

Fast Patching 기법을 이용한 주문형비디오 서비스의 구현

송은돈, 홍종우, 김홍익, 박승권
한양대학교 전자통신컴퓨터공학과

The Fast Patching Scheme for VoD Service

Song, Eun-Don, Hong, Jong-Woo, Kim, Hong-Ik, Park, Sung-Kwon

Hanyang University

{lallala, smart799, hongik, sp2996}@hanyang.ac.kr

요 약

최근 방송통신융합 기술에 대한 관심이 높아지고 있는 상황에서 주문형비디오 서비스에 대한 관심도 증가하고 있다. 본 논문에서는 주문형비디오 서비스의 구현을 위해 기존의 패칭 기법 중 Fast Data broadcasting 방식을 활용한 패칭 기법을 제안한다. 대역폭이 제한되어 있는 통신 환경에서 사용자는 요청한 비디오를 시청하기까지 일정 시간을 기다려야 하는데, 사용자 요청 시점 이전 데이터를 Fast Data broadcasting 방식을 적용한 패칭 채널을 통해 전송함으로써 대기 시간을 크게 줄이는 효과를 얻을 수 있다.

1. 서론

최근 방송통신의 융합의 기술에 대한 관심이 높아지고 있는 상황에서 방송의 서비스를 통신으로, 통신의 서비스를 방송으로 제공하려는 노력이 활발히 이루어지고 있다. 이에 따라 주문형비디오 서비스(Video on Demand: VoD)에 대한 관심도 크게 증가하고 있다. VoD를 구현하는 방법에는 사용자의 요구가 있을 때, 각 사용자에게 하나의 채널을 할당하여 원하는 비디오를 전송하는 RVoD(Real Video on Demand) 서비스와 VoD 시스템 대역에서 제한된 몇 개의 비디오를 전송하고 시청자가 이들

가운데 하나를 선택하여 보는 NVoD(Near Video on Demand) 서비스가 있다. 모든 사용자에게 채널을 하나씩 할당하여 원하는 방송을, 원하는 시간에 보여주는 RVoD 서비스가 가장 이상적인 VoD 서비스이지만, 통신 대역의 제약으로 실제 네트워크 환경에는 RVoD가 아닌 NVoD 방식으로 VoD 서비스가 구현된다.

NVoD 서비스의 구현 방법은 배칭(Batching) 기법과 패칭(Patching) 기법 두 가지가 있다. 배칭 기법은 비디오 데이터를 각 방식에 따라 나누고 배치한 세그먼트(데이터의 조각)들을 일괄적으로

반복 전송하는 기법을 말하며[1][2][3][4], 패칭 기법은 배칭 기법과 마찬가지로 세그먼트들을 일괄적으로 전송하되 사용자의 요청 이전 시점의 못 받은 데이터에 대해 별도의 채널을 통해 전송하는 덧대기 방식을 말한다[5][6]. 본 논문은 기존의 배칭 기법인 Fast Data broadcasting 기법을 패칭(덧대기)에 적용한 것에 그 주안점이 있다.

2. Fast Data Broadcasting

배칭 기법은 각 세그먼트의 크기를 다르게 하는 방법과 각 채널의 대역폭을 다르게 하는 방법으로 분류할 수 있는데 본 논문에서 응용된 Fast Data broadcasting 기법(이하 FD 기법)은 세그먼트의 크기도 같고, 채널의 대역폭도 같게 하여 전송하는 방법이다. FB 기법의 전송 방식은 다음과 같다[1].

