화한 것이다. N_M 이 불연속적인 값을 가지고 있으며, 따라서 N_M 을 최소화 하는 최적의 d 의 값 역시

1) $\lceil \cdot \rceil$ 은 올림을 의미한다.

불연속하게 된다.[12] 이때 동적채널을 사용할 경우 C 는 kT 시간 이후에 해제가 가능하므로 kT 시간의 비디오를 저장할 수 있어야 한다. $0 \leq k \leq d$ 이기 때문에 멀티캐스트만을 위해서 필요한 버퍼의 최대크기는 dT 이다. 또한 유니캐스트 채널 사용을 위하여 필요한 버퍼의 크기는 T 이다.[13-14] 따라서 제안된 방식에서 필요한 저장장치의 최대크기는 $(d+1)T$ 이다. 그리고 그림 6은 최적의 d 로부터 시스템의 성능을 최대화하는 최적의 N_{DM} , N_{SM} 그리고 N_M 값을 구하여 표시하였다.

2. 도착률에 따른 시청자의 평균 대기시간

주문형 비디오 시스템에서 시청자 요청은 λ requests/second의 포아송 프로세스(Poisson process)로 해석이 가능하다. 이유는 시청자의 요청이 균일하고 상호 독립적으로 도착한다고 가정할 수 있기 때문이다. 포아송 프로세스에서 요청이 임계값 δ 안에 도착할 확률은 δ/T 이다. 다시 말하면, 만약 시청자가 비디오 시청을 요구할 경우, 멀티캐스트 사용자는 $P_M = \delta/T$ 의 확률로 발생한다. 반면, 시청자가 비디오 시청을 요구할 경우, 유니캐스트 사용자는 $P_{DU} = (1 - \delta/T)$ 의 확률로 발생한다. VoD 시스템의 시청자 요청의 도착률이 포아송 프로세스이므로, 유니캐스트 사용자의 도착률도 포아송 프로세스이다. 유니캐스트 사용자의 도착률은 $\lambda_{DU} = \lambda P_{DU}$ 가 된다. 멀티캐스트 사용자의 경우, 시청자의 평균 대기시간은 β 의 평균이 된다. δ 안에 도착하는 요청은 0과 δ 사이에 균일하게 분포되므로, 멀티캐스트 사용자의 평균 대기시간은 다음과 같이 주어진다.

