

디지털 텔레비전 환경에서의 문맥인지 캐싱 정책

최지혁, 이해원, 김수연, 최양희
서울대학교 컴퓨터공학과

Context-aware Caching Strategy in Digital Television Environment

Jihyuk Choi, Haewon Lee, Sooyeon Kim and Yanghee Choi
Dept. of Computer Engineering, Seoul National Univ.

요 약

컴퓨터통신 분야 및 개인이동통신 분야 등에서 먼저 도입된 멀티미디어 데이터의 사용이 큰 호응을 얻으면서, 최근 디지털 텔레비전 환경에서 멀티미디어 데이터 방송서비스를 시행하려는 노력들이 이루어지고 있다. 이는 다수의 사용자를 겨냥한 방송서비스로 데이터가 개별 사용자의 요구에 의해서가 아닌 정해진 방송 순서에 의해 제공된다. 따라서 사용자들은 원하는 정보를 수신 할 때까지 대기하게된다. 이러한 응답시간 (response time)을 감소시키기 위해서, 문맥인지 캐싱 (context-aware caching) 정책과 캐쉬 점유 우선순위 결정에 핵심이 되는 두 노드사이의 거리 산출을 위한 방법을 제안한다. 또한 선입선출 (first-in-first-out) 정책과의 비교 모의실험 (simulation)을 통해, 제안한 문맥인지 캐싱 정책이 응답시간 감소 효과가 좋음을 확인하고, 이를 분석한다.

1. 서론

인터넷과 같은 컴퓨터통신 분야 및 개인이동통신 분야 등에서 먼저 도입된 멀티미디어 데이터의 사용이 큰 호응을 얻 으면서, 텔레비전 환경에서도 멀티미디어 데이터 방송서비스 를 시행하려는 노력들이 이루어지고 있다 [1], [2]. 이는 다수 의 사용자를 대상으로 하는 방송서비스로 정해진 방송 순서 에 의해 데이터를 전송하게 되므로, 임의의 시각부터 수신을 시작하게 되는 사용자들은 원하는 데이터를 수신할 때까지 긴 응답시간 (response time)을 겪게된다. 그러나 기존의 전 화나 텔레비전에 익숙한 사용자들은 긴 응답시간에 의한 지 연에 민감하리라 예상되는데, 이 문제의 해결방법 중의 하나 가 사용자 단말에서의 캐싱 (caching)이다.

캐싱에 의해 단말에 저장된 데이터는 사용자의 요구 즉시 제공가능하기에 응답시간을 줄일 수 있다. 일반적인 경우 단 말의 저장 용량의 크기는 단말의 가격조건을 고려하여 유한 하게 결정되므로, 방송되는 데이터 모두를 저장하는 것은 불 가능하다 따라서 캐싱에 의해 얻을 수 있는 성능 개선을 극대화하기 위한, 반입할 데이터와 교체할 데이터를 결정하는 적절한 캐싱 정책의 수립이 필요하다.

효율적인 데이터 방송서비스를 위한 여러 노력들이 이루어 져 왔다. Acharya 등은 Broadcast Disk 연구를 통해 위성망,

케이블 텔레비전망, VDSL (Very-high-speed Digital Subscriber Line) 등의 새로운 망 환경은 하향 채널이 상향 채널에 비해 상당히 큰 대역폭을 가지는 비대칭적 통신 환경 (asymmetric communication environment)으로 규정하고, 이 런 환경에서의 방송 및 캐싱 정책을 제시하였다 [3]. 또한, Vaidya 등은 수학적 접근에 의한 방송 스케줄 생성 알고리 즘을 제안하였다 [4].

하지만 기존의 연구들은 고정된 내용과 고정된 크기의 데 이터 아이템들이 반복적으로 방송된다고 가정하고 있어서 [3], 텔레비전 프로그램 내용과 관련된 다양한 크기의 다수의 정보가 전송되는 디지털 텔레비전의 방송 환경과 많은 차이 가 있다. 또한 단말 사용자의 응답시간을 줄일 수 있는 선반 입 (prefetching)이나 캐싱에 대한 연구는 상대적으로 덜 이루 어졌다.

본 논문에서는 이상과 같이 디지털 텔레비전 환경에 멀티 미디어 데이터 방송을 도입할 때 생길 수 있는 응답시간을 감소시키기 위한 캐싱 정책을 제안한다.