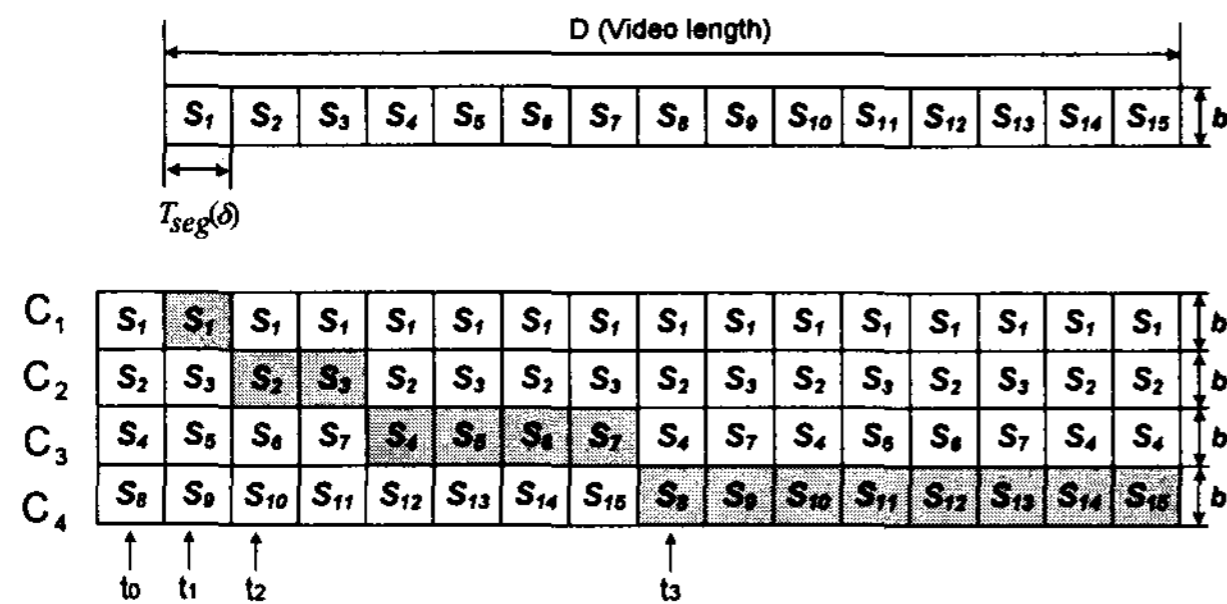


그림 1. Fast Data broadcasting의 세그먼트 분할과 채널 할당 ($N_{ch} = 4$ 일 경우)

2.1 서버의 동작

FB 기법은 비디오 데이터를 동일한 크기의 $2^{N_{ch}} - 1$ 개의 세그먼트들로 나누되, 첫 번째 채널, C_1 에는 첫 세그먼트를, 두 번째 채널, C_2 에는 다음 두 개 세그먼트를, 세 번째 채널, C_3 에는 다음 4개의 세그먼트를, 그리고 마지막 채널, $C_{N_{ch}}$ 에는 $2^{N_{ch}-1}$ 개의 세그먼트를 할당하여 반복 전송한다.

2.2 클라이언트의 동작

사용자는 자신이 요청한 비디오 데이터를 적

절히 받아 끊김 없는 재생을 위해 다음과 같은 방법으로 동작한다. 사용자가 그림 1의 t_0 시점에서 비디오를 요청하였을 경우, 현재 전송 중인 세그먼트가 모두 전송되기를 기다린 후, 다음 세그먼트의 시작 시점부터 데이터를 받기 시작한다. 따라서 일정 시간을 기다려야 하는 경우가 발생하는데 이때의 최대 대기 시간은 하나의 세그먼트가 지속되는 시간인

$$\delta = \frac{D}{2^{N_{ch}} - 1}$$

이 된다(여기서 D 는 프로그램의 재생시간(단위: 분), N_{ch} 는 총 채널 수). 그림 1은 한 프로그램당 4개의 채널을 사용할 경우($N_{ch} = 4$)의 FD 기법의 채널 할당을 나타낸다. 이 경우 전송되는 비디오가 재생시간(D)이 100분인 프로그램이라고 가정

하면, 사용자의 최대 대기 시간은 $\frac{100}{2^4 - 1}$ 인 6.7분

이 될 것이다.

사용자의 요청이 t_0 시점에 들어왔을 경우 C_1 을 통해 첫 번째 세그먼트, S_1 을, C_2 를 통해 두 번째, 세 번째 세그먼트, S_2, S_3 를, C_3 를 통해 네 번째 세그먼트부터 일곱 번째 세그먼트, $S_4 \sim S_7$ 를 받는다. 이 경우에서 S_1 의 전송이 끝나는 시점에서 다음 채널을 통해 S_2 를 받을 수 있고 S_3 의 전송이 끝나는 시점에서 다음 채널을 통해 S_4 를 받을 수 있기 때문에(그림 1의 음영 부분), 전송 데이터의 별도 저장 없이도 비디오의 연속 재생이 가능하다. 하지만 사용자의 요청이 t_1 에 들어왔을 경우, S_1 데이터를 받는 동안 다른 채널을 통해 S_2 를 저장하지 않으면, S_1 의 재생이 끝나는 시점에서 다음 세그먼트인 S_2 를 적절히 재생할 수 없게 된다. 사용자의 요청이 t_2 시점에 들어왔을 경우도, 다음 채널의 데이터를 미리 저장하지 않으면 S_3 의 재생이 끝나는 시점에서 S_4 를 연속 재생할 수 없을 것이다. 즉 위와 같은 경우에서 사용자는 프로그램의 연속 재생을 위해 비디오의 일정 부분을 저장할 필요가 있는데, 이때 저

장 데이터량을 버퍼 요구량이라고 한다. 그리고 버퍼 요구량이 최대가 되는 시점은 사용자의 요청이 t_3 시점에 들어온 경우이다. 이 경우는 8개의 세그먼트가 재생되는 시간 동안 15개의 세그먼트를 모두 받아야 하므로 이중 7개의 세그먼트를 저장하여야 한다. 이를 다시 말하면, $2^{N_{ch}}$ 번째 세그먼트가 전달되는 시점에 요청이 들어온 경우로 $2^{N_{ch}} - 1$ 개 세그먼트 데이터량을 저장하여야 연속 재생이 가능하게 된다. 따라서 최대 버퍼 요구량은

$$Z = (D - 2^{N_{ch}-1} \delta) b$$

이 된다(여기서, b 는 데이터 소모율(초당 재생되는 데이터량, 단위: bps)).

