

멀티캐스팅 기술을 사용한 효율적인 노래방 서비스

황 태 준[†] · 우 요 섭^{††} · 김 익 수^{††}

요 약

본 논문에서는 네트워크 지원뿐만 아니라 서버의 부하를 크게 경감시켜 MOD의 서비스 품질을 향상시키는 멀티캐스팅을 사용한 새로운 노래방 서비스 알고리즘을 제안한다. 제안된 효율적인 노래방 서비스 알고리즘은 유니캐스트로 전송되는 노래를 서비스 받음과 동시에 후-순위 예약 곡이 현재 서비스되고 있는 다른 클라이언트들의 예약 곡과 동일할 경우, 멀티캐스트 채널을 통해 자신의 버퍼에 저장하는 방법을 채택하였다. 따라서 유니캐스트 채널을 통한 서비스 종료에 이어 다음 순서의 예약 곡을 네트워크 자원을 통하지 않고 직접 버퍼로부터 서비스되는 방법을 사용한다.

개안된 효율적인 노래방 서비스를 위한 진송 알고리즘은 예약 곡이 많을수록 소요되는 채널의 수가 크게 감소될 수 있으며, 실시간으로 서비스될 수 있다. 본 알고리즘을 이용한 시뮬레이션을 통해 노래방 서비스 시스템의 성능이 크게 향상되었음을 확인하였다.

An Effective Noraebang Service using Multicasting Technology

Tai-Jun Hwang[†] · Yo-Seop Woo^{††} · Ik-Soo Kim^{††}

ABSTRACT

In this paper, we propose an effective Noraebang service algorithm using multicasting technology. The goal of this proposed Noraebang service is to reduce the load of server as well as network resources and to enhance the performance of MOD.

The effective Noraebang service algorithm proposes the new method that stores the post-ordered reserve songs in client's buffer through multicast channels when they are servicing for other clients' requests. Then they are serviced directly from the buffer in the order of request, not through the network and server, after the completion of the service pre-ordered requested song.

The number of channel required in Noraebang service which uses this algorithm can be decreased greatly when many songs are reserved, and the post-ordered songs can be serviced at real time by using buffer. The simulation results of this algorithm are confirmed that the performance of the proposed effective Noraebang service system increased significantly.

1. 서 론

멀티미디어 정보 시스템은 텍스트, 이미지, 음성과 비디오를 포함하는 다양한 형태의 데이터들을 저장, 통신 및 표현하는 기술을 통합하는 분야로서 현재 교

육, 의료와 오락 산업분야에 광범위하게 이용되고 있으며, 이는 앞으로도 계속 폭발적으로 증가될 전망이다. 이와 함께 멀티미디어 정보 서비스는 언제, 어디서나 사용자가 원하는 시점에 서비스가 제공되는 주문형 (On-demand) 서비스의 형태로 발전하고 있다[1, 2]. 주문형 서비스의 형태는 영화 프로를 비디오 테이프 대여점에서 빌려 보는 것과 동일하게 가정에서 시청할 수 있는 오락용의 VOD 서비스, 뉴스와 강의를 시청할

[†] 준 희 원 · 경문적립학교 건임장사
^{††} 정 희 원 · (시립)인천대학교 정보통신공학과 교수
논문집수 · 1999년 10월 23일, 실사완료 · 2000년 5월 31일

수 있는 NOD와 EOD 서비스 등으로 구분할 수 있다. 이와 함께 시청각 교육, 원격영상회의, 홈쇼핑, 대화형 영화, 전자게임, 지리정보 시스템, 전자서적, 전자도서관 등도 멀티미디어 처리 분야에서 급속도로 개발되고 있다. 현재의 컴퓨터, 통신, 응용 분야의 발전과 응용 양상을 관측해보면, 앞으로는 일부 특수한 분야를 제외하고, 거의 모든 응용분야에서 On-demand 서비스가 이루어질 것이다.

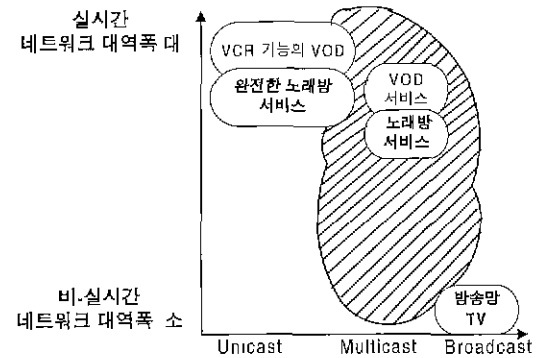
앞서 기술한 바와 같이 멀티미디어의 중요성과 처리 기술의 확보는 당연한 명제이며, 항상 새로운 요구는 새로운 기술의 발전을 필요로 해왔듯이 멀티미디어 데이터 처리 역시 수많은 기술적인 난관을 갖고 있다. VOD, NOD와 EOD 같은 주문형 서비스를 제공하기 위해서는 다양한 데이터 스트림(stream)들을 필요로 하고, 실시간 검색(real-time retrieval) 및 매우 방대한 양의 데이터를 저장하여야 하기 때문에 우선적으로 초고속 네트워크, 대용량의 저장장치와 데이터 압축기술 및 지능을 갖춘 멀티미디어 서버의 개발과 같은 기술적인 발전이 필수적으로 이루어져야 한다[2, 3]. 이와 같은 기술적인 발전과 함께 영화, 음악과 뉴스 및 원격교육 등을 주문형으로 서비스를 제공할 수 있는 MOD 시스템의 실현을 위한 노력이 가속화되고 있다[1, 3, 4, 10] 따라서 연속 매체(continuous media)인 비디오와 음성 신호는 물론 애니메이션과 같은 동영상 서비스를 실시간으로 제공할 수 있는 멀티미디어 서버의 설계에 대한 연구로서 음악과 동영상을 서비스하는 네트워크를 통한 노래방 서비스는 앞으로 수요가 크게 증가할 것으로 기대된다.