2. 데이터 방송 환경

기존의 텔레비전 방송과 연계하여 부가정보 방송을 실시할 경우 방송 채널에는 멀티미디어 데이터의 흐름이 계속되고 사용자는 단말을 켜는 임의의 순간부터 수신하게 된다. 따라

서 텔레비전 프로그램이 진행되는 일정 시간 동안 제공하고자 하는 데이터들을 단 일회 방송하는 것은 불충분하다. 즉, 반복적으로 전송해야 하는데, 이 때 모든 데이터를 같은 주기로 전송하기보다는 데이터가 접근될 확률과 크기에 따라 주기를 달리 하는 것이 평균 응답시간을 감소시키는 데 효과적이며, 개별 데이터의 방송 간격은 일정한 것이 바람직하다 [3], [4].

텔레비전 환경에서의 멀티미디어 데이터 방송서비스들에서는 [1], [2] 사용자들이 원하는 멀티미디어 데이터를 식별자(ID, Identifier)에 의해 요청하는 것이 아니라, 그림 1에서와 같이 방송 제공사 측에서 의도한 접근 흐름 중 한 경로를 따라가게 된다. 사용자가 접근 가능한 장면을 하나의 노드로 표현하고 그 장면에서 일회 분기함으로써 바로 접근될 수 있는 장면들을 그 노드의 자식 노드로 표현할 때, 가능한 접근 흐름은 그림 1에서와 같은 접근흐름트리 [5]로 표현될 수 있다.

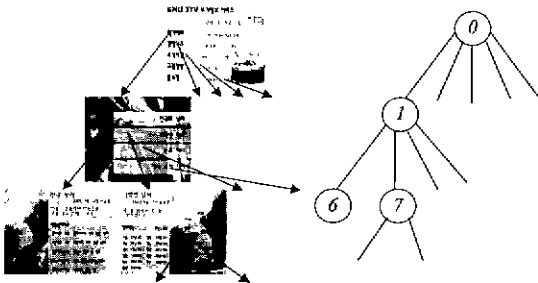


그림 1. 멀티미디어 데이터 방송서비스와 접근흐름트리

위와 같은 접근 흐름에 따른 경로 이동 외에 기존의 컴퓨터 환경에서의 웹브라우저에서처럼 루트 (root)로의 직접 이동이 가능하다고 가정한다. 또한, 데이터들의 방송 순서는 그림 2와 같이 이차원 라운드로빈 (two-dimensional round-robin) 알고리즘 [5]에 의해 결정된다고 가정한다.

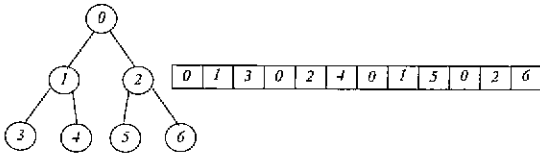


그림 2. 방송 스케줄 예 (높이-2 접근흐름트리의 경우)

3. 문맥인지 (context-aware) 캐싱 정책

3.1. 문맥인지 캐싱

사용자 단말로의 데이터 공급 경로는 방송 채널로 제한되기 때문에 데이터 접근 요구 시 단말의 저장장치에 데이터가 없을 경우 사용자는 해당 데이터의 다음 방송 시각까지 기다려야 한다 따라서 캐쉬의 접근 실패 비용은 응답시간이라고 볼 수 있다. 평균적인 캐쉬 접근 실패 비용을 감소시키기 위해서는 새로운 페이지를 반입하기 위해서 캐쉬에 존재하는 페이지를 추출해야 할 경우 사용자가 현재 서비스를 받고 있

는 위치에서 방문할 확률이 가장 낮은 페이지를 교체 대상 (replacement victim)으로 선택해야 한다.

고전적인 의미의 캐싱은 요구되었던 페이지를 빠른 메모리에 저장해 두었다가 재차 요구가 발생할 경우 빠른 시간 내에 접근할 수 있도록 하는 방법으로서 어떤 페이지에 대한 최초 접근 시에는 필수적 (compulsory)으로 캐쉬 접근 실패가 발생하게 된다 [6]. 그런데 디지털 텔레비전 환경에서의 데이터 방송서비스에서는 같은 페이지에 대한 접근이 여러 번 일어나기보다는 한번에 그칠 가능성이 높으므로 필수적 접근 실패를 줄이는 것이 매우 중요하다 필수적인 캐쉬 접근 실패를 줄이기 위해서 접근 가능한 데이터를 캐쉬에 선반입하는 정책이 도입될 수 있다. 사용자가 현재 서비스를 받고 있는 위치에서 가까운 미래에 접근이 예상되는 데이터에 대해 선반입을 수행한다.