3. Fast Patching Scheme

패칭 기법은 VoD 서비스를 위해 할당된 대역을 스테저드(Staggered) 채널과 패취(Patched) 채널로 나누어 전송하되, 스테저드 채널에서는 전체 비디오 데이터를 스테저드 주기(T_s) 간격으로 반복 전송하고, 패취 채널에서는 사용자의 요청 이전에 데이터 부분을 유니캐스트(Unicast) 혹은 브로드캐스트(Broadcast)로 전달하는 방식을 말한다. 이때 패취를 위해 각 사용자별로 하나의 채널을 할당하여 유니캐스트로 전송하는 JAS(Joint and Stream)[5] 방식과 비디오 데이터의 앞부분을 브로드캐스트하되 채널별 전송 슬롯의 길이는 같지만 대역폭이 다른 SBB(Stream Bundling Broadcasting) 방식, 채널별 대역폭은 같지만 채널별 전송 슬롯이 다른 BMP(Batched Multicast Patching) 방식이 있다[6]. 본 논문에서 제안한 FP 방식은 비디오의 스테저드 주기 부분에 FD 방식을 적용하여 분할하여 패칭한다.

3.1 서버의 동작

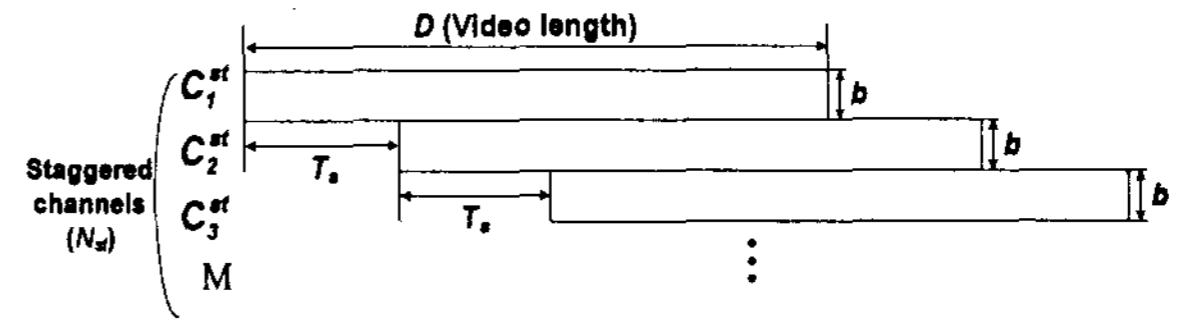


그림 2. FP 방식의 스테저드 채널

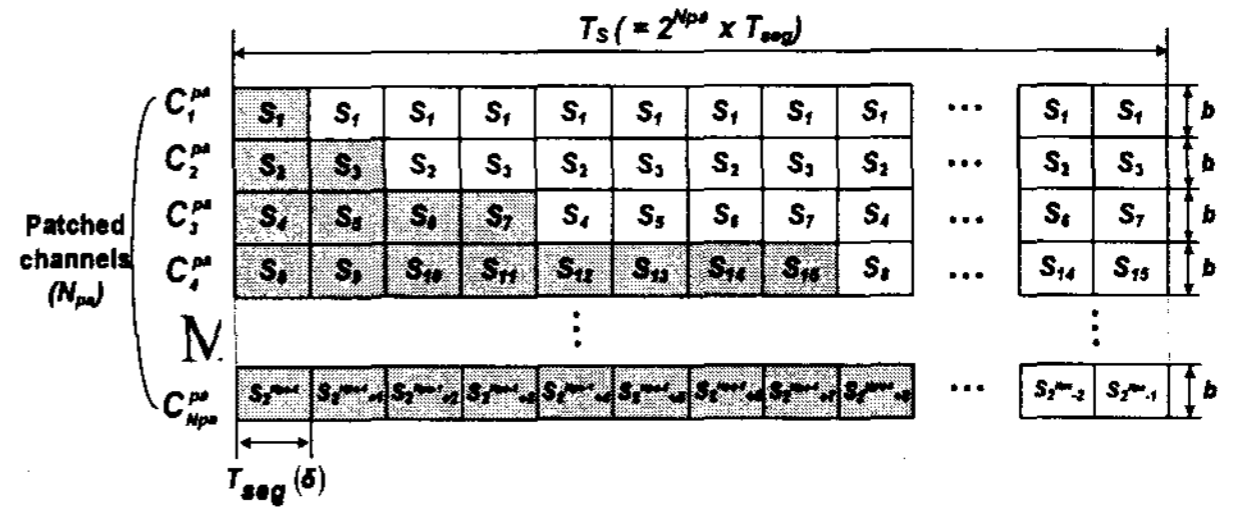


그림 3. FP 방식의 패취 채널과 세그먼트 분할

그림 2와 그림 3은 각각 FP 방식의 스테저드 채널과 패취 채널을 나타낸다. 그림 2의 FP 방식의 스테저드 채널은 일반적으로 알려져 있는 스테저드 방식으로 비디오를 전송한다. 이때 스테저드 주기(T_s)는 다음과 같이 결정된다.

$$T_s = \frac{D}{N_{st}}$$