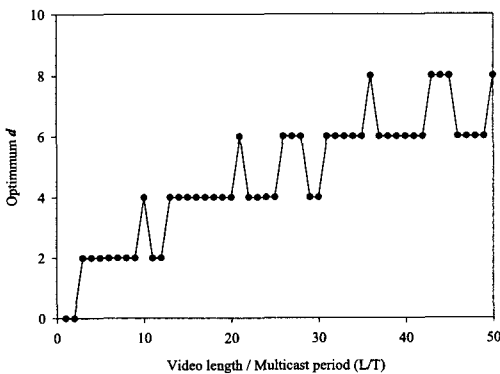


그림 5. 버퍼사이즈와 N_M 을 최소화하는 d

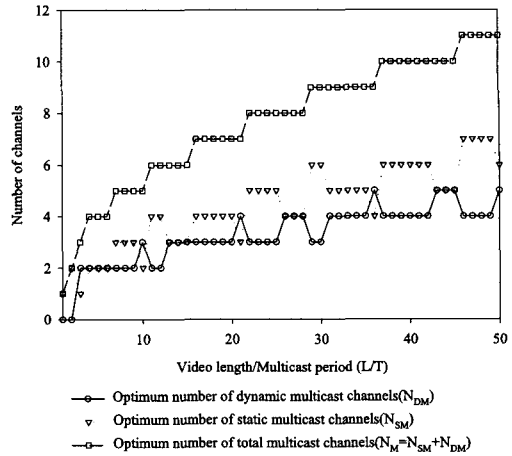


그림 6. 최적의 성능을 보이는 채널의 갯수

$$W_M(\delta) = \frac{\delta}{2}. \tag{7}$$

반면 유니캐스트 사용자의 평균대기시간은 2장에서 언급했듯이, 사용가능한 동적 유니캐스트 채널을 유니캐스트 큐에서 기다리는 시간이다. 이는 큐잉이론으로 분석이 가능하다. 유니캐스트 큐에서 기다리는 시간을 계산하기 위해서는 유니캐스트 사용자가 멀티캐스트 채널을 사용하기 전에 유니캐스트 채널을 사용하는 서비스 시간을 알아야 한다. 유니캐스트 사용자의 서비스 시간은 0과 $(T - \delta)$ 시간 사이에 균일하게 분포되어 있다. 따라서 평균 서비스 시간은 다음과 같이 표시된다.

$$T_s = \frac{(T - \delta)}{2}. \tag{8}$$

동적채널할당 방식에는 $N_{DM} = N - (N_{DM} + N_{SM})$ 개의 동적 유니캐스트 채널이 있다. 유니캐스트 사용자는 포아송 프로세스를 가진 FCFS(First come first serve) 다중 서버 큐로 볼 수 있다. 이와 같은 큐의 정확한 해석은 아직 이루어지고 있지 않으며, 근사값이 구해져 있다. 본 시스템에서는 Allen과 Cunnen에 의해서 구해진 $G/G/m$ 의 근사값을 적용하겠다 [17].

$$W_{DU}(\delta) \approx \frac{Ec(N_{DU}, u)}{N_{DU}(1 - \rho)} \left(\frac{C_A^2 + C_S^2}{2} \right) T_s. \tag{9}$$

수식 (9)에 사용된 변수를 설명하면 다음과 같다.

- ① $C_s^2 = 1$: the coefficient of variation for Poisson process
- ② $u = \lambda_u T_s$: the traffic intensity
- ③ $\rho = u / N_{DU}$: the server utilization
- ④ T_s : the average service time
- ⑤ C_s^2 : the coefficient of variation for uniformly distributed service time.

여기서 C_s^2 는 다음과 같이 값을 구할 수 있다.

$$C_s^2 = \frac{(T - \delta)^2}{12} \left(\frac{2}{(T - \delta)} \right)^2 = \frac{1}{3}. \quad (10)$$

⑥ $Ec(N_{DU}, u)$ 는 아래와 같이 주어진 Erlang C 함수이다.

$$Ec(N_{DU}, u) = \frac{\frac{u^{N_{DU}}}{m!}}{\frac{u^{N_{DU}}}{N_{DU}!} + (1 - \rho) \sum_{k=0}^{N_{DU}-1} \frac{u^{N_{DU}-k}}{k!}}. \quad (11)$$

결국 $W(\delta)$ 로 표시되는 시청자의 평균 대기시간은 다음과 같다.

$$W(\delta) = P_M W_M(\delta) + P_{DU} W_{DU}(\delta). \quad (12)$$

만약 δ 가 일정하다면, 도착률에 따라 $W_M(\delta)$ 의 $W_{DU}(\delta)$ 차이가 매우 클 수 있다. 이는 $W(\delta)$ 를 아주 크게 할 수 있으므로, 도착률에 따라 동적으로 임계값 δ 를 조정하는 것이 좋다. δ 의 조정은 현재 서비스 받고 있는 시청자에게 영향이 없다. 임계값의 조정은 다음과 같이 한다.

$$\delta = \min \{x \mid (w_M(x) - w_{DU}(x) \leq \epsilon, T \geq x \geq 0)\}. \quad (13)$$

그림 7은 80 채널 시스템과 100채널 시스템에서의 임계값 조정을 나타내었다. 여기서는 도착률이 증가하면서 어떻게 임계값을 조정할 것인가를 보여준다. 80 채널 시스템과 100 채널 시스템에서 모두 도착률이 증가하면서 임계값도 같이 증가함을 볼 수 있다. 이는 도착률이 커질수록, 다시 말하면 시청자의 수가 많아질수록 유니캐스트 사용자가 큐에서 대기하는 시간이 기하급수적으로 증가하기 때문에 더 많은 시청자를 멀티캐스트 사용자로 보내게 되는 것이다. 80 채널 시스템의 임계값이 100 채널 시스템의 임계값에