위의 캐쉬 교체 기법과 선반입을 결합한 캐싱 정책이 문맥인지 캐싱 (CAC, Context-Aware Caching)이다. CAC에서 사용자가 현재 서비스를 받고 있는 위치에서 가까운 미래에 방문할 노드에 대한 확률의 결정, 즉 접근예상은 접근흐름트리에서의 두 노드사이의 거리에 의해 이루어진다 새로운 방송 데이터를 수신하였을 때, CAC는 아래와 같이 동작한다.

```

if (새로 도착한 아이템이 캐쉬에 반입되어 있다) then
    캐쉬에 아무런 변화 없음;
else if (캐쉬에 새 아이템을 반입할 공간이 있다) then
    새 아이템을 반입;
else if ((캐쉬 내 아이템 중의 최대거리) > (새 아이템까지의 거리)) then (
    while (새 아이템을 반입할 공간이 부족하다)
        캐쉬 내 최대거리 아이템을 추출;
    새 아이템을 반입;
)
    
```

3.2. 두 노드사이의 거리 산출

3.1절에서 언급한 바와 같이 CAC에서 캐쉬 점유의 우선순위 (priority)는 접근흐름트리에서의 두 노드사이의 거리에 의해 결정된다. 하지만, 텔레비전 방송 프로그램에 따라 변하는 접근흐름트리의 전체 구조를 사용자 단말이 전부 파악하고 있는 것은 비효율적이라 할 수 있다. 따라서 접근흐름트리의 전체구조를 알지 못하는 상황에서 두 노드사이의 거리 산출이 요구되는데, 이는 방송 데이터들의 식별자를 그림 3과 같이 적절히 준다면 다음과 가능하다.

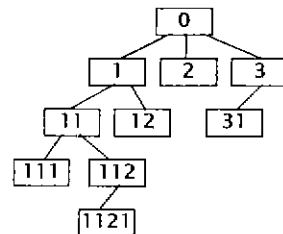


그림 3. 방송 데이터 식별자

- 1) 현재 서비스중인 노드가 0번인 경우 (예 0과 112의 거리)

$$(\text{두 노드사이의거리}) = \lfloor \log_{10} id \rfloor + 1$$
- 2) 현재 서비스중인 노드가 0번이 아닌 경우
 - 두 노드가 위치하는 서브트리에 따라 2가지로 나뉜다.
 - 다른 서브트리에 있는 경우 (예 1121과 31의 거리)
 - : 0번 노드로 직접이동 후 가는 경우가 최단거리가 된다.
 - $$(\text{두 노드사이의거리}) = \lfloor \log_{10} id \rfloor + 2$$
 - 같은 서브트리에 있는 경우
 - (예. 1121과 1의 거리, 1121과 111의 거리)

```

len1 = (int)floor(log10(id1)),
len2 = (int)floor(log10(id2)),

for(len1>=0 && len2>=0, len1--, len2--)
    if(id1/(int)pow(10, len1) != id2/(int)pow(10, len2))
        break;

dis1 = len1 + len2 + 2;
    
```

: 위에서 구한 경로이동만으로 이루어진 경우와 0번 노드로 직접이동 후 가는 경우의 거리의 최소값이 최단거리가 된다.

$$(\text{두 노드사이의거리}) = \min(\text{dis1}, \lfloor \log_{10} id \rfloor + 2)$$

4. 실험

문맥인지 캐싱 (CAC) 정책의 응답시간 감소 효과를 알아보기 위하여, 선입선출 (FIFO, First-In-First-Out) 방법과의 비교를 위한 모의실험 (simulation)을 수행하였다. 선입선출 방법은 비교적 간단하면서도 최근에 방송된 내용을 캐쉬에 보관한다는 점에서 설득력을 가지는 캐쉬 교체 정책이다.

메이커 방송을 위한 방송 대역폭은 2Mbps로 가정하였고, 방송 스케줄은 최대 차수 5, 최대 깊이 5인 접근흐름트리를 무작위로 생성하여 이차원 라운드로빈 알고리즘에 의해 결정하였다. 위와 같이 생성된 접근흐름트리는 115개의 노드를 갖는데, 각 노드의 사이즈는 평균 86.5KB, 표준편차 10KB의 정규분포 (normal distribution)를 따른다고 가정하였다.