여기서, N_{st} 는 스테저드 채널의 수를 의미한다.

그림 3과 같이, 패취 채널에 비디오 데이터의 앞 T_s 구간에 FB 방식을 적용하여 세그먼트 할당을 한다. 이때 2장에서 설명한 FB와 조금 다른 점은 $2^{N_{ch}} - 1$ 개의 세그먼트가 아닌 $2^{N_{ch}}$ 개 세그먼트로 분할한다는 것이다. 이때 하나 더 많아진 세그먼트로 인한 이득은 다음과 같다.

1. 짧아진 세그먼트 시간으로 인한 사용자 대기 시간이 감소한다.
2. 각 채널의 세그먼트들이 스테저드 구간 안에서 모두 2의 배수 만큼 반복 전송되므로, 공통의 반복 주기를 갖게 된다.

3.2 클라이언트의 동작

어떤 시점에 사용자의 요청이 발생했다고 가정하면 사용자의 장비는 요청이 발생한 시점 이후의 세그먼트는 스테거드 채널을 통해 받고, 이와 동시에 요청 발생 이전 시점의 세그먼트들은 패취 채널을 통해 받게 된다. FP 방식의 사용자 대기시간은 추가된 하나의 세그먼트로 인해

$$\delta = \frac{D}{2^{N_{ch}}}$$

이 된다. 패취 채널에서 데이터를 받는 동안 스테거드 채널을 통해서도 데이터를 받는 경우가 있으므로, FP 방식의 버퍼 요구량은 패취 채널을 통해 받아야 하는 데이터량이 된다. 버퍼 요구량이 최대가 되는 시점은 그림 4에서 보듯이 사용자의 요청이 스테거드 구간의 끝에서 2번째($2^{N_{ch}} - 1$) 세그먼트에 들어왔을 경우이다. 이때는 첫 번째 세그먼트부터 $2^{N_{ch}} - 1$ 번째 세그먼트까지 패취 채널을 통해 받아야 하므로 $2^{N_{ch}} - 1$ 개 세그먼트(그림 4의 음영)의 데이터량을 버퍼에 저장해야 한다.