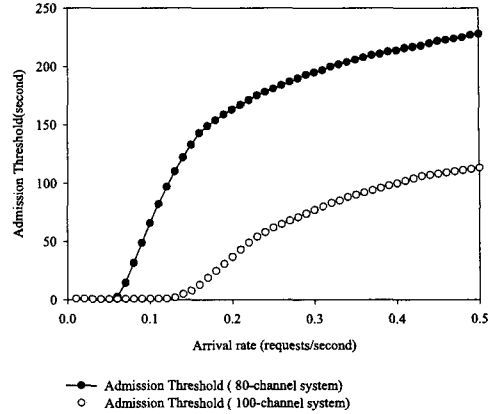


그림 7. 임계값 조정

비해 보다 빠르게 증가하고 있음도 확인할 수 있다.

IV. 모의 실험

동적 채널할당 방식을 모의실험하기 위해 본 논문에서는 비디오의 길이를 모두 120분으로 하였으며, 시청자의 도착률을 포아송 프로세스로 모델링하였다. 또한 시스템의 채널을 80개와 100개로 고정하여 모의 실험을 하였다. 테이블 1에 모의실험을 위한 파라미터들을 나타내었다. 본 논문에서는 시청자의 도착률에 따라 채널을 분할하는 방법과 평균 대기시간을 측정 하였다. 또한 시청자의 평균 대기시간을 기준으로 NVoD, TVoD, UVoD를 동적 채널할당 방식과 비교 하였다.

표 1. 모의실험 파라미터.

Parameters	Symbol	Value
Number of videos	M	10
Total number of channels	N	80 and 100
Number of dynamic unicast channels	N_{DU}	Not available
Number of dynamic multicast channels	N_{DM}	Not available
Number of static unicast channels	N_{SM}	Not available
Number of dynamic channels located between the closest static channels	d	Not available
Length of each video	L	120 minutes

1. 채널 분할 방법

N_{DM} , N_{DU} 그리고 N_{SM} 의 비율은 시청자의 대기시간과 밀접한 관계가 있으므로 모의실험을 통하여 채널들의 비율을 구하였다. 그림 8(a)와 그림 8(b)는 도착률을 0~0.5 requests/second 로 변화시키면서 채널의 비율을 구한 그림이다. 8(a)는 80 채널 시스템을 8(b)는 100 채널 시스템의 결과를 표시하였다. 이 그림들은 시스템의 성능을 최대화하기 위하여 얼마나 많은 N_{DM} , N_{DU} 그리고 N_{SM} 을 할당하여야 하는지에 대하여 보여주고 있다.

그림에서 우선 도착률이 증가함에 따라 N_{DU} 가 감소함을 볼 수 있다. 이는 유니캐스트 시청자의 대기시간이 도착률이 증가함에 따라 빠르게 증가하게 되고 따라서 동적 멀티캐스트 채널의 비율을 늘려 유니캐스트 시청자를 멀티캐스트 시청자로 전환시키는 것이 성능 향상에 유리하기 때문이다. 그렇게 함으로써 동적 채널할당 방식 시스템은 급격한 유니캐스트 시청자의 대기시간 증가를 방지하게 된다.

또한 N_D 의 비율이 N_{SM} 의 비율보다 항상 많은 것을 발견할 수 있다. 이는 동적채널의 수가 항상 고정 채널의 수보다 많게 유지된다는 것을 의미한다.