네트워크를 통한 노래방 서비스는 예약, 서비스, 역재생, 고속재생 및 정지 등의 서비스를 제공할 수 있는 대화형 VOD(interactive video-on demand) 서비스의 전 단계로서 단지 예약, 서비스, 정지 또는 취소의 기능만을 제공하기 때문에 대화형 VOD에 비해 적은 네트워크 대역폭으로 조만간 서비스가 상용화 될 것이다. 그리므로 네트워크를 통한 노래방 서비스는 다양한 주문형 서비스의 한 분야로서 이용될 수 있다[5, 6, 10].

양질의 서비스를 제공하는 기존의 노래방 서비스는 클라이언트와 서버 사이에 실 시간적인 서비스를 제공하기 위해 각 사용자에게 독립된 채널을 할당해야 하기 때문에 서비스 요청이 많을 경우 매우 방대한 네트워크 대역폭과 서버의 부하를 무시할 수 없으므로 현재 극히 제한적인 서비스만이 제공되고 있다. 따라서 VOD의 노래방 서비스는 제한된 채널 용량으로 서비

스 요청이 즉각적으로 이루어지지 않기 때문에 서비스 요청을 일정 시간 동안 지연시켜 서비스를 제공하고 있다[2, 6]. (그림 1)에 주문형 서비스 요청에 따른 각종 서비스의 실시간 적용의 정도와 네트워크 대역폭의 정도를 나타냈다.

(그림 1)에 나타낸 바와 같이 VCR 기능을 갖춘 완전한 대화형 노래방 서비스와 VOD 서비스는 실시간으로 질 높은 서비스를 제공하지만 한 개 채널이 특정 사용자에게 전적으로 할당되어야 하는 단점 때문에 실현에 큰 어려움이 있지만 멀티캐스팅 전송기법을 사용할 경우 한 개의 채널로 다수의 사용자가 동시에 서비스를 받을 수 있다는 것을 알 수 있다

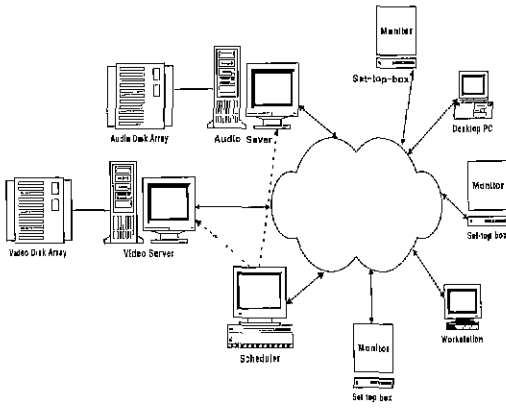


(그림 1) 서비스 종류에 따른 네트워크 대역폭과 실시간 서비스의 정도

본 논문에서 제안한 네트워크를 통한 노래방 서비스를 위한 멀티캐스팅 전송 알고리즘은 사용자측에 소량의 비피만으로 서비스가 실시간으로 제공될 수 있으며, 또한 여러 곳에 산재해 있는 노래방 또는 가정에서 다수의 사용자들이 서비스를 요청할 경우 특히 최근에 히트되고 있는 인기 있는 노래들은 분명히 서비스 요청이 폭주할 것이라는 자명한 사실에 기초를 두고 있다 따라서 노래방 사용자 그룹들이 서비스 요청에 앞서 미리 예약을 하고, 이어서 서비스를 요청할 경우 서버 측의 스케줄러는 각 사용자들의 서비스 요청을 받아 요청된 노래를 가운데 동일한 노래들을 그룹화하여 한 개의 단일 채널로 다수의 사용자 그룹에게 서비스하는 멀티캐스팅 기술을 사용한다. 이 경우 앞서서도 설명한바와 같이 부족한 네트워크 자원의 사용을 극대화할 수 있으며 이와 아울러 서버의 부하도 크게 경감시킬 수 있다

2. Music-on-Demand Server의 구성

부족한 네트워크 자원을 고갈시키지 않고 다수의 사용자가 동시에 서비스를 제공받을 수 있는 특징을 갖는 멀티캐스팅 전송 기법을 사용한 노래방 서비스를 지원하는 Music-on Demand 서버의 구성은 비디오/오디오 서버와 대용량의 저장장치 및 스케줄러(scheduler)로 구성되는 서버 측과 데이터의 전송을 담당하는 네트워크 부분 및 수신장치로 이루어진 클라이언트 부분으로 구성될 수 있으며, 이를 (그림 2)에 나타냈다