$$Z = (T_s - \delta)b$$

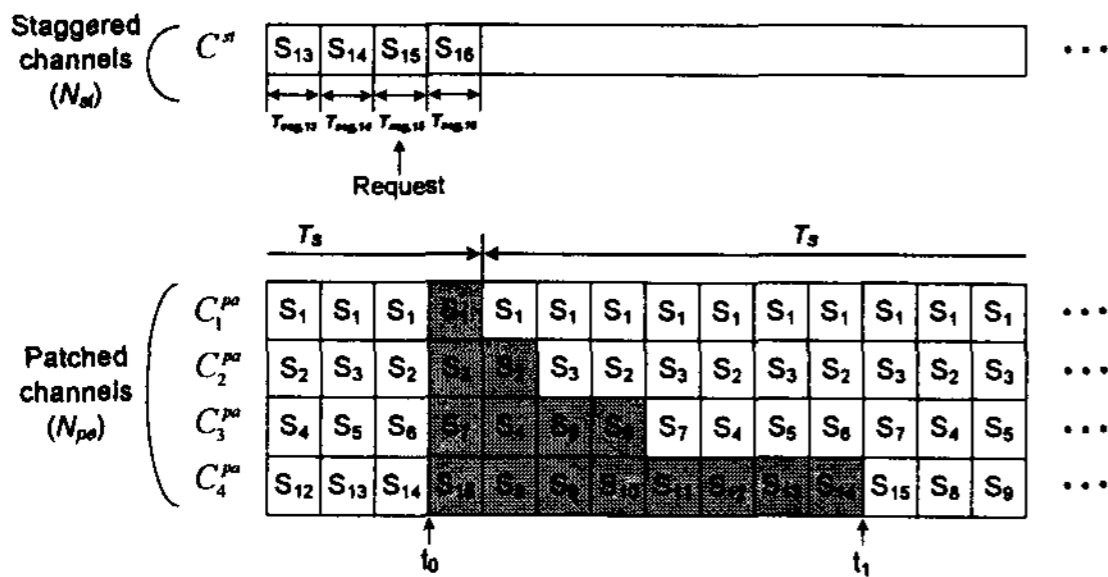


그림 4. FP 방식의 버퍼 요구량이 최대가 되는 시점

4. 성능의 비교

본 논문에서 제안된 방식의 성능을 검증하기 위해 패취 방식으로 먼저 제안되었던 JAS, SBB 방식을 사용자 대기 시간, 버퍼 요구량 측면에서 시뮬레이션 하였다. 실험 조건은 전송되는 프로그램의 길이를 100분, 스테거드 채널 수를 5개, 그리

고 사용자의 요청 도착율(request arrival/min)을 2로 설정하였다($D=100(\text{min})$, $N_{st}=5(T_s=20)$, $\lambda=2$).

