브로드캐스트 환경에서 데이터 접근 빈도를 고려한 효과적인 인덱스 배치 기법

박지현° 박광진 강상원 김종완 임석진 황종선
고려대학교
{hyun°, kjpark, swkang, wany, seokjin, hwang}@disys.korea.ac.kr

An Efficient Index Allocation Scheme Considering Data Access Frequencies in Mobile Broadcast Environments

JieHyun Park° KwangJin Park Sang-Won Kang Jongwan Kim SeokJin Im Chong-Sun Hwang
Department of Computer Science and Engineering, Korea University

요 약

이동 컴퓨팅 환경에서 통신 네트워크가 갖는 무선 채널 대역폭의 협소함과 이동 단말기의 에너지 제약으로 인해, 다수의 이동 클라이언트들에게 데이터를 전달할 때에는 다수의 클라이언트들의 동시 데이터 접근을 지원하는 브로드캐스트 방법을 사용함으로써 제약점들을 보완할 수 있다.

본 논문에서는 클라이언트의 에너지와 데이터에 대한 접근시간(access time)의 효율을 높이기 위해 데이터의 접근빈도(access frequency)를 고려한 브로드캐스트 방법과 브로드캐스트 인덱스를 추가하는 방법을 함께 반영한 DAF(Data Access Frequencies)브로드캐스팅 기법을 제안한다. DAF브로드캐스팅 기법은 데이터의 접근빈도를 고려한 인덱스를 교차하여 추가함으로써 접근빈도가 높은 데이터를 원하는 다수의 사용자에 대한 접근시간을 줄임으로써 모든 사용자의 평균 접근시간을 줄이는데 목적이 있다. 수학적 분석을 통해 DAF브로드캐스팅 기법을 평가하고 기존의 브로드캐스트 방법과 DAF브로드캐스팅 기법의 성능을 비교 분석한다.

1. 서 론

이동 통신 네트워크(mobile communication networks)의 발전으로 무선 채널(wireless channel)을 통해 클라이언트(client)들은 언제 어디서든 원하는 정보를 얻을 수 있게 되었다. 클라이언트들의 자유로운 이동성은 이동 컴퓨팅 환경(mobile computing environments)의 장점이지만 이로 인해 이동 단말기의 에너지 제한, 무선 통신 대역폭의 협소함, 높은 에러율, 잦은 접속 단절 등의 여러 가지 제약 조건이 발생한다[1]. 이러한 제약점을 가지는 이동 컴퓨팅 환경에서, 데이터 브로드캐스트(broadcast) 방법을 사용하면 다수의 이동 클라이언트의 질의처리를 개별적으로 하지 않으므로 무선 채널의 협소한 대역폭, 단말기의 제한된 에너지와 같은 제약점을 보완할 수 있다[7,9]. 브로드캐스트 환경은 서버가 클라이언트의 데이터 요구를 예측하고, 그 예측 결과에 따라 무선 통신망을 이용하여 데이터를 클라이언트들에게 전송하면 클라이언트들은 필요한 데이터를 선택적으로 수신한다.

이러한 방법에서는 모든 데이터에 대한 접근시간(access time)이 동일하지만 일반적으로 클라이언트들의 데이터 접근빈도는 모든 데이터에 대해 동일하지 않으므로 접근빈도를 고려한 브로드캐스트 방법이 필요하다. 데이터의 접근빈도만을 반영하여 전달할 경우, 클라이언트의 데이터에 대한 평균 접근시간은 단축되지만 인덱스가 없기 때문에 클라이언트가 원하는 데이터를 얻을 때까지 오랜 시간 동안 무선 채널을 듣고 있어야 하므로 많은 에너지를 소비하게 된다. 접근빈도를 고려하지 않은 인덱스를 추가하는 브로드캐스트의 경우에는 에너지 소비면에서는 효율적이지만 클라이언트가 원하는 데이터를 얻을 때까지 추가한 인덱스에 따른 지연이 발생한다.

본 논문에서는 데이터 접근빈도(access frequency)와 인덱스(index)를 모두 고려하여 데이터를 브로드캐스트 하는 방법인 DAF(Data Access Frequencies)브로드캐스팅 기법을 제안한다. DAF브로드캐스팅 기법은 일부 전체 데이터에 대한 인덱스 대신, 데이터의 접근빈도를 고려한 인덱스를 추가함으로써 다수의 사용자에 대한 평균 접근시간을 줄이는데 목적이 있다.