(그림 2) MOD 시스템 블록도

2.1 비디오/오디오 서버 및 저장장치

비디오/오디오 서버 및 저장장치는 노래의 특성 또는 사용자의 취향에 맞는 다양한 동영상 및 악기 반주를 포함한 다양한 오디오 신호와 가사 등을 저장하고 또한 네트워크에 이들 데이터 스트림을 전송한다. 동영상 스트림의 경우 뮤직비디오의 부채와 현재 네트워크의 낮은 대역폭으로 인하여 소수의 대표화면(전원 풍경, 수영복 등)으로 브로드캐스트 또는 사용자의 취향에 맞게 사용자 그룹 단위로 멀티캐스트 방식으로 전송된다. 데이터는 CD 음질의 MP3로 압축되어 있다.

2.2 스케줄러

스케줄러는 사용자 그룹들로부터 서비스 요청을 접수하여 채널 상태에 따른 서비스 가능 여부를 알려주고, 예약된 노래들에 대해 서비스 스케줄을 결정함과 동시에 멀티캐스팅 전송이 가능한 노래들을 선별하여 노래방 서버를 제어한다.

본 논문에서 제안한 노래방 서버의 스케줄러는 다음과 같은 특징을 갖고 있다. 클라이언트 A로부터 요청이 접수될 경우 서버는 지금까지 다른 클라이언트들의 예약 곡들과 상이한 단일-요청 곡일 경우 클라이언트 A에게 유니캐스트-채널을 할당하여 신청 곡을 서비스한다. 그러나 다른 클라이언트들의 신청 곡과 동일할 경우 스케줄러는 이들 곡에 대해 한 개의 멀티캐스트-채널을 할당하여 서비스를 제공하게 된다. 물론 클라이언트 A의 후-순위 예약 곡이 다른 클라이언트들의 선-순위 예약 곡과 동일할 경우에는 클라이언트 A에게 비퍼 저장 제어 메시지를 전송하며, 제어 메시지를 수신한 클라이언트들(A 포함)은 현재 서비스 중인 자신의 선-순위 예약 곡을 서비스 받음과 동시에 멀티캐스트-채널로 전송 중인 후-순위 예약 곡을 일단 비퍼에 저장하게 된다. 따라서 비퍼 저장 메시지를 수신한 클라이언트들은 선-순위 예약 곡들을 서비스 받은 후에는 서버와 네트워크를 통하지 않고 예약 순서에 따라 비퍼로부터 후-순위 예약 곡을 서비스 받게 된다. 물론 동일 곡을 선-순위로 예약했던 다른 클라이언트들은 동일한 멀티캐스트-채널을 사용하여 네트워크를 통해 직접 서비스 받게 된다.

이와 같이 스케줄러는 서비스를 요청한 클라이언트들의 서비스 가능 여부는 물론 예약 곡들간의 중복 여부를 판단하여 유니캐스트 또는 멀티캐스트 채널을 할당할 뿐만 아니라 비퍼 저장 제어 메시지도 전송하여야 하며, 또한 서비스를 요청한 모든 사용자 그룹에게 이와 같은 예약 곡에 대한 스케줄 결과를 전송한다. 이 밖에 서비스할 수 있는 채널 수의 제한으로 요청이 거부될 경우 거부되는 회수를 나타내기 위한 계수기를 보유하여야 하며, 이는 다음 스케줄링에서 우선 순위를 갖도록 제어한다.

2.4 네트워크

효과적인 멀티캐스트 전송을 구현하고, 사용자에게 양질의 QoS를 제공하기 위한 충분한 대역폭을 확보해야 한다. 여기서 충분한 대역폭이란 서버로부터 서비스되는 채널 당 소요되는 대역폭(오디오 신호의 경우 135Kbps)을 의미하며, 노래방 서버는 수백 개 이상의 채널을 지원할 수 있어야 한다. 이와 더불어 클라이언트로부터 서비스의 요청은 물론 정지 및 취소 기능과 같은 인터럽트 제어신호의 전달과 요청된 오디오/비디오 스트림의 전송과 서비스 요청의 허가 및 다른 채널

로 전송되는 다른 오디오 스트림의 캐칭(caching)과 같은 제반 제어신호를 전달할 수 있도록 양방향 통신을 제공하여야 한다.

2.5 수신장치

수신장치는 수신된 압축 비디오/오디오 스트림을 디코드하여 각각 모니터에 디스플레이 할과 동시에 스피커를 구동하여야 한다. 이와 동시에 다른 멀티캐스트 채널을 통하여 전송될 수 있는 후-순위 예약 곡들을 수신하여 버퍼에 저장하고, 선-순위 예약 곡들을 서비스 받은 후 자신이 예약한 순서에 따라 버퍼로부터 서비스를 제공받게 된다.