2. 시청자 평균 대기시간 비교

시청자의 도착률이 낮을 경우 동적 채널할당 방식의 평균 대기시간을 그림 9(a)와 9(b)에 나타내었으며, 이를 TVoD와 UVoD의 평균 대기시간과 비교하였다. 80 채널 시스템에서는 도착률을 0~0.1 requests/second로 변화시켰으며, 100 채널 시스템에서는 보다

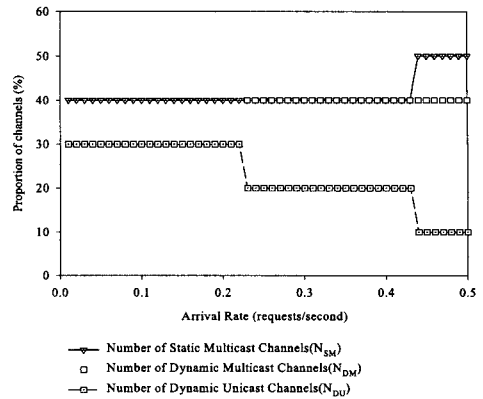


그림 8(b). 최적의 채널 분할(100 채널)
그림 8. 최적의 채널 분할

정확한 성능비교를 위하여 도착률을 0~0.2 requests/second 로 변화시키면서 비교하였다. 기대한

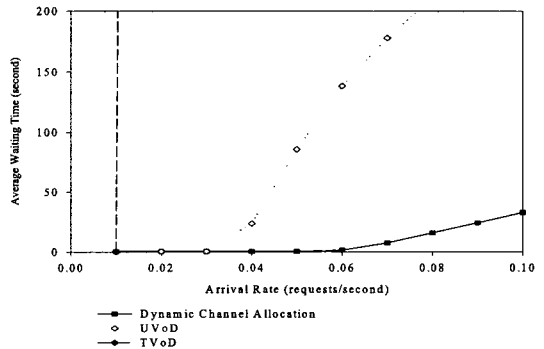


그림 9(a) 도착률이 낮은 경우 평균 대기시간(80 채널)

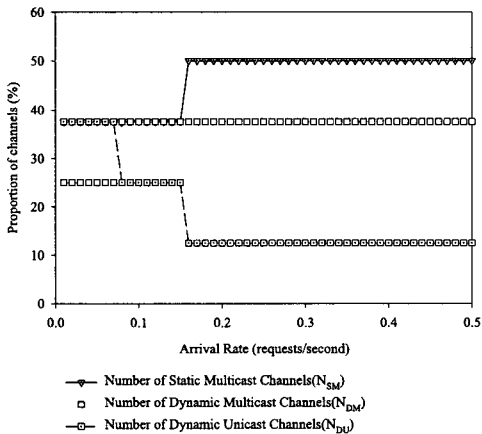


그림 8(a). 최적의 채널 분할(80 채널)

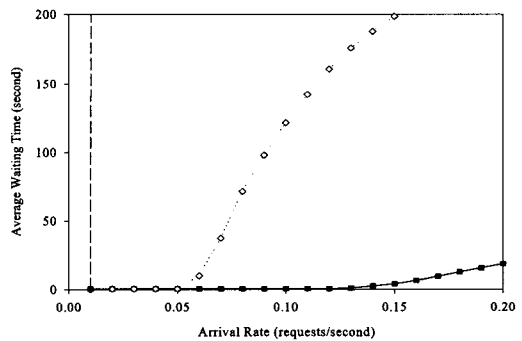


그림 9(b) 도착률이 낮은 경우 평균 대기시간(100 채널)
그림 9. 도착률이 낮은 경우 평균 대기시간 비교

대로 도착률이 증가함에 따라 세 방식의 평균 대기시간이 증가함을 발견할 수 있었다. TVoD는 80 채널 시스템에서는 도착률이 0.01을 넘을 경우 100 채널 시스템에서는 0.02 가 넘을 경우 과부하가 발생하였다. UVoD가 TVoD보다 성능이 좋았으며, 동적 채널 할당 방식이 UVoD보다 좋은 성능을 보였다. 그림 9(a)를 살펴보면, UVoD는 0.04 보다 작을 경우 평균 대기시간이 10초 이하가 되었다. 반면 동적 채널 할당 방식은 0.08이 되어서야 평균대기시간이 10초를 넘게 됨을 확인할 수 있었다. 도착률 0.07에서 동적 채널할당 방식의 평균대기 시간이 UVoD에 비해서 약 95% 감소하였다.