본 논문의 구성은 다음과 같다. 2장에서는 기존의 연구들에 대해 알아보고 3장에서 기존 연구의 문제점을 지적하고 새로운 데이터 브로드캐스트 방법을 제안한다. 4장에서는 수식을 통한 분석으로 제안된 방법의 성능을 평가하고 기존의 방법들과 비교하고, 5장에서 결론을 맺는다.

2. 관련 연구

브로드캐스트의 가장 큰 장점은 확장성(scalability)으로 이동 클라이언트의 수가 증가하여도 그에 따른 추가 통신비용이 거의 발생하지 않는다는 점이다[5]. 즉, 브로드캐스트는 일정량의 채널을 다수의 사용자가 공유하여 사용하기 때문에 제한적인 대역폭의 활용측면에서 매우 효율적이다. 또한 무선 통신 네트워크 환경에서 데이터를 다운링크 하는 것은 요청을 업링크 하는 것보다 더 적은 에너지를 필요로 하기 때문에 브로드캐스트를 사용할 경우 이동 단말기의 에너지 사용도 줄일 수 있다[8]. 데이터 브로드캐스트 구성방법에서 기존의 연구들은 크게 두 가지로 분류할 수 있다. 하나는 클라이언트들의 데이터 접근빈도를 고려해서 데이터들의 전송 횟수를 달리하여 평균 접근시간을 단축시킨 브로드캐스트 방법[2]이고 또 다른 연구 방향은 서버가 데이터와 데이터에 대한 인덱스를 함께 브로드캐스트 하고 클라이언트는 인덱스를 이용하여 원하는 데이터가 브로드캐스트 되는 시간동안에만 채널을 듣는 선택적 데이터 수신(selective tuning)을 가능하게 하여 제한된 에너지의 소비를 줄일 수 있도록 하는 것이다[3].

2.1 접근빈도를 고려한 브로드캐스트

데이터의 접근빈도 차이를 고려하여 데이터들의 전송 빈도를 달리함으로써, 클라이언트들이 데이터를 얻기 위해 기다려야 하는 시간을 차별하는 브로드캐스트 방법에서 데이터는 클라이언트의 상대적인 접근빈도에 따라 핫 데이터(hot data)와 콜드 데이터(cold data)로 분류된다[2]. 클라이언트들이 자주 참조하는 핫 데이터는 다른 데이터들보다 전송 횟수를 많이 하고, 그렇지 않은 콜드 데이터는 전송 횟수를 적게 하여 그림 1의

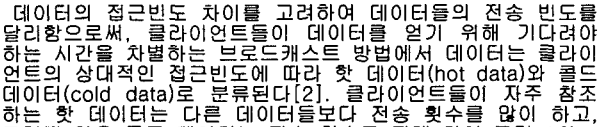

그림 1 . 데이터의 접근빈도를 고려한 브로드캐스트

형태로 데이터를 브로드캐스트 할 수 있다. 핫 데이터를 먼저 전송한 후에 콜드 데이터를 전송하는 방식은 핫 데이터가 콜드 데이터보다 자주 전송됨으로써 접근빈도는 만족시키지만, 핫

데이터를 얻기 위해 기다려야 하는 시간이 일정하지 않기 때문에 그림 1과 같이 전송 빈도가 다른 데이터들을 서로 교차시켜 브로드캐스트 하면 클라이언트가 핫 데이터를 받기 위해 기다려야 하는 접근시간을 일정하게 조절할 수 있다.

2.2 인덱스를 이용한 브로드캐스트

이동 단말기는 에너지가 제한되어 있기 때문에 에너지의 소비를 줄이기 위해 선택적으로 데이터 수신이 가능하도록 데이터들의 브로드캐스트 주기에 인덱스를 배치하고 중복시키는 방법이 있다[3,6]. 이러한 브로드캐스트 방법에서 구성요소는 그림

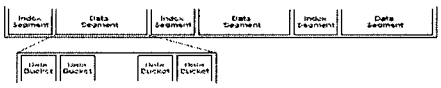

그림 2. 브로드캐스트 구성요소

2와 같다. 브로드캐스트 채널을 통해서 전송되는 가장 작은 논리적 단위를 버켓(bucket)이라고 한다. 인덱스 세그먼트는 데이터들의 키 값들을 가지고 있고, 모든 데이터 버켓은 다음 인덱스에 대한 포인터(pointer)와 다음 브로드캐스트 주기의 오프셋(offset)을 가지고 있다.