따라서 제안된 방식은 부족한 네트워크 자원을 고갈시키지 않고, 즉 새로운 채널의 할당 없이 예약된 노래를 버퍼로부터 직접 서비스 받을 수 있으며, 이때 서버는 버퍼에 이미 저장되어 있는 노래에 대해 새로이 디스크 접근을 하지 않게 되므로 서버의 부하도 크게 경감시키게 된다.

3. 채널관리 알고리즘

네트워크를 통한 노래방 서비스는 차세대 주문형 서비스의 총아로 불리어지는 VCR 기능을 갖춘 대화형 VOD 서비스와는 달리 노래의 예약과 서비스 개시, 정지 및 취소만을 수행하기 때문에 대화형 VOD 서비스에 비해 상대적으로 단순화된다. 따라서 멀티캐스팅 전송 기술을 사용한 노래방 서비스에 대한 알고리즘은 서버 측과 클라이언트 측을 구분하여 각각 <표 1>과 <표 2>와 같이 구현된다.

<표 1>의 Step 3의 수신은 예약 곡이 다른 클라이언트의 예약 곡과 중복되지 않을 경우 서버로부터 단일-채널(unicast channel)로 데이터를 수신하는 경우와 요청 곡이 다른 클라이언트들과 동일한 경우로서 첫째, 선-순위 예약 곡일 경우 다른 클라이언트들과 함께 멀티캐스트 채널로 예약 곡을 수신하게 된다. 둘째, 후-순위 예약 곡일 경우에는 서버로부터 버퍼 저장 제어 메시지를 수신하고, 이어서 멀티캐스트 채널로 전송되는 후-순위 예약 곡을 버퍼에 저장하게 된다. 마지막으로 선-순위 예약 곡이 이미 다른 채널로 서비스 중인 경우에는 서버로부터 이미 서비스된 부분을 단일-채널로 전송 받아 서비스 받음과 동시에 버퍼 저장 메시지를 전송하여 멀티캐스트 채널로 전송 중인 나머지

부분을 자신의 버퍼에 저장하게 된다.

<표 1> 클라이언트 알고리즘

Step 1	클라이언트 ID, 노래 ID의 전송
Step 2	서비스 인증 ID 수신, 인증 ID를 수신 못할 경우 Step1로 이동
Step 3	수신 <ul style="list-style-type: none"> 1 단일 요청 곡 : Unicast 채널로 수신 2 다른 클라이언트들과 동일 요청 곡 <ul style="list-style-type: none"> ① 선-순위 예약 곡 : Multicast 채널로 수신 ② 후-순위 예약 곡 : 서버로부터 버퍼 저장 메시지 수신 <p>Multicast 채널로 전송되는 노래를 버퍼에 저장</p> <ul style="list-style-type: none"> ③ 이미 서비스 중인 선 순위 예약 곡 : Unicast 채널로 수신 <p>서버로부터 버퍼 저장 메시지 수신</p> <p>Multicast 채널로 전송되는 노래를 버퍼에 저장</p>
Step 4	서비스 <ul style="list-style-type: none"> 1 단일 요청 곡 : Unicast 채널로 A/V 플레이어 동작 2 동일 요청 곡 <ul style="list-style-type: none"> ① 선-순위 예약 곡 : Multicast 채널로 직접 서비스 개시 ② 후-순위 예약 곡 : ① 종료 후 버퍼로부터 서비스 개시 ③ 이미 서비스 중인 선 순위 예약 곡 : Unicast 채널 서비스 개시 <p>버퍼로부터 서비스 개시</p>
Step 5	추가 예약 노래 ID 전송
Step 6	Unicast 채널로부터 서비스 종료 토큰을 수신, 채널 이동

<표 2> 서버 알고리즘

Step 1	클라이언트 ID, 노래 ID를 클라이언트로부터 수신 채널에 여유가 있으면 서비스 인증 ID 송신, 채널의 여유가 없으면 대기, 거부계수기를 1로 세트
Step 2	배열에 클라이언트 ID, 노래 ID 저장
Step 3	스케줄러 매칭 비교 <ul style="list-style-type: none"> 1 동일 예약 곡 ==> Multicast 채널 할당 2 단일-예약 곡 ==> Unicast 채널 할당 <ul style="list-style-type: none"> ① Unicast 채널의 Multicast 채널로 변환 : 서비스 개시 동안 Step 5로부터 예약 곡이 동일할 경우 ② Multicast 채널의 해지 : ①에서 Unicast 서비스가 Multicast 서비스 보다 길 경우 ③ Unicast 채널 해지 : ①에서 Multicast 서비스가 Unicast 서비스보다 긴 경우
Step 4	송신/서비스 <ul style="list-style-type: none"> 1 Unicast 채널 서비스 2 Multicast 채널 서비스 <ul style="list-style-type: none"> ① Multicast 채널로 직접 서비스 ② 버퍼 저장 메시지 송신 : 후 순위 예약 곡이 서비스될 경우 ③ 버퍼 저장 완료 메시지 송신
Step 5	추가 예약 노래 ID 수신, Step 2로 이동