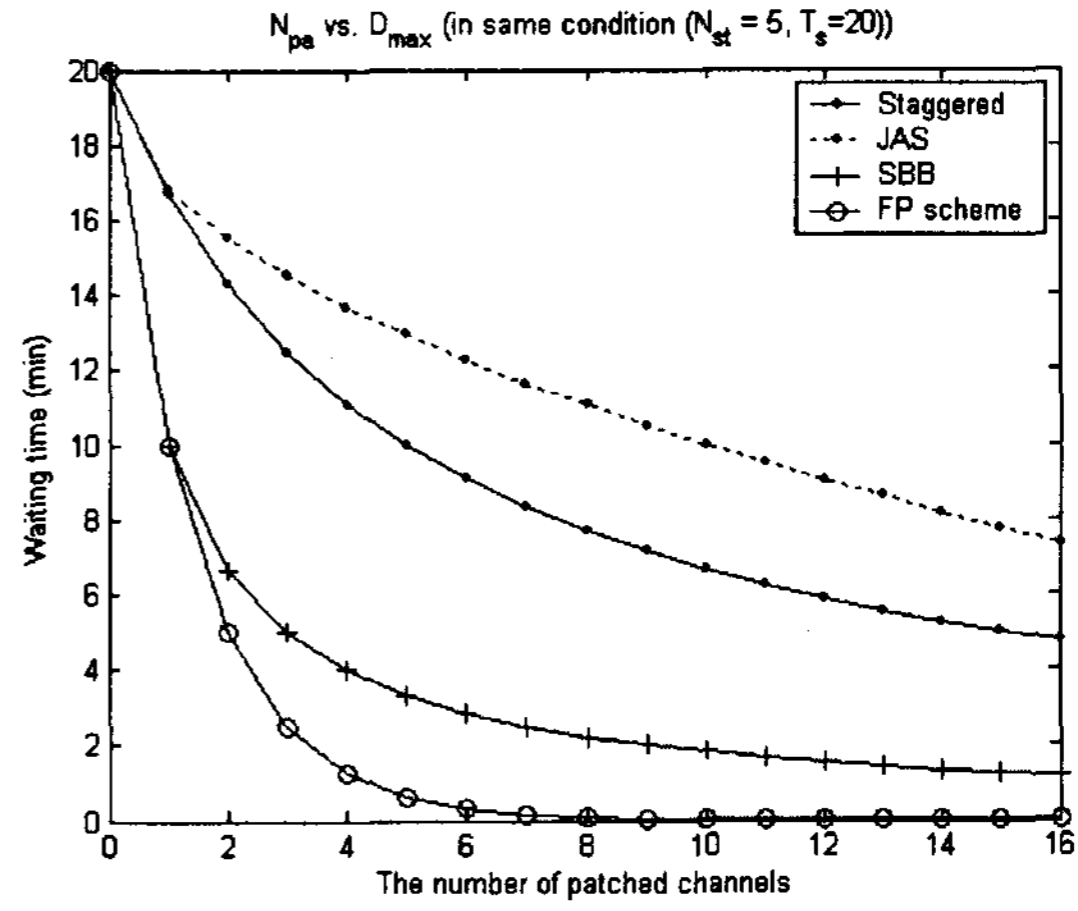


그림 5. 각 방식별 사용자 대기 시간의 비교

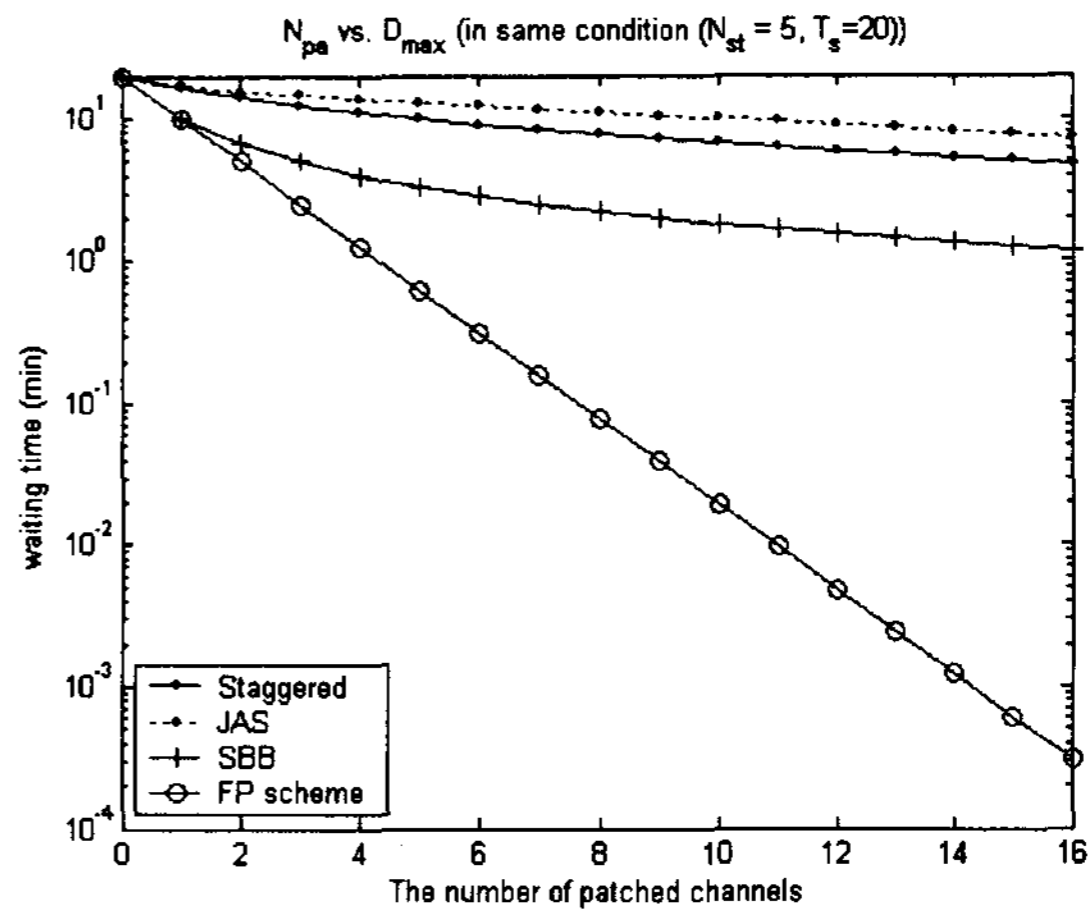


그림 6. 각 방식별 사용자 대기 시간의 비교 (로그스케일)

그림 5와 6은 각 방식별 할당 채널 수에 따른 사용자의 대기 시간을 비교한 결과이다. 스테거드 방식, JAS, SBB 방식 모두 대역 증가함에 따라 대기 시간이 선형적으로 감소하는 경향을 보이지만 제안된 FP 방식은 대역 증가에 따라 지수적으로 감소함을 알 수 있다. 그림 5를 통해 FP 방식의 경우 패취 채널이 5개 할당되는 경우 사용자

의 대기 시간은 1분 미만임을 알 수 있다.