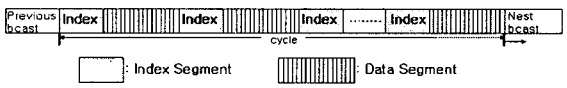

그림 3. (1,m)인덱싱

그림 3은 클라이언트가 반복되는 인덱스를 통해 원하는 데이터에 대한 키 값을 얻어 데이터가 브로드캐스트 되는 시간을 파악한 후, 원하는 데이터가 브로드캐스트 되는 시간에 깨어나 선택적으로 원하는 데이터를 수신할 수 있는 (1,m)인덱싱을 나타낸다. 이와 같이 브로드캐스트 되는 데이터 사이에 전체 데이터에 대한 인덱스를 배치한 방법은 클라이언트의 에너지 효율을 높인다.

3. DAF: Data Access Frequencies 브로드캐스팅 기법

이 장에서는 이동 컴퓨팅 환경에서 데이터 브로드캐스트 주기에 전체 데이터에 대한 인덱스뿐만 아니라 데이터 접근빈도를 고려한 인덱스를 교차하여 배치하는 DAF브로드캐스팅 기법을 제안한다. [2]에서 제안하는 방법(그림 1)은 데이터의 접근빈도 차이를 고려하여 데이터를 전송하는 횟수를 구분함으로써 핫 데이터에 대한 접근시간은 상대적으로 줄이지만 클라이언트가 원하는 데이터를 얻을 때까지 오랜 시간 동안 무선 채널을 듣고 있어야 하므로 많은 에너지를 소비한다. [3]에서 제안하는 (1,m)인덱싱(그림 3)에서는 접근빈도를 고려하지 않은 인덱스를 추가하여 에너지 소비 면에서는 효율적이지만 추가된 인덱스의 비효율적인 중복에 따른 부가적인 지연이 발생한다.

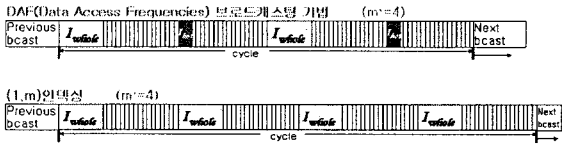

그림 4 . 제안기법인 DAF브로드캐스팅 기법과 (1,m)인덱싱 비교

본 논문에서는 데이터의 브로드캐스트 주기에 전체 데이터의 인덱스를 모두 추가하지 않고 데이터의 접근빈도를 고려한 일부 데이터에 대한 인덱스를 교차로 추가하여 배치하는 브로드캐스트 방법을 새로이 제안한다. 전체 데이터에 대한 인덱스의 전송횟수를 줄이는 대신, 핫 데이터에 대한 인덱스를 추가한 접근빈도를 고려한 인덱싱의 경우 그림 4와 같이 브로드캐스트 주기가 (1,m)인덱싱에 비해 짧아지기 때문에 많은 사용자들이 자주 원하는 데이터에 대한 인덱스를 얻기 위한 시간이 줄어들고, 따라서 전체 데이터들에 대한 접근시간이 줄어든다. 그림 5는 앞서 설명한 브로드캐스트 주기를 구성하는 방법을 명확히 하기 위해 기술한 알고리즘이다. 본 논문의 인덱스를 구성하기 위한 각 데이터에 대한 접근빈도와 핫 데이터, 콜드 데이터의 비율은 지프분포(Zipf distribution)를 따른다[4]. 표 1은 그림 5,6과 4장의 수식에서 사용할 파라미터이다.

index construction.
D sorted in the descending order according to access frequency;
D_{hot} is the number of n/(s+1)data which hold a high rank;
The remaining data is called D_{cold} = D-D_{hot};
I_{whole} is defined as the index of D;
I_{hot} is defined as the index of D_{hot};
BS_I is defined as broadcast cycle.