2장에서 언급한 바와 같이 사용자들의 방송서비스 사용은 접근흐름트리에서의 이동으로 나타내어진다. 한 노드에서 각각의 자식 노드들로 경로 이동하는 확률은 서로 같으며, 부모 노드나 루트 노드로 돌아 갈 확률은 자식 노드들로 갈 확률의 반이며, 한 노드에 머무는 시간은 평균 10초의 지수분포 (exponential distribution)를 따른다고 가정하였다.

그림 4는 FIFO, CAC 두 캐싱 정책에 대하여 캐쉬 크기와 응답시간의 관계를 나타낸 것이다. 두 경우 모두 캐쉬 크기가 증가함에 따라 평균 응답시간이 감소하고 있으나, CAC 경우가 감소 효과가 월등히 좋을 수 있다. 그 이유는 그림 5에서 확인 할 수 있다. 그림 5는 접근흐름트리의 노드 레벨에 따른 접근 성공률 (hit ratio)을 나타낸 것이다. FIFO의 경우 단순히 시간적 순서에 의해 캐싱이 이루어지므로, 자주 방송

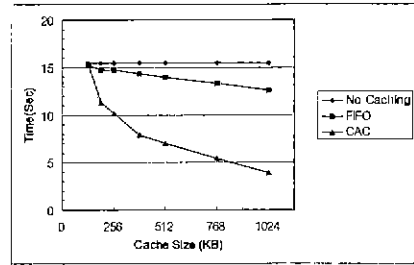


그림 4. 캐쉬 크기에 따른 응답시간

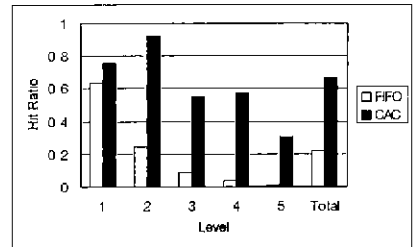


그림 5. 노드 레벨에 따른 접근 성공률 (캐쉬 크기가 512KB인 경우)

되지 않는 큰 레벨의 노드들의 경우 거의 캐싱이 이루어지지 않는 반면, CAC의 경우는 어느 정도 캐싱이 이루어지고 있다. 레벨이 큰 노드들의 경우는 자주 방송되지 않기 때문에, 캐싱이 안 이루어진 경우 응답시간이 매우 길어지게 되므로, 레벨이 큰 노드들의 접근 성공률이 전체적인 응답시간에 미치는 영향이 크게 된다. 따라서 CAC가 FIFO에 비하여 응답시간 감소에 있어 좋은 효과를 나타내는 것이다.

5. 결론

메이커가 개별 사용자의 요구에 의해서가 아니라 방송 순서에 의해 제공될 때, 사용자들은 원하는 정보를 수신 할 때까지 대기하게 된다. 이러한 응답시간을 줄이기 위해서 문맥인지 캐싱 (CAC) 정책과 CAC에서 캐쉬 점유 우선순위 결정에 핵심이 되는 두 노드사이의 거리 산출을 위한 방법을 제안하였다. 또한 모의실험을 통해, CAC가 응답시간 감소에 좋은 성능을 나타냄을 확인하였다.

참고문헌

- [1] OpenTV 홈페이지, <http://www.opentv.com>
- [2] Canal+ 홈페이지, <http://www.cplus.fr>
- [3] S. Acharya, M. Franklin and S. Zdonik, "Dissemination-based Data Delivery using Broadcast Disks," IEEE Personal Communications, vol. 2, no. 6, pp. 50-60, Dec. 1995.
- [4] N. H. Vaidya and S. Hameed, "Data Broadcast in Asymmetric Wireless Environment," Workshop on Satellite-based Information Services (WOSBIS), New York, Nov. 1996
- [5] 김수연, 최양희, "데이터 방송에서의 효율적인 캐싱 정책," 한국정보과학회 논문지 제출, <http://mmlab.snu.ac.kr/~sykm/broadcast/jkiss98.doc>
- [6] J. L. Hennessy and D. A. Patterson, Computer Architecture : a Quantitative Approach, Morgan Kaufmann Publishers, 1996.