그림 9(b)를 살펴보면, 동적 채널할당 방식의 성능이 그림 9(b)에 비해서 더 나아졌음을 확인할 수 있다. UVoD는 도착률이 0.06보다 작을 경우 평균대기 시간이 10초 이하였으며, 동적 채널할당 방식은 도착률이 0.18이 되어서야 10초 이상이 됨을 알 수 있다.

도착률이 높을 경우를 고려하여 이와 비슷한 그림을 그림 10(a)와 10(b)에 표시하였다. 여기서 UVoD와 동적 채널할당 방식의 평균대기시간이 높은 도착률에서는 증가 속도가 점점 둔화됨을 확인할 수 있다. 앞서 설명한 대로 N_{DM} , N_{SM} 과 N_{DU} 의 비율 조정이 이를 가능하게 한 것이다.

대기시간 비교시청자의 도착률이 낮을 경우 동적 채널할당 방식의 평균 대기시간을 그림 9(a)와 9(b)에 나타내었으며, 이를 TVoD와 UVoD의 평균 대기시간과 비교하였다. 80 채널 시스템에서는 도착률을 0~0.1 requests/second로 변화시켰으며, 100 채널 시스템에서는 보다 정확히 성능을 비교하기 위하여 도착률을 0~0.2 requests/second 로 변화시키면서 성

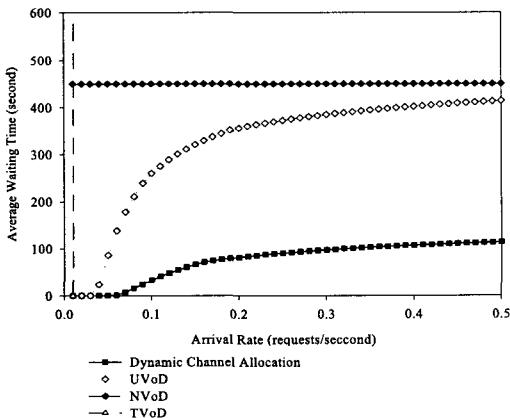


그림 10(a) 도착률이 높은 경우 평균 대기시간(80 채널)

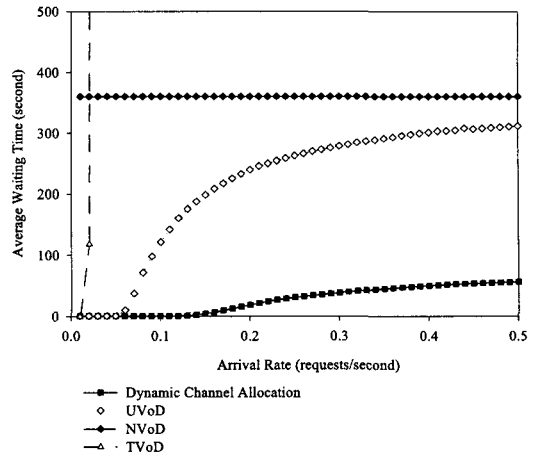


그림 10(b) 도착률이 높은 경우 평균 대기시간(100 채널)
그림 10. 도착률이 높은 경우 평균 대기시간 비교

능을 비교하였다. 기대한대로 도착률이 증가함에 따라 세 방식의 평균 대기시간이 증가함을 발견할 수 있었다. TVoD는 80 채널 시스템에서는 도착률이 0.01을 넘을 경우 100 채널 시스템에서는 0.02 가 넘을 경우 과부하가 발생하였다. UVoD가 TVoD보다 성능이 좋았으며, 동적 채널할당 방식이 UVoD보다 좋은 성능을 보였다.

그림 9(a)를 살펴보면, UVoD는 0.04 보다 작을 경우 평균 대기시간이 10초 이하가 되었다. 반면 동적 채널할당 방식은 0.08이 되어서야 평균대기시간이 10초를 넘게 됨을 확인할 수 있었다. 도착률 0.07에서 동적 채널할당 방식의 평균대기 시간이 UVoD에 비해서 약 95% 감소하였다.