Algorithm BroadcastScheduling
//indices are distributed in broadcast cycle.
input: D, data set;
output: BS_I, broadcast schedule with index;
procedure:

```
count = 1
for (i=1; i≤m*; i++) do
    if count is odd then
        add I_whole to BS_I; //홀수 번째 I = I_whole
    else
        add I_hot to BS_I; //짝수 번째 I = I_hot
    end if
    add d_((i-1)n/m* n/m*)+1 to d_(in/m*) to BS_I;
      // D/m* 만큼의 데이터를 인덱스의 뒤에 추가
    count++;
end for
return BS_I;
```

그림 5. DAF브로드캐스팅 기법의 주기를 구성하는 알고리즘

표 1. 파라미터

m^*	the number of replications of Indices
BS_I	Broadcast Schedule with Index
D	$\{d_1, d_2, d_3, \cdots, d_n\}$, n= $\lvert D_{hot}\rvert+\lvert D_{cold}\rvert$
D_{hot}	$\{D_{hot}\in D \mid d_{h\_1}, d_{h\_2}, d_{h\_3}, \cdots, d_{h\_n/(s+1)}\}$
D_{cold}	$\{D_{cold}\in D \mid d_{c\_1}, d_{c\_2}, d_{c\_3}, \cdots, d_{c\_n\ [n/(s+1))}\}$
I_{whole}	Index of D
I_{hot}	Index of D_{hot}
D_{hot} : D_{cold}	1 : s (ratio of the number of data)
k	the number of levels in the I_{whole} tree
k'	the number of levels in the I_{hot} tree

그림 6은 클라이언트가 원하는 데이터를 다운로드 하는 프로토콜이다. 모든 데이터 버켓은 전체 데이터에 대한 인덱스와 핫 데이터에 대한 인덱스의 포인터, 두 개를 포함하고 클라이언트는

```
Mobile client requests for data with key Q
Tune into the broadcast channel
Read the bucket
Retrieve Pointers and Offset
Go to doze mode
(1) if data with key Q is D_hot then
        Tune in again index segment(I_whole or I_hot)
    else
        Tune in again index segment(I_whole)
    end if
    Read the bucket in the index segment

    Go to doze mode
    Tune in again when data with key Q are broadcasted
    Download data with key Q

(2) if Q = key have broadcasted then
        Go to doze mode
        Repeat from (1) in next broadcast
    end if
```

그림 6. 클라이언트의 다운로드 프로토콜

doze mode와 active mode로 동작한다. doze mode란 브로드캐스트 채널로 어떤 데이터가 들어오는지 확인하지 않고 최소의 전력만을 사용하여 작업하는 상태이고 active mode는 브로드캐스트 채널을 통해 원하는 데이터를 수신하기 위해 청취(tune)하는 상태로 에너지를 소비하는 상태를 말한다.

4. 수학적 성능 평가

이 장에서는 수학적 분석을 통해 DAF브로드캐스팅 기법을 평가하고 기존의 브로드캐스트 방법인 (1,m)인덱싱과 DAF브로드캐스팅 기법의 성능을 비교 분석한다. 브로드캐스트 방법의 성능은 접근시간(access time)과 튜닝시간(tuning time), 두 가지 기준에 의해 평가된다. 여기서 시간은 버켓의 수로 측정한다. 접근시간은 probe wait과 bcast wait을 합한 시간이며, probe wait은 클라이언트가 브로드캐스트 채널을 듣기 시작해서 원하는 데이터에 대한 인덱스 정보를 얻게 될 때까지의 평균시간이고, bcast wait은 인덱스를 듣은 시점으로부터 원하는 데이터를 얻을 때까지 소요되는 평균 시간을 나타낸다. 튜닝시간은 클라이언트가 실제로 채널을 듣는 시간을 나타낸다. 즉, 클라이언트가 데이터 수신을 위해 에너지를 사용하는 시간을 나타낸다.