Step 4는 서비스 부분으로서 Step 3에서 수신된 노래에 대한 서비스이다. 우선 단일-요청 곡은 단일-채널로 수신되는 노래를, 그리고 동일 요청 곡일 경우로

서 첫째, 선-순위 예약 곡일 경우 멀티캐스트 채널로 직접 오디오/비디오(A/V) 플레이어에 서비스를 개시하게 된다. 그러나 후-순위 예약 곡일 경우에는 선-순위 예약 곡의 서비스가 종료된 다음 채널을 양도하고 자신의 버퍼에 이미 전송된 후-순위 예약 곡에 대한 서비스가 개시된다 또한 마지막으로 다른 클라이언트들에게 이미 다른 채널(예 채널 k)로 서비스가 진행 중인 선-순위 예약 곡은 서버로부터 채널의 여유가 있을 경우 이미 서비스된 부분만을 단일-채널을 통해 서비스가 이루어짐과 동시에 채널 k 로 전송되는 노래를 자신의 버퍼에 저장하여 단일-채널을 통한 서비스 시점과 버퍼에 저장된 부분이 일치할 때 클라이언트는 단일-채널을 양도하고 노래의 나머지 부분은 버퍼를 통해 서비스 받게 된다.

또한 Step 5는 한 곡의 서비스가 종료될 시점에서 다시 노래를 예약하는 단계를 의미하며, Step 6은 Step 4로부터 단일-채널로 서비스되는 단일-예약 곡이 종료될 경우 현재 서비스된 단일-채널을 양도하고, Step 4에서 설명한 바와 같이 Unicast 서비스 중에 이미 다른 채널로 멀티캐스트 서비스된(또는 되는) 다른 후-순위 예약 곡은 서버를 통하지 않고 버퍼로부터 서비스 되는 단계이다.

<표 2>에 나타난 서버 알고리즘 Step 3의 스케줄러 부분은 클라이언트들의 예약 곡들은 요청 순서에 따라 2차원 배열 큐(Queue)에 저장하게 되며, 서비스 채널 수의 제한 때문에 서비스 요청 순서에 따라 우선 순위를 결정하게 된다. 또한 스케줄러는 서비스를 요청한 클라이언트 C_1 부터 C_N 까지의 예약 곡들 가운데 동일 예약 곡 여부를 검사하여(클라이언트 당 예약 곡 수를 $R_N = 5$ 로 5곡씩 예약하도록 하였다) 중복-예약 곡들은 Multicast-채널을 통해서 그리고 단일-예약 곡들은 Unicast-채널로 서비스되도록 한다.

Step 3의 단일-예약 곡의 경우 Unicast-채널의 할당은 몇 가지로 구분될 수 있으며, 우선 서비스가 종료될 때까지 다른 클라이언트로부터 동일 곡의 서비스 요청이 없는 경우로서 이때 Unicast-채널은 서비스가 종료될 때까지 유지된다. 그러나 서비스 도중 Step 5로부터 클라이언트 K 가 동일 곡을 예약할 경우 채널 사용의 효율성을 위하여 서버는 이미 서비스 중인 Unicast-채널을 Multicast-채널로 변환하여 서비스를 계속하게 된다. 이에 앞서 서버는 클라이언트 K 에게 버퍼 저장 메시지를 전송하며, 이때 Multicast-채널로 전송되는 부분은

버퍼에 저장하게 된다 이와 동시에 서버는 클라이언트 K 에게 새로운 Unicast-채널로 이미 서비스된 부분만을 서비스하고, 클라이언트 K 는 채널을 양도함과 동시에 나머지 부분은 버퍼를 통해 서비스 받게 된다.

Multicast-채널의 해지는 클라이언트 K 의 예약이 있기 전에 서비스된 부분이 Multicast로 서비스된 부분보다 길 경우에 발생하는 경우로서 클라이언트 K 는 버퍼에 저장을 완료한 후에도 자신에게 이미 할당되어 있는 Unicast-채널로 서비스를 받고 이의 종료 후에는 채널을 양도하고 버퍼로부터 나머지 부분에 대한 서비스를 계속 받게 된다. 또한 Unicast-채널의 해지는 앞의 경우와 반대되는 경우로서 클라이언트 K 에 있어서 Multicast로 서비스되는 부분이 더 긴 경우에 해당된다.

Step 4의 송신/서비스의 Unicast-채널 서비스는 단일-예약 곡일 경우이며, Multicast-채널 서비스의 직접 서비스는 클라이언트들 사이에 동일-예약 곡이 존재하여 이들의 서비스 시점이 동일할 경우이다. 또한 버퍼 저장 제어 메시지 송신의 예로서 첫 번째는 클라이언트가 단일-예약 곡을 Unicast-채널로 서비스 받는 동안 후 순위로 예약된 곡이 멀티캐스트 채널을 통해 전송될 경우 클라이언트는 전송 스트림을 버퍼에 저장하게 되며, 두 번째는 Step 3에서 Unicast-채널의 Multicast-채널로 변환하는 경우로서 단일-예약 곡이 서비스되고 있는 사이에 Step 5로부터 다른 클라이언트 K 가 서비스 중인 노래를 예약하는 경우로서 현재 시점 이후 부분을 Multicast-채널을 통하여 버퍼에 저장하도록 함과 동시에 서버는 클라이언트 K 에게 Unicast-채널을 할당하여 이미 서비스된 부분을 서비스하도록 한다. 따라서 클라이언트 K 는 노래를 예약하기 전에 이미 서비스된 일정부분은 Unicast-채널로 서비스 받고, 그이후의 부분은 버퍼를 통해 서비스가 계속 이루어지게 된다.