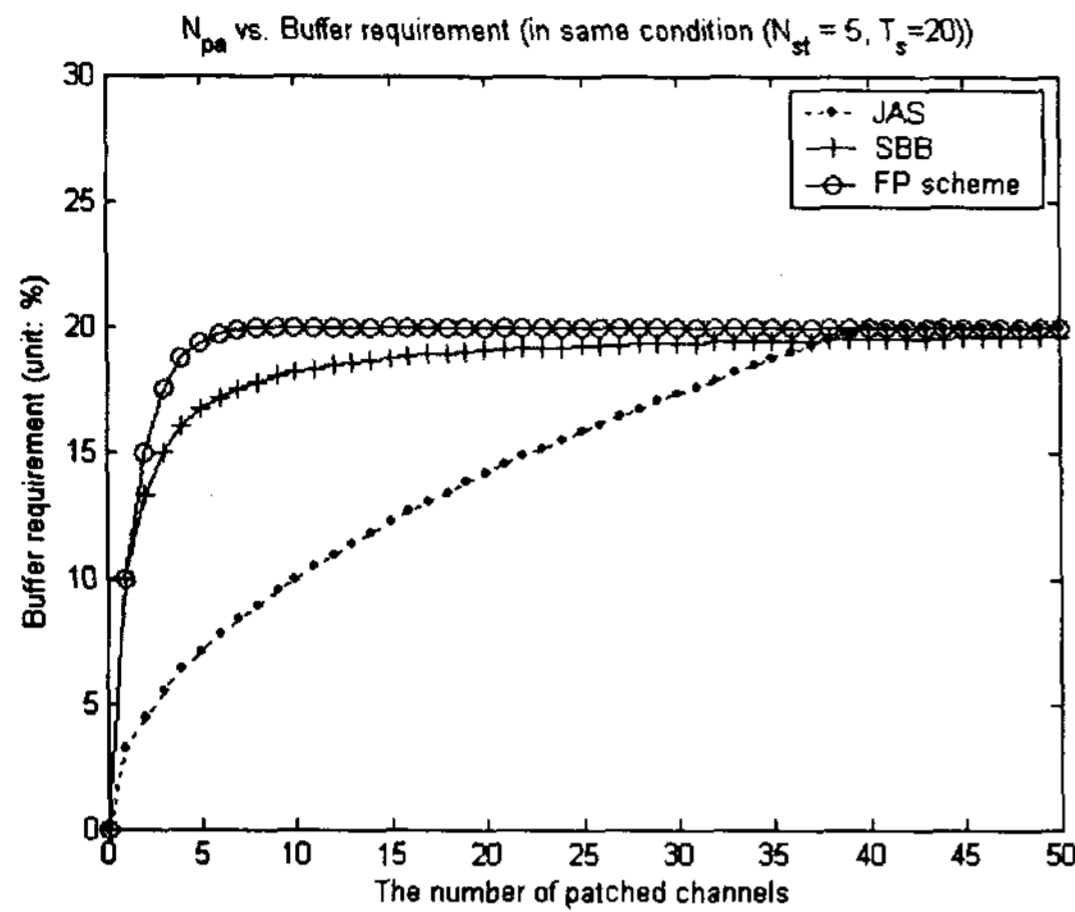


그림 7. 각 방식별 버퍼 요구량의 비교

그림 7은 각 방식별 대역폭에 따른 버퍼 요구량을 시뮬레이션 한 결과이다. 4장에서 언급한 바와 같이 패칭 방식의 버퍼 요구량은 $(T_s - \delta)$ 시간 동안의 데이터량인데, 채널 수가 증가할수록 각 방식 모두 대기 시간이 0으로 수렴하여 버퍼 요구량은 T_s 값인 20으로 수렴함을 확인할 수 있다.