DAF브로드캐스팅 기법의 접근시간을 수식으로 표현하면 수식 (1)~(5)와 같다. 수식 (1),(2)의 probe wait(D_{hot})과 probe wait(D_{cold})은 각각 D_{hot}와 D_{cold}에 대한 probe wait이고, 수식 (3)의 bcast wait은 전체 주기의 $\frac{1}{2}$이며, 수식 (4)의 접근시간(D_{hot})은 probe wait(D_{hot})과 bcast wait을 합한 시간이다. 수식 (5)의 접근시간(평균)은 D_{hot}와 D_{cold}, 두 데이터에 대한 평균 접근시간을 나타낸다.

$$\text{probe wait}(D_{hot}) = \frac{1}{4}\left(I_{whole} + I_{hot} + \frac{2D}{m^*}\right) \quad (1)$$

$$\text{probe wait}(D_{cold}) = \frac{1}{2}\left(I_{whole} + I_{hot} + \frac{2D}{m^*}\right) \quad (2)$$

$$\text{bcast wait} = \frac{1}{2}\left(\frac{m^* \times I_{whole}}{2} + \frac{m^* \times I_{hot}}{2} + D\right) + C \quad (3)$$

$$\text{접근시간}(D_{hot}) = \frac{1}{4}[(m^*+1) \times I_{whole} - (m^*+1) \times I_{hot} + (\frac{2}{m^*}+2) \times D] + C \quad (4)$$

$$\text{접근시간}(\text{평균}) = \frac{m^*}{4} \times (I_{whole} + I_{hot}) + \frac{3}{10} \times (I_{whole} - I_{hot}) + \frac{3D}{5m^*} + \frac{D}{2} + C \quad (5)$$

DAF브로드캐스팅 기법의 튜닝시간을 수식으로 표현하면 수식 (6)~(8)과 같다. 수식 (6),(7)의 튜닝시간(D_{hot})과 튜닝시간(D_{cold})은 각각 D_{hot}와 D_{cold}에 대한 튜닝시간이고, 수식 (8)의 튜닝시간(평균)은 D_{hot}와 D_{cold}, 두 데이터에 대한 평균 튜닝시간을 나타낸다.

$$\text{튜닝시간}(D_{hot}) = 1 + \frac{k-k'}{2} + C \quad (6)$$

$$\text{튜닝시간}(D_{cold}) = 1 + \frac{2k+k'}{2} + C \quad (7)$$

$$\text{튜닝시간}(\text{평균}) = 1 + \frac{6k+5k'}{10} + C \quad (8)$$

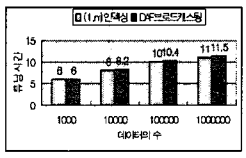

그림 7. 튜닝시간 비교

수식 (9)~(12)는 DAF브로드캐스팅 기법과 기존의 (1,m)인덱싱의 성능을 수학적으로 비교하기 위해 DAF브로드캐스팅 기법의 수식에서 접근시간=$\frac{1}{2}[(m^*+1) \times I_{whole} + \frac{m^*+1}{m^*} \times D] + C$, 튜닝시간= $1+k+C$ 인 (1,m)인덱싱의 수식을 뺀 것을 정리한 것이다.

$$\text{접근시간}(D_{hot}) = -\frac{1}{4}[(m^*+1) \times (I_{whole} - I_{hot})] \quad (9)$$

$$\text{접근시간}(\text{평균}) = -[I_{whole} \times (\frac{m^*}{4} + \frac{1}{5}) + \frac{D}{10m^*}] + [I_{hot} \times (\frac{m^*}{4} + \frac{3}{10})] \quad (10)$$

$$\text{튜닝시간}(D_{hot}) = -\frac{k-k'}{2} \quad (11)$$

$$\text{튜닝시간}(\text{평균}) = -\frac{4k-5k'}{10} \quad (12)$$

그림 7을 통해 알 수 있듯이 튜닝시간은 기존의 기법과 거의 차이가 없다. 그림 8은 수식 (4),(5)를 토대로 그래프를 이용해서 DAF브로드캐스팅 기법의 접근시간을 기존 기법과 비교한 것이다. 그래프를 통해, 브로드캐스트 주기가 짧아짐으로 인해 m^* 값의 변화에 따른 접근시간이 기존의 (1,m)인덱싱에 비해 완만히 증가하고, 모든 m^*값에서 기존의 기법인 (1,m)인덱싱보다 DAF브로드캐스팅 기법의 성능이 더 좋아진 것을 알 수 있다. 또한 데이터 수의 변화에도 DAF브로드캐스팅 기법이 기존기법보다 접근시간이 줄어듦을 알 수 있다. active mode일 때는 250mW/버켓 정도를 소비하고 active mode와 비교해서 doze mode일 때 사용하는 에너지는 아주 적지만 클라이언트가 doze mode일 때도 50μW/버켓 정도의 에너지를 소비하므로 접근시간을 줄이는 것은 클라이언트의 에너지 소비 또한 줄인다.