클라이언트의 Set-top-box에 삽입되는 버퍼는 VCR 기능을 갖춘 대화형 VOD의 경우보다 훨씬 단순하며, 구현이 매우 용이하기 때문에 버퍼의 용량은 몇 개의 곡만을 저장할 수 있는, 즉 MP3의 경우 최대 4곡을 저장할 분량인 20MB 정도만이 필요할 것이며, 이는 시스템을 구현하는데 문제점이 없으리라 판단된다[10].

스케줄러에서 서비스 요청에 대해 채널을 할당하는 예로서 <표 3>으로부터 클라이언트 C_1 의 첫 번째 예약 곡 22의 서비스가 종료되기 전에 클라이언트 C_2 와 C_3 의 서비스 요청이 동시에 인증 받은 후 클라이언트

C2의 첫 번째 예약 곡 10번이 서비스 될 경우, 그리고 노래 10번은 3개 클라이언트들이 이미 예약되어 있으므로 클라이언트 C1과 C3은 각기 Unicast-채널 1과 3을 통해 각기 노래 22와 8이 서비스됨과 동시에 채널 2는 Multicast-채널로 노래 10번이 서비스되고 있으므로 서버는 클라이언트 C1과 C3에 버퍼 저장 메시지를 전송하여 Multicast-채널 2를 통해 전송되는 스트림을 각각의 버퍼에 저장하도록 제어한다 따라서 클라이언트 C1은 첫 번째 예약 곡 노래 22를 서비스 받은 다음 채널 1을 양도하고 버퍼로부터 예약 곡 10을 서비스 받게 되며, 이의 종료와 함께 다시 3번째 예약 곡 35를 서비스 받기 위해서 다시 계속해서 채널을 요청하게 된다. 클라이언트 C3은 클라이언트 C1과 같이 2, 3번째 예약 곡 17과 49를 서비스 받은 후 버퍼로부터 10번 곡을 4번째로 서비스 받게 된다. 또한 <표 3>으로부터 서버는 Multicast-채널을 사용하여 동일한 예약 곡 10번 곡을 서비스 한 후에는 Unicast-채널로 변환하여 클라이언트 C2의 두 번째 예약 곡 41을 Unicast 서비스하게 된다.

<표 3> 클라이언트 대 예약 순서 2차원 배열

클라이언트(C _N) \ 예약 순서(R _N)	1	2	3	4	5
C ₁	22	10	35	47	54
C ₂	10	41	30	20	35
C ₃	8	17	49	10	27
...
C _N	35	9	30	40	22

그리고 클라이언트로부터 서비스 요청이 폭주할 때 채널의 여유가 없을 경우 요청은 거부되고 이때 거부 계수기가 1로 세트되며 이는 채널의 여유가 생길 때 우선 순위를 갖게 되어 거부 계수기가 1인 사용자에게 서비스 요청을 허락(인증 ID 전송)한다

이상과 같은 멀티캐스팅 알고리즘을 기본으로 사용자 그룹들의 예약된 노래가 서비스되며, 이는 Unicast 서비스에 비해 서비스되는 비율이 크게 향상된다.

4. 시뮬레이션과 성능 분석

제안된 알고리즘을 사용하여 시뮬레이션 결과를 다음과 같은 환경에서 성능을 평가하였다.

서버가 제공할 수 있는 N개의 노래 곡목 ID 가운데 클라이언트들로부터 1번째로 자주 선택되는 노래의 확

률을 구하기 위하여 Zipf 분포를 사용한다[8, 9]. Zipf 분포에서 서버로부터 제공되는 N개의 노래 곡목 가운데 클라이언트들로부터 1번째로 자주 선택되는 확률 p_1 는 $p_i = Z/i$ 로 나타낼 수 있으며, 여기서 $Z = 1/(1+1/2+1/3+\dots+1/N)$ 이다.