그림 9(b)를 살펴보면, 동적 채널할당 방식의 성능이 그림 9(b)에 비해서 더 나아졌음을 확인할 수 있다. UVoD는 도착률이 0.06보다 작을 경우 평균대기 시간이 10초 이하였으며, 동적 채널할당 방식은 도착률이 0.18이 되어서야 10초 이상이 됨을 알 수 있다.

도착률이 높을 경우를 고려하여 이와 비슷한 그림을 그림 10(a)와 10(b)에 표시하였다. 여기서 UVoD와 동적 채널할당 방식의 평균대기시간이 높은 도착률에서는 증가 속도가 점점 둔화됨을 확인할 수 있다. 앞서 설명한 대로 N_{DM} , N_{SM} 과 N_{DU} 의 비율 조정이 이를 가능하게 한 것이다.

그림 10(a)는 80 채널 시스템의 경우를 나타내었는데, 동적 채널할당 방식의 평균 대기시간이 도착률이 0.5에서 114 초였는데 이는 413초의 평균 대기시간을

가진 UVoD와 비교하면 약 363%의 성능개선을 보인 것이다. 그림 10(b)을 살펴보면 100채널 시스템은 같은 도착률에서 더 좋은 성능 개선을 보였는데, 약 551%의 성능개선을 보였다.

V. 결론

본 논문은 동적 채널 할당을 이용하여 TVoD와 NVoD의 장점을 상호결합하는 결합형 VoD 방식에 대해서 제안하였다. 동적 채널할당 방식은 고정 멀티캐스트 채널, 동적 멀티캐스트 채널, 동적 유니캐스트 채널 그리고 시청자 측의 버퍼를 이용하여 동적채널들을 해제하고 재사용함으로써 성능 향상을 하였다. 또한 멀티캐스트 채널, 동적 멀티캐스트 채널 그리고 동적 유니캐스트 채널의 비율이 성능에 영향을 미치므로 성능을 최대화할 수 있는 비율을 모의실험을 통하여 제시하였다. 그리고 동적 채널할당 방식의 성능을 TVoD, NVoD 그리고 UVoD와 비교하여 동적 채널할당 방식의 성능의 우수함을 보였다.

제안된 방식은 사용자측의 단말기에 버퍼가 필요하고, 멀티캐스트의 전송시간 간격을 사용자측에서 알아야 하며, 채널관리가 복잡해질 수 있다는 것이 단점이라 할 수 있다. 그러나 현재 시청자의 단말기가 점점 발전하고 있으며, 하드디스크와 프로세서를 탑재한 시청자 측의 단말기가 이미 상용화되어 보급되고 있는 추세이므로 사용자 단말기의 버퍼에 관련해 큰 문제가 없을 것으로 사료된다. 또한 비디오 콘텐츠의 디지털화와 단말기의 프로세서 탑재로 인하여 복잡한 채널관리가 가능해질 것이다. 향후 시스템의 가용 채널의 수가 적을 경우에 효과적으로 주문형비디오 서비스 시스템을 구현하는 방법에 대한 연구가 필요하다.

참 고 문 헌

[1] D. Deloddere, W. Verbiest, and H. Verhille, "Interactive video on demand," *IEEE Communication Magazine*, vol. 32, pp. 82-88, May 1994.

[2] S. Viswanathan and T. Imielinski. "Pyramid Broadcasting for video on demand service," In *IEEE Multimedia Computing and Networking Conference*, Vol. 2417, San Jose, California, pp 66-77, 1995.

[3] K. A. Hua and S. Sheu. "Skyscraper

Broadcasting: a new broadcasting scheme for metropolitan video on demand systems," In *SIGCOMM 97*, pp. 89-100, Sept.1997.

[4] Li-Shen Juhn, Li-Ming Tseng, "Fast data broadcasting and receiving scheme for popular video service," *IEEE Transactions on Broadcasting*, vol. 44, pp. 100-105, Mar. 1998.