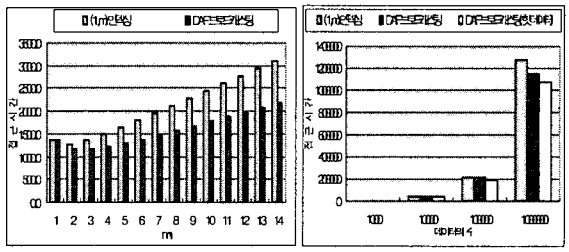

그림 8. 그래프를 통한 성능 비교 분석

5. 결론

이동 컴퓨팅 환경에서 데이터 전송을 위해 사용되는 기존의 브로드캐스트 기법들은 에너지의 효율성은 고려하지 않고 데이터의 접근빈도만을 고려하거나, 접근빈도는 고려하지 않은 인덱스를 배치하고 중복하는 방법으로 연구되었다. 본 논문에서는 브로드캐스트 전송 주기에 접근빈도를 고려한 인덱스를 교차하여 추가함으로써 에너지 효율성을 증가시키고 평균 접근시간을 줄이는 DAF브로드캐스팅 기법을 제안한다. 이 기법은 접근빈도를 고려한 인덱스를 전송함으로써 클라이언트의 에너지 소비를 줄인다. 또한 전체데이터에 대한 인덱스를 중복하는 대신 핫 데이터에 대한 인덱스를 교차시켜 중복시키는 방법으로 전체 브로드캐스트 주기를 줄여, 핫 데이터를 원하는 다수의 클라이언트들의 접근시간을 단축시킴으로써 데이터에 대한 평균 접근시간 줄인다.

6. 참고 문헌

[1] G.H. Forman and J. Zahorjan, "The Challenges of Mobile Computing," IEEE Computer 27(4), 1994, pp. 38-47.

[2] S. Acharya, R. Alonso, M. Franklin, and S. Zdonik, "Broadcast Disks: Data Management for Asymmetric Communications Environments," in Proceedings of the ACM SIGMOD Conference on Management of Data, San Jose, California, May 1995, pp. 199-210.

[3] T. Imielinski, S. Viswanathan, and B.R. Badrinath, "Data on Air: Organization and Access," IEEE Transactions of Data and Knowledge Engineering, vol. 9, no. 3, 1997, pp. 353-372.

[4] Chih-Hao Hsu, Guanling Lee, and Arbee L.P. Chen, "Index and Data Allocation on Multiple Broadcast Channels Considering Data Access Frequencies," in Proceedings of Third International Conference on Mobile Data Management, 2002, pp. 87-92.

[5] Pin-Kwang Eng, "Disseminating Data in Unreliable Wireless Environment," in Proceedings of Third International Conference on Mobile Data Management, 2002, pp. 157-158.

[6] Ming-Syan Chen, Kun-Lung Wu, and Philip S. Yu, "Optimizing Index Allocation for Sequential Data Broadcasting in Wireless Mobile Computing," Proceedings in IEEE Transactions on Knowledge and Data Engineering, 2003, pp. 161-173.

[7] A. Celik, Ping Ding, and J.A. Holliday, "Data broadcasting with Data Item Locality and Client Mobility," in Proceedings of the IEEE International Conference on Mobile Data Management, 2004, pp. 166.

[8] Andrew Y. Ho, and Dik Lun Lee, "Data Indexing for Heterogeneous Multiple Broadcast Channel," in Proceedings of the IEEE International Conference on Mobile Data Management, 2004, pp.274-283.

[9] B. Zheng, W.C. Lee, and D.L. Lee, "Search Continuous Nearest Neighbors On the Air," in Proceeding of the First Annual International Conference on Mobile and Ubiquitous Systems: Networking and Services, 2004