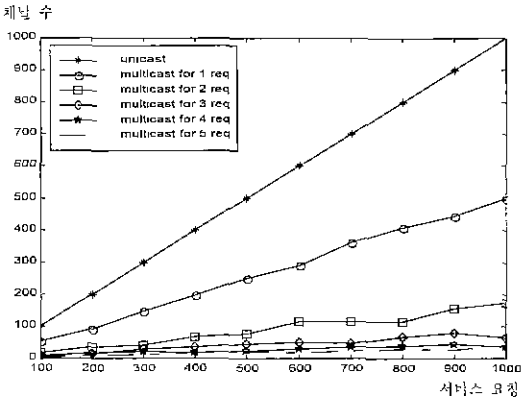
따라서 클라이언트들로부터 최고 인기 곡($i=1$)의 요청 확률은 다른 곡에 비해서 크기 때문에($p_1 \gg p_N$) 클라이언트들이 자주 선택하는 노래, 즉 인기 곡들의 ID는 낮은 ID번호를 보유하여야 함을 알 수 있다. 이로부터 클라이언트들이 i번째로 자주 요청하는 노래의 서비스 요청 율은 $\lambda_i = \lambda p_i$ 이고, 여기서 λ 는 분당 서비스 요청 율이다[8]

시뮬레이션에 사용되는 서버가 제공할 수 있는 노래의 곡목 수는 $N=5000$ 곡이고, 서비스 요청 율 λ 는 분당 100~1000회로 가정한다. 그리고 5000곡의 노래 가운데 서비스 요청이 빈번한 인기 곡들은 다른 곡들에 비해 선택 확률에 대한 가중치(weight value)가 높아야 하기 때문에 앞에서 구한 선택 확률인 p_i 를 제공하는 노래 가운데 선택되는 각각의 노래에 대한 가중치로 사용하였다.

따라서 클라이언트들의 서비스 요청으로부터 인기 곡의 선택 확률을 상대적으로 높여 시뮬레이션을 수행하였으며, 여기서 클라이언트들로부터 가중치에 의해 서비스 요청된 예약 곡들에 대해 스케줄러에서 서비스를 위한 스케줄링을 수행하여 단일 곡들은 기존의 방법과 동일하게 Unicast-채널을 할당하고, 중복된 동일 곡의 경우에 대해서는 Multicast-채널을 할당하여 서비스를 제공하는 시뮬레이션 방법을 도입하였다

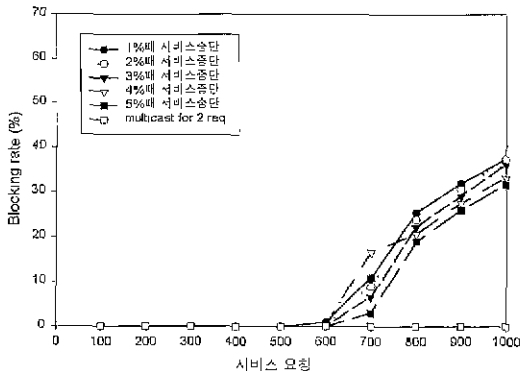
(그림 3)은 서비스 요청 율에 따른 요구되는 채널 수의 결과를 나타냈으며, (그림 3)에 나타난 바와 같이 본 논문에서 제안한 멀티캐스팅 기술을 사용한 알고리즘을 사용하여 서비스 요청 율 $\lambda=100$ 에서 $\lambda=1000$ 인 경우로서 한 개 곡만을 요청할 때 기존의 Unicast 서비스에 비해 채널 수가 49.6%로 줄어들 수 있음을 확인하였다 또한 5개 곡을 예약할 경우에도 Unicast 시나리오에서 요구되는 채널의 수에 비해 두 번째 예약 곡의 경우는 소요 채널의 수가 16.3%로 줄어들었고, 세 번째 예약 곡의 경우 8.3%로, 그리고 네 번째 예약 곡의 경우 5.7%로 감소될 수 있음을 확인하였다 또한 다섯 번째 예약 곡의 경우에는 불과 요구되는 채널의 수가 3.7%로 줄어들게 되었음을 확인하여 제안된 멀티캐스팅 전송 알고리즘의 성능이 크게 향상되고 있음을

알 수 있다.



(그림 3) 서비스 요청에 대한 채널 수

또한 노래방 서버로부터 서비스의 개시 후 클라이언트가 서비스 중단을 요구할 경우 Unicast-채널로 서비스를 제공하였을 경우 일시적으로 채널을 양도한 다음 다시 요청을 개시하기 때문에 서비스가 차단될 수도 있다. 또한 Multicast-채널로 서비스 도중에 중단을 요구할 경우에는 이와 같은 서비스 차단의 시간이 길어질 수도 있기 때문에 노래방 서버가 300개의 서비스 채널을 보유할 경우에 노래방 서버로부터의 서비스 차단율(Blocking rate)의 결과를 (그림 4)에 나타냈다.



(그림 4) 서비스 요청에 따른 Blocking 율

(그림 4)에 나타낸 바와 같이 제안된 멀티캐스팅 전송 알고리즘은 클라이언트들이 서비스 개시에 앞서서 5곡을 예약하고, 서버로부터 예약 곡을 서비스 받는 도중에 이를 중단하는 율이 1~5%일 경우 클라이언트

들로부터 첫 번째 신청 곡만을 제외하고 모두 서비스가 이루어질 수 있음을 나타내고 있다.