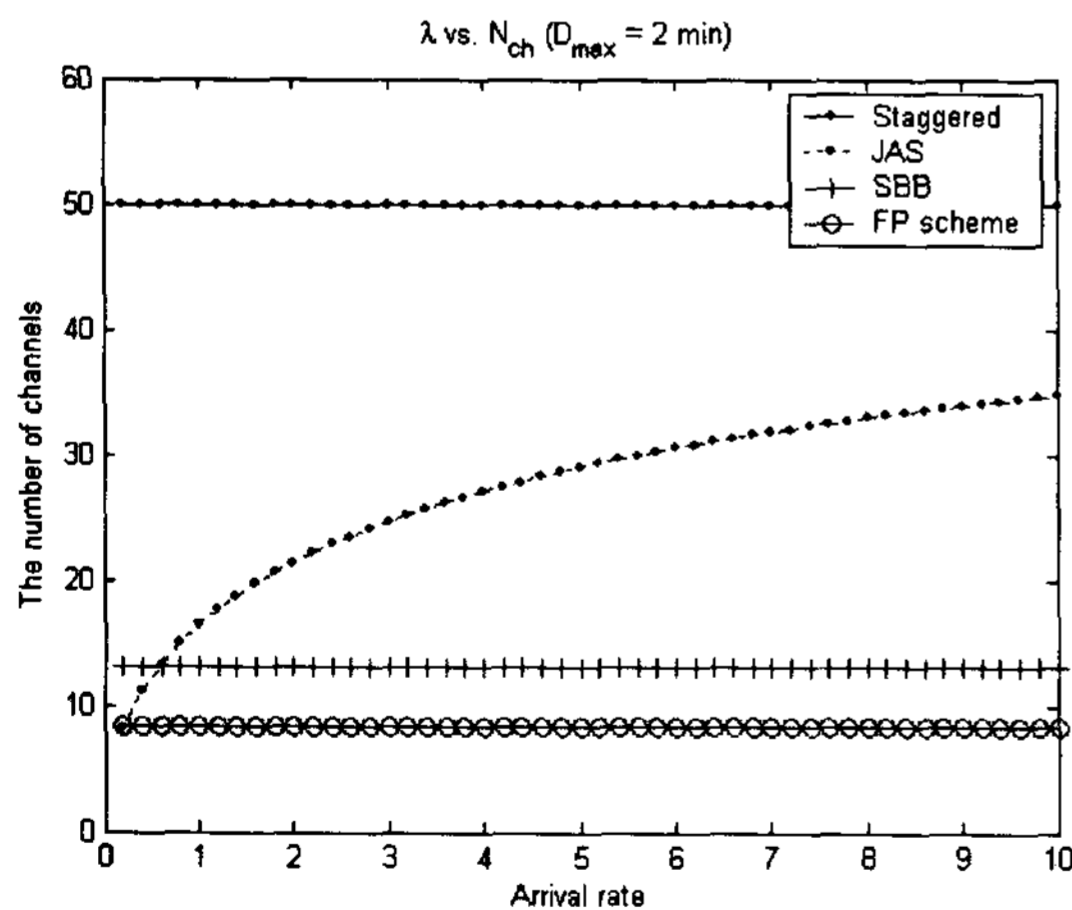


그림 8. 요청 도착율에 따른 필요 채널 수의 변화 ($D_{max} = 2min$)

그림 8은 사용자의 요청 도착율에 따른 필요 채널

수의 변화를 나타낸 그래프이다. 요청 도착율은 사용자의 요구 발생 빈도를 나타내는 수치이다. 그래프에서 보듯 스테거드, SBB, FP 방식은 사용자 수에 상관없이 대기 시간을 2분으로 설정하여 시스템을 설계할 경우, 필요 채널의 수는 일정하지만 JAS 방식의 경우는 사용자의 요청이 많아지면 필요한 채널 수도 증가함을 알 수 있다. 이는 JAS 방식의 유니캐스트 전송에 기인한다.

4. 결론

VoD 시스템 설계에 있어 가장 이상적인 경우는 요청하는 모든 사용자에게 하나씩의 채널을 할당하여, 원하는 프로그램을 원하는 시간에 전송하는 것이다. 하지만 대역 제한된 통신 환경에서의 VoD 서비스 구현은 프로그램별로 제한된 대역을 할당하여 이루어진다. 이로 인해 사용자 측에서는 원하는 프로그램을 요청하더라도 일정 시간을 기다리고, 전송된 데이터의 일부를 저장 후 재생해야 하는 번거로움과 기술적인 문제를 해결해야만 한다. 이를 극복하고자 VoD 배칭 기법 및 패칭 기법의 여러 방식들이 제안되어 왔다. 본 논문에서는 기존의 패칭 방식에 사용자의 요청 이전의 데이터를 Fast Data broadcasting 방식을 적용하여 전달함으로써 사용자의 대기 시간을 크게 줄일 수 있는 방안을 제시하였다. 제안된 방식을 적용한 시스템의 대기 시간은 대역폭 증가에 따라 선형적 감소를 보이는 여타의 패칭 방법과 달리, 지수적으로 감소하는 우수한 결과를 보여준다.

[참고문헌]

- [1] Li-Shen Juhn and Li-Ming Tseng, "Fast Data Broadcasting and Receiving Scheme for Popular Video Service," *IEEE Transactions on Broadcasting*, Vol.44, pp.100-105, Mar.1998.
- [2] L.S.Juhn and L. M. Tseng, "Harmonic Broadcasting for Video-on-Demand Service," *IEEE Trans. Broadcast*, Vol.43, no.3, pp.268- 271, Sep. 1997.

- [3] L-S. Juhn and L-M Tseng, "Staircase Data Broadcasting and Receiving scheme for Hot Video Service," *IEEE Trans, Consumer Electronics*, vol. 43, no. 4, pp. 1110-1117, Nov. 1997.
- [4] S. Viswanathan, T. Imielinski, "Pyramid Broadcasting for Video on Demand Service," *SPIE*, Vol. 2417, pp. 66-77, 1995.
- [5] S. H. Gary Chan and S. H. Ivan Yeung, "Client Buffer Techniques for Scalable Video Broadcasting over Broadband Networks with Low User Delay," *IEEE Transactions on Broadcasting*, Vol. 48, No. 1, Mar. 2002.
- [6] Salahuddin A. Azad, Manzur Murshed, and Laurence S. Dooley, "A Novel Batched Multicast Patching Scheme for Video Broadcasting with Low User Delay," *Signal Processing and Information Technology*, 2003, pp. 339-342, Dec. 2003.