[5] T. C. Chiueh and C. H. Lu, "A periodic broadcasting approach to video on demand service," *Proc. SPIE*, vol. 2615, pp. 162-169, 1996.

[6] C. C. Aggarwal, J. L. Wolf, and P. S. Yu, "A permutation-based pyramid broadcasting scheme for video on demand systems," In *Proc. International Conference Multimedia Computing and Systems*, pp. 118-126, Jun. 1996.

[7] D. L. Eager and M. K. Vernon, "Dynamic skyscraper broadcasts for video on demand," *MIS'98*, Istanbul, Turkey, pp. 18-32, Sep. 1998.

[8] L. S. Juhn, L. M. Tseng, "Harmonic broadcasting for video on demand service," *IEEE Transactions on Broadcasting*, vol. 43, pp. 268-271, Sep. 1997.

[9] J.-F. Paris, S.-W. Carter, and D.-D. Long, "A low bandwidth broadcasting protocol for video on demand." *MASCOTS' 98*, pp 127-132, Jul. 1998.

[10] Z. Y. Yang, L. S. Juhn, L. M. Tseng, "On optimal broadcasting scheme for popular video service," *IEEE Transactions on Broadcasting*, vol. 45, pp. 318-322, Sep. 1999.

[11] K. C. Almeroth and M. H. Ammar, "The use of multicast delivery to provide a scalable and interactive video on demand service," *IEEE Journal on Selected Areas in Communications*, vol. 14, pp. 1110-1122, Aug. 1996.

[12] Sukwon Lee, Kyungjin Seo, Sungkwon Park, "Improving Channel Efficiency for Popular Video Service Using Dynamic Channel Broadcasting," *IEICE Transaction on Communications*, vol. E87-B, no.10, pp. 3068-3075, October. 2004.

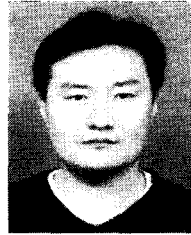
[13] Jack. Y. B. Lee, "UVoD - A unified architecture for video on demand services,"

IEEE communication. letters, vol. 3, no. 9, pp. 277-279, Sept. 1999.

- [14] Jack. Y. B. Lee, "On a unified architecture for video-on-demand services," IEEE Transactions on Multimedia, vol. 4, Issue 1, pp. 38 -47, March 2002.
- [15] W. F. Poon, J. Feng, "Batching policy for video-on-demand in multicast environment," Electronics Letters, Vol. 36, pp. 1329 -1330, 20 July 2000.
- [16] 조진성, 신현식, "큐잉 모델을 이용한 대규모 주문형 비디오 서버의 성능분석", 한국통신학회 논문지, 제23권 1호, pp 155-161, 1998년 1월
- [17] A. O. Allen, Probability, Statistics and Queueing Theory With Computer Science Applications, 2nd ed. New York: Academic, 1990.

이 석 원 (Suk Won Lee)

정회원



1996년 2월 : 한양대학교 전자통신공학과 졸업
 2000년 2월 : 한양대학교 전자통신전파공학과 공학석사
 2000년 3월~현재: 한양대학교 전자통신전파공학과 박사과정

<관심분야> CATV Multimedia Systems, UWB communications

박 승 권 (Sung-Kwon Park)

정회원



1982년 2월: 한양대학교 전자통신공학과 공학사
 1983년 8월: Stevens Institute of Technology, 전자공학과 공학석사
 1987년 12월: Rensselaer Polytechnic Institute, 전자공학과 공학박사

1984년 1월~1987년 8월: Rensselaer Polytechnic Institute, Electrical, Computer and Systems Engineering Dept., Research Assistant
 1987년 9월~1992년 8월: Tennessee Technological University, Electrical Engineering Dept., 조교수
 1992년 9월~1993년 1월: Tennessee Technological University, Electrical Engineering Dept., 부교수
 1993년 3월~현재: 한양대학교 공과대학 전자통신컴퓨터공학부, 교수

<관심분야> 지능형 데이터 방송, CATV Multimedia Systems, Digital Signal Processing