5. 결 론

본 논문에서 제안한 네트워크를 통한 노래방 서비스 알고리즘은 부족한 네트워크 자원을 멀티캐스팅 전송 기법을 사용하여 요구되는 채널의 수를 크게 줄일 수 있음을 확인하였으며, 이와 동시에 서버의 부하도 크게 감소시켜 클라이언트가 5곡을 예약할 경우 Unicast 전송에서 소요되는 채널의 수에 비해서 최대 37%까지 줄일 수 있음을 확인하였다. 따라서 시스템의 성능은 기존의 시스템에 비해 전체적으로 5배 이상(5.98배)으로 최대화시킬 수 있었으며, 사용되는 버퍼의 용량이 20KB 정도로 작기 때문에 시스템 구현에 경제적 문제점이 없으리라 판단된다 또한 서비스 개시 후 클라이언트들의 서비스 중단 율에 따른 노래방 서버의 서비스 차단 율은 첫 번째 예약 곡만을 제외하고 모두 정상적으로 서비스될 수 있음을 확인하였다.

앞으로 노래의 인기도에 따른 서비스 요청 분포를 보다 정확하게 측정할 수 있어야 할 것이며, 이를 토대로 실제 시스템이 구현될 경우 노래방은 물론 원격 교육 서비스에 이용될 수 있으리라 사료되고, VCR 기능을 갖춘 대화형 VOD 서버의 기초로도 유용하게 사용될 수 있을 것이다.

본 연구는 한국 과학 재단 지정 인천대학교 멀티미디어 연구센터의 지원에 의한 것입니다.

참 고 문 헌

- [1] P. Basu, R. Krishnan and T. D. C. Little, "Optimal Stream Clustering Problems in Video-on-Demand," Proc. Parallel and Distributed Computing and Systems, LasVegas, NV, Oct. 1998
- [2] K. Almeroth and M. Ammar, "The Interactive Multimedia Jukbox(IMJ) : A New Paradigm for the On-Demand Delivery of Audio/Video," the 7th International WWW conference, Brisbane, Australia, April 1998.
- [3] L. Golubchik, J. Lui and R. Muntz, "Adaptive Piggybacking : A Novel Technique for Data Sharing

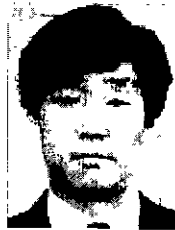
in Video-on-Demand Storage Servers." ACM Multimedia Systems Journal, Vol.4, No.3, pp.140-155, 1996.

- [4] J. Caviedes, "A Technological Perspective of Anytime, Anywhere Education," ALN Magazine Vol 2, Issue 1, 1998
- [5] Y. J. Lee, D. H. C. Du and W. H. Ma, "SESAME : A Scalable and Extensible Architecture for Multimedia Entertainment," IEEE 20th Int'l Computer Software and Applications Conference (COMPSAC96) Seoul, Korea, Aug 1996.
- [6] K. Almeroth and M. Ammar. "On the Use of Multicast Delivery to Provide a Scalable and Interactive Video-on-Demand Service," Journal on Selected Areas of Communications(JSAC), Aug. 1996.
- [7] D. Jadav, C. Srinilta and A. Choudhary, "An Evaluation of Design Tradeoffs in a High Performance Media-on-Demand Server," ACM Multimedia Systems Journal, Jan 1997.
- [8] P. Mundur, R. Simon and A. Sood, "Integrated Admission Control in Hierarchical Video-on-Demand Systems," IEEE Multimedia Systems Vol.1, pp.220-225, 1999.
- [9] K. A. Hua, Y. Cai and S. Sheu, "Patching : A Multicast Technique for True Video-on-Demand Services," ACM Multimedia 98 the 6th ACM Int'l Multimedia Conf., pp 191-200, 1998.
- [10] Backhyun Kim, Seungchan Moon, Iksoo Kim and Yoseop Woo, "A Buffering Algorithm for Providing The Truly Interactive Video-On-Demand Services," Proceeding of International Conference on Parallel and Distributed Processing Technology and Applications(PDPTA 99). pp.211-217, 1999.



황 태 준

e-mail : tjhwang@isis.inchon.ac.kr
 1997년 인천대학교 전자계산학과
 1999년 인천대학교 대학원 전자계산학과(공학석사)
 1999년 인천대학교 대학원 박사과정 정보통신공학과
 2000년~현재 경문직업학교 전임강사
 관심분야 : 멀티미디어, 데이터베이스 암호화



우 요 섭

e-mail : yswwoo@lion.inchon.ac.kr
 1986년 한양대학교 전자통신공학과
 1988년 한양대학교 대학원 전자통신공학과(공학석사)
 1992년 한양대학교 대학원 전자통신공학과(공학박사)
 1992년~1994년 인천대학교 정보통신공학과 조교수
 1994년~현재 (시립)인천대학교 정보통신공학과 부교수
 관심분야 : 한국어 정보처리, 멀티미디어 정보검색, VOD



김 익 수

e-mail : iskim@lion.inchon.ac.kr
 1978년 동국대학교 전자공학과
 1981년 동국대학교 대학원(공학석사)
 1985년 동국대학교 대학원(공학박사)
 1988년~1994년 인천대학교 정보통신공학과 부교수
 1993년~1994년 North Carolina State Univ. 객원교수
 1994년~현재 (시립)인천대학교 정보통신공학과 교수
 관심분야 : 병렬처리, 컴퓨터 구조, VOD, 및 ATM 스위치